



TESIS - BM185407

**USE CASE POINT VS. FUNCTION POINT SEBAGAI  
EFFORT DAN COST ESTIMATION  
PENGEMBANGAN PERANGKAT LUNAK DALAM  
AGILE FRAMEWORK PADA PT. VASCOMM SOLUSI  
TEKNOLOGI**

**LIA NINDA SAFITRI  
09211750054014**

**Dosen Pembimbing:  
Prof. Dr. techn. Drs. M. Isa Irawan, M.T.**

**Departemen Manajemen Teknologi  
Fakultas Desain Kreatif Dan Bisnis Digital  
Institut Teknologi Sepuluh Nopember  
2020**

## LEMBAR PENGESAHAN TESIS

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar

**Magister Manajemen Teknologi (M.MT)**

di

**Institut Teknologi Sepuluh Nopember**

Oleh:

**LIA NINDA SAFITRI**

**NRP: 09211750054014**

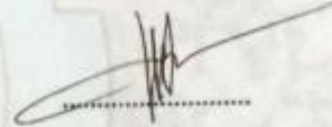
**Tanggal Ujian: 23 Januari 2020**

**Periode Wisuda: Maret 2020**

Disetujui oleh:

**Pembimbing:**

1. Prof. Dr. Techn. Drs. M. Isa Irawan, M.T.  
NIP: 196312251989031001



**Penguji:**

1. Dr. Techn. Ir. R. V. Hari Ginardi, M.Sc.  
NIP: 196505181992031003



2. Faizal Mahananto, S.Kom., M.Eng., Ph.D  
NIP: 5200201301010

Kepala Departemen Manajemen Teknologi  
Fakultas Desain Kreatif dan Bisnis Digital

**Prof. Ir. I Nyoman Pujawan, M.Eng, Ph.D, CSCP**  
NIP: 196912311994121076



# **USE CASE POINT VS. FUNCTION POINT SEBAGAI EFFORT DAN COST ESTIMATION PENGEMBANGAN PERANGKAT LUNAK DALAM AGILE FRAMEWORK PADA PT. VASCOMM SOLUSI TEKNOLOGI**

Nama : Lia Ninda Safitri  
NRP : 09211750054014  
Pembimbing : Prof. Dr. Techn. Drs. M. Isa Irawan, M.T.

## **ABSTRAK**

Manajemen biaya perangkat lunak adalah salah satu faktor terpenting dalam menentukan keberhasilan proyek pengembangan perangkat lunak. Satu tahap dalam perencanaan proyek pengembangan perangkat lunak adalah estimasi usaha dan biaya. Penganggaran berdasarkan usaha juga dilakukan secara transparan dan akuntabel.

Salah satu masalah yang dihadapi PT. Vascomm Solusi Teknologi dalam pengerjaan suatu proyek adalah perhitungan estimasi biaya belum memiliki standar. Perbedaan biaya estimasi dan biaya eksekusi terjadi pada tahun 2019. Hal ini dikarenakan kurang tepatnya perhitungan estimasi usaha yang berkenaan langsung dengan lama periode dan biaya pengembangan.

PT. Vascomm Solusi Teknologi menggunakan Framework Agile-Scrum sebagai pengembangan perangkat lunaknya dan Object Oriented Programming pada sebagian besar proyeknya. Dibutuhkan model perhitungan untuk PT. Vascomm Solusi Teknologi yang sesuai dengan model pengembangannya. Use Case Point dapat dijadikan sebagai alat ukur usaha pengembangan perangkat lunak sekaligus dijadikan dasar untuk perhitungan biaya pengembangan. Use Case Point juga mempertimbangkan faktor teknis dan lingkungan.

Oleh karena itu, hal baru dalam penelitian ini adalah bagaimana menerapkan Use Case Point pada Scrum Framework untuk mendapatkan estimasi usaha dan biaya yang lebih akurat. Penelitian ini bertujuan untuk membandingkan hasil perhitungan realita dengan estimasi, Use Case Point dan Function Point dan perhitungan sebenarnya yang ada di PT. Vascomm Solusi Teknologi. Hasilnya didapatkan bahwa Use Case Point dapat diterapkan untuk perhitungan estimasi biaya pengembangan. Use Case Point dibandingkan Realita memiliki nilai deviasi yang lebih kecil yaitu 4-8%. Function Point dibandingkan Realita memiliki nilai deviasi yang lebih besar yaitu sekitar 86-169%. Realita-Estimasi dengan nilai deviasi 14-17%. Use Case Point menekan selisih biaya estimasi dan menjadi lebih akurat.

**Kata kunci:** *Agile Development, Estimasi Usaha, Estimasi Biaya, Proyek Pengembangan Perangkat Lunak, Scrum, Use Case Point.*



# **USE CASE POINT AND FUNCTION POINT AS SOFTWARE DEVELOPMENT EFFORT AND COST ESTIMATION ON AGILE FRAMEWORK AT PT. VASCOMM SOLUSI TEKNOLOGI**

Student's Name : Lia Ninda Safitri  
Student ID : 09211750054014  
Supervisor : Prof. Dr. Techn. Drs. M. Isa Irawan, M.T.

## **ABSTRACT**

*Software cost management is one of the most important factors that determine the success of software development project. One step in software development project planning is effort and cost estimation. Budgeting based on effort must be transparent and accountable.*

*One of the problems faced by PT. Vascomm Solusi Teknologi in software development is the unavailability of cost estimation calculation standard. The difference between estimation and actual cost occurred in 2019. It is caused by the low accuracy of effort estimation calculation that is directly related to the length of period and development cost.*

*PT. Vascomm Solusi Teknologi uses Agile-scrum Framework as a model of software development and Object Oriented Programming in most of the projects. The calculation model that suits its development model is needed. Use case point can be used as measurement instrument for software development effort, also as a base for calculating the development cost. Use case point is not only calculating the number of features, but also considering technical and environmental factors.*

*Thus, the novelty of this research is how to apply the use case point in scrum framework to obtain more accurate effort and cost estimations. This research aims to compare the result of use case point with function point calculation and the actual calculation PT. Vascomm Solusi Teknologi. The results show that Use Case Point can be applied for calculating development cost estimation. The deviation of cost estimation compared to actual cost for Use Case Point, Function Point and, Estimation are 4-8%, 86-169%, and 14-17%, respectively. Use Case point is able to decrease the gap of estimation cost to be more accurate.*

**Kata kunci:** *Agile Development, Cost Estimation, Effort Estimation, Software Development Project, Scrum, Use Case Point.*



## KATA PENGANTAR

Puji syukur kehadirat Allah SWT, hanya dengan curahan rahmat dan hidayah-Nya penulis dapat menyelesaikan Proposal Tesis yang berjudul *Use Case Point* sebagai Perangkat lunak *Effort* dan *Cost Estimation* dalam *Agile Framework* pada PT. Vascomm Solusi Teknologi. Tesis ini diajukan untuk memenuhi prasyarat untuk menyelesaikan studi magister di Program Studi Magister Manajemen Teknologi, Bidang Manajemen Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya. Dalam penyelesaian Tesis ini, penulis mendapatkan banyak dukungan moral dan material dari banyak pihak. Atas bantuan tersebut penulis ingin menyampaikan penghargaan dan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Ir. I Nyoman Pujawan., M.Eng., Ph.D., CSCP selaku Ketua Program Studi Magister Manajemen Teknologi.
2. Bapak Dr. Techn.Ir. R.V. Hari Ginardi, M.Sc. selaku dosen wali.
3. Bapak Prof. Dr. Techn. Drs. M. Isa Irawan, M.T. selaku dosen pembimbing tesis yang telah meluangkan waktu, tenaga dan pikiran dalam memberikan bimbingan, pengarahan, dan ilmu pengetahuan.
4. Seluruh dosen pengajar yang telah memberikan pengajaran dan ilmu yang begitu banyak. Serta seluruh karyawan MMT-ITS yang telah membantu selama masa perkuliahan. Terima kasih atas ilmu yang telah diajarkan.
5. Orang tua (Sunardi dan Kasrieni) dan keluarga yang memberikan doa dan dukungannya selama ini.
6. Teman-teman MTI angkatan 2018 yang selalu memotivasi, mengingatkan, memberi masukan kepada penulis dalam penyusunan Tesis ini.
7. Semua pihak yang tidak dapat disebutkan satu persatu, yang telah banyak memberikan berbagai macam bantuan dalam penyusunan Tesis ini.

Surabaya, Agustus 2019

Lia Ninda Safitri





## DAFTAR ISI

|   |      |
|---|------|
| LEMBAR PENGESAHAN TESIS .....                           | ii   |
| ABSTRAK .....   | iii  |
| ABSTRACK .....  | v    |
| KATA PENGANTAR .....                                    | vii  |
| DAFTAR ISI.....   | ix   |
| DAFTAR GAMBAR .....                                     | xiii |
| DAFTAR TABEL.....                                       | xv   |
| BAB 1    PENDAHULUAN .....                              | 1    |
| 1.1    Latar Belakang .....                             | 1    |
| 1.2    Rumusan Masalah .....                            | 4    |
| 1.3    Batasan Masalah.....                             | 4    |
| 1.4    Tujuan Penelitian.....                           | 5    |
| 1.5    Manfaat Penelitian.....                          | 5    |
| 1.6    Kontribusi Penelitian .....                      | 5    |
| 1.7    Sistematika Penulisan.....                       | 6    |
| BAB 2    KAJIAN PUSTAKA.....                            | 7    |
| 2.1    Related Work.....                                | 7    |
| 2.2    PT. Vascomm Solusi Teknologi.....                | 8    |
| 2.2.1    Kontak dan Kantor Vascomm.....                 | 8    |
| 2.2.2    Product dan Solusi Vascomm .....               | 9    |
| 2.3    Manajemen Proyek.....                            | 10   |
| 2.3.1    Metodologi Proyek IT .....                     | 12   |
| 2.4 <i>Software Development Life Cycle (SDLC)</i> ..... | 12   |
| 2.5    Project <i>Cost</i> Management .....             | 13   |

|       |  |    |
|-------|--|----|
| 2.5.1 | <i>Software Cost Estimation</i> .....            | 13 |
| 2.6   | <i>Agile Framework</i> .....                     | 14 |
| 2.6.1 | Scrum Method .....                               | 14 |
| 2.7   | <i>Use Case Point</i> .....                      | 16 |
| 2.7.1 | <i>Unadjusted Actor Weights (UAW)</i> .....      | 16 |
| 2.7.2 | <i>Unadjusted Use Case Weights (UUCW)</i> .....  | 16 |
| 2.7.3 | <i>Unadjusted Use Case Point (UUCP)</i> .....    | 17 |
| 2.7.4 | <i>Technical Complexity Factors (TCF)</i> .....  | 17 |
| 2.7.5 | <i>Environment Complexity Factor (ECF)</i> ..... | 18 |
| 2.7.6 | <i>Estimasi Effort / Usaha</i> .....             | 18 |
| 2.8   | <i>Function Point</i> .....                      | 19 |
| 2.8.1 | <i>Unjusted Function Point Complexity</i> .....  | 19 |
| 2.8.2 | <i>Unjusted Weight Complexity</i> .....          | 20 |
| 2.8.3 | <i>Unadjusted Function Point</i> .....           | 21 |
| 2.8.4 | <i>Technical Complexity Factor</i> .....         | 21 |
| 2.8.5 | <i>Function Point</i> .....                      | 22 |
| 2.8.6 | <i>Estimasi Effort / Usaha</i> .....             | 22 |
| 2.8.7 | <i>Estimasi Biaya</i> .....                      | 23 |
| BAB 3 | <b>METODOLOGI PENELITIAN</b> .....               | 25 |
| 3.1   | Studi Literatur .....                            | 26 |
| 3.2   | Pengumpulan data .....                           | 26 |
| 3.2.1 | Wawancara .....                                  | 27 |
| 3.2.2 | Lembar Penilaian .....                           | 27 |
| 3.3   | Analisa Hasil .....                              | 28 |
| 3.3.1 | <i>Use Case Point</i> .....                      | 28 |
| 3.3.2 | <i>Function Point</i> .....                      | 29 |

|   |  |     |
|---|--|-----|
| 3.4   | Pembahasan dan Evaluasi .....  | 31  |
| 3.5   | Kesimpulan dan Saran .....   | 31  |
| BAB 4   | ANALISA DAN PERHITUNGAN.....   | 33  |
| 4.1   | Pembahasan Aplikasi .....  | 33  |
| 4.1.1   | Aplikasi 1 : CEKLOK.ID.....  | 33  |
| 4.1.2   | Aplikasi 2 : COMMJUNCTION.....   | 36  |
| 4.2   | Perhitungan <i>Use Case Point</i> .....  | 37  |
| 4.2.1   | Aplikasi 1 : CEKLOK.ID.....  | 38  |
| 4.2.2   | Aplikasi 2 : COMMJUNCTION.....   | 50  |
| 4.3   | Perhitungan Function Point .....   | 61  |
| 4.3.1   | Aplikasi 1 : CEKLOK.ID.....  | 61  |
| 4.3.2   | Aplikasi 2 : COMMJUNCTION.....   | 71  |
| BAB 5   | PEMBAHASAN & EVALUASI.....   | 81  |
| 5.1   | Analisis Perbandingan Hasil Durasi Pengerjaan dan Biaya Pengembangan antar Metode..... | 81  |
| BAB 6   | KESIMPULAN.....  | 85  |
| 6.1   | Kesimpulan.....  | 85  |
| 6.2   | Saran.....   | 86  |
| DAFTAR PUSTAKA  | .....  | 87  |
| LAMPIRAN PENILAIAN TECHNICAL DAN ENVIRONMENT COMPLEXITY FACTOR PADA <i>USE CASE POINT</i> | .....  | 89  |
| LAMPIRAN PENILAIAN TECHNICAL COMPLEXITY FACTOR PADA <i>FUNCTION POINT</i>                 | .....  | 95  |
| LAMPIRAN HASIL PENILAIAN COMPLEXITY FACTOR CEKLOK   | .....  | 104 |
| Technical Complexity Factor pada Metode Use Case Point                                    | .....  | 104 |
| Environment Complexity Factor pada Metode Use Case Point                                  | .....  | 105 |

|   |     |
|---|-----|
| Technical Complexity Factor pada Metode Function Point.....         | 106 |
| LAMPIRAN HASIL PENILAIAN COMPLEXITY FACTOR COMMJUNCTION<br>.....    | 107 |
| Technical Complexity Factor pada Metode Use Case Point .....        | 107 |
| Environment Complexity Factor pada Metode Use Case Point.....       | 108 |
| Technical Complexity Factor pada Metode Function Point.....         | 109 |
| LAMPIRAN PERHITUNGAN SPRINT RATE DAN DAY RATE CEKLOK                | 110 |
| LAMPIRAN PERHITUNGAN SPRINT RATE DAN DAY RATE<br>COMMJUNCTION ..... | 111 |

## DAFTAR GAMBAR

|   |    |
|---|----|
| Gambar 1.1 Gap Software Development Cost Vascomm.....                           | 3  |
| Gambar 2.1 Kantor Vascomm .....   | 9  |
| Gambar 2.2 Vascomm sebagai Information Technology .....                         | 9  |
| Gambar 2.3 Vascomm sebagai Startup Factory .....                                | 10 |
| Gambar 2.4 Kerangka Kerja Scrum .....   | 15 |
| Gambar 3.1 Alur Penelitian.....   | 25 |
| Gambar 3.2 Langkah <i>Use Case Point</i> .....                                  | 28 |
| Gambar 4.1 Use Case CEKLOK.....   | 38 |
| Gambar 4.2 Use Case COMMJUNCTION.....   | 50 |
| Gambar 5.1 Deviasi Biaya Pengembangan CEKLOK terhadap Biaya Realita ....        | 82 |
| Gambar 5.2 Deviasi Biaya Pengembangan COMMJUNCTION terhadap Biaya Realita ..... | 82 |



## DAFTAR TABEL

|   |    |
|---|----|
| Tabel 2.1 Penelitian Terdahulu .....  | 7  |
| Tabel 2.2 Tipe dan Bobot Aktor .....  | 16 |
| Tabel 2.3 Tipe dan Bobot Use Case pada <i>Use Case Point</i> .....              | 17 |
| Tabel 2.4 Technical Factors .....   | 17 |
| Tabel 2.5 Environment Complexity .....  | 18 |
| Tabel 2.6 Bobot Kompleksitas External Input .....                               | 21 |
| Tabel 2.7 Bobot Kompleksitas External Output dan External Inquiries .....       | 21 |
| Tabel 2.8 Bobot Kompleksitas Logical Internal dan External Interface File ..... | 21 |
| Tabel 2.9 Perhitungan Unadjusted Function Point .....                           | 21 |
| Tabel 2.10 Bobot Technical Complexity Factor .....                              | 22 |
| Tabel 2.11 Technical Factors .....  | 22 |
| Tabel 2.12 Jone's First Order Estimate Exponent .....                           | 23 |
| Tabel 4.1 Rangkuman Aplikasi CEKLOK .....                                       | 33 |
| Tabel 4.2 Rangkuman Aplikasi COMMJUNCTION .....                                 | 36 |
| Tabel 4.3 Deskripsi Aktor pada Aplikasi 1 .....                                 | 39 |
| Tabel 4.4 Perhitungan Unadjusted Actor Weight (UAW) .....                       | 40 |
| Tabel 4.5 Daftar Tipe Use Case Aplikasi 1 .....                                 | 40 |
| Tabel 4.6 Perhitungan Unadjusted Use Case Weight (UUCW) Aplikasi 1 .....        | 46 |
| Tabel 4.7 Perhitungan Technical Complexity Factors Aplikasi 1 .....             | 47 |
| Tabel 4.8 Perhitungan Environmental Complexity Factors Aplikasi 1 .....         | 48 |
| Tabel 4.9 Deskripsi Aktor pada Aplikasi 2 .....                                 | 51 |
| Tabel 4.10 Perhitungan Unadjusted Actor Weight (UAW) .....                      | 51 |
| Tabel 4.11 Daftar Tipe Use Case Aplikasi 2 .....                                | 52 |
| Tabel 4.12 Perhitungan Unadjusted Use Case Weight (UUCP) Aplikasi 2 .....       | 57 |
| Tabel 4.13 Perhitungan Technical Complexity Factors Aplikasi 2 .....            | 58 |
| Tabel 4.14 Perhitungan Environmental Complexity Factors Aplikasi 2 .....        | 59 |
| Tabel 4.15 Function Type CEKLOK .....   | 61 |
| Tabel 4.16 Penentuan Bobot Tipe Fungsi Pengguna EI .....                        | 63 |
| Tabel 4.17 Penentuan Bobot Tipe Fungsi Pengguna EO .....                        | 64 |
| Tabel 4.18 Penentuan Bobot Tipe Fungsi Pengguna EQ .....                        | 64 |

|   |    |
|---|----|
| Tabel 4.19 Penentuan Bobot Tipe Fungsi Pengguna ILF .....       | 65 |
| Tabel 4.20 Penentuan Bobot Tipe Fungsi Pengguna EIF .....       | 67 |
| Tabel 4.21 Perhitungan UFP .....                                | 68 |
| Tabel 4.22 Hasil Technical Complexity Factors.....              | 69 |
| Tabel 4.23 Estimasi Biaya CEKLOK .....                          | 71 |
| Tabel 4.24 Function Type COMMJUNCTION .....                     | 71 |
| Tabel 4.25 Penentuan Bobot Tipe Fungsi Pengguna EI .....        | 73 |
| Tabel 4.26 Penentuan Bobot Tipe Fungsi Pengguna EO.....         | 74 |
| Tabel 4.27 Penentuan Bobot Tipe Fungsi Pengguna EQ.....         | 74 |
| Tabel 4.28 Penentuan Bobot Tipe Fungsi Pengguna ILF .....       | 75 |
| Tabel 4.29 Penentuan Bobot Tipe Fungsi Pengguna EIF .....       | 75 |
| Tabel 4.30 Perhitungan UFP .....                                | 76 |
| Tabel 4.31 Hasil Technical Complexity Factors.....              | 77 |
| Tabel 4.32 Estimasi Biaya COMMJUNCTION .....                    | 79 |
| Tabel 5.1 Perbandingan Hasil Metode Aplikasi CEKLOK .....       | 81 |
| Tabel 5.2 Perbandingan Hasil Metode Aplikasi COMMJUNCTION ..... | 81 |
| Tabel 5.3 Selisih Biaya antar Metode .....                      | 82 |



# **BAB 1**

## **PENDAHULUAN**

Pada Bab ini akan dijelaskan mengenai beberapa hal dasar dalam pembuatan proposal penelitian yang meliputi latar belakang, perumusan masalah, tujuan, manfaat, kontribusi penelitian, dan batasan masalah.

### **1.1 Latar Belakang**

Setiap tahunnya jumlah pengeluaran untuk teknologi informasi terus meningkat. Melalui lembaga penelitian, International Data Center (IDC) Indonesia, anggaran belanja untuk teknologi informasi dan komunikasi (TIK) pada 2016 diperkirakan tumbuh 8,3% yaitu sekitar Rp 197,4 triliun (Dewi, Prassida, Sholiq, & Subriadi, 2016).

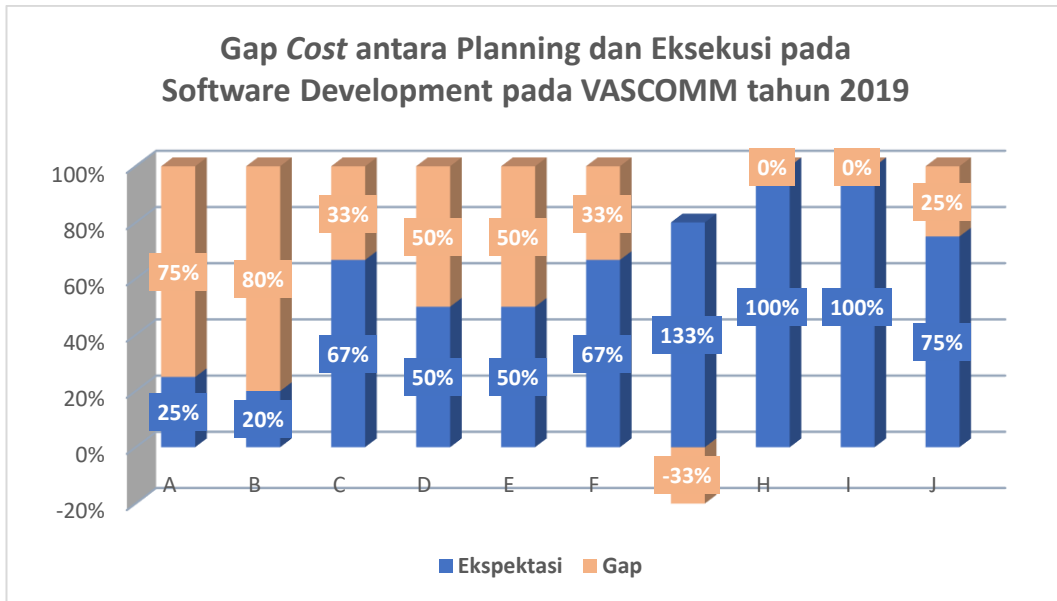
Peneliti McKinsey dalam (Dewi, Prassida, Sholiq, & Subriadi, 2016) menyatakan bahwa penyebab kegagalan proyek teknologi informasi, khususnya perangkat lunak, sebesar 66% karena anggaran yang melebihi batas, 33% karena lembur, dan sisanya 17% karena nilai / manfaatnya kurang. Penelitian yang sama dari Standish Group, hingga 2015 ada total 71% mengatakan pengembangan teknologi informasi gagal disebabkan oleh beberapa faktor, termasuk kurangnya dukungan dari eksekutif, kematangan organisasi, keterlibatan pengguna, untuk tujuan ambigu perusahaan. Salah satu bentuk dukungan eksekutif adalah kemampuan untuk memperkirakan sumber daya yang akan digunakan selama proyek. Di Indonesia, metode estimasi upaya proyek pengembangan perangkat lunak tidak terlalu menjadi perhatian khusus. Berdasarkan pengamatan, hanya prinsip keadilan yang digunakan selama penentuan biaya pengembangan perangkat lunak (Dewi, Prassida, Sholiq, & Subriadi, 2016).

Di dalam manajemen proyek terdapat proses perencanaan, pengorganisasian, pengarahan dan pengendalian sumber daya organisasi agar tercapainya tujuan dengan sumber daya tertentu dan waktu tertentu (Zahiroh, Saputra, & Herlambang, 2018). Selain itu manajemen proyek harus mempertimbangkan tiga aspek yaitu ruang lingkup (*scope*), waktu (*time*) dan biaya (*Cost*) agar proyek dapat dikatakan sukses (Schwalbe, 2014).

Manajemen biaya/*cost* pada pengembangan perangkat lunak adalah salah satu faktor terpenting dalam menentukan keberhasilan pengembangan industri perangkat lunak (Fan & al, 2009; Qina & Fang, 2011). Estimasi merupakan antisipasi dari sumber daya yang membutuhkan penyelesaian dalam pengembangan perangkat lunak dengan memperhatikan lingkungan dan metode (Yuliansyah, Qudsiyah, Zahrotun, & Arfiani, 2018). Satu tahap dalam perencanaan proyek pengembangan perangkat lunak adalah estimasi usaha/*effort* (Sholih, Dewi, & Subriadi, 2017). Estimasi biaya merupakan proses memprediksi usaha yang dibutuhkan untuk mengembangkan perangkat lunak (Fan & al, 2009), karena sebagai hasil yang tidak berwujud, penganggaran berdasarkan usaha juga harus dilakukan secara transparan dan akuntabel (Dewi, Subriadi, & Sholih, A Modification Complexity Factor in Function Points Method for Software Cost Estimation Towards Public Service Application, 2017).

PT. Vascomm Solusi Teknologi berdiri sejak tahun 2006 dan selalu mengalami peningkatan setiap tahunnya dalam melakukan pengembangan perangkat lunak dalam bentuk website maupun mobile apps baik milik instansi maupun perusahaan. Oleh karena itu, perlu adanya manajemen proyek perangkat lunak yang baik agar teknologi informasi yang dikembangkan sesuai dengan kebutuhan pengguna.

Masalah yang dihadapi PT. Vascomm Solusi Teknologi dalam pengerjaan suatu proyek yaitu dalam perhitungan estimasi biaya belum memiliki standar perhitungan. Pada tahun 2019 60% proyek PT. Vascomm Solusi Teknologi harus mengalami kemunduran karena pengembangan aplikasi perpanjangan waktu dari timeline awal perencanaan. Kurang tepatnya estimasi penyelesaian pekerjaan pada pengembangan perangkat lunak menjadi kendalanya. Perpanjangan timeline ini membuat biaya pengembangan perangkat lunak menjadi membengkak. Dapat dilihat pada Gambar 1.1 tentang grafik biaya pengembangan perangkat lunak. Bagan berwarna biru merupakan biaya/waktu awal perencanaan dan bagan bagian atas merupakan selang (gap) dengan eksekusi. Jika dirata-rata selang (gap) biaya pengembangan perangkat lunak proyek sebesar 29%.



**Gambar 1.1 Gap Software Development Cost Vascomm**

- |                      |                          |
|----------------------|--------------------------|
| A. BLINK             | F. Smart Residence       |
| B. BlinkPay          | G. INKOWAPI              |
| C. BNI Syariah HL    | H. QRIS Management       |
| D. Bansos            | I. Redesign Portal Agent |
| E. VA Debit Kemenkeu | J. Hadeer                |

PT. Vascomm Solusi Teknologi sendiri menggunakan *Framework Agile-Scrum* sebagai model pengembangan perangkat lunaknya dan memakai *Object Oriented Programming* pada sebagian besar proyeknya.

*Use Case Point* (UCP) merupakan metode yang dapat dijadikan sebagai alat untuk mengukur usaha pengembangan Perangkat lunak yang selanjtnya dapat digunakan untuk perhitungan biaya pengembangan. UCP menjadi semakin populer untuk perhitungan estimasi usaha dan biaya pengembangan perangkat lunak sebagaimana dibuktikan oleh banyak penelitian tentang UCP dengan respon cukup baik. Studi ini meliputi yang berikut: (i) UCP lebih baik dari yang diharapkan oleh para ahli, penyimpangan UCP adalah 19% sedangkan penyimpangan para ahli 20%; (ii) Penyimpangan UCP sekitar 9% ketika diuji untuk sekitar 200 proyek pengembangan perangkat lunak; dan (iii) penyimpangan UCP hanya 6,89% pada pengembangan perangkat lunak proyek skala kecil-menengah (Dewi, Prassida, Sholiq, & Subriadi, 2016).

Pada penelitian yang dilakukan oleh (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018) dan (Kang, Choi, & Baik, 2010) bahwa *Use Case Point* terbukti dapat dijadikan alat ukur estimasi usaha pengembangan software dengan *Framework Agile-Scrum* menggunakan Story point. Berbagai metode estimasi telah diterapkan dalam *Agile Development*, meskipun sebagian besar metode ini tidak menghasilkan nilai akurasi prediksi yang baik. Namun, Story Point dan *Use Case Point* adalah teknik yang mendekati dengan metode spesifikasi persyaratan perangkat lunak seperti User story dan Use case. Spesifikasi ini adalah yang paling umum digunakan. Berdasarkan penelitian, Story Point adalah metode yang telah banyak diimplementasikan dalam *Framework Scrum* tidak dengan *Use Case Point* (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018). Oleh karena itu, hal baru dalam penelitian ini adalah bagaimana menerapkan *Use Case Point* pada *Scrum Framework*. Ini merupakan peningkatan untuk memanfaatkan estimasi usaha sekaligus biaya perangkat lunak dalam proyek pengembangan perangkat lunak.

Berdasarkan latar belakang yang ada di atas serta penelitian yang dilakukan, maka diharapkan dapat mengangkat bahasan tentang estimasi perhitungan usaha dan biaya pengembangan Perangkat lunak menggunakan *Use Case Point* sebagai bahan pertimbangan PT. Vascomm Solusi Teknologi. Kemudian akan dilakukan perbandingan antara metode *Use Case Point* dengan biaya proyek sebenarnya.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah disebutkan maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana hasil penerapan metode *Use Case Point* dalam mengestimasi biaya untuk pengembangan pada PT. Vascomm Solusi Teknologi?
2. Bagaimana hasil perbandingan perbandingan evaluasi biaya menggunakan metode *Use Case Point* dan Biaya proyek sesungguhnya?

## **1.3 Batasan Masalah**

Batasan masalah ditentukan dalam penelitian ini adalah sebagai berikut:

1. Proyek yang digunakan merupakan proyek yang dikerjakan oleh PT. Vascomm Solusi Teknologi.
2. Ruang lingkup penelitian ini hanya dibatasi sampai tiga tahap *Software Development Life Cycle* (SDLC) yaitu pada tahap *Planning, Analysis, Design* dan *Implementation*.
3. Pembuatan *User Story, Use Case Diagram* dan berdasarkan hasil wawancara dan validasi dengan pihak PT. Vascomm Solusi Teknologi.
4. Menggunakan 2 Project Pengembangan Aplikasi yaitu CEKLOK dan COMMJUNCTION

#### **1.4 Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah sebagai berikut :

1. Menjelaskan hasil penerapan metode *Use Case Point* dalam mengestimasi biaya untuk pengembangan proyek pada PT. Vascomm Solusi Teknologi.
2. Menjelaskan hasil perbandingan perbandingan evaluasi biaya menggunakan metode *Use Case Point*, Biaya proyek sesungguhnya.

#### **1.5 Manfaat Penelitian**

Bahan pertimbangan PT. Vascomm Solusi Teknologi untuk perbaikan dan evaluasi estimasi usaha dan biaya terhadap proyek-proyek yang dijalankan, terutama untuk proyek yang sejenis.

#### **1.6 Kontribusi Penelitian**

Kontribusi dari penelitian ini adalah :

1. Kontribusi Teori :
  - Poin-poin yang dijadikan acuan pada *Use Case Point* adalah hasil penelitian sebelumnya.
  - Menghitung estimasi biaya proyek secara otomatis berdasar estimasi usaha dari nilai *Use Case Point* .
2. Kontribusi Praktis :
  - Mendapatkan nilai estimasi usaha dan biaya proyek PT. Vascomm Solusi Teknologi.

- Sebagai bahan pertimbangan manajerial PT. Vascomm Solusi Teknologi untuk kebijakan keputusan terhadap proyek kedepannya.

## **1.7 Sistematika Penulisan**

Berikut adalah sistematika penulisan yang digunakan pada peneltian ini:

### **Bab I Pendahuluan**

Bab ini menyajikan mengenai latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, metodologi, dan sistematika penulisan.

### **Bab II Tinjauan Pustaka**

Bab ini menyajikan tentang kajian literatur mengenai teori dan penelitian sebelumnya yang telah dilakukan yang mendasari penelitian.

### **Bab III Metode Penelitian**

Bab ini menyajikan metode dan langkah-langkah yang dilakukan oleh peneliti dalam melakukan penelitian ini.

### **Bab IV Hasil Penelitian dan Pembahasan**

Bab ini menyajikan mengenai pembahasan hasil penelitian yang telah dilakukan dan analisis hasil penelitian yang diperoleh.

### **Bab V Kesimpulan dan Saran**

Bab ini menyajikan kesimpulan dan saran yang didapatkan dari pembahasan hasil penelitian.

## BAB 2

### KAJIAN PUSTAKA

Bab ini menjelaskan materi penunjang penelitian, kajian dan perbandingan antara penelitian sebelumnya yang terkait masalah yang dibahas pada penelitian ini.

#### 2.1 Related Work

Terdapat beberapa penelitian yang menjadi dasar penelitian ini, berikut penelitian-penelitian yang dimaksud tertera pada Tabel 2.1:

**Tabel 2.1 Penelitian Terdahulu**

| No. | Judul  | Penulis  | Hasil Penelitian   |
|-----|--|--|--|
| 1   | Implementation of <i>Use Case Point</i> as Software <i>Effort</i> Estimation in Scrum  | (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018) | Hasil kolaborasi <i>Use Case Point</i> dan Scrum <i>Framework</i> akan simulasikan menggunakan sistem manajemen proyek berbasis web. Implementasi dari <i>Use Case Point</i> dan Scrum dapat dilakukan dengan mengaitkan User Story pada Product Backlog dengan Use Case pada <i>Use Case Point</i> dan dengan memodifikasi atribut User Story ke Fungsionalitas dan menambahkan atribut transaksi ketika mendefinisikan User Story pada Product Backlog . |
| 2   | Perbandingan Evaluasi Biaya Pengembangan Sistem Antrian RSUD Dr. Soetrasno Rembang Menggunakan Metode <i>Use Case Point</i> dan <i>Function Point</i> (Studi Kasus: CV Pabrik Teknologi) | (Zahiroh, Saputra, & Herlambang, 2018)           | Perbandingan antara hasil estimasi biaya menggunakan <i>Use Case Point</i> , <i>Function Point</i> dan <i>Guesstimate</i> pada proyek dengan tipe pengembangan <i>waterfall</i> .  |

|   |  |                             |  |
|---|--|-----------------------------|--|
| 3 | <i>Effort Distribution to Estimate Cost in Small to Medium Software Development Project with Use Case Points</i> | (Primandari & Sholiq, 2016) | Temuan dari penelitian ini adalah nilai persentase upaya untuk tiga kategori fase pengembangan perangkat lunak pada proyek kecil-menengah di Indonesia. Hasil lain dari penelitian ini adalah langkah-langkah untuk menentukan perkiraan biaya pengembangan perangkat lunak dengan metode <i>Use Case Points</i> . |
| 4 | <i>Effort and Cost Allocation in Medium to Large Software Development Projects</i>                               | (Saleh, 2011)               | Pedoman untuk alokasi biaya dan upaya berdasarkan aktivitas pengembangan perangkat lunak dengan metode <i>Use Case Points</i> .  |

## 2.2 PT. Vascomm Solusi Teknologi

Perjalanan bisnis Vascomm berkecimpung di industri teknologi sudah dimulai sejak 13 tahun lalu, tepatnya pada April 2006. Mulanya, perusahaan bernama CV Alcomindo Jaya (CV. AJ). Hingga saat ini, perusahaan yang telah mengganti namanya menjadi PT. Vascomm Solusi Teknologi ini mencoba pendekatan baru dalam merencanakan, mengembangkan konsep dan gagasan, serta mengoperasionalkannya.

Sebagai sebuah learning organization, Vascomm terus berbenah diri mengembangkan model bisnisnya dengan dukungan talenta muda, generasi millennial yang punya segudang karakteristik. Ini terwujud melalui tujuan jangka panjang Vascomm selain sebagai IT solution, juga menjadi startup factory yang akan melahirkan banyak startup berbasis teknologi.

### 2.2.1 Kontak dan Kantor Vascomm

Kantor Pusat Vascomm berada di Perkantoran Gateway Blok A No 12, Jl. Raya Waru, Sawotratap Kab Sidoarjo, Jawa Timur 61254 – Indonesia. Dan



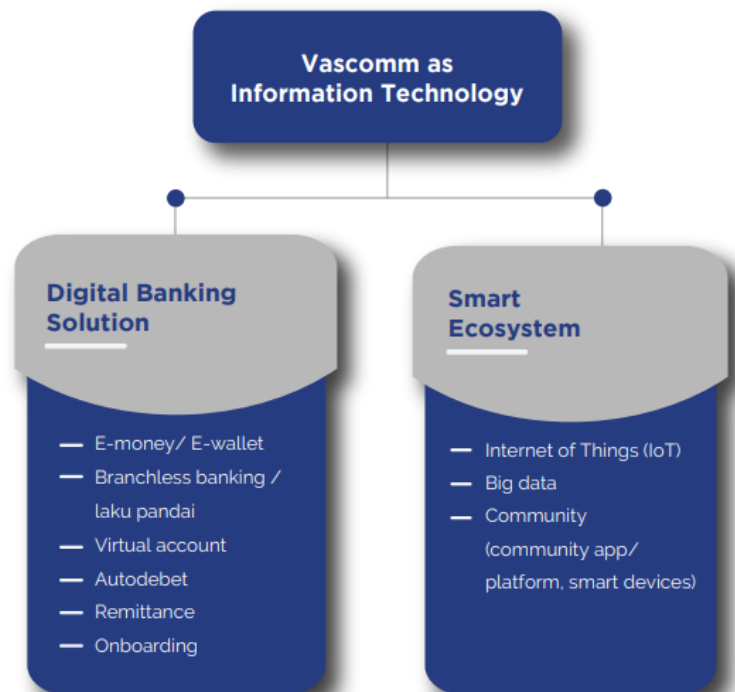
memiliki beberapa kantor cabang yaitu di Jakarta, Yogyakarta dan Malang seperti pada Gambar 2.1 Kantor Vascomm.



**Gambar 2.1 Kantor Vascomm**

### 2.2.2 Product dan Solusi Vascomm

Vascomm memiliki dua peran, yaitu sebagai Information Technology dan sebagai StartUp Factory. Sebagai Information Technology melakukan kegiatan untuk Digital Banking Solution dan Smart Ecosystem seperti pada Gambar 2.2.



**Gambar 2.2 Vascomm sebagai Information Technology**

Sebagai StartUp Factory melakukan kegiatan Inkubator, Kaselerator dan Venture Capital seperti pada Gambar 2.3



**Gambar 2.3 Vascomm sebagai Startup Factory**

### 2.3 Manajemen Proyek

Manajemen merupakan proses perencanaan, pengorganisasian, memimpin dan mengendalikan kegiatan anggota dan sumber daya lainnya agar tercapainya sasaran organisasi yang telah ditentukan (Soeharto, 2001 disitasi dari (Noerlina, 2008).

Secara garis besar, proyek memiliki empat tahapan berikut: 1) Tahap Konsepsi, pada tahap ini dilakukan penyusunan, perumusan gagasan, analisis pendahuluan dan melakukan studi kelayakan; 2) Tahap Pendefinisian, menyiapkan rencana proyek secara detail dan menentukan spesifikasi proyek secara rinci; 3) Tahap Akuisisi, melakukan desain, pengadaan fasilitas pendukung dan material untuk tahap selanjutnya, produksi serta implementasi dan 4) Tahap Operasi,

merupakan tahap akhir suatu proyek dimana proyek diserahkan kepada pengguna (Santosa, 2003 disitasi dari (Noerlina, 2008)).

Manajemen proyek merupakan kegiatan yang melibatkan perencanaan, pemantauan, pengendalian anggota proyek dan peristiwa yang terjadi dalam pengembangan perangkat lunak dari konsep operasional awal hingga akhir (Pressman, 2010).

Manajemen proyek merupakan kegiatan mengatasi kendala, mengembangkan perangkat lunak dengan kualitas tinggi dan memastikan proyek yang dijalankan telah sesuai dengan yang direncanakan. Selain itu kriteria keberhasilan dalam manajemen proyek memang bervariasi tergantung dari proyek yang dijalankan, namun untuk sebagian besar proyek, tujuan terpenting adalah: 1) Menyerahkan hasil proyek kepada pelanggan tepat pada waktunya; 2) Perangkat lunak yang dikembangkan memenuhi kebutuhan pelanggan; 3) Menjaga agar tim pengembang nyaman dan bekerja dengan baik dan 4) Hindari biaya berlebih dalam anggaran (Sommerville, 2011).

Manajemen proyek terdapat 10 bidang pengetahuan yaitu :

1. Manajemen ruang lingkup proyek (*Scope*), melakukan pendefinisian dan pengelolaan seluruh pekerjaan yang harus diselesaikan agar proyek berjalan dengan sukses.
2. Manajemen waktu proyek (*Time*), melakukan perkiraan, penjadwalan dan memastikan waktu yang diperlukan untuk menyelesaikan proyek.
3. Manajemen biaya proyek (*Cost*), melakukan persiapan dan pengelolaan anggaran untuk proyek.
4. Manajemen mutu proyek (*Quality*), memastikan proyek dilakukan sesuai dengan kebutuhan dan memastikan kebutuhan yang telah didefinisikan sudah terpenuhi.
5. Manajemen sumber daya manusia (*Human resource*), mengelola seluruh sumber daya manusia (*SDM*) yang terlibat dalam proyek secara efektif.
6. Manajemen komunikasi (*Communications*), mengumpulkan, menyebarkan dan menyimpan informasi proyek.

7. Manajemen resiko (Risk), identifikasi, analisis serta menanggapi resiko yang terkait dengan proyek.
8. Manajemen pengadaan (Procurement), yaitu kegiatan pengadaan barang dan jasa untuk proyek.
9. Manajemen pemangku kepentingan proyek (Stakeholder), identifikasi dan analisis mengenai keterlibatan stakeholder dalam proyek.
10. Manajemen integrasi proyek (Integration), memastikan seluruh fungsi apakah mempengaruhi atau dipengaruhi dengan bidang-bidang lainnya (Schwalbe, 2014).

### **2.3.1 Metodologi Proyek IT**

Metodologi menyediakan rencana tingkat strategis untuk mengelola dan mengendalikan proyek TI. Metodologi sebagai acuan untuk memulai, merencanakan, dan mengembangkan sistem informasi. Metodologi terdapat fase, hasil, proses, alat, dan bidang pengetahuan untuk mendukung proyek TI yaitu:

Fase 1: Mengkonseptualisasikan dan Menginisialisasi

Fase 2: Mengembangkan Piagam Proyek dan Rencana Proyek yang Rinci

Fase 3: Jalankan dan Kontrol Proyek

Fase 4: Tutup Proyek

Fase 5: Evaluasi Keberhasilan Proyek

### **2.4 *Software Development Life Cycle (SDLC)***

SDLC atau *Software Development Life Cycle* merupakan urutan kegiatan pengembangan sistem dari satu kegiatan ke kegiatan berikutnya. SDLC merupakan bagian dari PLC karena banyak dari kegiatan pengembangan sistem informasi terjadi di tahap Execute. *Software Development Life Cycle* memiliki lima fase yaitu Planning, Analysis, Design, Implementation dan Maintenance.

1. **Planning**, mengidentifikasi dan menanggapi masalah atau kesempatan dan menggabungkan proses project management dengan kegiatan system *Development*. Perencanaan memastikan bahwa tujuan, lingkup, anggaran,

jadwal, teknologi, *system Development*, metode dan tools yang dibutuhkan berada di tempat.

2. **Analysis**, mencoba menyelidiki masalah atau kesempatan secara lebih detail. Umumnya, sistem analis akan bertemu dengan berbagai stakeholders (pengguna, manager, pelanggan) untuk mempelajari lebih lanjut tentang masalah atau kesempatan.
3. **Design**, tim proyek menggunakan requirements sebagai masukan untuk merancang arsitektur yang mendukung sistem. Arsitektur ini meliputi merancang jaringan, konfigurasi perangkat keras, database, antarmuka pengguna dan program perangkat lunak.
4. **Implementation**, mencakup pengembangan dan pembangunan sistem, pengujian serta instalasi. Selain itu juga memastikan dokumentasi harus ada di tempat.
5. **Maintenance and Support**, untuk tahap ini mungkin tidak termasuk tahapan proyek saat ini, sehingga masih menjadi pertimbangan penting. Setelah pengerjaan perubahan sistem selesai dilakukan, bentuk pemeliharaan dan peningkatan terhadap sistem sering diminta untuk memperbaiki kesalahan yang ditemukan dalam sistem (Marchewka, 2003).

## 2.5 Project Cost Management

Dalam manajemen biaya proyek, memperkirakan biaya atau tugas tertentu dengan melibatkan perkiraan durasi melibatkan lima langkah yaitu:

1. Mendefinisikan sumber daya yang dibutuhkan untuk melakukan pekerjaan
2. Menentukan jumlah sumber daya yang dibutuhkan
3. Mendefinisikan biaya penggunaan masing-masing sumber
4. Menghitung biaya tugas atau aktivitas
5. Memastikan bahwa sumber daya setara (Marchewka, 2003).

### 2.5.1 Software Cost Estimation

Estimasi biaya perangkat lunak memperkirakan estimasi usaha (biasanya secara person-month), durasi proyek (dalam waktu kalender) dan biaya (bisa dalam satuan dollar, rupiah atau mata uang lain). Sebagian besar metode estimasi biaya

menghasilkan perkiraan usaha yang kemudian dapat dikonversi menjadi durasi proyek dan biaya. Konversi menjadi biaya bisa dilakukan dengan menghitung rata-rata gaji per staff yang terlibat dalam proyek lalu mengalikannya dengan perkiraan usaha yang dibutuhkan.

Proses estimasi biaya melibatkan tujuh langkah yaitu:

1. Mendefinisikan tujuan estimasi biaya
2. Menentukan kebutuhan data dan sumber daya;
3. Mendefinisikan kebutuhan perangkat lunak
4. Mendetailkan sebanyak mungkin kriteria perangkat lunak yang layak.
5. Gunakan beberapa teknik estimasi biaya untuk memanfaatkan kombinasi.
6. Bandingkan perkiraan yang berbeda.
7. Setelah proyek dimulai, pantau biaya aktual serta kemajuannya.

## **2.6 Agile Framework**

Dalam ekonomi modern, kondisi pasar berubah dengan cepat, kebutuhan pelanggan dan pengguna akhir berkembang, dan ancaman kompetitif baru muncul tanpa peringatan. Praktisi harus mendekati rekayasa perangkat lunak dengan cara yang memungkinkan mereka tetap gesit untuk mendefinisikan proses lean, bermanuver, yang dapat mengakomodasi kebutuhan bisnis modern.

Filosofi *Agile* untuk rekayasa perangkat lunak menekankan empat masalah utama:

- Pentingnya mengatur diri sendiri tim yang memiliki kontrol atas pekerjaan yang lakukan.
- Komunikasi dan kolaborasi antar anggota tim, antar praktisi dan pelanggan.
- Pengakuan bahwa perubahan mewakili peluang.
- Penekanan pada penyerahan perangkat lunak yang cepat dan memuaskan pelanggan.

### **2.6.1 Scrum Method**

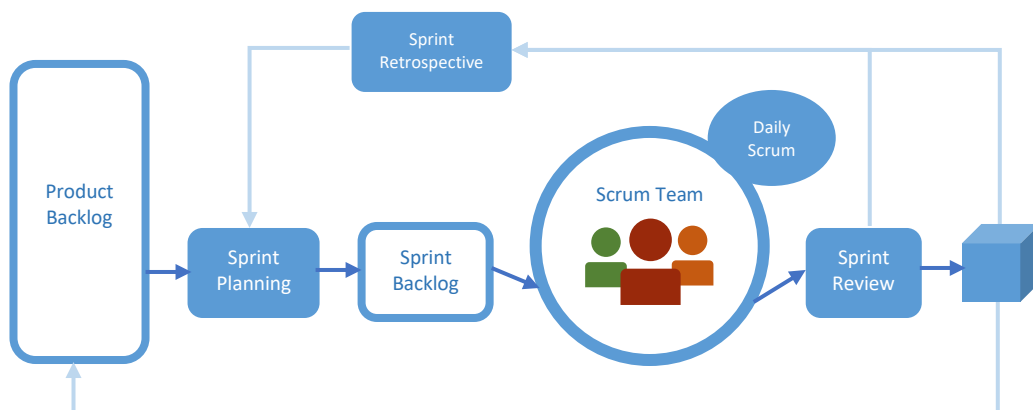
Scrum adalah kerangka kerja dengan proses yang sangat sederhana untuk mengelola proyek yang rumit juga dalam Pengembangan Perangkat Lunak *Agile*. Beberapa metode estimasi upaya pada kerangka Scrum termasuk Story Point.

Scrum adalah kerangka kerja di mana orang dapat mengatasi masalah adaptif yang kompleks, sementara secara produktif dan kreatif memberikan produk dengan nilai setinggi mungkin. Scrum sendiri adalah kerangka kerja sederhana untuk kolaborasi tim yang efektif pada produk yang kompleks. Rekan pencipta Scrum, Ken Schwaber dan Jeff Sutherland telah menulis *The Scrum Guide* untuk menjelaskan Scrum dengan jelas dan ringkas. Panduan ini berisi definisi Scrum. Definisi ini terdiri dari peran, peristiwa, artefak, dan aturan Scrum yang mengikatnya. Scrum merupakan metode yang:

- Ringan
- Mudah dimengerti
- Sulit dikuasai

### Kerangka kerja Scrum

Scrum itu sederhana. Scrum mengimplementasikan metode ilmiah secara empirisme. Scrum menggantikan pendekatan algoritmik yang diprogramkan dengan pendekatan heuristik, dengan mempertimbangkan orang-orang dan organisasi mandiri untuk menghadapi ketidakpastian dan memecahkan masalah yang kompleks. Gambar 2.4 di bawah ini mewakili Scrum in Action seperti yang dijelaskan oleh Ken Schwaber dan Jeff Sutherland dalam buku mereka *Software in 30 Days* (Scrum.org., 2019).



**Gambar 2.4 Kerangka Kerja Scrum**

## 2.7 Use Case Point

Metode *Use Case Point* pertama kali dipopulerkan oleh Gustav Karner pada tahun 1993 yang merupakan turunan dari metode Function Point Analysis (FPA) yang bertujuan untuk menyediakan metode estimasi sederhana dengan berorientasi pada objek proyek perangkat lunak (Karner, 1993 disitasi dari (Primandari & Sholih, 2016)). Dalam metode *Use Case Point*, terdapat beberapa komponen yang diperlukan. Berikut penjelasan komponen-komponen yang diperlukan:

### 2.7.1 *Unadjusted Actor Weights (UAW)*

Jenis aktor diklasifikasikan menjadi tiga kategori yaitu Simple, Average dan Complex (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018). Bobot tiap tipe aktor telah dijelaskan pada Tabel 2.2.

- Simple actor = merupakan aktor yang berinteraksi melalui API seperti Command Prompt.
- Average actor = aktor yang berinteraksi melalui protokol seperti TCP/IP, FTP, HTTP atau disebut aktor yang melakukan penyimpanan data (file, RDBMS).
- Complex Actor = aktor yang berinteraksi melalui GUI atau halaman web (Zahiroh, Saputra, & Herlambang, 2018).

**Tabel 2.2 Tipe dan Bobot Aktor**

| <b>Tipe Aktor</b> | <b>Bobot</b> |
|-------------------|--------------|
| Simple            | 1            |
| Average           | 2            |
| Complex           | 3            |

Sumber : (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018)

### 2.7.2 *Unadjusted Use Case Weights (UUCW)*

Setiap Use Case diklasifikasikan ke dalam salah satu dari tiga jenis yaitu *Simple*, *Average* dan *Complex*. Simple Use Case mengandung maksimum 3 transaksi. Average Use Case berisi antara 4 sampai 7 transaksi dan Complex Use Case mengandung lebih dari 7 transaksi. Berikut daftar tipe aktor dengan bobotnya pada Tabel 2.3 :



**Tabel 2.3 Tipe dan Bobot Use Case pada *Use Case Point***

| <b>Type Use Case</b> | <b>Deskripsi</b>       | <b>Bobot</b> |
|----------------------|------------------------|--------------|
| Simple               | <= 3 transaksi         | 1            |
| Average              | 4 sampai 7 transaksi   | 2            |
| Complex              | Lebih dari 7 transaksi | 3            |

Sumber : (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018)

Penelitian ini menggunakan aturan penentuan transaksi berdasarkan actor goals atau disebut semantic point. Jika melihat transaksi dari semantic point, sebagian besar use case memiliki goals atau tujuan yang sama. Semantic point telah membedakan 12 jenis transaksi yaitu *Create, Retrieve, Update, Delete, Link, Delete Link, Asynchronous Retrieve, Dynamic Retrieve, Dynamic Retrieve, Transfer, Check Object, Complex Internal Activity* dan *Change State* (Ochodek et al., 2011).

### 2.7.3 Unadjusted Use Case Point (UUCP)

Unadjusted Use Case Point (UUCP) merupakan penjumlahan *Unadjusted Actor Weights* (UAW) dengan Undjusted Use Case Weights (UUCW) seperti pada rumus 2.1 berikut :

$$UUCP = UAW + UUCW \quad (2.1)$$

### 2.7.4 Technical Complexity Factors (TCF)

Berikut merupakan daftar Technical Factors untuk perhitungan Technical Complexity Factors beserta bobotnya:

**Tabel 2.4 Technical Factors**

| <b>Technical Factor</b> |                                      | <b>Bobot</b> |
|-------------------------|--------------------------------------|--------------|
| T1                      | Distributed System Required          | 2            |
| T2                      | Response Time                        | 2            |
| T3                      | End User Efficiency                  | 1            |
| T4                      | Complex Internal Processing Required | 1            |
| T5                      | Reusable Code                        | 1            |
| T6                      | Easy to Install                      | 0,5          |
| T7                      | Easy to Use                          | 0,5          |
| T8                      | Portable                             | 2            |
| T9                      | Easy to Change                       | 1            |
| T10                     | Concurrent                           | 1            |
| T11                     | Security Features                    | 1            |
| T12                     | Access for Third Parties             | 1            |
| T13                     | Special Training Required            | 1            |

Sumber : (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018)

Technical Complexity Factors (TCF) seperti yang tertera pada Tabel 2.4 berisi 13 faktor yang memiliki pengaruh besar pada kinerja proyek. Setiap nilai faktor merupakan pengaruh terhadap produktivitas perangkat lunak yang memiliki peringkat antara 0 sampai 5 (0 berarti tidak ada pengaruh dan 5 berarti pengaruh yang kuat). TCF dihitung dengan mengalikan nilai setiap bobot faktor (T1-T13) kemudian menambahkan semua angka-angka untuk mendapatkan jumlah yang disebut TF lalu menerapkan rumus :

$$TCF = 0,6 + (0,01 * TF) \quad (2.2)$$

### 2.7.5 Environment Complexity Factor (ECF)

Berikut merupakan daftar Technical Factors untuk perhitungan Environment Complexity Factors :

**Tabel 2.5 Environment Complexity**

| Environment Factor |                                | Bobot |
|--------------------|--------------------------------|-------|
| F1                 | Familiarity With The Project   | 1,5   |
| F2                 | Application Experience         | 0,5   |
| F3                 | OO Programming Experience      | 1     |
| F4                 | Lead Analyst Capability        | 0,5   |
| F5                 | Motivation                     | 1     |
| F6                 | Stable Requirements            | 2     |
| F7                 | Part Time Staff                | -1    |
| F8                 | Difficult Programming Language | -1    |

Sumber : (Yuliansyah, Qudsiah, Zahrotun, & Arfiani, 2018)

Environment Factor (EF) dihitung dengan mengalikan nilai faktor (F1-F8) seperti pada Tabel 2.5 dengan bobot nilai masing-masing kemudian dijumlahkan untuk mendapatkan jumlah yang disebut EF seperti yang ditunjukkan pada rumus:

$$ECF = 1.4 + (-0,03 * EF) \quad (2.3)$$

### 2.7.6 Estimasi Effort / Usaha

Pada langkah terakhir didapatkan hasil *Use Case Point* dengan mengalikan *Unadjusted Use Case Point* dengan *Technical Complexity Factor*.

$$UCP = UUCP * TCF * ECF \quad (2.4)$$

Untuk mengubah nilai UCP menjadi nilai usaha yaitu *Hours of Effort* maka diperlukan pengembangan perangkat lunak sesuai dengan metrik *Use Case Point*.

Dengan asumsi  $p$  adalah staff hours per use case maka rumus Hours of *Effort* adalah sebagai berikut : (Zahiroh, Saputra, & Herlambang, 2018).

$$\text{Hours of Effort} = UCP * p \quad (2.5)$$

## 2.8 *Function Point*

Konsep dari Function Point pertama kali diperkenalkan oleh Allan J. Albercht di tahun 1979 yang kemudian dilakukan perbaikan di tahun 1984. Pada tahun 1986 sejak berdirinya Internasional Function Point Counting Practices (IFPUG), beberapa versi dari Function Point Counting Practices Manual diterbitkan. Function Point adalah metrik karena memberikan ukuran pada produk di awal siklus pengembangan. Function Point mewakili upaya untuk mengembangkan fitur yang diinginkan (Kang, Choi, & Baik, 2010).

Metode Function Point merupakan metode yang berorientasi pada ukuran fungsi perangkat lunak untuk menganalisis estimasi biaya serta usaha yang dibutuhkan dalam pengembangan perangkat lunak (Zahiroh, Saputra, & Herlambang, 2018).

### 2.8.1 *Unjusted Function Point Complexity*

Mengklasifikasikan tipe fungsi pengguna berdasarkan karakteristik kompleksitas yang dimiliki. Elemen tipe fungsi pengguna ada lima tipe yaitu:

1. External Input (EI) merupakan proses dasar yang memproses data atau mengendalikan informasi yang berasal dari luar batas aplikasi. Tujuan utama EI adalah mempertahankan satu atau lebih ILF dan/atau mengubah perilaku sistem.
2. External Outputs (EO) merupakan sebuah proses yang melakukan pengiriman data atau informasi yang ditujukan kepada user. Tujuan utama EO ialah menyajikan informasi kepada pengguna melalui pemrosesan secara logika. Logika pemrosesan harus mengandung setidaknya satu rumus atau perhitungan matematis, membuat data turunan, mempertahankan satu atau lebih ILF atau mengubah perilaku sistem.
3. External Inquiry (EQ) merupakan sebuah proses yang melakukan pengiriman data atau informasi diluar batas sistem seperti menampilkan data dilayar user.

Tujuan utama EQ adalah menyajikan informasi kepada pengguna melalui pengambilan informasi data atau kontrol dari ILF dan EIF. Logika pemrosesan tidak berisi rumus atau perhitungan matematis dan tidak menghasilkan data turunan, serta tidak ada ILF yang dipertahankan selama pemrosesan dan tidak mengubah perilaku sistem.

4. Internal Logical File (ILF) merupakan sekelompok data informasi pengguna yang diidentifikasi dan tersimpan dalam batas aplikasi. Tujuan utama ILF adalah mempertahankan data yang dipelihara melalui satu atau lebih proses dasar pada aplikasi.
5. External Interface Files (EIF) merupakan sekumpulan data atau informasi yang dapat dikenali dan direferensikan oleh aplikasi, namun dipelihara dalam batas aplikasi lain. Tujuan utama EIF adalah menyimpan data yang direferensikan melalui satu atau lebih proses dasar di dalam batas aplikasi. Artinya, sebuah EIF yang dihitung untuk sebuah aplikasi harus berada dalam ILF aplikasi lain (Zahiroh, Saputra, & Herlambang, 2018).

### **2.8.2 *Unjusted Weight Complexity***

Penentuan bobot kompleksitas Function Point yang terdiri dari:

1. Data Element Type (DET) yaitu merupakan jenis elemen data yang bersifat unik yang dapat dikenali dan tidak berulang.
2. Record Element Type (RET) yaitu sub kelompok elemen data pengguna yang dikenali dalam ILF atau EIF.
3. File Type Reference (FTR) yaitu berkas logical internal yang dibaca atau dikelola oleh fungsi transaksional atau file antarmuka eksternal yang dibaca oleh fungsi transaksional.

Kemudian untuk mempermudah dalam mengidentifikasi FTR maka dilakukan dengan cara mengidentifikasi ILF dan EIF dikarenakan memiliki jumlah yang sama. Bobot kompleksitas tiap elemen fungsi disajikan pada Tabel 2.6, Tabel 2.7 dan Tabel 2.8 :

**Tabel 2.6 Bobot Kompleksitas External Input**

| FTR | DET     |         |         |
|-----|---------|---------|---------|
|     | 1-4     | 5-15    | 16 >    |
| 0-1 | Low     | Low     | Average |
| 2   | Low     | Average | High    |
| =3  | Average | High    | High    |

**Tabel 2.7 Bobot Kompleksitas External Output dan External Inquiries**

| FTR | DET     |         |         |
|-----|---------|---------|---------|
|     | 1-5     | 6-19    | 20 >    |
| 0-1 | Low     | Low     | Average |
| 2-3 | Low     | Average | High    |
| 4 > | Average | High    | High    |

**Tabel 2.8 Bobot Kompleksitas Logical Internal dan External Interface File**

| RET | DET     |         |         |
|-----|---------|---------|---------|
|     | 1-19    | 20-50   | 51 >    |
| 1   | Low     | Low     | Average |
| 2-5 | Low     | Average | High    |
| >=6 | Average | High    | High    |

Sumber : (Marchewka, 2003)

### 2.8.3 *Unadjusted Function Point*

Unadjusted Function Point, nilai dari setiap pembobotan kompleksitas pada elemen Function Point. Bobot kompleksitas berdasarkan kriteria dari setiap kategori ditunjukkan pada Tabel 2.9 :

**Tabel 2.9 Perhitungan Unadjusted Function Point**

| Elemen FP                       | Low     | Average  | High     | Total |
|---------------------------------|---------|----------|----------|-------|
| External Input                  | 3 x ... | 4 x ...  | 6 x ...  |       |
| External Output                 | 4 x ... | 5 x ...  | 7 x ...  |       |
| External Inquiries              | 3 x ... | 4 x ...  | 6 x ...  |       |
| Internal Logical File           | 7 x ... | 10 x ... | 15 x ... |       |
| External Interface Files        | 5 x ... | 7 x ...  | 10 x ... |       |
| Unadjusted Function Point (UFP) |         |          |          |       |

Sumber : (Marchewka, 2003)

### 2.8.4 *Technical Complexity Factor*

Identifikasi karakteristik perangkat lunak atau Technical Complexity Factor (TCF) memiliki nilai pada tiap faktor dengan bobot antara 0 sampai 5. Tabel 2.10

menjelaskan influence pada score yang diberikan, dan Tabel 2.11 merupakan daftar subject dan grade score yang bisa dipilih.

**Tabel 2.10 Bobot Technical Complexity Factor**

| Score | System Influence      |
|-------|-----------------------|
| 0     | No Influence          |
| 1     | Incidental Influence  |
| 2     | Moderate Influence    |
| 3     | Average Influence     |
| 4     | Significant Influence |
| 5     | Strong Influence      |

**Tabel 2.11 Technical Factors**

| No. | Subject                     | Grade |   |   |   |   |
|-----|-----------------------------|-------|---|---|---|---|
|     |                             | 1     | 2 | 3 | 4 | 5 |
| 1   | Data Communications         | 1     | 2 | 3 | 4 | 5 |
| 2   | Distributed Data Processing | 1     | 2 | 3 | 4 | 5 |
| 3   | Performance                 | 1     | 2 | 3 | 4 | 5 |
| 4   | Heavily Used Configuration  | 1     | 2 | 3 | 4 | 5 |
| 5   | Transaction Rate            | 1     | 2 | 3 | 4 | 5 |
| 6   | Online Data Entry           | 1     | 2 | 3 | 4 | 5 |
| 7   | End-User Efficiency         | 1     | 2 | 3 | 4 | 5 |
| 8   | Online Update               | 1     | 2 | 3 | 4 | 5 |
| 9   | Complex Processing          | 1     | 2 | 3 | 4 | 5 |
| 10  | Reusability                 | 1     | 2 | 3 | 4 | 5 |
| 11  | Installation Ease           | 1     | 2 | 3 | 4 | 5 |
| 12  | Operation Ease              | 1     | 2 | 3 | 4 | 5 |
| 13  | Multiple Sites              | 1     | 2 | 3 | 4 | 5 |
| 14  | Facilitate Change           | 1     | 2 | 3 | 4 | 5 |
|     | Total                       |       |   |   |   |   |

Sumber : (Zahiroh, Saputra, & Herlambang, 2018)

$$TCF = 0.65 + 0.01 \times RCAF \quad 2.6$$

### 2.8.5 Function Point

Function Point sebuah perangkat lunak yang akan dikembangkan yaitu dengan cara memasukkan nilai Unadjusted Function Point dan Technical Complexity Factor pada rumus berikut :

$$FP = UFP * TCF \quad 2.7$$

### 2.8.6 Estimasi Effort / Usaha

Menghitung usaha yang diperlukan untuk pengembangan perangkat lunak berdasarkan person-month. Perhitungan ini menggunakan fungsi estimasi

eksponensial oleh Jones ada pada persamaan 2.8 dengan  $f$  merupakan nilai Function Point,  $j$  merupakan nilai Jone's First Order seperti tertera pada

Tabel 2.12.

$$m = f^{3*j}/27 \quad 2.8$$

**Tabel 2.12 Jone's First Order Estimate Exponent**

| Kind of Software | Organization's Skill/Abilities |         |                |
|------------------|--------------------------------|---------|----------------|
|                  | Best In Class                  | Average | Worst In Class |
| System           | 0.43                           | 0.45    | 0.48           |
| Business         | 0.41                           | 0.43    | 0.46           |
| Shrink-wrap      | 0.39                           | 0.42    | 0.45           |

Sumber : (Zahiroh, Saputra, & Herlambang, 2018)

Jenis-jenis perangkat lunak yaitu: (1) System, yaitu operating systems, device drivers, compiler, embedded software, firmware, real-time system, dan lain-lain; (2) Bussines, in house system for an organization, payroll, stock control, accounting system, dan lain-lain; (3) Shrink-wrap, packed and sold commercially.

Kemampuan atau keterampilan organisasi yaitu: (1) Best in Class, yaitu perusahaan atau orang-orang yang memiliki kemampuan diatas rata-rata dengan pengalaman yang cukup banyak dalam pengembangan perangkat lunak; (2) Average, yaitu perusahaan atau orang-orang yang memiliki kemampuan rata-rata dalam menangani pengembangan perangkat lunak; (3) Worst in class, yaitu perusahaan atau orang-orang yang memiliki kemampuan tidak cukup baik dan tidak memiliki pengalaman dalam menangani pengembangan perangkat lunak. Menghitung lama durasi proyek menggunakan Persamaan 2.9 dengan  $s$  merupakan schedule months for optimal schedule,  $f$  merupakan nilai effort dan  $j$  merupakan nilai Jone's First Order.

$$s = f^j \quad 2.9$$

### 2.8.7 Estimasi Biaya

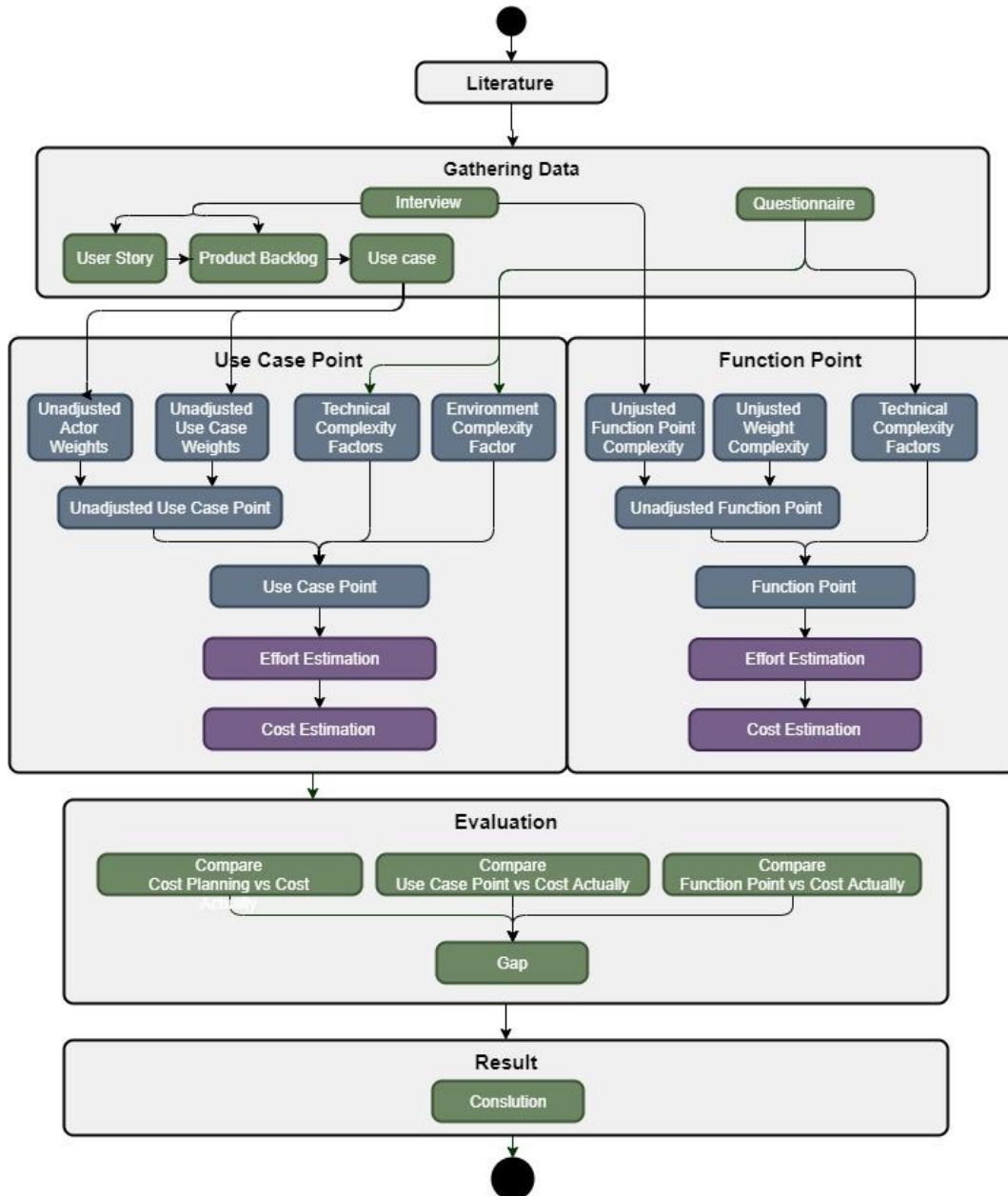
*Effort* / usaha yang telah didapatkan pada tahap sebelumnya dibagi menjadi tiga aktivitas yang dijalankan dalam pengembangan perangkat lunak yaitu *Software Development*, *Ongoing Activity* dan *Quality dan Testing*.

1. *Software Development*, meliputi analisis kebutuhan pengguna seperti permintaan dan spesifikasi, desain, implementasi, pengujian integrasi yang mencapai sekitar 42,5% dari total usaha.
2. *Ongoing Activity*, atau disebut aktivitas yang berkesinambungan seperti manajemen proyek, manajemen konfigurasi, dokumentasi, penerimaan dan penyebaran yang mencapai sekitar 20,82% dari total usaha.
3. *Quality dan Testing*, aktivitas yang berhubungan dengan kualitas dan pengujian seperti pengujian integrasi, penjaminan kualitas, evaluasi dan pengujian yang mencapai sekitar 36,68% dari total usaha (Primandari & Sholih, 2016; Saleh, 2011).



## BAB 3 METODOLOGI PENELITIAN

Bab ini menguraikan tahap yang akan dilakukan pada penelitian. Pada Gambar 3.1 menjelaskan metodologi penelitian secara umum terdiri dari beberapa tahap.



**Gambar 3.1 Alur Penelitian**

Penjelasan tahapan metode penelitian diatas akan diterangkan secara terperinci pada sub bab berikut.

### **3.1 Studi Literatur**

Suatu penelitian tentu saja selalu diawali dengan studi literatur sebagai dasar proses pengkajian yang berkaitan dengan topik penelitian yang dilakukan. Pada penelitian ini, referensi yang digunakan oleh penulis adalah berbagai macam literatur yang berhubungan dengan Manajemen Proyek IT, *Use Case Point*, Agile-Scrum, serta estimasi usaha dan biaya proyek pengembangan perangkat lunak.

Dari literatur–literatur di atas diharapkan dapat memberikan gambaran yang cukup lengkap dan memberikan kontribusi kerangka kerja dari penelitian ini dalam melakukan evaluasi proses bisnis berdasar indikator keberlanjutan perusahaan. Adapun literatur – literatur tersebut dapat pada jurnal melalui portal Sciendirect, IEEE, portal perusahaan ritel serta ebook.

### **3.2 Pengumpulan data**

Tahapan dari observasi penelitian ini adalah melakukan penelusuran dan peninjauan ke PT. Vascomm Solusi Teknologibeserta proyek yang akan dijadikan studi kasus. Pada penelitian ini, data yang digunakan yaitu data kualitatif dan kuantitatif. Data kualitatif merupakan data yang bersifat deskriptif atau data yang dikategorikan seperti diagram, catatan-catatan lapangan atau wawancara ketika penelitian dilakukan. Sedangkan data kuantitatif merupakan data yang bersifat kuantitatif atau angka-angka statistik yang dapat berbentuk variabel-variabel dan operasionalisasinya menggunakan skala ukuran tertentu (Sarwono, 2006).

Data kuantitatif yang didapat yaitu lembar penilaian Technical dan Environment Complexity Factor. Sedangkan data kualitatif yang didapat merupakan hasil wawancara yang dibutuhkan untuk proses pembuatan User Story dan Use Case Diagram. Data yang dibutuhkan diperoleh langsung dari PT. Vascomm Solusi Teknologi.

Selain itu tujuan pengumpulan data ini juga untuk mengetahui informasi mengenai manajemen proyek yang ada di PT. Vascomm Solusi Teknologi,

informasi tentang proyek yang diselesaikan dan alokasi SDM serta timeline yang dibutuhkan dalam proyek.

### **3.2.1 Wawancara**

Penelitian ini membutuhkan pengumpulan data melalui wawancara, yaitu membutuhkan ketersediaan dari Project Manager untuk dimintai keterangan mengenai proyek yang akan diselesaikan. Wawancara ini berisi beberapa pertanyaan mengenai masalah yang dihadapi ketika pengerjaan proyek, sistem yang akan dikembangkan seperti analisis kebutuhan fungsionalitas sistem yang hasilnya akan diubah ke bentuk User Story dan Use Case karena penelitian ini membutuhkan Use Case Diagram untuk menentukan bobot dari *Unadjusted Use Case Point*. Selain itu dalam wawancara ini menanyakan beberapa hal mengenai proyek tersebut seperti cara penentuan nilai proyek, alokasi SDM dan waktu guna memahami perencanaan proyek yang ada pada PT. Vascomm Solusi Teknologi.

### **3.2.2 Lembar Penilaian**

Penelitian ini juga menggunakan lembar penilaian untuk mengambil data di PT. Vascomm Solusi Teknologi yang terlampir pada Lampiran. Mekanisme penyebaran lembar penilaian untuk metode *Use Case Point* menggunakan tiga tahapan.

Penyebaran lembar penilaian. Lembar penilaian disebarkan kepada responden yang dalam penelitian ini adalah tim pengembang dari PT. Vascomm Solusi Teknologi yang akan mengembangkan proyek. Adapun target responden kuisisioner adalah tim terkait proyek yang terpilih, terdiri dari:

1. Project Manager / Team Leader
2. Product Owner / Business Analyst
3. BackEnd Developer
4. FrontEnd Developer
5. System Analyst
6. Quality Assurance

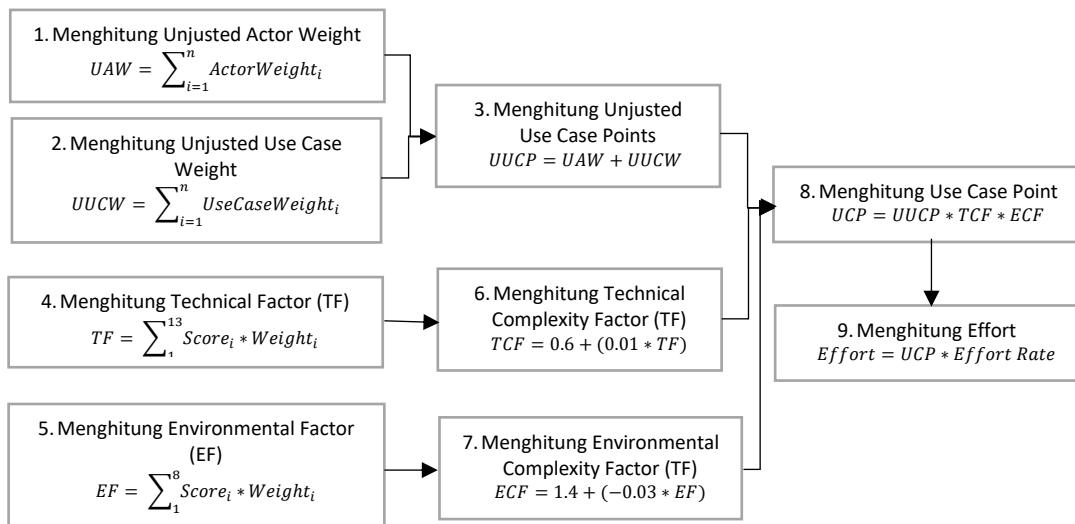
Lembar penilaian disebar untuk mendapatkan informasi terkait perhitungan nilai TCF dan ECF pada *Use Case Point*.

Tahap terakhir yaitu mendapatkan skor pada setiap lembar penilaian untuk metode *Use Case Point*. Pada metode *Use Case Point*, skor di setiap indikator akan dikalikan dengan bobot yang hasil akhirnya berupa nilai TF untuk lembar penilaian Technical Complexity Factor dan nilai EF untuk lembar penilaian Environment Complexity Factor.

### 3.3 Analisa Hasil

#### 3.3.1 Use Case Point

Analisis hasil penelitian ini dilakukan berdasarkan penjadwalan pengerjaan proyek. Kemudian menggunakan metode perhitungan *Use Case Point* dengan langkah-langkah seperti pada Gambar 3.2 berikut :



**Gambar 3.2 Langkah Use Case Point**

Sumber : (Primandari & Sholih, 2016)

#### 1. *Unadjusted Actor Weights (UAW)*

Langkah pertama adalah mengidentifikasi dan mengklasifikasikan jenis aktor. Jenis aktor diklasifikasikan menjadi tiga kategori yaitu Simple, Average dan Complex dan memiliki bobot yang telah dijelaskan pada tabel 2.2. *Unadjusted Actor Weights (UAW)* diperoleh dari berapa banyak aktor dari masing-masing tipe aktor kemudian dikali dengan total bobot faktor masing-masing.

#### 2. *Unadjusted Use Case Weights (UUCW)*

*Unadjusted Use Case Weights* (UUCW) dihitung dengan mengalikan jumlah bobot seperti pada tabel 2.3, dengan masing-masing Use Case lalu dijumlahkan.

### **3. Unadjusted Use Case Point (UUCP)**

*Unadjusted Use Case Point* (UUCP) merupakan penjumlahan *Unadjusted Actor Weights* (UAW) dengan *Unadjusted Use Case Weights* (UUCW) seperti pada persamaan 2.1

### **4. Technical Complexity Factors (TCF) dan Environment Complexity Factor (ECF)**

Langkah berikutnya adalah menghitung data *Technical Complexity Factors* (TCF) dan *Environment Complexity Factor* (ECF) yang didapatkan dari kuisioner kepada tim proyek pengembang perangkat lunak sebelumnya. Perhitungan TCF berdasarkan persamaan 2.2 dengan daftar bobot pada tabel 2.4. sedangkan Perhitungan ECF berdasarkan persamaan 2.3 dengan daftar bobot pada tabel 2.5.

### **5. Use Case Point**

Muara dari perhitungan metode *Use Case Point* adalah titik ini, yaitu perhitungan *Use Case Point*. *Use Case Point* sendiri didapatkan dari persamaan 2.4 pada Bab 2.

### **6. Estimasi Usaha**

Untuk mendapatkan nilai usaha dari nilai *Use Case Point* adalah dengan perhitungan persamaan 2.5. Kemudian jumlah waktu yang didapatkan akan dibagi menjadi jumlah sprint dan jam kerja yang ada di Vascomm.

### **7. Estimasi Biaya**

Setelah itu estimasi biaya yang didapatkan merupakan hasil estimasi usaha yang dihitung dengan berdasar aktivitas pengembangan perangkat lunak berdasar waktu, personil dan standar biaya yang dibutuhkan.

## **3.3.2 Function Point**

### **1. Unjusted Function Point Complexity**

Langkah pertama adalah membagi tipe fungsi pengguna berdasarkan karakteristik kompleksitas yang dimiliki. Elemen tipe fungsi pengguna ada

lima tipe yaitu EI, EO, EQ, ILF dan EIF. Elemen tipe tersebut dihitung dan dijumlahkan setiap tipenya.

## **2. *Unjusted Weight Complexity***

Menentukan bobot kompleksitas yang terdiri dari Data Element Type (DET), Record Element Type (RET) dan File Type Reference (FTR) sesuai pada Tabel 2.6, Tabel 2.7 dan Tabel 2.8. Kemudian hasilnya dijumlahkan.

## **3. *Unadjusted Function Point***

Selanjutnya adalah menghitung Unadjusted Function Point, dengan cara nilai tiap pembobotan kompleksitas di elemen Function Point yang diperoleh dikalikan dengan masing-masing bobot kompleksitas. Bobot kompleksitas berdasarkan kriteria dari setiap kategori ditunjukkan pada Tabel 2.9

## **4. *Technical Complexity Factor***

Langkah selanjutnya, melakukan identifikasi karakteristik perangkat lunak atau Technical Complexity Factor (TCF). Kemudian untuk menentukan nilai TCF, yang perlu dilakukan adalah menentukan faktor teknis dengan cara memberikan nilai pada tiap faktor dengan bobot antara 0 sampai 5 sesuai dengan Tabel 2.10 dan Tabel 2.11. kemudian dihitung sesuai rumus 2.6

## **5. *Function Point***

Setelah memperoleh nilai Unadjusted Function Point dan nilai Technical Complexity Factor maka langkah selanjutnya menentukan nilai dari Function Point seperti pada rumus 2.7.

## **6. *Estimasi Effort / Usaha***

**Setelah memperoleh nilai Function Point, maka langkah selanjutnya menghitung usaha yang diperlukan untuk pengembangan perangkat lunak berdasarkan person-month. Perhitungan ini menggunakan fungsi estimasi eksponensial oleh Jones ada pada persamaan 2.8. Jenis-jenis perangkat lunak dan nilai Jone's First Order ada pada**

Tabel 2.12

## **7. *Estimasi Biaya***

Effort / usaha yang telah didapatkan pada tahap sebelumnya dibagi menjadi tiga aktivitas yang dijalankan dalam pengembangan perangkat lunak yaitu Software Development.

### **3.4 Pembahasan dan Evaluasi**

Evaluasi dengan cara membandingkan hasil cost / biaya pengembangan perangkat lunak versi realita dengan versi ekspekasi, hasil perhitungan UCP di Scrum dan Function Point. Dari hasil ini akan dapat didapatkan gap atau deviasi nilai cost pengembangan perangkat lunak yang paling mendekati realita.

### **3.5 Kesimpulan dan Saran**

Setelah dilakukannya evaluasi, maka dapat diambil kesimpulan apakah UCP dapat dijadikan solusi untuk estimasi effort dan cost dari pengembangan pernakgkat lunak tersebut. Dari kesimpulan yang terlampir dapat dijadikan bahan pertimbangan bagi PT. Vascomm Solusi Teknologi dalam mengestimasi biaya.





## BAB 4

### ANALISA DAN PERHITUNGAN

#### 4.1 Pembahasan Aplikasi

Berikut akan dibahas tentang dua aplikasi yang menjadi bahan perhitungan, dimana akan dijelaskan tentang nama, jenis, status, lama pengerjaan, biaya pengembangan dan *user story* masing-masing aplikasi

##### 4.1.1 Aplikasi 1 : CEKLOK.ID

CEKLOK adalah *Human Capital Informaiton System* untuk mengorganisir tata kelola dan tata laksana manajemen Sumber Daya Manusia di perusahaan. Sebagai alat untuk mendukung pengambilan keputusan dari divisi HR / HCM. Berikut rangkuman kondisi pengerjaan Aplikasi CEKLOK pada Tabel 4.1.

**Tabel 4.1 Rangkuman Aplikasi CEKLOK**

| Status                          | Result    |                | Estimasi  |                |
|---------------------------------|-----------|----------------|-----------|----------------|
| On-Going<br>(soft launch Maret) | 36 sprint | Rp 320.869.565 | 30 sprint | Rp 267.391.304 |

Kebutuhan dari awal recruitment, karyawan masuk perusahaan sampai karyawan resign pun dapat dikelola oleh CEKLOK. Feature assessment juga menarik karena adanya perhitungan penilaian karyawan menggunakan algoritma yang didasari oleh tes psikologi. Dilakukan secara online melalui website dan mobile app sehingga data terupdate real-time, otomatisasi perhitungan seperti jam kerja, cuti, ijin, penggajian, kebutuhan perjalanan dinas dsb dilakukan oleh sistem.

Karyawan dapat absen, monitoring daily information, dan mengajukan ijin, perdin maupun kebutuhan lain yang difasilitasi oleh HR / HCM secara online. Penanggulangan data overload, waktu operasional yang lama dan salah penilaian karyawan dijawab oleh sistem ini.

### *User Story CEKLOK.ID*

Register dilakukan seorang user untuk mendaftar sendiri menjadi end user dari sistem HRIS / E-Recruitment / Assessment / Travel Agent. Admin HRIS dapat memasukkan data karyawan dengan mengunggah data sesuai dengan template yang dapat di unduh. Sehingga user yang telah registrasi dan terdaftar untuk dapat masuk ke sistem. Dengan memasukkan username dan password. Sistem akan membedakan peran sesuai dengan data yang dimiliki. Semua user dapat login via website. User yang dapat melakukan login via mobile adalah user dari HRIS, tepatnya sebagai karyawan. Dapat mengakses daily mobile HRIS features.

User dapat merubah password login jika dirasa dibutuhkan atau jika lupa password saat akan login. User Admin dapat mengatur hak akses setiap user berdasar grading dan status dan dapat mengelola data karyawan yang telah tersimpan dalam DB. Pengaturan data berupa Departemen, Grading, Job title, Karyawan, Status, Status Non-Aktif, Broadcast Mail, Template dan Benefit. User Admin dapat mengelola absen karyawan. Karyawan dapat melakukan check-in/out, dan melihat informasi terkait absen. Pengelolaan berupa Geolocation, Check In Frontend dan Leave & Permission Approval.

User admin dapat mengelola cuti dan lembur dari segi pengaturan, approval dan checking. Dan karyawan dapat mengelola cuti dengan pengajuan dan checking. Dan karyawan (SPV) dapat mengelola lembur dengan pengajuan, laporan dan checking.

Kode surat dapat ditambah, dilihat dan dirubah oleh user admin. Pengelolaan surat menyurat seperti Nomor Surat, Bank Template dan Broadcast surat dapat dilakukan user admin dan karyawan. Kegiatan perjalanan dinas difasilitasi pada sistem HRIS. Dimana pengaturan tentang Request, Reschedule, Cancel, History, Pembelian Tiket, Notification, Transaction, Payment, Reimbursement, Limitasi, Jenis dan Jenis Price dapat dilakukan.

User admin dapat mengelola aset dari segi pengaturan, approval dan checking. Dan karyawan (SPV) dapat mengelola aset dengan pengajuan, pengajuan maintenance dan checking. Man Power Planning (MPP) dan Man Power Needs (MPN) dapat diatur dan ditambahkan rancangan anggaran yang dibutuhkan.

Mengatur payroll dapat dilakukan lewat HRIS ini. Mengelola Employee Payroll, pembayaran, Raise, Download Slip, Tunjangan, Setting Program dan THR.

Program pelatihan disediakan dan terbagi menjadi 2 yaitu pelatihan internal dan eksternal. Dimana admin dapat mengatur, menjadwalkan, mengumumkan budgeting dan checking. Sedangkan karyawan dapat request dan checking. Pelatihan internal meliputi Existing Program List, Create Program, Edit Program, History Program, Detail Program, Apply Program, Program Completion dan Detail Result.

Pelatihan eksternal meliputi Request Training, List Request Training, Broadcast on change status dan Setting Budget. Aktivitas ini merupakan lanjutan dari pelatihan, dimana dalam ini dapat melakukan Edit Questionnaire + Score + Input Description, Approval Process, Follow Up Training dan mengelola Questionnaire. Pengaturan API dapat dilakukan berupa kegiatan List Follow Up dan Detail Follow Up API. Begitu juga pengaturan SOP oleh user, disediakan fitur Bank SOP, Submit SOP, Approval SOP, Audit SOP, List SOP, Detail SOP dan Download SOP

Aktivitas recruitment dilakukan secara online, admin difasilitasi untuk mengelola lowongan yang dibuka perusahaan. Pengaturan berupa open/close, informasi, view, tracking, schedule test dan penerimaan.

Untuk user dapat melakukan registrasi untuk membuat akun E-recruitment. Login lewat landing page untuk menjadi user e-rec, kemudian melengkapi data diri sehingga dapat apply lowongan.

Admin dapat memberikan penilaian kepada karyawan secara sistematis berdasarkan sisi psikologi yang akan diselaraskan dengan KPI perusahaan. Karyawan dapat melakukan tes yang diberikan dan melihat score kemampuan masing-masing.

Admin berinteraksi dengan agen travel agent serta dapat melakukan pengelolaan Request dan Transaction

#### 4.1.2 Aplikasi 2 : COMMJUNCTION

COMMJUNCTION merupakan platform yang dapat digunakan oleh komunitas apapun. Berikut rangkuman kondisi pengerjaan Aplikasi COMMJUNCTION pada Tabel 4.2.

**Tabel 4.2 Rangkuman Aplikasi COMMJUNCTION**

| Status                          | Result    |                | Estimasi  |                |
|---------------------------------|-----------|----------------|-----------|----------------|
| On-Going<br>(soft launch Maret) | 28 sprint | Rp 200.869.565 | 24 sprint | Rp 208.043.478 |

Wadah untuk menampung komunitas apapun. Dimana komunitas dapat memiliki aplikasi sendiri yang dapat mengatur aktivitas-aktivitas komunitas dimanapun dan kapanpun. Sebagai komunitas user akan mendapatkan value lebih dari modul-modul yang ditawarkan seperti berkomunikasi dengan member, memberikan berita internal komunitas, dapat kustomisasi app yang disesuaikan dengan keunikan komunitas sendiri dan pastinya dapat mengelola secara online merupakan hal menakjubkan yang dapat dimiliki tanpa harus bersusah payah membuat app sendiri. Jadi tidak perlu lagi mengundang member dan memberikan informasi satu persatu kepada member komunitas. Semua pembayaran dalam satu pintu. All in one App for all Community.

#### *User Story COMMJUNCTION*

Komunitas memiliki aplikasi sendiri dimana dapat melakukan kustomisasi sesuai keunikan komunitas dan mengatur aktivitas-aktivitas komunitas dimanapun dan kapanpun. Tidak perlu lagi mengundang dan memberikan informasi kepada satu-persatu member karena merupakan hal yg melelahkan. Berkomunikasi dengan member, memberikan berita internal komunitas, melakukan pembayaran dengan mudah tanpa harus bersusah payah membuat app sendiri.

COMMJUNCTION merupakan sebuah platform untuk komunitas apapun, di Indonesia. Aplikasi berupa Website dan Mobile membuat COMMJUNCTION dapat dibuat dan digunakan dimanapun dan kapanpun. Komunitas akan memiliki seseorang untuk menjadi admin, dimana admin merupakan orang yang memiliki kendali untuk mengatur dan memilih platform milik komunitas. Penggunaan fitur-

fitur tergantung pada modul-modul yang dipilih komunitas, maka harga juga tergantung penggunaan modul.

Modul-modul yang ditawarkan memenuhi kebutuhan aktivitas komunitas. Setelah Admin registrasi dan membayar pilihan paket modul, admin dapat mengelola anggota (subscriber), setingan dan tampilan aplikasi untuk komunitas. Anggota dapat login jika telah register dan menikmati fitur-fitur sesuai dengan yang dipilih.

Tersedia fitur Single Sign In untuk meningkatkan keamanan sistem pada Login. Dashboard akan memberikan gambaran terhadap user (admin, admin komunitas dan anggota komunitas). admin, admin komunitas dan anggota komunitas dapat mengelola profil masing-masing. admin dan admin komunitas dapat mengelola Subscriber, Report, modul, user dan log dimana report Transaction, Module Report dan reconciliation didapatkan oleh mereka. admin sendiri dapat mengelola Payment dan Pricing.

User dapat customize module yang nantinya akan dijaikan template, dengan minimal spesifikasi tersedia dashboard, Module Management, Setting, Log Management, Payment Management dan API. Sedangkan pada awal rilis sekarang modul yang disediakan adalah NEWS, EVENT dan FRIENDS. dimana didalam modul tersebut telah tersedia masing-masing Dashboard, setting, Module Management, Log Management beserta Open API.

Keluhan-keluhan seperti harus mengundang member atau menyebarkan informasi broadcast satu per satu merupakan hal yang melelahkan. Dengan COMMJUNCTION itu semua dapat dilakukan dengan mudah, dimanapun dan kapanpun. Cukup satu aplikasi untuk semua kegiatan komunitas.

#### **4.2 Perhitungan *Use Case Point***

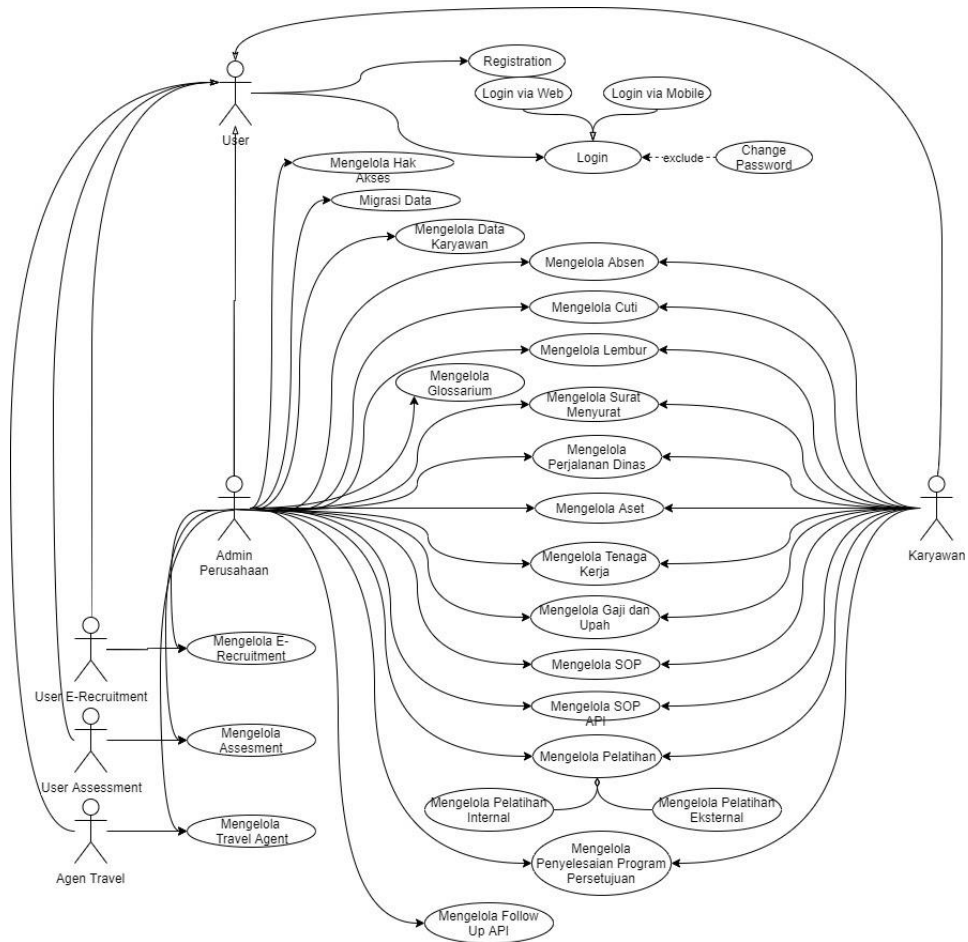
Perhitungan menggunakan Use Case Point dilakukan pada kedua aplikasi dengan runut yang telah dijelaskan pada Bab 3 dan ketentuan pada Bab 2.

## 4.2.1 Aplikasi 1 : CEKLOK.ID

Langkah awal perhitungan menggunakan Use Case Point adalah dengan membuat Use Case Diagram, berikut adalah penjabaran Use Case untuk aplikasi CEKLOK.

### 4.2.1.1 Use Case

Use Case Diagram berdasarkan User Story yang telah divalidasi oleh tim pengembang dari PT. Vascomm Solusi Teknologi. Berikut Use Case Diagram CEKLOK digambarkan pada Gambar 4.1 Use Case CEKLOK.



**Gambar 4.1 Use Case CEKLOK**

Diagram ini dibutuhkan dalam penelitian ini untuk menghitung estimasi proyek CEKLOK menggunakan metode Use Case Point.

#### 4.2.1.2 Unadjusted Actor Weights (UAW)

Nilai Unadjusted Actor Weight (UAW) didapat dengan mengidentifikasi dan mengklasifikasikan jenis aktor. Jenis aktor diklasifikasikan menjadi tiga kategori yaitu; Simple, Average dan Complex. Simple actor merupakan merupakan aktor yang berinteraksi melalui API seperti Command Prompt dengan bobot 1. Average actor adalah aktor yang berinteraksi melalui protokol seperti TCP/IP, FTP, HTTP atau disebut aktor yang melakukan penyimpanan data (file, RDBMS) dengan bobot 2. Sedangkan Complex Actor merupakan aktor yang berinteraksi melalui GUI atau halaman web dengan bobot 3 (Ribu, 2001).

Aktor-aktor yang akan dikategorikan merupakan hasil dari analisis Use Case pada Aplikasi CEKLOK. Karena kebutuhan aktor yang lebih mudah bekerja dengan Web page untuk menjalankan aplikasi maka terdapat 5 aktor yang berperan dalam sistem yang dikategorikan dalam tipe Complex. Penjelasan mengenai peran aktor-aktor dalam berinteraksi dengan sistem yang disajikan pada Tabel 4.3.

**Tabel 4.3 Deskripsi Aktor pada Aplikasi 1**

| Aktor            | Deskripsi   | Tipe    |
|------------------|---|---------|
| Admin Perusahaan | User dari perusahaan (bagian HRD/HCM) yang berperan sebagai admin untuk mengelola user karyawan dan privilege sistem sesuai aturan perusahaan | Complex |
| Karyawan         | User yang menggunakan sistem yang telah terdaftar dalam sebuah perusahaan   | Complex |
| Recruitment User | User yang menggunakan modul recruitment untuk melamar pekerjaan   | Complex |
| Travel User      | User dari pihak travel yang bekerja sama untuk modul travel dan perjalanan bisnis   | Complex |
| Assessment User  | User yang menggunakan modul assessment untuk mengikuti test psikologi yang berhubungan dengan dunia kerja                                     | Complex |

Setelah melakukan pengklasifikasian aktor pada Use Case CEKLOK langkah selanjutnya yaitu melakukan perhitungan total Unadjusted Actor Weight (UAW) dengan mengalikan bobot dengan jumlah aktor sesuai dengan tipe aktor yang akan disajikan pada Tabel 4.4.

**Tabel 4.4 Perhitungan Unadjusted Actor Weight (UAW)**

| Tipe             | Bobot | Jumlah | Bobot x Jumlah |
|------------------|-------|--------|----------------|
| Simple           | 1     | 0      | 0              |
| Average          | 2     | 0      | 0              |
| Complex          | 3     | 5      | 15             |
| <b>Total UAW</b> |       |        | <b>15</b>      |

#### 4.2.1.3 Daftar Use Case dan Unadjusted Use Case Weights (UUCW)

Nilai Unadjusted Use Case Weight (UUCW) didapat dari hasil menentukan kategori use case yaitu sebagai simple, medium atau complex yang ditentukan berdasarkan jumlah transaksi sukses dan alternatif. Untuk use case simple memiliki  $\leq 3$  transaksi dalam use case dengan bobot kompleksitasnya adalah 5, untuk use case average memiliki 4-7 transaksi dalam use case dengan bobot kompleksitasnya adalah 10 sedangkan use case complex memiliki  $> 7$  transaksi dalam use case dengan bobot kompleksitasnya adalah 15 (Sholiq, 2015).

Penelitian ini menggunakan aturan penentuan transaksi berdasarkan actor goals atau disebut semantic point. Jika melihat transaksi dari semantic point, sebagian besar use case memiliki goals atau tujuan yang sama. Semantic point telah membedakan 12 jenis transaksi yaitu Create, Retrieve, Update, Delete, Link, Delete Link, Asynchronous Retrieve, Dynamic Retrieve, Dynamic Retrieve, Transfer, Check Object, Complex Internal Activity dan Change State (Ochodek et al., 2011).

Berikut merupakan analisis jumlah transaksi yang ada pada use case yang disajikan pada Tabel 4.5.

**Tabel 4.5 Daftar Tipe Use Case Aplikasi 1**

| No. | Use Case                     | Tipe    |
|-----|------------------------------|---------|
| 1   | Karyawan Registrasi personal | Average |
| 2   | Migrate Data                 | Simple  |
| 3   | Web Login                    | Simple  |
| 4   | Mobile Login                 | Simple  |
| 5   | Logout                       | Simple  |
| 6   | Hak Akses                    | Simple  |
| 7   | Forgot Password              | Average |
| 8   | Change Password              | Average |
| 9   | Add Departement              | Simple  |
| 10  | Update Departement           | Simple  |



| No. | Use Case                                 | Type    |
|-----|--|---------|
| 11  | Delete Departement                       | Simple  |
| 12  | Add Grading                              | Simple  |
| 13  | Update Grading                           | Simple  |
| 14  | Delete Grading                           | Simple  |
| 15  | Add Job Title                            | Simple  |
| 16  | Update Job Title                         | Simple  |
| 17  | Delete Job Title                         | Simple  |
| 18  | Add Karyawan Baru                        | Simple  |
| 19  | Detail Karyawan                          | Simple  |
| 20  | Update Info Karyawan                     | Simple  |
| 21  | Update Status Karyawan                   | Simple  |
| 22  | Grading Karyawan                         | Simple  |
| 23  | Rotasi Karyawan                          | Simple  |
| 24  | Peringatan Karyawan                      | Simple  |
| 25  | Add Status Karyawan                      | Simple  |
| 26  | Set Cuti Karyawan by Status              | Simple  |
| 27  | Checklist                                | Simple  |
| 28  | Un-Check process                         | Simple  |
| 29  | Broadcast info                           | Simple  |
| 30  | Add Broadcast Address                    | Simple  |
| 31  | Remove Broadcast Address                 | Simple  |
| 32  | Add Template                             | Simple  |
| 33  | Remove Template                          | Simple  |
| 34  | Edit Template                            | Simple  |
| 35  | Add Benefit                              | Simple  |
| 36  | Remove Benefit                           | Simple  |
| 37  | Edit Benefit                             | Simple  |
| 38  | Checklist & Upload Data                  | Simple  |
| 39  | Lock check-in/check-out                  | Simple  |
| 40  | View Position                            | Average |
| 41  | Check In Frontend                        | Simple  |
| 42  | Informasi Cuti                           | Simple  |
| 43  | Informasi Token                          | Simple  |
| 44  | Automated Start Cuti                     | Simple  |
| 45  | Setting count start nya berapa           | Simple  |
| 46  | Setting bulan per kapan (cut off)        | Simple  |
| 47  | Checklist probasi hitung cuti atau tidak | Simple  |
| 48  | Add Cuti                                 | Simple  |
| 49  | Remove Cuti                              | Simple  |
| 50  | View All Data Cuti                       | Simple  |
| 51  | Detail Request Cuti                      | Simple  |
| 52  | Proses Request Lembur                    | Simple  |
| 53  | Proses Approval                          | Simple  |

| No. | Use Case                                  | Type    |
|-----|---|---------|
| 54  | Proses Finish Overtime                    | Average |
| 55  | View All Data                             | Simple  |
| 56  | Filter Data                               | Simple  |
| 57  | Setting Approval                          | Simple  |
| 58  | Setting Harga Overtime                    | Simple  |
| 59  | Notifikasi Approval - Email               | Simple  |
| 60  | Notifikasi Approval - Mobile - Push Notif | Simple  |
| 61  | Notifikasi Finish - Email                 | Simple  |
| 62  | Notifikasi Finish - Mobile - Push Notif   | Simple  |
| 63  | Add New Kode Surat                        | Simple  |
| 64  | List Kode Surat                           | Simple  |
| 65  | Edit Kode Surat                           | Simple  |
| 66  | Request Nomor Surat                       | Simple  |
| 67  | History Request Nomor Surat               | Simple  |
| 68  | History Nomor Surat                       | Simple  |
| 69  | Approval Nomor Surat                      | Simple  |
| 70  | Upload New Template                       | Simple  |
| 71  | Edit File Template                        | Simple  |
| 72  | Download Template                         | Simple  |
| 73  | Broadcast Surat                           | Simple  |
| 74  | Re-Broadcast Surat                        | Simple  |
| 75  | List Surat                                | Simple  |
| 76  | Detail Surat                              | Simple  |
| 77  | Request Perdin                            | Simple  |
| 78  | Reschedule Perdin                         | Simple  |
| 79  | Cancel Perdin                             | Simple  |
| 80  | List Data Perdin                          | Simple  |
| 81  | Detail Perdin                             | Simple  |
| 82  | Approval / Reject Perdin Admin            | Simple  |
| 83  | Approval / Reject Perdin SPV              | Simple  |
| 84  | Upload Ticket                             | Simple  |
| 85  | Confirm Data & Approval Ticket - SPV      | Simple  |
| 86  | Confirm Data & Approval Ticket - User     | Simple  |
| 87  | Notif Ticket Purchased - SPV              | Simple  |
| 88  | Notif Ticket Purchased - User             | Simple  |
| 89  | History Transaksi tiket                   | Simple  |
| 90  | Download Reconcile Report                 | Simple  |
| 91  | Payment Confirmation                      | Simple  |
| 92  | Upload Reimbursement                      | Simple  |
| 93  | Approval Reimbursement                    | Simple  |
| 94  | Payment Confirmation                      | Simple  |
| 95  | Add reimburse type                        | Simple  |
| 96  | Edit reimburse type                       | Simple  |

| No. | Use Case                      | Type    |
|-----|-------------------------------|---------|
| 97  | Set limit dan rules tertulis  | Simple  |
| 98  | Set time limit request perdin | Simple  |
| 99  | Add Perdin                    | Simple  |
| 100 | Edit Perdin                   | Simple  |
| 101 | Add Jenis Prices              | Simple  |
| 102 | Edit Jenis Prices             | Simple  |
| 103 | List Data Request Ticket      | Simple  |
| 104 | Input Data Ticket             | Simple  |
| 105 | Approval Payment              | Simple  |
| 106 | Transaction                   | Simple  |
| 107 | Request Asset                 | Simple  |
| 108 | List Request Assets           | Simple  |
| 109 | History Request Asset         | Simple  |
| 110 | Notifikasi email              | Simple  |
| 111 | Create step status            | Simple  |
| 112 | Add Asset                     | Simple  |
| 113 | Edit Asset                    | Simple  |
| 114 | Delete Asset                  | Simple  |
| 115 | View Information Asset Mobile | Simple  |
| 116 | List Done Request             | Simple  |
| 117 | Process Done Request          | Simple  |
| 118 | History Asset                 | Simple  |
| 119 | Request Maintenance Asset     | Simple  |
| 120 | View List Asset Maintenance   | Simple  |
| 121 | Approve Maintenance           | Simple  |
| 122 | Finish Maintenance            | Simple  |
| 123 | History Maintenance           | Simple  |
| 124 | Add annual MPP                | Simple  |
| 125 | Edit MPP                      | Simple  |
| 126 | Lock MPP                      | Simple  |
| 127 | Approval annual MPP           | Simple  |
| 128 | History MPP                   | Simple  |
| 129 | Create New Request            | Simple  |
| 130 | Approval Request              | Simple  |
| 131 | Add Position                  | Simple  |
| 132 | Edit Position                 | Simple  |
| 133 | Activate/Deactivate           | Simple  |
| 134 | Landing Page Pre Login        | Average |
| 135 | Setting Landing Page          | Simple  |
| 136 | Register Account              | Complex |
| 137 | Verifikasi Account            | Simple  |
| 138 | Web Login                     | Simple  |
| 139 | View Informasi kandidat       | Simple  |

| No. | Use Case                        | Type    |
|-----|---------------------------------|---------|
| 140 | Edit data diri                  | Simple  |
| 141 | List lowongan active            | Simple  |
| 142 | Detail lowongan                 | Simple  |
| 143 | History Lowongan                | Simple  |
| 144 | Proses Apply                    | Simple  |
| 145 | View Schedule                   | Simple  |
| 146 | Proses Reschedule               | Simple  |
| 147 | Forgot Password                 | Average |
| 148 | Change Password                 | Average |
| 149 | Dashboard Admin                 | Simple  |
| 150 | Management Lowongan             | Simple  |
| 151 | View Lowongan                   | Simple  |
| 152 | Add Lowongan                    | Simple  |
| 153 | Edit Lowongan                   | Simple  |
| 154 | History Lowongan                | Simple  |
| 155 | List Kandidat                   | Simple  |
| 156 | View Kandidat                   | Simple  |
| 157 | Add Status                      | Simple  |
| 158 | Assign Status                   | Simple  |
| 159 | Create Schedule                 | Simple  |
| 160 | Detail Schedule                 | Simple  |
| 161 | View All Schedule               | Simple  |
| 162 | Absence Schedule                | Simple  |
| 163 | Close Schedule                  | Simple  |
| 164 | Management Assesment            | Complex |
| 165 | List Employee & History Payroll | Simple  |
| 166 | Detail Payroll Status           | Simple  |
| 167 | Payment Confirmation            | Simple  |
| 168 | Payment Status                  | Simple  |
| 169 | Payment Process                 | Average |
| 170 | Add Payment choice              | Simple  |
| 171 | Setting Payment choice          | Simple  |
| 172 | New Raise                       | Simple  |
| 173 | Edit Raise                      | Simple  |
| 174 | Approval Raise                  | Simple  |
| 175 | Download Slip                   | Simple  |
| 176 | Management Tunjangan            | Complex |
| 177 | Add Program                     | Simple  |
| 178 | Edit Program                    | Simple  |
| 179 | Deactivate Program              | Simple  |
| 180 | Perhitungan THR                 | Simple  |
| 181 | Request Training                | Simple  |
| 182 | Detail Training Request         | Simple  |

| No. | Use Case                                       | Type    |
|-----|--|---------|
| 183 | Approval Request                               | Simple  |
| 184 | Payment Approval Request                       | Simple  |
| 185 | History Request Training                       | Simple  |
| 186 | Broadcast on change status                     | Simple  |
| 187 | Add Annual Budget                              | Simple  |
| 188 | Edit Annual Budget                             | Simple  |
| 189 | Training Package List                          | Simple  |
| 190 | Apply Training                                 | Simple  |
| 191 | Confirm Training                               | Simple  |
| 192 | Create Program                                 | Simple  |
| 193 | Edit Program                                   | Simple  |
| 194 | History Program                                | Simple  |
| 195 | Detail Program                                 | Simple  |
| 196 | Apply Program                                  | Simple  |
| 197 | Upload Follow Up                               | Simple  |
| 198 | Input Questionnaire                            | Average |
| 199 | Detail Result                                  | Simple  |
| 200 | Edit Questionnaire + Score + Input Description | Average |
| 201 | Email notification broadcast                   | Simple  |
| 202 | Upload Training Completion                     | Simple  |
| 203 | Approval Training Completion                   | Simple  |
| 204 | Create Questionnaire                           | Simple  |
| 205 | Edit Questionnaire                             | Simple  |
| 206 | Active/Deactivate                              | Simple  |
| 207 | List Follow Up                                 | Simple  |
| 208 | Detail Follow Up                               | Simple  |
| 209 | List SOP                                       | Simple  |
| 210 | Detail SOP                                     | Simple  |
| 211 | Submit SOP                                     | Simple  |
| 212 | Approval SOP                                   | Simple  |
| 213 | Audit SOP                                      | Simple  |
| 214 | List SOP                                       | Simple  |
| 215 | Detail SOP                                     | Simple  |
| 216 | Download SOP                                   | Simple  |

Setelah mengelompokkan skenario menjadi sejumlah transaksi dan dapat diketahui tipe use casenya, maka langkah selanjutnya melakukan perhitungan Unadjusted Use Case Weight (UUCP) dengan mengalikan bobot dengan jumlah use case sesuai dengan tipe yang akan disajikan pada Tabel 4.6.

**Tabel 4.6 Perhitungan Unadjusted Use Case Weight (UUCW) Aplikasi 1**

| <b>Tipe</b>       | <b>Bobot</b> | <b>Jumlah</b> | <b>Bobot x Jumlah</b> |
|-------------------|--------------|---------------|-----------------------|
| Simple            | 1            | 202           | 202                   |
| Average           | 2            | 11            | 22                    |
| Complex           | 3            | 3             | 9                     |
| <b>Total UUCW</b> |              |               | <b>233</b>            |

#### **4.2.1.4 Unadjusted Use Case Point (UUCP)**

Kemudian setelah mendapatkan nilai Unadjusted Actor Weight (UAW) dan nilai Unadjusted Use Case Weight (UUCW) langkah selanjutnya melakukan perhitungan nilai Unadjusted Use Case Point (UUCP) berdasarkan persamaan yang terlampir pada Bab 2 rumus 2.1

$$\mathbf{UUCP = UAW + UUCW}$$

$$= 15 + 233$$

$$= 248$$

Nilai **Unadjusted Use Case Point (UUCP)** sebesar **248** didapat dari hasil penjumlahan nilai Unadjusted Actor Weight (UAW) dan nilai Unadjusted Use Case Weight (UUCW).

#### **4.2.1.5 Technical Complexity Factors (TCF)**

Pada tahap ini dilakukan penilaian melalui penyebaran lembar penilaian, perhitungan ini menilai skor yang diisi oleh tim pengembang pada setiap faktor teknis. Dalam pelaksanaan penyebaran lembar penilaian ini melibatkan responden dari tim pengembang CEKLOK. Rekap hasil dan perhitungan pada masing-masing faktor teknis yang disajikan pada Tabel 4.7. Perhitungan skor untuk setiap faktor didapat dari hasil pengisian lembar penilaian oleh tim pengembang kemudian dilakukan penjumlahan dan mencari rata-rata dari setiap faktor. Untuk rekap seluruh hasil pengisian lembar penilaian oleh tim pengembang terlampir pada Lampiran Hasil Penilaian Complexity Factor CEKLOK.

**Tabel 4.7 Perhitungan Technical Complexity Factors Aplikasi 1**

|                  | Technical Complexity Factors         | Bobot                       | Skor  | Bobot x Skor  |
|------------------|--------------------------------------|-----------------------------|-------|---------------|
| T1               | Distributed System Required          | 2                           | 4     | 8             |
| T2               | Response Time                        | 2                           | 3,875 | 7,75          |
| T3               | End User Efficiency                  | 1                           | 3,375 | 3,375         |
| T4               | Complex Internal Processing Required | 1                           | 3,375 | 3,375         |
| T5               | Reusable Code                        | 1                           | 3,875 | 3,875         |
| T6               | Easy to Install                      | 0,5                         | 3,125 | 1,5625        |
| T7               | Easy to Use                          | 0,5                         | 4,375 | 2,1875        |
| T8               | Portable                             | 2                           | 4     | 8             |
| T9               | Easy to Change                       | 1                           | 3,625 | 3,625         |
| T10              | Concurrent                           | 1                           | 3,25  | 3,25          |
| T11              | Security Features                    | 1                           | 4,125 | 4,125         |
| T12              | Access for Third Parties             | 1                           | 3     | 3             |
| T13              | Special Training Required            | 1                           | 2,625 | 2,625         |
| Total TF         |                                      |                             |       | 54,75         |
| <b>TOTAL TCF</b> |                                      | <b>0,6 + (0,01 * 54,75)</b> |       | <b>1,1475</b> |

Nilai **TCF** pada Tabel 4.7 sebesar **1,475** didapat dari rumus perhitungan  $TCF = 0,6 + (0,01 * TF)$ . Dimana TF adalah hasil penjumlahan dari seluruh perkalian skor dan bobot pada 13 faktor.

#### **4.2.1.6 Environment Complexity Factor (ECF)**

Pada tahap ini juga dilakukan penilaian melalui penyebaran lembar penilaian, perhitungan ini menilai skor yang diisi oleh tim pengembang pada setiap faktor lingkungan. Dalam pelaksanaan penyebaran lembar penilaian ini melibatkan responden dari tim pengembang CEKLOK.

Rekap hasil dan perhitungan pada masing-masing faktor lingkungan yang disajikan pada Tabel 4.8. Perhitungan skor untuk setiap faktor didapat dari hasil pengisian lembar penilaian oleh tim pengembang kemudian di lakukan penjumlahan dan mencari rata-rata dari setiap faktor. Untuk rekap seluruh hasil pengisian lembar penilaian oleh tim pengembang terlampir pada Lampiran Hasil Penilaian Complexity Factor CEKLOK.

**Tabel 4.8 Perhitungan Environmental Complexity Factors Aplikasi 1**

|                  | Environmental Complexity Factors | Bobot                        | Skor  | Bobot x Skor |
|------------------|----------------------------------|------------------------------|-------|--------------|
| E1               | Familiarity with The Project     | 1,5                          | 3,125 | 4,6875       |
| E2               | Application Experience           | 0,5                          | 3,625 | 1,8125       |
| E3               | OO Programming Experience        | 1                            | 3,5   | 3,5          |
| E4               | Lead Analyst Capability          | 0,5                          | 3,5   | 1,75         |
| E5               | Motivation                       | 1                            | 3,75  | 3,75         |
| E6               | Stable Requirements              | 2                            | 3,625 | 7,25         |
| E7               | Part Time Staff                  | -1                           | 1,625 | -1,625       |
| E8               | Difficult Programming Language   | -1                           | 1,875 | -1,875       |
| Total EF         |                                  |                              |       | 19,25        |
| <b>TOTAL TCF</b> |                                  | <b>1,4 + ( - 0,03*19,25)</b> |       | 0,82         |

Nilai ECF pada sebesar 0,82 didapat dari rumus perhitungan  $ECF = 1,4 + (- 0,03 * EF)$ . Dimana EF adalah penjumlahan dari seluruh perkalian skor dan bobot pada 8 faktor.

#### 4.2.1.7 Use Case Point

Sebelum dilakukan perhitungan hours of effort, harus dihitung terlebih dahulu nilai Use Case Point (UCP) yang didapatkan dari perkalian antara nilai UUCP, nilai TCF dan nilai ECF sesuai dengan persamaan berikut ini.

$$\begin{aligned}
 \text{Use Case Point (UCP)} &= \text{UUCP} * \text{TCF} * \text{ECF} \\
 &= 248 * 1,15 * 0,82 \\
 &= 234,07
 \end{aligned}$$

Nilai **UCP** sebesar **234,07** didapat dari hasil perkalian antara nilai UUCP, nilai TCF dan nilai ECF.

#### 4.2.1.8 Estimasi Usaha

Selanjutnya mengubah nilai Use Case Point (UCP) yang telah didapat menjadi nilai effort yaitu Hours of Effort, maka nilai UCP harus dikalikan dengan nilai staff-hour per Use Case Point berdasarkan teori Karner (1993) dengan p bernilai 20 *staff hours*. Perhitungan effort yang dibutuhkan untuk pengerjaan aplikasi CEKLOK ini dapat diketahui dengan menggunakan persamaan berikut ini.

$$\text{Hours of Effort} = \text{UCP} \times p$$



$$= 234,07 \times 20$$

$$= 4681,34 \text{ jam}$$

Berdasarkan persamaan di atas, nilai 234,06 merupakan nilai Use Case Point (UCP) yang didapat dari hasil penjumlahan antara nilai Unadjusted Use Case Point (UUCP) dan nilai Technical and Environment Complexity Factor (TCF dan ECF). Hasil dari nilai staff-hour per Use Case Point, sehingga dalam proyek ini memiliki waktu kerja sebanyak 4681,34 jam.

#### **4.2.1.9 Estimasi Biaya**

Estimasi biaya / cost pada metode Scrum didasarkan pada seberapa banyak sprint yang dibutuhkan untuk membangun software / aplikasi tersebut. Setiap proyek pengembangan software pada scrum memiliki biaya Man Power per Sprint atau biasa disebut Sprint Rate. Sprint Rate ini adalah biaya yang harus dikeluarkan setiap Sprint-nya tergantung pada banyaknya personel Scrum Team dan Payroll Rate yang dibutuhkan.

Sesuai kesepakatan Scrum Team, Sprint dilakukan seminggu sekali atau lima hari kerja (Sabtu - Minggu libur). Untuk pengembangan aplikasi CEKLOK, Sprint Rate yang dibutuhkan adalah sebesar **Rp 8.913.043,00** (berdasarkan perhitungan pada Lampiran Perhitungan *Sprint Rate* dan *Day Rate* CEKLOK).

Berdasarkan hours of effort yang didapatkan sebelumnya yaitu 4681,34 jam. Dibutuhkan sekitar 195 hari kerja atau 8,1 bulan pengerjaan. Jumlah sprint yang dibutuhkan adalah sekitar 39 sprint. Maka jumlah estimasi biaya yang diperlukan:

$$\text{Estimasi biaya} = 39 \times \text{Rp } 8.913.043,00$$

$$= \text{Rp } 347.608.696,00$$

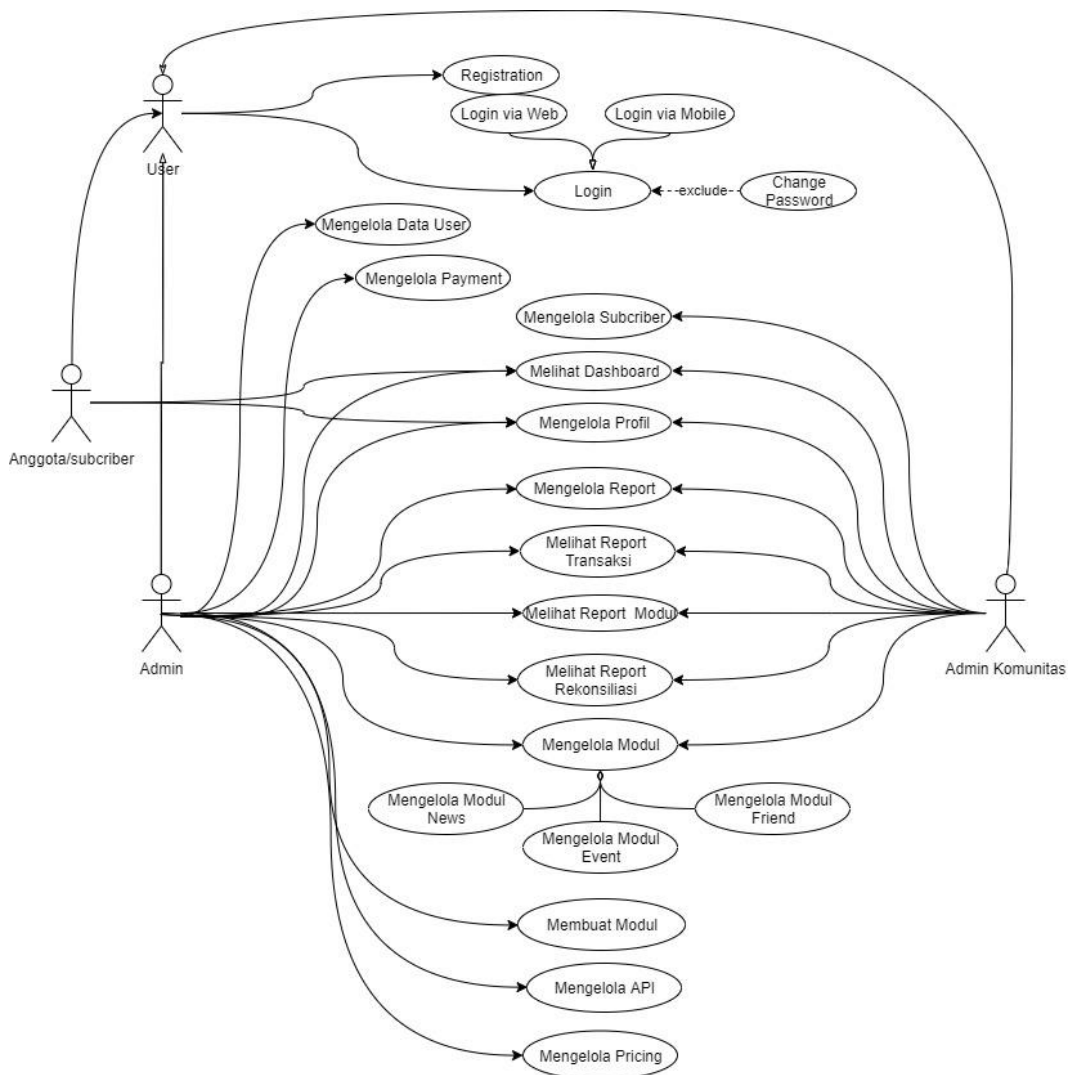
Jadi total estimasi biaya pengembangan CEKLOK berdasarkan Use Case Point yaitu sebesar **Rp 347.608.696,00**.

## 4.2.2 Aplikasi 2 : COMMJUNCTION

Langkah awal perhitungan menggunakan Use Case Point adalah dengan membuat Use Case Diagram, berikut adalah penjabaran Use Case untuk aplikasi COMMJUNCTION.

### 4.2.2.1 Use Case COMMJUNCTION

Use Case Diagram berdasarkan User Story yang telah divalidasi oleh tim pengembang dari PT. Vascomm Solusi Teknologi dan kemudian melakukan validasi Use Case Diagram. Diagram ini dibutuhkan dalam penelitian ini untuk menghitung estimasi proyek CEKLOK menggunakan metode Use Case Point. Berikut Use Case Diagram CEKLOK digambarkan pada Gambar 4.2.



Gambar 4.2 Use Case COMMJUNCTION

#### 4.2.2.2 *Unadjusted Actor Weights (UAW)*

Nilai *Unadjusted Actor Weight* (UAW) didapat dengan mengidentifikasi dan mengklasifikasikan jenis aktor. Jenis aktor diklasifikasikan menjadi tiga kategori yaitu; Simple, Average dan Complex. Simple actor merupakan merupakan aktor yang berinteraksi melalui API seperti Command Prompt dengan bobot 1. Average actor adalah aktor yang berinteraksi melalui protokol seperti TCP/IP, FTP, HTTP atau disebut aktor yang melakukan penyimpanan data (file, RDBMS) dengan bobot 2. Sedangkan Complex Actor merupakan aktor yang berinteraksi melalui GUI atau halaman web dengan bobot 3 (Ribu, 2001).

Aktor-aktor yang akan dikategorikan merupakan hasil dari analisis Use Case pada Aplikasi COMMJUNCTION. Karena kebutuhan aktor yang lebih mudah bekerja dengan Web page untuk menjalankan aplikasi maka terdapat 6 aktor yang berperan dalam sistem yang dikategorikan dalam tipe Complex. Penjelasan mengenai peran aktor-aktor dalam berinteraksi dengan sistem yang disajikan pada Tabel 4.9.

**Tabel 4.9 Deskripsi Aktor pada Aplikasi 2**

| Aktor             | Deskripsi  | Tipe    |
|-------------------|--|---------|
| Super Admin       | User yang mengelola sistem dan pengaturan user admin, pricing, payment   | Complex |
| Admin Komunitas   | User dari komunitas yang berperan sebagai admin untuk mengelola user dan privilege sistem sesuai kebutuhan komunitas | Complex |
| Anggota Komunitas | User yang menggunakan sistem yang telah terdaftar dalam komunitas dan membayar                                       | Complex |

Setelah melakukan pengklasifikasian aktor pada Use Case COMMJUNCTION langkah selanjutnya yaitu melakukan perhitungan total Unadjusted Actor Weight (UAW) dengan mengalikan bobot dengan jumlah aktor sesuai dengan tipe aktor yang akan disajikan pada Tabel 4.10.

**Tabel 4.10 Perhitungan Unadjusted Actor Weight (UAW)**

| Tipe             | Bobot | Jumlah | Bobot x Jumlah |
|------------------|-------|--------|----------------|
| Simple           | 1     | 0      | 0              |
| Average          | 2     | 0      | 0              |
| Complex          | 3     | 3      | 9              |
| <b>Total UAW</b> |       |        | <b>9</b>       |

#### 4.2.2.3 Daftar Use Case dan Unadjusted Use Case Weights (UUCW)

Nilai Unadjusted Use Case Weight (UUCW) didapat dari hasil menentukan kategori use case yaitu sebagai simple, medium atau complex yang ditentukan berdasarkan jumlah transaksi sukses dan alternatif. Untuk use case simple memiliki  $\leq 3$  transaksi dalam use case dengan bobot kompleksitasnya adalah 5, untuk use case average memiliki 4-7 transaksi dalam use case dengan bobot kompleksitasnya adalah 10 sedangkan use case complex memiliki  $> 7$  transaksi dalam use case dengan bobot kompleksitasnya adalah 15 (Sholiq, 2015).

Penelitian ini menggunakan aturan penentuan transaksi berdasarkan actor goals atau disebut semantic point. Jika melihat transaksi dari semantic point, sebagian besar use case memiliki goals atau tujuan yang sama. Semantic point telah membedakan 12 jenis transaksi yaitu Create, Retrieve, Update, Delete, Link, Delete Link, Asynchronous Retrieve, Dynamic Retrieve, Dynamic Retrieve, Transfer, Check Object, Complex Internal Activity dan Change State (Ochodek et al., 2011).

Berikut merupakan analisis jumlah transaksi yang ada pada use case yang disajikan pada Tabel 4.11.

**Tabel 4.11 Daftar Tipe Use Case Aplikasi 2**

| No. | Use Case                          | Tipe    |
|-----|-----------------------------------|---------|
| 1   | Administration Registration       | Complex |
| 2   | Payment Process (on some pricing) | Simple  |
| 3   | Payment Verification              | Simple  |
| 4   | Verify payment                    | Simple  |
| 5   | Email Blast                       | Simple  |
| 6   | Email Activation                  | Simple  |
| 7   | Google SSO                        | Simple  |
| 8   | Facebook SSO                      | Simple  |
| 9   | Custom Domain Portal              | Simple  |
| 10  | Custom Portal Page Looks          | Simple  |
| 11  | Portal Form                       | Simple  |
| 12  | Forgot Password                   | Average |
| 13  | Subscriber Registration Form      | Simple  |
| 14  | Community daily activity (A)      | Simple  |
| 15  | Community daily transaction (A)   | Simple  |
| 16  | Refresh Button (A)                | Simple  |

| No. | Use Case                         | Type   |
|-----|----------------------------------|--------|
| 17  | Total Community (A)              | Simple |
| 18  | Total Subscribers (A)            | Simple |
| 19  | Total Transaction Count (A)      | Simple |
| 20  | Total Transaction Number (A)     | Simple |
| 21  | Refresh Button (A)               | Simple |
| 22  | Top 5 Community (A)              | Simple |
| 23  | Top 5 Transaction (A)            | Simple |
| 24  | Top 5 Transaction Nominal (A)    | Simple |
| 25  | Pending Community Request (A)    | Simple |
| 26  | Community daily activity (CA)    | Simple |
| 27  | Community daily transaction (CA) | Simple |
| 28  | Refresh Button (CA)              | Simple |
| 29  | Total Community (CA)             | Simple |
| 30  | Total Subscribers (CA)           | Simple |
| 31  | Total Transaction Count (CA)     | Simple |
| 32  | Total Transaction Number (CA)    | Simple |
| 33  | Refresh Button (CA)              | Simple |
| 34  | Top 5 Subscriber (CA)            | Simple |
| 35  | Top 5 Transaction (CA)           | Simple |
| 36  | Pending Subscriber Request (CA)  | Simple |
| 37  | Module-Base General Info         | Simple |
| 38  | Basic Information                | Simple |
| 39  | Quick Menus                      | Simple |
| 40  | Module-Base General Info         | Simple |
| 41  | View List Subscriber             | Simple |
| 42  | View List Pending Subscriber     | Simple |
| 43  | Filter Subscriber                | Simple |
| 44  | Detail Subscriber                | Simple |
| 45  | Edit Subscriber                  | Simple |
| 46  | Approval Subscriber              | Simple |
| 47  | List All Transaction             | Simple |
| 48  | Filter Transaction               | Simple |
| 49  | Detail Transaction               | Simple |
| 50  | Download Transaction             | Simple |
| 51  | List All Module Activity         | Simple |
| 52  | Filter Activity                  | Simple |
| 53  | Download Module Activity         | Simple |
| 54  | List Transaction Reconcile       | Simple |
| 55  | Filter Transaction               | Simple |
| 56  | Download Transaction             | Simple |
| 57  | Connect New Module               | Simple |
| 58  | Display Module                   | Simple |
| 59  | Setting Module                   | Simple |

| No. | Use Case                | Type    |
|-----|-------------------------|---------|
| 60  | Pending Payment         | Simple  |
| 61  | History Payment         | Simple  |
| 62  | Payment Detail          | Simple  |
| 63  | Filter Payment          | Simple  |
| 64  | Download Payment        | Simple  |
| 65  | Payment Process         | Average |
| 66  | Payment Confirmation    | Simple  |
| 67  | Add New Method          | Simple  |
| 68  | Edit Payment Method     | Simple  |
| 69  | Choose Payment Method   | Simple  |
| 70  | Create User             | Simple  |
| 71  | List User               | Simple  |
| 72  | Detail User             | Simple  |
| 73  | Edit User               | Simple  |
| 74  | Create New User Type    | Simple  |
| 75  | Edit Priviledge         | Simple  |
| 76  | Edit Profile            | Simple  |
| 77  | Change Password         | Simple  |
| 78  | Logout                  | Simple  |
| 79  | View Log Activity       | Simple  |
| 80  | Filter Log Activity     | Simple  |
| 81  | Download Log Activity   | Simple  |
| 82  | Display Pricing Type    | Simple  |
| 83  | Edit Pricing Type       | Simple  |
| 84  | View Fee Sharing Status | Simple  |
| 85  | Edit Fee Sharing        | Simple  |
| 86  | Chart Information       | Simple  |
| 87  | List Information        | Simple  |
| 88  | View Data Custom Module | Simple  |
| 89  | View Data               | Simple  |
| 90  | Create Data             | Simple  |
| 91  | Updating Data           | Simple  |
| 92  | Custom Management       | Average |
| 93  | Dashboard information   | Simple  |
| 94  | Module Behaviour        | Simple  |
| 95  | Enable/Disable Module   | Simple  |
| 96  | Module Activity         | Simple  |
| 97  | API Activity            | Simple  |
| 98  | Payment Activy          | Simple  |
| 99  | Payment Management      | Average |
| 100 | View Data API           | Simple  |
| 101 | Create Data API         | Simple  |
| 102 | Update Data API         | Simple  |

| No. | Use Case                                      | Type   |
|-----|---|--------|
| 103 | Latest News                                   | Simple |
| 104 | Top Visited News                              | Simple |
| 105 | Top Loved News                                | Simple |
| 106 | List All News                                 | Simple |
| 107 | Create News                                   | Simple |
| 108 | Detail News                                   | Simple |
| 109 | Edit News                                     | Simple |
| 110 | Scrape News                                   | Simple |
| 111 | Publish / Unpublish News                      | Simple |
| 112 | Set News as Headline                          | Simple |
| 113 | Set Headline Number                           | Simple |
| 114 | Enable / Disable Visitor Count View           | Simple |
| 115 | Enable / Disable Like Count View              | Simple |
| 116 | Enable / Disable Share Count View             | Simple |
| 117 | Dashboard display checklist                   | Simple |
| 118 | Create Subscriber Activity Log                | Simple |
| 119 | Create Administrator Log                      | Simple |
| 120 | View All News                                 | Simple |
| 121 | View Parsed News                              | Simple |
| 122 | View Headlines                                | Simple |
| 123 | Like News                                     | Simple |
| 124 | View Detailed News                            | Simple |
| 125 | Visit Count                                   | Simple |
| 126 | Share Count                                   | Simple |
| 127 | Latest Events                                 | Simple |
| 128 | Top Attended Events                           | Simple |
| 129 | Top Visited Events                            | Simple |
| 130 | View All Events                               | Simple |
| 131 | Create Event                                  | Simple |
| 132 | Edit Event                                    | Simple |
| 133 | Detail Events                                 | Simple |
| 134 | Publish / Unpublish Event                     | Simple |
| 135 | Close Event                                   | Simple |
| 136 | View Attendance Confirmation                  | Simple |
| 137 | View detail attendance confirmation           | Simple |
| 138 | Approve Attendee                              | Simple |
| 139 | Approve Attendee Payment Confirmation         | Simple |
| 140 | Generate QR & Attendance Information          | Simple |
| 141 | Review Attendance Information (by QR or Code) | Simple |
| 142 | Attendance Payment Option                     | Simple |
| 143 | Attendance Confirmation Option                | Simple |
| 144 | Dashboard display checklist                   | Simple |

| No. | Use Case                           | Type   |
|-----|------------------------------------|--------|
| 145 | Create Subscriber Activity Log     | Simple |
| 146 | Create Administrator Log           | Simple |
| 147 | View All Events                    | Simple |
| 148 | Detail Events                      | Simple |
| 149 | Attending Events                   | Simple |
| 150 | Attending Payment                  | Simple |
| 151 | Confirm Payment                    | Simple |
| 152 | Detail Attending Information       | Simple |
| 153 | Pending Friend Confirmation List   | Simple |
| 154 | Top Friends                        | Simple |
| 155 | Total Friends                      | Simple |
| 156 | Friend Suggestion List             | Simple |
| 157 | View Friend list                   | Simple |
| 158 | Search Friend                      | Simple |
| 159 | Filter Friend Search               | Simple |
| 160 | View Friend Profile                | Simple |
| 161 | Add Friend                         | Simple |
| 162 | View Pending confirmation list     | Simple |
| 163 | Approve / Disapprove Friend        | Simple |
| 164 | Chat Friend (Whatsapp)             | Simple |
| 165 | Friend Suggestion List             | Simple |
| 166 | Friend Suggestion enable/disable   | Simple |
| 167 | Friend Confirmation enable/disable | Simple |
| 168 | Dashboard display checklist        | Simple |
| 169 | Create Subscriber Activity Log     | Simple |
| 170 | Create Administrator Log           | Simple |
| 171 | View Friend List                   | Simple |
| 172 | Search Friend                      | Simple |
| 173 | Filter Search                      | Simple |
| 174 | Add Friend                         | Simple |
| 175 | View Friend Profile                | Simple |
| 176 | View Pending Confirmation list     | Simple |
| 177 | Approval Confirmation list         | Simple |

Setelah mengelompokkan skenario menjadi sejumlah transaksi dan dapat diketahui tipe use casenya, maka langkah selanjutnya melakukan perhitungan Unadjusted Use Case Weight (UUCP) dengan mengalikan bobot dengan jumlah use case sesuai dengan tipe yang akan disajikan pada Tabel 4.12.



**Tabel 4.12 Perhitungan Unadjusted Use Case Weight (UUCP) Aplikasi 2**

| <b>Tipe</b>      | <b>Bobot</b> | <b>Jumlah</b> | <b>Bobot x Jumlah</b> |
|------------------|--------------|---------------|-----------------------|
| Simple           | 1            | 172           | 172                   |
| Average          | 2            | 4             | 8                     |
| Complex          | 3            | 1             | 3                     |
| <b>Total UAW</b> |              |               | <b>183</b>            |

#### **4.2.2.4 Unadjusted Use Case Point (UUCP)**

Kemudian setelah mendapatkan nilai Unadjusted Actor Weight (UAW) dan nilai Unadjusted Use Case Weight (UUCW) langkah selanjutnya melakukan perhitungan nilai Unadjusted Use Case Point (UUCP) berdasarkan persamaan yang terlampir pada Bab 2 rumus 2.1

$$\mathbf{UUCP = UAW + UUCW}$$

$$= 9 + 183$$

$$= 192$$

Nilai **Unadjusted Use Case Point (UUCP)** sebesar **192** didapat dari hasil penjumlahan nilai Unadjusted Actor Weight (UAW) dan nilai Unadjusted Use Case Weight (UUCW).

#### **4.2.2.5 Technical Complexity Factors (TCF)**

Pada tahap ini dilakukan penilaian melalui penyebaran lembar penilaian, perhitungan ini menilai skor yang diisi oleh tim pengembang pada setiap faktor teknis. Dalam pelaksanaan penyebaran lembar penilaian ini melibatkan responden dari tim pengembang COMMJUNCTION.

Rekap hasil dan perhitungan pada masing-masing faktor teknis yang disajikan pada Tabel 4.13. Perhitungan skor untuk setiap faktor didapat dari hasil pengisian lembar penilaian oleh tim pengembang kemudian di lakukan penjumlahan dan mencari rata-rata dari setiap faktor. Untuk rekap seluruh hasil pengisian lembar penilaian oleh tim pengembang terlampir pada Lampiran Hasil Penilaian Complexity Factor COMMJUNCTION.

**Tabel 4.13 Perhitungan Technical Complexity Factors Aplikasi 2**

|                  | Technical Complexity Factors         | Bobot                       | Skor  | Bobot x Skor |
|------------------|--------------------------------------|-----------------------------|-------|--------------|
| T1               | Distributed System Required          | 2                           | 4,167 | 8,333        |
| T2               | Response Time                        | 2                           | 4,000 | 8,000        |
| T3               | End User Efficiency                  | 1                           | 3,167 | 3,167        |
| T4               | Complex Internal Processing Required | 1                           | 3,833 | 3,833        |
| T5               | Reusable Code                        | 1                           | 4,167 | 4,167        |
| T6               | Easy to Install                      | 0,5                         | 3,000 | 1,500        |
| T7               | Easy to Use                          | 0,5                         | 4,167 | 2,083        |
| T8               | Portable                             | 2                           | 4,167 | 8,333        |
| T9               | Easy to Change                       | 1                           | 3,833 | 3,833        |
| T10              | Concurrent                           | 1                           | 3,167 | 3,167        |
| T11              | Security Features                    | 1                           | 4,333 | 4,333        |
| T12              | Access for Third Parties             | 1                           | 2,667 | 2,667        |
| T13              | Special Training Required            | 1                           | 2,833 | 2,833        |
| Total TF         |                                      |                             |       | 56,25        |
| <b>TOTAL TCF</b> |                                      | <b>0,6 + (0,01 * 56,25)</b> |       | 1,163        |

Nilai **TCF** pada Tabel 4.13 sebesar **1,163** didapat dari rumus perhitungan  $TCF = 0,6 + (0,01 * TF)$ . Dimana TF adalah hasil penjumlahan dari seluruh perkalian skor dan bobot pada 13 faktor.

#### **4.2.2.6 Environment Complexity Factor (ECF)**

Pada tahap ini juga dilakukan penilaian melalui penyebaran lembar penilaian, perhitungan ini menilai skor yang diisi oleh tim pengembang pada setiap faktor lingkungan. Dalam pelaksanaan penyebaran lembar penilaian ini melibatkan responden dari tim pengembang COMMJUNCTION.

Rekap hasil dan perhitungan pada masing-masing faktor lingkungan yang disajikan pada Tabel 4.14. Perhitungan skor untuk setiap faktor didapat dari hasil pengisian lembar penilaian oleh tim pengembang kemudian di lakukan penjumlahan dan mencari rata-rata dari setiap faktor. Untuk rekap seluruh hasil pengisian lembar penilaian oleh tim pengembang terlampir pada Lampiran Hasil Penilaian Complexity Factor COMMJUNCTION.

**Tabel 4.14 Perhitungan Environmental Complexity Factors Aplikasi 2**

|                  | Environmental Complexity Factors | Bobot                       | Skor  | Bobot x Skor |
|------------------|----------------------------------|-----------------------------|-------|--------------|
| E1               | Familiarity with The Project     | 1,5                         | 3,667 | 5,5          |
| E2               | Application Experience           | 0,5                         | 3,333 | 1,67         |
| E3               | OO Programming Experience        | 1                           | 3     | 3            |
| E4               | Lead Analyst Capability          | 0,5                         | 3,667 | 1,83         |
| E5               | Motivation                       | 1                           | 4     | 4            |
| E6               | Stable Requirements              | 2                           | 3,5   | 7            |
| E7               | Part Time Staff                  | -1                          | 0     | 0            |
| E8               | Difficult Programming Language   | -1                          | 2     | -2           |
| Total EF         |                                  |                             |       | 21           |
| <b>TOTAL TCF</b> |                                  | <b>1,4 + ( - 0,03 * 21)</b> |       | <b>0,77</b>  |

Nilai ECF pada sebesar 0,77 didapat dari rumus perhitungan  $ECF = 1,4 + (-0,03 * EF)$ . Dimana EF adalah penjumlahan dari seluruh perkalian skor dan bobot pada 8 faktor.

#### **4.2.2.7 Use Case Point**

Sebelum dilakukan perhitungan hours of effort, harus dihitung terlebih dahulu nilai Use Case Point (UCP) yang didapatkan dari perkalian antara nilai UUCP, nilai TCF dan nilai ECF sesuai dengan persamaan berikut ini.

$$\begin{aligned}
 \text{Use Case Point (UCP)} &= \text{UUCP} * \text{TCF} * \text{ECF} \\
 &= 192 * 1,163 * 0,77 \\
 &= 171,86
 \end{aligned}$$

Nilai **UCP** sebesar **171,86** didapat dari hasil perkalian antara nilai UUCP, nilai TCF dan nilai ECF.

#### **4.2.2.8 Estimasi Usaha**

Selanjutnya mengubah nilai Use Case Point (UCP) yang telah didapat menjadi nilai effort yaitu Hours of Effort, maka nilai UCP harus dikalikan dengan nilai staff-hour per Use Case Point berdasarkan teori Karner (1993) dengan bernilai 20 *staff hours*. Perhitungan effort yang dibutuhkan untuk pengerjaan aplikasi COMMJUNCTION ini dapat diketahui dengan menggunakan persamaan berikut ini.

$$\begin{aligned}
\text{Hours of Effort} &= \text{UCP} \times p \\
&= 171,86 \times 20 \\
&= 3437,28 \text{ jam}
\end{aligned}$$

Berdasarkan persamaan diatas, nilai 171,86 merupakan nilai Use Case Point (UCP) yang didapat dari hasil penjumlahan antara nilai Unadjusted Use Case Point (UUCP) dan nilai Technical and Environment Complexity Factor (TCF dan ECF). Hasil dari nilai staff-hour per Use Case Point, sehingga dalam proyek ini memiliki waktu kerja sebanyak 3437,28 jam.

#### 4.2.2.9 *Estimasi Biaya*

Estimasi biaya / cost pada metode Scrum didasarkan pada seberapa banyak sprint yang dibutuhkan untuk membangun software / aplikasi tersebut. Setiap proyek pengembangan software pada scrum memiliki biaya Man Power per Sprint atau biasa disebut Sprint Rate. Sprint Rate ini adalah biaya yang harus dikeluarkan setiap Sprint-nya tergantung pada banyaknya personel Scrum Team dan Payroll Rate yang dibutuhkan.

Sesuai kesepakatan Scrum Team, Sprint dilakukan seminggu sekali atau lima hari kerja (Sabtu - Minggu libur). Untuk pengembangan aplikasi COMMJUNCTION, Sprint Rate yang dibutuhkan adalah sebesar **Rp 7.173.913,00** (berdasarkan perhitungan pada Lampiran Perhitungan *Sprint Rate* dan *Day Rate* COMMJUNCTION).

Berdasarkan hours of effort yang didapatkan sebelumnya yaitu 4737,97 jam. Dibutuhkan sekitar 143,22 hari kerja atau 5,96 bulan pengerjaan. Jumlah sprint yang dibutuhkan adalah sekitar 29 sprint. Maka jumlah estimasi biaya yang diperlukan:

$$\begin{aligned}
\text{Estimasi biaya} &= 29 \times \text{Rp } 7.173.913,00 \\
&= \text{Rp } 208.043.478,00
\end{aligned}$$

Jadi total estimasi biaya pengembangan COMMJUNCTION berdasarkan Use Case Point yaitu sebesar **Rp 208.043.478,00**.

### 4.3 Perhitungan Function Point

Perhitungan menggunakan Function Point dilakukan pada kedua aplikasi dengan runut yang telah dijelaskan pada Bab 3 dan ketentuan pada Bab 2.

#### 4.3.1 Aplikasi 1 : CEKLOK.ID

Langkah awal perhitungan menggunakan Function Point adalah dengan menganalisis tipe fungsi dan elemennya, kemudian menghitung elemen-elemen tersebut, berikut adalah penjabaran tipe fungsi untuk aplikasi CEKLOK.

##### 4.3.1.1 Analisis Tipe Fungsi

Hasil analisis dari story point menghasilkan identifikasi lima tipe fungsi pengguna. Hasil yang didapatkan yaitu External Input (EI), External Output (EO), External Inquiry (EQ), Internal Logical Files (ILF) dan External Interface Files (EIF). Hasil tipe fungsi pengguna yang diidentifikasi dapat di lihat pada Tabel 4.15.

**Tabel 4.15 Function Type CEKLOK**

| External Input (EI)   | External Output (EO)             | External Inquiry (EQ)                  | Internal Logical File (ILF) | External Interface File (EIF) |
|-----------------------|----------------------------------|--|-----------------------------|-------------------------------|
| mengelola login       | menampilkan informasi rotasi     | menampilkan data atau daftar karyawan  | admin                       | Payment Confirmation          |
| mengelola hak akses   | menampilkan informasi peringatan | menampilkan data atau daftar cuti      | benefit                     | Approval Payment              |
| mengelola password    | menampilkan informasi broadcast  | menampilkan data atau daftar grade     | checklist                   | Transaction                   |
| mengelola departement | menampilkan informasi cuti       | menampilkan data atau daftar job-title | company                     | Verifikasi Account            |
| mengelola grade       | menampilkan informasi token      | menampilkan data atau daftar surat     | department                  |                               |
| mengelola job title   | mengirim notifikasi              | menampilkan data atau daftar tiket     | employee                    |                               |
| mengelola karyawan    | mengirim broadcast               | menampilkan data atau daftar reconcile | grade                       |                               |
| mengelola template    | menampilkan informasi perdin     | menampilkan data atau daftar perdin    | invoice                     |                               |

| External Input (EI)        | External Output (EO)                | External Inquiry (EQ)                     | Internal Logical File (ILF) | External Interface File (EIF) |
|----------------------------|-------------------------------------|---|-----------------------------|-------------------------------|
| mengelola benefit          | menampilkan informasi e-recruitment | menampilkan data atau daftar aset         | job_title                   |                               |
| mengelola cuti             | menampilkan informasi payment       | menampilkan data atau daftar maintenance  | modul                       |                               |
| mengelola lembur           | menampilkan informasi payroll       | menampilkan data atau daftar mpp          | payment                     |                               |
| mengelola surat            | menampilkan informasi benefit       | menampilkan data atau daftar lowongan     | payment method              |                               |
| mengelola perdin           | menampilkan informasi training      | menampilkan data atau daftar schedule     | payment type                |                               |
| mengelola reimbursement    |                                     | menampilkan data atau daftar user recruit | pricing                     |                               |
| mengelola tiket            |                                     | menampilkan data atau daftar payroll      | priviledge                  |                               |
| mengelola aset             |                                     | menampilkan data atau daftar slip         | role                        |                               |
| mengelola mpp              |                                     | menampilkan data atau daftar training     | status contract             |                               |
| mengelola recruitment      |                                     | menampilkan data atau daftar sop          | violation                   |                               |
| mengelola user recruitment |                                     |   | cuti                        |                               |
| mengelola training         |                                     |   | absence type                |                               |
| mengelola kuisisioner      |                                     |   | absence                     |                               |
| mengelola sop              |                                     |   | overtime                    |                               |
| mengelola payroll          |                                     |   | glossarium                  |                               |
| mengelola maintenance      |                                     |   | letter                      |                               |
| mengelola lowongan         |                                     |   | perdin                      |                               |
| mengelola schedule         |                                     |   | travel agent                |                               |
|                            |                                     |   | asset                       |                               |
|                            |                                     |   | man power                   |                               |
|                            |                                     |   | recruitment                 |                               |

| External Input (EI) | External Output (EO) | External Inquiry (EQ) | Internal Logical File (ILF) | External Interface File (EIF) |
|---------------------|----------------------|-----------------------|-----------------------------|-------------------------------|
|                     |                      |                       | assesment                   |                               |
|                     |                      |                       | payroll                     |                               |
|                     |                      |                       | training                    |                               |
|                     |                      |                       | approval program completion |                               |
|                     |                      |                       | sop                         |                               |
| 26                  | 13                   | 18                    | 34                          | 4                             |

#### 4.3.1.2 Unadjusted Function Point

Berdasarkan hasil analisis tipe fungsi pengguna pada tahap sebelumnya, dilakukan penentuan bobot kompleksitas Function Point dari masing-masing tipe fungsi pengguna dengan aturan Data Element Type (DET), Record Element Type (RET), File Type Referenced (FTR). Hasil perhitungan penentuan bobot kompleksitas Function Point dari masing-masing tipe fungsi pengguna dengan aturan DET, RET dan FTR yang diidentifikasi berdasarkan analisis pada Aplikasi CEKLOK, ditunjukkan pada Tabel 4.16 – 4.19.

**Tabel 4.16 Penentuan Bobot Tipe Fungsi Pengguna EI**

| Tipe Fungsi                | DET               | FTR             | Kom  |
|----------------------------|-------------------|-----------------|------|
| <b>External Input (EI)</b> | Data karyawan     | admin           | HIGH |
|                            | username&password | benefit         |      |
|                            | previledge        | department      |      |
|                            | data departement  | employee        |      |
|                            | data grade        | grade           |      |
|                            | data job title    | glossarium      |      |
|                            | data cuti         | job_title       |      |
|                            | data template     | payment         |      |
|                            | data benefit      | priviledge      |      |
|                            | data cuti         | status_contract |      |
|                            | data lembur       | cuti            |      |
|                            | data surat        | absence         |      |
|                            | data perdin       | overtime        |      |
|                            | data reimbursment | letter          |      |
|                            | data tiket        | perdin          |      |
|                            | data asset        | travel_agent    |      |
|                            | data mpp          | asset           |      |
| data user e-rec            | man_power         |                 |      |
| data lowongan              | recruitment       |                 |      |

| Tipe Fungsi | DET                 | FTR                 | Kom |
|-------------|---------------------|---------------------|-----|
|             | status              | assesment           |     |
|             | schedule            | training            |     |
|             | data tunjangan      | sop                 |     |
|             | data training       |                     |     |
|             | data quitionare     |                     |     |
|             | data SOP            |                     |     |
|             | <b>Total DET 25</b> | <b>Total FRT 22</b> |     |

**Tabel 4.17 Penentuan Bobot Tipe Fungsi Pengguna EO**

| Tipe Fungsi                 | DET                 | FTR                 | Kom  |
|-----------------------------|---------------------|---------------------|------|
| <b>External Output (EO)</b> | data karyawan       | employee            | HIGH |
|                             | data cuti           | cuti                |      |
|                             | info broadcast      | letter              |      |
|                             | data token          | travel_agent        |      |
|                             | data surat          | recruitment         |      |
|                             | data tiket          | payroll             |      |
|                             | data user e-rec     | payment             |      |
|                             | status              | benefit             |      |
|                             | payroll             | perdin              |      |
|                             | payment             | training            |      |
|                             | data tunjangan      |                     |      |
|                             | data tiket          |                     |      |
|                             | data training       |                     |      |
|                             | <b>Total DET 13</b> | <b>Total FRT 10</b> |      |

**Tabel 4.18 Penentuan Bobot Tipe Fungsi Pengguna EQ**

| Tipe Fungsi                  | DET                | FTR                 | Kom  |
|------------------------------|--------------------|---------------------|------|
| <b>External Inquiry (EQ)</b> | data karyawan      | employee            | HIGH |
|                              | password karyawan  | cuti                |      |
|                              | data cuti          | letter              |      |
|                              | data surat         | perdin              |      |
|                              | data perdin        | training            |      |
|                              | data asset         | asset               |      |
|                              | data mpp           | man_power           |      |
|                              | data e-recruitment | recruitment         |      |
|                              | schedule           | sop                 |      |
|                              | data reconcile     |                     |      |
|                              | data training      |                     |      |
|                              | data sop           |                     |      |
|                              |                    | <b>Total DET 12</b> |      |



**Tabel 4.19 Penentuan Bobot Tipe Fungsi Pengguna ILF**

| Tipe Fungsi                        | DET            | RET                         | Kom  |
|------------------------------------|----------------|-----------------------------|------|
| <b>Internal Logical File (ILF)</b> | id_admin       | admin                       | HIGH |
|                                    | company_id     | benefit                     |      |
|                                    | company_code   | checklist                   |      |
|                                    | user_name      | company                     |      |
|                                    | password       | department                  |      |
|                                    | old_password   | employee                    |      |
|                                    | name           | grade                       |      |
|                                    | gender         | invoice                     |      |
|                                    | pob            | job_title                   |      |
|                                    | dob            | modul                       |      |
|                                    | address        | payment                     |      |
|                                    | kelurahan      | payment_method              |      |
|                                    | kecamatan      | payment_type                |      |
|                                    | city           | pricing                     |      |
|                                    | province       | priviledge                  |      |
|                                    | zipcode        | role                        |      |
|                                    | msisdn         | status_contract             |      |
|                                    | tlp            | violation                   |      |
|                                    | email          | cuti                        |      |
|                                    | status         | absence_type                |      |
|                                    | created_at     | absence                     |      |
|                                    | updated_at     | overtime                    |      |
|                                    | action_by      | glossarium                  |      |
|                                    | ip_address     | letter                      |      |
|                                    | id_benefit     | perdin                      |      |
|                                    | id_checklist   | travel_agent                |      |
|                                    | code_checklist | asset                       |      |
|                                    | id_company     | man_power                   |      |
|                                    | pricing_id     | recruitment                 |      |
|                                    | invoice_id     | assesment                   |      |
|                                    | company_code   | payroll                     |      |
|                                    | name           | training                    |      |
|                                    | address        | approval-program-completion |      |
|                                    | kelurahan      | sop                         |      |
|                                    | kecamatan      | admin                       |      |
|                                    | city           |                             |      |
|                                    | province       |                             |      |
|                                    | zipcode        |                             |      |
|                                    | msisdn         |                             |      |
|                                    | tlp            |                             |      |
| email                              |                |                             |      |
| payment_status                     |                |                             |      |

| Tipe Fungsi | DET                  | RET | Kom |
|-------------|----------------------|-----|-----|
|             | payment_at timestamp |     |     |
|             | id_department        |     |     |
|             | department_code      |     |     |
|             | id_employee          |     |     |
|             | priviledge_id        |     |     |
|             | role_id              |     |     |
|             | department_id        |     |     |
|             | department_upload    |     |     |
|             | job_title_id         |     |     |
|             | job_title_upload     |     |     |
|             | grade_id             |     |     |
|             | grade_upload         |     |     |
|             | benefit_id           |     |     |
|             | benefit_upload       |     |     |
|             | company_code         |     |     |
|             | employee_code        |     |     |
|             | user_name            |     |     |
|             | id_grade             |     |     |
|             | id_invoice           |     |     |
|             | id_job_title         |     |     |
|             | id_modul             |     |     |
|             | id_payment           |     |     |
|             | invoice              |     |     |
|             | id_payment_method    |     |     |
|             | id_payment_type      |     |     |
|             | id_pricing           |     |     |
|             | id_priviledge        |     |     |
|             | id_role              |     |     |
|             | id_status_contract   |     |     |
|             | code_status_contract |     |     |
|             | name                 |     |     |
|             | description text     |     |     |
|             | status               |     |     |
|             | leave_setting        |     |     |
|             | id_violation         |     |     |
|             | employee_id          |     |     |
|             | code_violation       |     |     |
|             | id_cuti              |     |     |
|             | user_id              |     |     |
|             | code_cuti            |     |     |
|             | id_absence_type      |     |     |
|             | code_absence_type    |     |     |
|             | id_absence           |     |     |
|             | id_overtime          |     |     |

| Tipe Fungsi | DET                             | RET                 | Kom |
|-------------|---------------------------------|---------------------|-----|
|             | code_overtime                   |                     |     |
|             | id_glossarium                   |                     |     |
|             | code_glossarium                 |                     |     |
|             | id_letter                       |                     |     |
|             | code_letter                     |                     |     |
|             | id_perdin                       |                     |     |
|             | id_travel_agent                 |                     |     |
|             | id_asset                        |                     |     |
|             | id_man_power                    |                     |     |
|             | id_recruitment                  |                     |     |
|             | id assesment                    |                     |     |
|             | id_payroll                      |                     |     |
|             | id_training                     |                     |     |
|             | id_approval-program-completion  |                     |     |
|             | id_sop                          |                     |     |
|             | code_travel_agent               |                     |     |
|             | code_asset                      |                     |     |
|             | code_man_power                  |                     |     |
|             | code_recruitment                |                     |     |
|             | code assesment                  |                     |     |
|             | code_payroll                    |                     |     |
|             | code_training                   |                     |     |
|             | code aproval-program-completion |                     |     |
|             | code_sop                        |                     |     |
|             | <b>Total DET 74</b>             | <b>Total RET 34</b> |     |

**Tabel 4.20 Penentuan Bobot Tipe Fungsi Pengguna EIF**

| Tipe Fungsi                          | DET               | RET            | Kom |
|--------------------------------------|-------------------|----------------|-----|
| <b>External Interface File (EIF)</b> | id_payment        | employee       | LOW |
|                                      | company_id        | company        |     |
|                                      | invoice           | payment        |     |
|                                      | from_bank_name    | payment_method |     |
|                                      | from_rek_name     | payment_type   |     |
|                                      | from_rek_no       |                |     |
|                                      | id_payment_method |                |     |
|                                      | payment_type_id   |                |     |
|                                      | to_bank_name      |                |     |
|                                      | to_rek_name       |                |     |
|                                      | to_rek_no         |                |     |
|                                      | id_payment_type   |                |     |

| Tipe Fungsi | DET                 | RET                | Kom |
|-------------|---------------------|--------------------|-----|
|             | id_employee         |                    |     |
|             | user_name           |                    |     |
|             | password            |                    |     |
|             | old_password        |                    |     |
|             | <b>Total DET 16</b> | <b>Total RET 5</b> |     |

Hasil perhitungan bobot kompleksitas Function Point pada Tabel 4.16 terhadap tipe fungsi EI mendapatkan bobot kompleksitas High, bobot ini diperoleh berdasarkan total perhitungan DET dan FTR dengan aturan sesuai tabel 2.6. Pada Tabel 4.17 dan Tabel 4.18 menunjukkan tipe fungsi EO dan EQ mendapatkan bobot kompleksitas High, bobot ini diperoleh berdasarkan total perhitungan DET dan FTR dengan aturan sesuai tabel 2.7. Tabel 4.19 menunjukkan tipe fungsi ILF mendapatkan bobot kompleksitas High dan Tabel 4.20 menunjukkan tipe fungsi EIF mendapatkan bobot kompleksitas Low, bobot ini diperoleh berdasarkan total perhitungan DET dan RET dengan aturan sesuai tabel 2.8. Dalam melakukan perhitungan Un-adjusted Funtion Point (UFP) yang berdasarkan hasil tabel sebelumnya akan dijelaskan pada Tabel 4.21 berikut.

**Tabel 4.21 Perhitungan UFP**

| Tipe         | Jumlah | Perhitungan Bobot Kompleksitas UFP |     |     | Bobot Kompleksitas | Nilai Kompleksitas | UFP        |
|--------------|--------|------------------------------------|-----|-----|--------------------|--------------------|------------|
|              |        | DET                                | RET | FTR |                    |                    |            |
| EI           | 26     | 25                                 | -   | 22  | HIGH               | 6                  | 156        |
| EO           | 13     | 13                                 | -   | 10  | HIGH               | 7                  | 91         |
| EQ           | 18     | 12                                 | -   | 9   | HIGH               | 6                  | 108        |
| ILF          | 34     | 74                                 | 34  | -   | HIGH               | 15                 | 510        |
| EIF          | 4      | 16                                 | 5   | -   | LOW                | 5                  | 20         |
| <b>Total</b> |        |                                    |     |     |                    |                    | <b>885</b> |

Hasil Un-adjusted Funtion Point (UFP) pada aplikasi CEKLOK adalah sebesar **885** berdasarkan perhitungan yang dicantumkan pada Tabel 4.21.

#### **4.3.1.3 Technical Complexity Factor (TCF)**

Dalam menentukan nilai Technical Complexity Factor (TCF) dilakukan dengan cara mengisi lembar penilaian berdasarkan 14 karakteristik umum sistem sesuai pada Bab 2. Penilaian dilakukan dengan cara memberi skor pada masing-

masing karakteristik dengan skor nol (0) sampai lima (5) sesuai dengan aturan Tabel 2.10. Lembar penilaian disebar kepada responden. Hasil lembar penilaian Technical Complexity Factor (TCF) dapat dilihat pada Tabel 4.22.

**Tabel 4.22 Hasil Technical Complexity Factors**

|     | Technical Complexity Factors | Skor        |
|-----|------------------------------|-------------|
| T1  | Data Communications          | 4,00        |
| T2  | Distributed Data Processing  | 3,75        |
| T3  | Performance                  | 3,63        |
| T4  | Heavily Used Configuration   | 2,88        |
| T5  | Transaction Rate             | 3,50        |
| T6  | Online Data Entry            | 4,75        |
| T7  | End-User Efficiency          | 4,25        |
| T8  | Online Update                | 2,75        |
| T9  | Complex Processing           | 3,50        |
| T10 | Reusability                  | 4,00        |
| T11 | Installation Ease            | 2,75        |
| T12 | Operation Ease               | 2,63        |
| T13 | Multiple Sites               | 3,88        |
| T14 | Facilitate Change            | 2,75        |
|     | Total TF                     | 49,00       |
|     | <b>0,65 + (0,01 * 49)</b>    | <b>1,14</b> |

Untuk mendapatkan nilai TCF digunakan perhitungan menggunakan rumus persamaan 2.6.

$$\begin{aligned}
 \text{TCF} &= 0.65 + (0.01 \times \text{RCAF}) \\
 &= 0.65 + (0.01 \times 49) \\
 &= 1.14
 \end{aligned}$$

#### **4.3.1.4 Function Point**

Untuk menentukan nilai function point pada aplikasi CEKLOK dilakukan perkalian dari hasil UFP dengan TCF menggunakan rumus persamaan 2.7

$$\begin{aligned}
 \text{Function Point} &= \text{UFP} \times \text{TCF} \\
 &= 885 \times 1.14 \\
 &= 1008,9
 \end{aligned}$$

Nilai Function Point untuk aplikasi CEKLOK adalah **1008,9**

#### **4.3.1.5 Estimasi Usaha**

Setelah nilai Function Point perangkat lunak A didapatkan, dilakukan perhitungan estimasi Effort menggunakan rumus persamaan 2.8 dan persamaan 2.9. Dimana FP dan f adalah nilai function point dan j adalah eksponen jones didapatkan dari Tabel 2.12 yang didapatkan bahwa aplikasi CEKLOK merupakan Shrink-wrap karena bersifat commercially dan Average karena memiliki kemampuan rata-rata dalam menangani pengembangan perangkat lunak.

$$\begin{aligned}\text{Effort} &= FP^{3*j} / 27 \\ &= 1008,9^{3*0.42} / 27 \\ &= 226 \text{ Person-months} \\ s &= f^j \\ &= 1008,9^{0.42} \\ &= 18,26 \text{ Months}\end{aligned}$$

Dari hasil perhitungan didapatkan bahwa aplikasi CEKLOK membutuhkan 226 Person-months dan sekitar 18 bulan pengerjaan.

#### **4.3.1.6 Estimasi Biaya**

Langkah selanjutnya dalam perhitungan total biaya perangkat lunak A ialah merubah nilai effort dari bulan menjadi satuan hari, dalam penelitian ini jumlah hari pada satu bulan kerja ialah 20 hari kerja. Dilakukan perkalian antara nilai schedule dengan jumlah hari kerja dalam satu bulan yaitu 20 hari kerja yang menghasilkan 365 hari kerja. Mengacu pada penelitian Kassem Sales (2011) dilakukan pembagian nilai Effort kedalam tiga kelompok aktivitas. Dalam penentuan nilai Person dilakukan dengan mengalikan %Effort dengan nilai Effort dan dalam penentuan nilai Days dilakukan dengan mengalikan %Effort dengan schedule yang telah dirubah menjadi satuan hari yaitu 365 hari.

Standar gaji yang digunakan untuk tiap aktivitas mengacu pada Man Days PT. Vascomm Solusi Teknologi sesuai kebutuhan Aplikasi CEKLOK. Dalam menentukan posisi yang dibutuhkan pada tiap fasenya dilakukan berdasarkan teori

yang terlampir pada Bab 2. Fase yang dihitung adalah fase pertama yaitu Software Development sesuai dengan batasan penelitian dan berdasarkan perhitungan pada Lampiran Perhitungan *Sprint Rate* dan *Day Rate* CEKLOK.

**Tabel 4.23 Estimasi Biaya CEKLOK**

| Kelompok Aktivitas                 | %Effort | Person | Days | Role       | Cost/Day   | Total                 |
|------------------------------------|---------|--------|------|------------|------------|-----------------------|
| <b>Software Development</b>        |         |        |      |            |            |                       |
| <b>Requirement</b>                 | 7,5%    | 17     | 27   | SA/PO      | Rp 217.391 | Rp 100.808.038        |
| <b>Spesification</b>               | 7,5%    | 17     | 27   | SA/PO      | Rp 217.391 | Rp 100.808.038        |
| <b>Design</b>                      | 10%     | 23     | 37   | SA & UI/UX | Rp 173.913 | Rp 143.371.597        |
| <b>Implementation</b>              | 10%     | 23     | 37   | SE         | Rp 195.652 | Rp 161.292.944        |
| <b>Acceptance &amp; deployment</b> | 7,5%    | 17     | 27   | SE         | Rp 195.652 | Rp 90.727.281         |
| <b>Total</b>                       | 42,5%   | 96     | 155  |            |            | <b>Rp 597.007.898</b> |

Hasil perhitungan estimasi biaya pembuatan pada aplikasi CEKLOK menggunakan adalah sebesar **Rp. 597.007.898** seperti pada Tabel 4.23.

#### 4.3.2 Aplikasi 2 : COMMJUNCTION

Langkah awal perhitungan menggunakan Function Point adalah dengan menganalisis tipe fungsi dan elemennya, kemudian menghitung elemen-elemen tersebut, berikut adalah penjabaran tipe fungsi untuk aplikasi COMMJUNCTION.

##### 4.3.2.1 Analisis Tipe Fungsi

Hasil analisis dari story point menghasilkan identifikasi lima tipe fungsi pengguna. Hasil yang didapatkan yaitu External Input (EI), External Output (EO), External Inquiry (EQ), Internal Logical Files (ILF) dan External Interface Files (EIF). Hasil tipe fungsi pengguna yang diidentifikasi dapat di lihat pada Tabel 4.24.

**Tabel 4.24 Function Type COMMJUNCTION**

| External Input (EI) | External Output (EO)            | External Inquiry (EQ)     | Internal Logical File (ILF) | External Interface File (EIF)     |
|---------------------|---------------------------------|---------------------------|-----------------------------|-----------------------------------|
| mengelola login     | menampilkan informasi community | menampilkan data activity | priviledge                  | Payment Process (on some pricing) |

| External Input (EI)         | External Output (EO)                 | External Inquiry (EQ)        | Internal Logical File (ILF) | External Interface File (EIF) |
|-----------------------------|--------------------------------------|------------------------------|-----------------------------|-------------------------------|
| mengelola registrasi        | menampilkan informasi subscriber     | menampilkan data transaction | transaction type            | Payment Verification          |
| mengelola halaman portal    | menampilkan informasi transaction    | menampilkan data request     | jenis comm                  | Verify payment                |
| mengelola password          | menampilkan informasi modul activity | menampilkan data subscriber  | cust jenis comm             | Email Blast                   |
| mengelola subscriber        | menampilkan informasi payment        | menampilkan data modul       | community                   | Email Activation              |
| mengelola modul             | menampilkan informasi news           | menampilkan data reconcile   | users                       | Google SSO                    |
| mengelola payment           | menampilkan informasi Visit Count    | menampilkan data payment     | comm admin                  | Facebook SSO                  |
| mengelola metode pembayaran | menampilkan informasi event          | menampilkan data user        | payment method              | Payment Process               |
| mengelola user              | menampilkan informasi friends        | menampilkan data pricing     | payment type                | Payment Confirmation          |
| mengelola previlege         | menampilkan informasi attended       | menampilkan data module      | pricing                     | Confirm Payment               |
| mengelola tipe user         |                                      | menampilkan data API         | feature type                | Chat Friend (Whatsapp)        |
| mengelola profile           |                                      | menampilkan data news        | feature                     |                               |
| mengelola pricing           |                                      | menampilkan data event       | module endpoint             |                               |
| mengelola data              |                                      | menampilkan data attendance  | payment grand               |                               |
| mengelola custom            |                                      | menampilkan data friends     | payment invoice             |                               |
| mengelola API               |                                      |                              | comm feature                |                               |
| mengelola news              |                                      |                              | payment verification        |                               |
| mengelola friends           |                                      |                              | bank                        |                               |
| mengelola event             |                                      |                              | cust portal login           |                               |
| mengelola Enable / Disable  |                                      |                              | routes                      |                               |



| External Input (EI)        | External Output (EO) | External Inquiry (EQ) | Internal Logical File (ILF) | External Interface File (EIF) |
|----------------------------|----------------------|-----------------------|-----------------------------|-------------------------------|
| mengelola dashboard        |                      |                       | comm payment grand          |                               |
| mengelola attendance event |                      |                       | comm payment invoice        |                               |
| mengelola fitur            |                      |                       | comm payment type           |                               |
|                            |                      |                       | comm payment method         |                               |
|                            |                      |                       | membership                  |                               |
|                            |                      |                       | membership feature          |                               |
|                            |                      |                       | comm subscriber             |                               |
|                            |                      |                       | cust sub param              |                               |
|                            |                      |                       | cust sub paramval           |                               |
| <b>23</b>                  | <b>10</b>            | <b>15</b>             | <b>29</b>                   | <b>11</b>                     |

#### 4.3.2.2 Unadjusted Function Point

Berdasarkan hasil analisis tipe fungsi pengguna pada tahap sebelumnya, dilakukan penentuan bobot kompleksitas Function Point dari masing-masing tipe fungsi pengguna dengan aturan Data Element Type (DET), Record Element Type (RET), File Type Referenced (FTR) yang ditunjukkan pada tabel , tabel dan tabel . Hasil perhitungan penentuan bobot kompleksitas Function Point dari masing-masing tipe fungsi pengguna dengan aturan DET, RET dan FTR yang diidentifikasi berdasarkan analisis pada Aplikasi COMMJUNCTION, ditunjukkan pada Tabel 4.16 – 4.19.

**Tabel 4.25 Penentuan Bobot Tipe Fungsi Pengguna EI**

| Tipe Fungsi                | DET                 | FTR              | Kom  |
|----------------------------|---------------------|------------------|------|
| <b>External Input (EI)</b> | Data user           | priviledge       | HIGH |
|                            | username & password | transaction_type |      |
|                            | previdence          | jenis_comm       |      |
|                            | data subscriber     | cust_jenis_comm  |      |
|                            | data custom         | community        |      |
|                            | data payment        | users            |      |
|                            | data tipe payment   | comm_admin       |      |
|                            | data pricing        | payment_method   |      |
|                            | data API            | payment_type     |      |
|                            | data news           | pricing          |      |
| data friends               | feature_type        |                  |      |

| Tipe Fungsi | DET                 | FTR                 | Kom |
|-------------|---------------------|---------------------|-----|
|             | data evet           | feature             |     |
|             | data komunitas      | module_endpoint     |     |
|             | data fitur          | payment_grand       |     |
|             | data modul          | comm_feature        |     |
|             |                     | cust_portal_login   |     |
|             |                     | routes              |     |
|             |                     | membership          |     |
|             |                     | membership_feature  |     |
|             |                     | comm_subscriber     |     |
|             | <b>Total DET 15</b> | <b>Total FRT 20</b> |     |

**Tabel 4.26 Penentuan Bobot Tipe Fungsi Pengguna EO**

| Tipe Fungsi                 | DET                 | FTR                 | Kom  |
|-----------------------------|---------------------|---------------------|------|
| <b>External Output (EO)</b> | Data user           | transaction_type    | HIGH |
|                             | data subscriber     | jenis_comm          |      |
|                             | data komunitas      | cust_jenis_comm     |      |
|                             | data payment        | community           |      |
|                             | data transaksi      | users               |      |
|                             | data modul          | comm_admin          |      |
|                             | data news           | feature             |      |
|                             | data friends        | module_endpoint     |      |
|                             | data event          | payment_grand       |      |
|                             | data attendance     | comm_subscriber     |      |
|                             | <b>Total DET 10</b> | <b>Total FRT 10</b> |      |

**Tabel 4.27 Penentuan Bobot Tipe Fungsi Pengguna EQ**

| Tipe Fungsi                  | DET                 | FTR                  | Kom  |
|------------------------------|---------------------|----------------------|------|
| <b>External Inquiry (EQ)</b> | data aktivitas      | transaction_type     | HIGH |
|                              | data transaksi      | cust_jenis_comm      |      |
|                              | data subscriber     | community            |      |
|                              | data modul          | users                |      |
|                              | data reconcile      | pricing              |      |
|                              | data payment        | module_endpoint      |      |
|                              | data user           | payment_grand        |      |
|                              | data pricing        | comm_feature         |      |
|                              | data modul          | payment_verification |      |
|                              | data API            | cust_portal_login    |      |
|                              | data news           | routes               |      |
|                              | data friends        | membership           |      |
|                              | data event          | comm_subscriber      |      |
|                              | data attendance     |                      |      |
| <b>Total DET 14</b>          | <b>Total FRT 17</b> |                      |      |

**Tabel 4.28 Penentuan Bobot Tipe Fungsi Pengguna ILF**

| Tipe Fungsi                        | DET                     | RET                  | Kom  |
|------------------------------------|-------------------------|----------------------|------|
| <b>Internal Logical File (ILF)</b> | id_privilege            | privilege            | HIGH |
|                                    | id_transaction_type     | transaction_type     |      |
|                                    | id_jenis_comm           | jenis_comm           |      |
|                                    | id_cust_jenis_comm      | cust_jenis_comm      |      |
|                                    | id_community            | community            |      |
|                                    | id_users                | users                |      |
|                                    | id_comm_admin           | comm_admin           |      |
|                                    | id_payment_method       | payment_method       |      |
|                                    | id_payment_type         | payment_type         |      |
|                                    | id_pricing              | pricing              |      |
|                                    | id_feature_type         | feature_type         |      |
|                                    | id_feature              | feature              |      |
|                                    | id_module_endpoint      | module_endpoint      |      |
|                                    | id_payment_grand        | payment_grand        |      |
|                                    | id_payment_invoice      | payment_invoice      |      |
|                                    | id_comm_feature         | comm_feature         |      |
|                                    | id_payment_verification | payment_verification |      |
|                                    | id_bank                 | bank                 |      |
|                                    | id_cust_portal_login    | cust_portal_login    |      |
|                                    | id_routes               | routes               |      |
|                                    | id_comm_payment_grand   | comm_payment_grand   |      |
|                                    | id_comm_payment_invoice | comm_payment_invoice |      |
|                                    | id_comm_payment_type    | comm_payment_type    |      |
|                                    | id_comm_payment_method  | comm_payment_method  |      |
|                                    | id_membership           | membership           |      |
|                                    | id_membership_feature   | membership_feature   |      |
|                                    | id_comm_subscriber      | comm_subscriber      |      |
|                                    | id_cust_sub_param       | cust_sub_param       |      |
|                                    | id_cust_sub_paramval    | cust_sub_paramval    |      |
|                                    | logo                    |                      |      |
|                                    | password                |                      |      |
|                                    | sso_id                  |                      |      |
|                                    | active                  |                      |      |
|                                    | status                  |                      |      |
| <b>Total DET 34</b>                | <b>Total FRT 29</b>     |                      |      |

**Tabel 4.29 Penentuan Bobot Tipe Fungsi Pengguna EIF**

| Tipe Fungsi                          | DET                     | RET            | Kom |
|--------------------------------------|-------------------------|----------------|-----|
| <b>External Interface File (EIF)</b> | id_payment_method       | email          | LOW |
|                                      | id_payment_type         | google         |     |
|                                      | id_pricing              | facebook       |     |
|                                      | id_payment_grand        | whatsapp       |     |
|                                      | id_payment_invoice      | payment_method |     |
|                                      | id_payment_verification | payment_type   |     |

| Tipe Fungsi | DET                     | RET                  | Kom |
|-------------|-------------------------|----------------------|-----|
|             | id_comm_payment_grand   | pricing              |     |
|             | id_comm_payment_invoice | payment_grand        |     |
|             | id_comm_payment_type    | payment_invoice      |     |
|             | id_comm_payment_method  | payment_verification |     |
|             |                         | comm_payment_grand   |     |
|             |                         | comm_payment_invoice |     |
|             |                         | comm_payment_type    |     |
|             |                         | comm_payment_method  |     |
|             | <b>Total DET 10</b>     | <b>Total RET 14</b>  |     |

Hasil perhitungan bobot kompleksitas Function Point pada Tabel 4.25 terhadap tipe fungsi EI mendapatkan bobot kompleksitas High, bobot ini diperoleh berdasarkan total perhitungan DET dan FTR dengan aturan sesuai tabel 2.6. Pada Tabel 4.26 dan Tabel 4.27 menunjukkan tipe fungsi EO dan EQ mendapatkan bobot kompleksitas High, bobot ini diperoleh berdasarkan total perhitungan DET dan FTR dengan aturan sesuai tabel 2.7. Tabel 4.28 menunjukkan tipe fungsi ILF mendapatkan bobot kompleksitas High dan Tabel 4.29 menunjukkan tipe fungsi EIF mendapatkan bobot kompleksitas Average, bobot ini diperoleh berdasarkan total perhitungan DET dan RET dengan aturan sesuai tabel 2.8. Dalam melakukan perhitungan Un-adjusted Function Point (UFP) yang berdasarkan hasil tabel sebelumnya akan dijelaskan pada Tabel 4.30 berikut.

**Tabel 4.30 Perhitungan UFP**

| Tipe         | Jumlah | Perhitungan Bobot Kompleksitas UFP |     |     | Bobot Kompleksitas | Nilai Kompleksitas | UFP        |
|--------------|--------|------------------------------------|-----|-----|--------------------|--------------------|------------|
|              |        | DET                                | RET | FTR |                    |                    |            |
| EI           | 23     | 15                                 |     | 20  | HIGH               | 6                  | 138        |
| EO           | 10     | 10                                 |     | 10  | HIGH               | 7                  | 70         |
| EQ           | 15     | 14                                 |     | 13  | HIGH               | 6                  | 90         |
| ILF          | 29     | 34                                 | 29  |     | HIGH               | 15                 | 435        |
| EIF          | 11     | 10                                 | 14  |     | AVERAGE            | 7                  | 77         |
| <b>Total</b> |        |                                    |     |     |                    |                    | <b>810</b> |

Hasil Un-adjusted Function Point (UFP) pada aplikasi CEKLOK adalah sebesar **810** berdasarkan perhitungan yang dicantumkan pada Tabel 4.21.

### 4.3.2.3 *Technical Complexity Factor (TCF)*

Dalam menentukan nilai Technical Complexity Factor (TCF) dilakukan dengan cara mengisi lembar penilaian berdasarkan 14 karakteristik umum sistem. Penilaian dilakukan dengan cara memberi skor pada masing-masing karakteristik dengan skor nol (0) sampai lima (5) sesuai dengan aturan Tabel 2.10. Lembar penilaian disebar kepada responden. Hasil lembar penilaian Technical Complexity Factor (TCF) dapat dilihat pada Tabel 4.31.

**Tabel 4.31 Hasil Technical Complexity Factors**

|     | Technical Complexity Factors | Skor  |
|-----|------------------------------|-------|
| T1  | Data Communications          | 4,00  |
| T2  | Distributed Data Processing  | 4,00  |
| T3  | Performance                  | 4,33  |
| T4  | Heavily Used Configuration   | 3,33  |
| T5  | Transaction Rate             | 3,50  |
| T6  | Online Data Entry            | 5,00  |
| T7  | End-User Efficiency          | 4,17  |
| T8  | Online Update                | 3,17  |
| T9  | Complex Processing           | 3,17  |
| T10 | Reusability                  | 4,67  |
| T11 | Installation Ease            | 2,50  |
| T12 | Operation Ease               | 3,33  |
| T13 | Multiple Sites               | 4,00  |
| T14 | Facilitate Change            | 3,33  |
|     | Total TF                     | 52,50 |
|     | <b>0,65 + (0,01 * 52,5)</b>  | 1,18  |

Untuk mendapatkan nilai TCF digunakan perhitungan menggunakan rumus persamaan 2.6.

$$\begin{aligned} \text{TCF} &= 0.65 + (0.01 \times \text{RCAF}) \\ &= 0.65 + (0.01 \times 52,5) \\ &= 1.18 \end{aligned}$$

#### 4.3.2.4 *Function Point*

Untuk menentukan nilai function point pada aplikasi COMMJUNCTION dilakukan perkalian dari hasil UFP dengan TCF menggunakan rumus persamaan 2.7

$$\begin{aligned}\text{Function Point} &= \text{UFP} \times \text{TCF} \\ &= 810 \times 1.18 \\ &= 951,75\end{aligned}$$

Nilai Function Point untuk aplikasi CEKLOK adalah **951,75**

#### 4.3.2.5 *Estimasi Usaha*

Setelah nilai Function Point perangkat lunak A didapatkan, dilakukan perhitungan estimasi Effort menggunakan rumus persamaan 2.8 dan persamaan 2.9. Dimana FP dan f adalah nilai function point dan j adalah eksponen jones didapatkan dari Tabel 2.12 yang didapatkan bahwa aplikasi CEKLOK merupakan Shrink-wrap karena bersifat commercially dan Average karena memiliki kemampuan rata-rata dalam menangani pengembangan perangkat lunak.

$$\begin{aligned}\text{Effort} &= FP^{3*j} / 27 \\ &= 951,75^{0.42} / 27 \\ &= 210 \text{ Person-months} \\ \mathbf{s} &= f^j \\ &= 951,75^{0.42} \\ &= 17,82 \text{ Months}\end{aligned}$$

Dari hasil perhitungan didapatkan bahwa aplikasi CEKLOK membutuhkan 210 Person-months dan sekitar 18 bulan pengerjaan.

#### 4.3.2.6 *Estimasi Biaya*

Langkah selanjutnya dalam perhitungan total biaya perangkat lunak A ialah merubah nilai effort dari bulan menjadi satuan hari, dalam penelitian ini jumlah hari pada satu bulan kerja ialah 20 hari kerja. Dilakukan perkalian antara nilai schedule dengan jumlah hari kerja dalam satu bulan yaitu 20 hari kerja yang menghasilkan

365 hari kerja. Mengacu pada penelitian Kassem Sales (2011) dilakukan pembagian nilai Effort kedalam tiga kelompok aktivitas. Dalam penentuan nilai Person dilakukan dengan mengalikan %Effort dengan nilai Effort dan dalam penentuan nilai Days dilakukan dengan mengalikan %Effort dengan schedule yang telah dirubah menjadi satuan hari yaitu 365 hari.

Standar gaji yang digunakan untuk tiap aktivitas mengacu pada Man Days PT. Vascomm Solusi Teknologi sesuai kebutuhan Aplikasi COMMJUNCTION. Dalam menentukan posisi yang dibutuhkan pada tiap fasenya dilakukan berdasarkan teori yang terlampir pada Bab 2. Fase yang dihitung adalah fase pertama yaitu Software Development sesuai dengan batasan penelitian dan berdasarkan perhitungan pada Lampiran Perhitungan *Sprint Rate* dan *Day Rate* COMMJUNCTION.

**Tabel 4.32 Estimasi Biaya COMMJUNCTION**

| Kelompok Aktivitas                 | %Effort | Person | Days       | Role      | Cost/Day              |
|------------------------------------|---------|--------|------------|-----------|-----------------------|
| <b>Requirement</b>                 | 16      | 27     | SA/PO      | Rp217.391 | Rp 91.400.541         |
| <b>Spesification</b>               | 16      | 27     | SA/PO      | Rp217.391 | Rp 91.400.541         |
| <b>Design</b>                      | 21      | 36     | SA & UI/UX | Rp173.913 | Rp 129.992.030        |
| <b>Implementation</b>              | 21      | 36     | SE         | Rp195.652 | Rp 146.240.940        |
| <b>Acceptance &amp; deployment</b> | 16      | 27     | SE         | Rp195.652 | Rp 82.260.529         |
| <b>Total</b>                       | 89      | 151    |            |           | <b>Rp 541.294.581</b> |

Hasil perhitungan estimasi biaya pembuatan pada aplikasi COMMJUNCTION menggunakan adalah sebesar **Rp. 541.294.581** seperti pada Tabel 4.32.





## BAB 5

### PEMBAHASAN & EVALUASI

#### 5.1 Analisis Perbandingan Hasil Durasi Pengerjaan dan Biaya Pengembangan antar Metode

Hasil analisis dari penelitian ini yaitu perbandingan antara biaya Realita dengan Estimasi, Use Case Point dan Function Point. Perbandingan ini berisi data berupa durasi pengerjaan dan biaya total yang dibutuhkan untuk pengembangan CEKLOK dan COMMJUNCTION. Membandingkan hasil perhitungan antara durasi dan biaya baik menggunakan Realita, Estimasi, Use Case Point dan Function Point pada Tabel 5.1 untuk CEKLOK dan Tabel 5.2 untuk COMMJUNCTION.

**Tabel 5.1 Perbandingan Hasil Metode Aplikasi CEKLOK**

|               | Realita        | Estimasi       | Use Case Point | Function Point |
|---------------|----------------|----------------|----------------|----------------|
| <b>Durasi</b> | 36 sprint      | 30 sprint      | 39 sprint      | 76 sprint      |
| <b>Biaya</b>  | Rp 320.869.565 | Rp 267.391.304 | Rp 347.608.696 | Rp 597.007.898 |

**Tabel 5.2 Perbandingan Hasil Metode Aplikasi COMMJUNCTION**

|               | Realita        | Estimasi       | Use Case Point | Function Point  |
|---------------|----------------|----------------|----------------|-----------------|
| <b>Durasi</b> | 28 sprint      | 24 sprint      | 29 sprint      | 72 sprint       |
| <b>Biaya</b>  | Rp 200.869.565 | Rp 172.173.913 | Rp 208.043.478 | Rp. 541.294.581 |

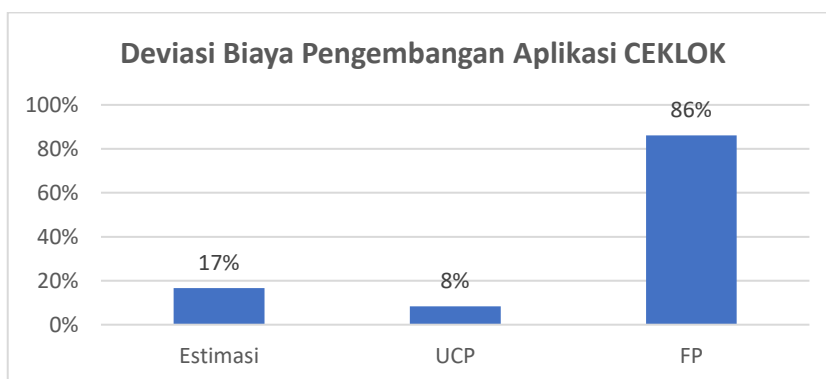
Terlihat pada Tabel 5.1 dan Tabel 5.2 menunjukkan kesenjangan antara realitas biaya, estimasi, Use Case Point dan Function Point. Pada perhitungan untuk aplikasi CEKLOK maupun COMMJUNCTION metode Use Case Point memiliki selisih biaya dengan nominal yang lebih kecil dibandingkan dengan biaya estimasi, sedangkan metode Function Point membutuhkan biaya dengan nominal yang lebih besar dibandingkan dengan biaya estimasi. Selisih biaya tersebut terlihat pada Tabel 5.3 berikut.

**Tabel 5.3 Selisih Biaya antar Metode**

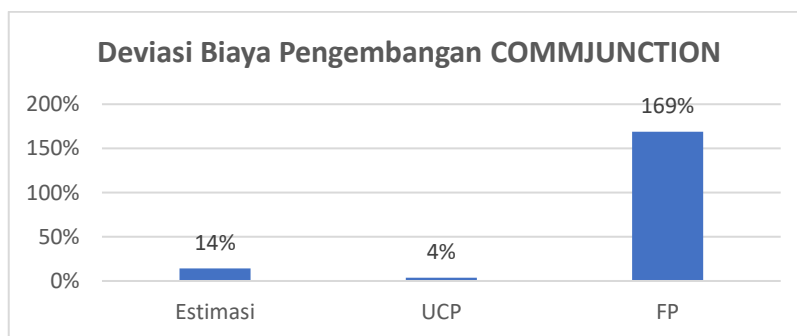
|                     | Estimasi      | Use Case Point | Function Point |
|---------------------|---------------|----------------|----------------|
| <b>CEKLOK</b>       | Rp 53.478.261 | Rp 26.739.131  | Rp 276.138.333 |
| <b>COMMJUNCTION</b> | Rp 28.695.652 | Rp 7.173.913   | Rp 340.425.016 |

Hal ini disebabkan karena metode Function Point dapat menentukan kebutuhan SDM pada setiap fase sehingga jika di total maka menghasilkan estimasi biaya yang lebih besar. Berbeda dengan metode Use Case Point yang mengasumsikan bahwa di setiap fase membutuhkan 1 orang.

Pada metode Function Point membutuhkan durasi pengerjaan lebih lama sedangkan metode Use Case Point membutuhkan durasi pengerjaan lebih sedikit. Hal ini disebabkan karena metode Use Case Point menggunakan dua jenis lembar penilaian yang memiliki indikator bernilai positif dan negatif sehingga mempengaruhi durasi pengerjaan. Metode Function Point hanya menggunakan satu jenis lembar penilaian dan memiliki analisis dengan elemen yang lebih banyak.



**Gambar 5.1 Deviasi Biaya Pengembangan CEKLOK terhadap Biaya Realita**



**Gambar 5.2 Deviasi Biaya Pengembangan COMMJUNCTION terhadap Biaya Realita**

Berdasarkan Gambar 5.1 dan Gambar 5.2, dapat dilihat bahwa biaya Realita dibandingkan UCP memiliki nilai deviasi yang lebih kecil yaitu 4-8%, sedangkan bahwa Realita dibandingkan FP memiliki nilai deviasi yang lebih besar yaitu sekitar 86-169% dibandingkan dengan biaya Realita-Estimasi dengan nilai deviasi 14-17%. Ini membuktikan bahwa UCP dapat menekan deviasi anggaran menjadi lebih kecil atau dengan kata lain, lebih akurat. UCP dapat digunakan untuk estimasi biaya dalam model pengembangan Scrum.



## **BAB 6**

### **KESIMPULAN**

#### **6.1 Kesimpulan**

Use Case Point dapat diterapkan untuk menghitung biaya pengembangan perangkat lunak pada PT. Vascomm Solusi Teknologi dengan cara yaitu pertama menghitung Unadjusted Actor Weights (UAW) yaitu mengidentifikasi dan mengklasifikasikan jenis aktor, Unadjusted Use Case Weights (UUCW) yaitu mengalikan jumlah bobot dengan masing-masing Use Case, Unadjusted Use Case Point (UUCP) yaitu penjumlahan Unadjusted Actor Weights (UAW) dengan Unadjusted Use Case Weights (UUCW). Kuisioner disebar dan dianalisis hasilnya berdasarkan Technical Complexity Factors (TCF) dan Environment Complexity Factor (ECF). kemudian mendapatkan Use Case Point.

Untuk mendapatkan nilai usaha dari nilai Use Case Point adalah dengan menghitung dengan staff hours. Kemudian jumlah waktu yang didapatkan akan dibagi menjadi jumlah sprint dan jam kerja yang ada di Vascomm. Setelah itu estimasi biaya yang didapatkan merupakan hasil estimasi usaha yang dihitung dengan berdasar aktivitas pengembangan perangkat lunak berdasar waktu, personil dan standar biaya yang dibutuhkan.

Hasilnya didapatkan bahwa Use Case Point dapat diterapkan untuk perhitungan estimasi biaya pengembangan. Use Case Point dibandingkan Realita memiliki nilai deviasi yang lebih kecil yaitu 4-8%. Function Point dibandingkan Realita memiliki nilai deviasi yang lebih besar yaitu sekitar 86-169%. Realita-Estimasi dengan nilai deviasi 14-17%. Use Case Point menekan selisih biaya estimasi dan menjadi lebih akurat. UCP mengurangi penyimpangan biaya pengembangan yang tidak akurat menggunakan estimasi konvensional sebesar 9-10%. Hasil perhitungan dapat menjadi dasar dari total biaya proyek yang mendasari jadwal waktu dan harga proyek. Semakin kecil penyimpangan, semakin akurat pengembangan estimasi biaya, semakin baik manajemen proyek perangkat lunak yang bekerja. Dengan demikian, kegagalan / kerugian proyek dapat diminimalkan.

Perhitungan UCP pada SCRUM dapat dilakukan bisa dijadikan acuan untuk bahan pertimbangan / acuan proyek selanjutnya (misalkan untuk dasar mengikuti tender).

## **6.2 Saran**

Pada penelitian selanjutnya dapat dilakukan penelitian yang lebih mendalam terkait dengan penerapan poin dari Use Case Point untuk setiap Sprint pada Manajemen Scrum. Dimana hasil poin pada Use Case Poin dijadikan value/nilai yang akan dicapai pada setiap Sprint yang dilakukan untuk membangun perangkat lunak. Dengan cara membagi Use Case Point dengan total Sprint yang dibutuhkan, hasilnya akan terlihat berapa rata-rata poin yang dibutuhkan setiap sprint yang dilakukan.

## DAFTAR PUSTAKA

- Dewi, R. S., Prassida, G. F., Sholih, & Subriadi, A. P. (2016). UCPabc as an Integration Model for Software Cost Estimation. *2nd International Conference on Science in Information Technology (ICSITech)* (pp. 187-192). IEEE.
- Dewi, R. S., Subriadi, A. P., & Sholih. (2017). A Modification Complexity Factor in Function Points Method for Software Cost Estimation Towards Public Service Application. *4th Information Systems International Conference 2017* (pp. 415–422). Bali: ELSEVIER. doi:10.1016/j.procs.2017.12.172
- Fan, & al, e. (2009). *Software Cost Estimation*. Hongkong: The Hongkong Polytechnic University.
- Kang, S., Choi, O., & Baik, J. (2010). Model-based Dynamic Cost Estimation and Tracking Method for Agile Software Development. *9th IEEE/ACIS International Conference on Computer and Information Science* (pp. 743-748). Taejon: IEEE. doi:10.1109/ICIS.2010.126
- Marchewka, J. (2003). *Information Technology Project Management Providing Measurable Organizational Value*. New Delh: Wiley india Pvt. Ltd. doi:978-0471392033
- Noerlina. (2008). Perencanaan Manajemen Proyek Sistem Informasi dan Teknologi Informasi Online Bisnis. *Jurnal Piranti Warta, XI*, 440-450.
- Pressman, R. S. (2010). *Software engineering : a practitioner's approach — 7th ed.* US: McGraw-Hill.
- Primandari, P. L., & Sholih. (2016). Effort Distribution to Estimate Cost in Small to Medium Software Development Project with Use Case Point. *The Third Information Systems International Conference* (pp. 78-85). Surabaya: ELSEVIER. doi:10.1016/j.procs.2015.12.107
- Qina, X., & Fang, M. (2011). Summarization of Software Cost Estimation. *Procedia Engineering*, 3027 – 3031.

- Saleh, K. (2011). Effort and Cost Allocation in Medium to Large Software Development Projects. *Internasional Journal of Computers*, 74-79.
- Schwalbe, K. (2014). *Information Technology Project Management (7th edition)*. United State of America: Augburgs College.
- Scrum.org. (2019). *What is Scrum?* Retrieved from Scrum.Org.: <https://www.scrum.org/resources/what-is-scrum>
- Sholiq, Dewi, R. S., & Subriadi, A. P. (2017). A Comparative Study of Software Development Size Estimation Method: UCPabc vs Function Points. *4th Information Systems International Conference 2017* (pp. 470–477). Bali: ELSEVIER. doi:10.1016/j.procs.2017.12.179
- Sommerville, I. (2011). *Software Engineering Ninth Edition*. USA: Addison-Wesley.
- Yuliansyah, H., Qudsiah, S., Zahrotun, L., & Arfiani, I. (2018). Implementation of use case point as software effort estimation in Scrum. *International Conference on Engineering and Applied Technology (ICEAT)*. I. Yogyakarta: IOP Publishing. doi:10.1088/1757-899X/403/1/012085
- Zahiroh, D. F., Saputra, M. C., & Herlambang, A. D. (2018). Perbandingan Evaluasi Biaya Pengembangan Sistem Antrian RSUD Dr. Soetrasno Rembang Menggunakan Metode Use Case Point dan Function Point (Studi Kasus: CV Pabrik Teknologi). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 369-379. Retrieved from <http://j-ptiik.ub.ac.id/index.php/j-ptiik>



## **LAMPIRAN PENILAIAN TECHNICAL DAN ENVIRONMENT COMPLEXITY FACTOR PADA *USE CASE POINT***

Nama : .....

Jabatan : .....

Telp : .....

Jenis Kelamin : .....

Pengalaman bekerjasama :  1x       >1x       >2x

dengan tim proyek

Tgl pengisian lembar penilaian: .....

### **PETUNJUK UMUM**

Survei dengan cara penyebaran lembar penilaian ini merupakan salah satu cara yang digunakan untuk mengetahui beberapa faktor yang berhubungan dengan estimasi biaya proyek terpilih.

Hasil dari penyebaran lembar penilaian ini akan digunakan penulis sebagai bahan untuk tugas akhir (tesis). Mohon kerjasama Bapak/Ibu/Saudara dalam memberikan penilaian dengan jujur dan apa adanya. Terimakasih atas perhatiannya.

Petunjuk Pengisian Lembar Penilaian:

1. Sebelum mengisikan penilaian Bapak/Ibu/Saudara dimohon untuk membaca gambaran umum perusahaan dan kondisi proyek terpilih.
2. Untuk pengisian nilai, Bapak/Ibu/Saudara dituntut untuk mengisi berdasarkan pengetahuan atau pengalaman yang dimiliki terkait dengan pengerjaan sebuah proyek terpilih.
3. Nilai yang diberikan hanya satu penilaian saja yaitu pada range 0 (no influence) sampai 5 (strong influence) (dicentang pada kolom yang telah disediakan).

**LEMBAR PENILAIAN TECHNICAL COMPLEXITY FACTOR PADA  
USE CASE POINT**

Indikator 0 (No Influence), 1 (Incidental Influence), 2 (Moderate Influence), 3  
(Average Influence), 4 (Significant Influence), 5 (Very Influence)

| Faktor Teknis |   | Deskripsi   | Nilai |   |   |   |   |   |
|---------------|---|---|-------|---|---|---|---|---|
|               |   |   | 0     | 1 | 2 | 3 | 4 | 5 |
| <b>T1</b>     | <b>Kebutuhan Sistem Terdistribusi</b>         | Menginformasikan apakah pemrosesan data terdistribusi diperlukan dalam sistem. Angka yang lebih tinggi mewakili kebutuhan arsitektur yang semakin kompleks.   |       |   |   |   |   |   |
| <b>T2</b>     | <b>Pentingnya Waktu Respon</b>                | Menentukan efisiensi sistem yang berkaitan dengan waktu respon terhadap event, flow, dan lain-lain. Nilai yang lebih tinggi menunjukkan pentingnya waktu respon.  |       |   |   |   |   |   |
| <b>T3</b>     | <b>Efisiensi Pengguna Akhir</b>               | Mendefinisikan efisiensi untuk pengguna akhir. Artinya, apakah sistem ini dikembangkan untuk mengoptimalkan efisiensi pengguna atau hanya mengutamakan kemampuan sistem? Nilai yang semakin tinggi menggambarkan sistem yang meningkatkan efisiensi pengguna. |       |   |   |   |   |   |
| <b>T4</b>     | <b>Kebutuhan Kompleksitas Proses Internal</b> | Menentukan apakah operasi yang bersifat complicated yang terkait dengan pengolahan data dan penggunaan algoritma diperlukan? Database yang sederhana akan memiliki nilai rendah.  |       |   |   |   |   |   |
| <b>T5</b>     | <b>Transaction Rate Penggunaan Ulang Kode</b> | Menginformasikan apakah elemen atau kode sistem yang dihasilkan akan digunakan kembali.   |       |   |   |   |   |   |
| <b>T6</b>     | <b>Kemudahan Instalasi</b>                    | Mengenai metode instalasi dan kemudahan instalasi bagi pengguna, apakah user-friendly atau menunjukkan keikutsertaan seorang spesialis untuk proses instalasi dan proses mempersiapkan pemasokan awal   |       |   |   |   |   |   |

| Faktor Teknis |                                     | Deskripsi  | Nilai |   |   |   |   |   |
|---------------|-------------------------------------|--|-------|---|---|---|---|---|
|               |                                     |  | 0     | 1 | 2 | 3 | 4 | 5 |
|               |                                     | perangkat lunak. Semakin tinggi tingkat kompetensi pengguna, maka semakin rendah nilai yang diberikan.   |       |   |   |   |   |   |
| <b>T7</b>     | <b>Kemudahan Penggunaan</b>         | Mengenai kemudahan dalam penggunaan dan kemudahan untuk mempelajari bagaimana menggunakan sistem. Semakin tinggi nilai yang diberikan, maka semakin besar pentingnya kemudahan penggunaan.   |       |   |   |   |   |   |
| <b>T8</b>     | <b>Dukungan Cross Platform</b>      | Menginformasikan apakah aplikasi harus beroperasi di lingkungan tertentu. Semakin banyak platform yang harus didukung (bisa versi browser, perangkat mobile, atau Windows/OSX/Unix), maka semakin tinggi nilai yang diberikan  |       |   |   |   |   |   |
| <b>T9</b>     | <b>Kemudahan Terhadap Perubahan</b> | Menentukan apakah sistem yang akan dibangun mudah untuk dikembangkan lebih lanjut di masa depan. Semakin tinggi nilai yang diberikan, maka menggambarkan sistem yang dikembangkan membutuhkan kustomisasi atau perubahan lebih di masa depan.                                |       |   |   |   |   |   |
| <b>T10</b>    | <b>Tingginya Konkurensi</b>         | Menginformasikan apakah sistem dapat mengatasi masalah konkurensi yang terjadi? (Misalnya mengatasi penguncian database). Semakin tinggi nilai yang diberikan, maka menandakan tingginya perhatian yang harus diberikan untuk menyelesaikan masalah pada data atau aplikasi. |       |   |   |   |   |   |
| <b>T11</b>    | <b>Kustomisasi Keamanan</b>         | Mendefinisikan apakah sistem memerlukan pengamanan akses terhadap sistem atau data? Semakin tinggi nilai yang diberikan, maka harus melakukan kustomisasi keamanan lebih terhadap sistem yang dikembangkan.  |       |   |   |   |   |   |
| <b>T12</b>    | <b>Ketergantungan</b>               | Apakah sistem membutuhkan penggunaan kontrol dari pihak ketiga?  |       |   |   |   |   |   |

| Faktor Teknis |                               | Deskripsi   | Nilai |   |   |   |   |   |
|---------------|-------------------------------|---|-------|---|---|---|---|---|
|               |                               |   | 0     | 1 | 2 | 3 | 4 | 5 |
|               | <b>pada Kode Pihak Ketiga</b> | Misalnya penggunaan ulang kode atau kode pihak ketiga dapat mengurangi usaha yang dibutuhkan sistem untuk memberikan solusi terhadap permasalahan sistem. Jika semakin penting membutuhkan kode pihak ketiga, maka semakin rendah nilai yang diberikan. |       |   |   |   |   |   |
| <b>T13</b>    | <b>Pelatihan Pengguna</b>     | Menentukan kebutuhan untuk pelatihan bagi pengguna. Semakin tinggi nilai yang diberikan, maka semakin lama waktu yang dibutuhkan pengguna untuk mencapai tingkat penguasaan aplikasi.   |       |   |   |   |   |   |

**LEMBAR PENILAIAN ENVIRONMENT COMPLEXITY FACTOR PADA  
USE CASE POINT**

Indikator 0 (No Influence), 1 (Incidental Influence), 2 (Moderate Influence), 3  
(Average Influence), 4 (Significant Influence), 5 (Very Influence)

| Faktor Teknis |   | Deskripsi   | Nilai |   |   |   |   |   |
|---------------|---|---|-------|---|---|---|---|---|
|               |   |   | 0     | 1 | 2 | 3 | 4 | 5 |
| <b>E1</b>     | <b>Familiaritas / Kedekatan dalam Mengenal Proyek</b> | Menginformasikan sejauh mana tim mengetahui domain dan teknis dari solusi terhadap masalah pelanggan. Selain itu mengetahui tentang metodologi dimana proyek dilaksanakan misal SCRUM serta pengetahuan tentang bahasa pemodelan sistem misal UML. Semakin tinggi nilai yang diberikan, maka semakin tinggi tingkat pengalaman. |       |   |   |   |   |   |
| <b>E2</b>     | <b>Pengalaman Terhadap Aplikasi</b>                   | Mengenai pengalaman tim dalam mengembangkan perangkat lunak. Semakin tinggi nilai yang diberikan mewakili pengalaman yang lebih banyak. Jika pengembangan sistem ini merupakan pengalaman baru, maka akan bernilai 0.   |       |   |   |   |   |   |
| <b>E3</b>     | <b>Pengalaman Pemrograman Berorientasi Obyek</b>      | Pengalaman dalam merancang aplikasi berorientasi objek yang berkaitan dengan kemampuan mendesain aplikasi berorientasi objek serta kemampuan untuk menggunakan alat pendukung untuk merancang sistem. Semakin tinggi nilai yang diberikan, maka menandakan pengalaman yang lebih terhadap OO Programming.                       |       |   |   |   |   |   |
| <b>E4</b>     | <b>Kemampuan Menganalisis</b>                         | Menentukan kemampuan analisis mengenai requirement dari client dan memiliki pengetahuan terkait masalah yang sedang dipecahkan. Semakin tinggi nilai yang diberikan mewakili semakin tingginya kemampuan dan pengetahuan.   |       |   |   |   |   |   |
| <b>E5</b>     | <b>Motivasi</b>                                       | Menilai motivasi tim untuk terlibat dalam tugas yang ditugaskan. Nilai yang tinggi menunjukkan besarnya motivasi.   |       |   |   |   |   |   |

| Faktor Teknis |   | Deskripsi  | Nilai |   |   |   |   |   |
|---------------|---|--|-------|---|---|---|---|---|
|               |   |  | 0     | 1 | 2 | 3 | 4 | 5 |
| <b>E6</b>     | <b>Stabilitas Persyaratan</b>               | Mendefinisikan jika persyaratan tidak sering mengalami perubahan. Nilai yang tinggi menandakan kebutuhan akan perubahan yang tinggi.   |       |   |   |   |   |   |
| <b>E7</b>     | <b>Staff yang Bekerja Paruh Waktu</b>       | Menentukan apakah ada staff yang bekerja paruh waktu dalam tim (misalnya magang, pelajar).   |       |   |   |   |   |   |
| <b>E8</b>     | <b>Tingkat Kesulitan Bahasa Pemrograman</b> | Menentukan tingkat kesulitan dalam mempelajari bahasa pemrograman sistem yang akan diimplementasikan. <ul style="list-style-type: none"> <li>0. Easy, dalam 1 minggu bahasa pemrograman dapat di pahami</li> <li>1. Setidaknya membutuhkan 2 minggu untuk memahami bahasa pemrograman</li> <li>2. Setidaknya membutuhkan 1 bulan untuk memahami bahasa pemrograman</li> <li>3. Membutuhkan pelatihan khusus untuk memahami bahasa pemrograman</li> <li>4. Membutuhkan pelatihan khusus dan membutuhkan bantuan selama proyek berlangsung untuk memahami bahasa pemrograman</li> <li>5. Difficult, bahasa pemrograman hanya dapat dipahami oleh orang yang berpengalaman</li> </ul> |       |   |   |   |   |   |

**LAMPIRAN PENILAIAN TECHNICAL COMPLEXITY  
FACTOR PADA *FUNCTION POINT***

Nama : .....

Jabatan : .....

Telp : .....

Jenis Kelamin : .....

Pengalaman bekerjasama :  1x       >1x       >2x

dengan tim proyek

Tgl pengisian lembar penilaian: .....

**PETUNJUK UMUM**

Survei dengan cara penyebaran lembar penilaian ini merupakan salah satu cara yang digunakan untuk mengetahui beberapa faktor yang berhubungan dengan estimasi biaya proyek terpilih.

Hasil dari penyebaran lembar penilaian ini akan digunakan penulis sebagai bahan untuk tugas akhir (tesis). Mohon kerjasama Bapak/Ibu/Saudara dalam memberikan penilaian dengan jujur dan apa adanya. Terimakasih atas perhatiannya.

Petunjuk Pengisian Lembar Penilaian:

1. Sebelum mengisikan penilaian Bapak/Ibu/Saudara dimohon untuk membaca gambaran umum perusahaan dan kondisi proyek terpilih.
2. Untuk pengisian nilai, Bapak/Ibu/Saudara dituntut untuk mengisi berdasarkan pengetahuan atau pengalaman yang dimiliki terkait dengan pengerjaan sebuah proyek terpilih.

Nilai yang diberikan hanya satu penilaian saja yaitu pada range 0 (no influence) sampai 5 (strong influence) (dicentang pada kolom yang telah disediakan).

**LEMBAR PENILAIAN TECHNICAL COMPLEXITY FACTOR PADA  
FUNCTION POINT**

| Faktor Teknis |  | Deskripsi  | Nilai    |          |          |          |          |          |  |  |
|---------------|--|--|----------|----------|----------|----------|----------|----------|--|--|
|               |  |  | 0        | 1        | 2        | 3        | 4        | 5        |  |  |
| <b>T1</b>     | <b>Data Communication</b>  | Mendeskripsikan sejauh mana aplikasi berkomunikasi langsung dengan prosesor komputer.            |          |          |          |          |          |          |  |  |
|               | Pemilihan Nilai untuk T1:<br>0. Aplikasi yang dikembangkan merupakan batch processing* atau aplikasi yang dikembangkan merupakan stand-alone*<br>1. Aplikasi bersifat batch processing namun memiliki remote data entry* atau remote printing*<br>2. Aplikasi bersifat batch processing namun memiliki remote data entry* dan remote printing*<br>3. Aplikasi meliputi sekumpulan data online atau TP (teleprocessing)* dari front-end ke batch processing atau sistem query<br>4. Aplikasi lebih dari sekedar front-end, tapi hanya mendukung satu jenis TP (misalnya: entri data clientserver atau entri data melalui internet, tetapi tidak keduanya)<br>5. Aplikasi lebih dari front end dan mendukung lebih dari satu jenis TP (misalnya: entri data client-server dan entri data melalui internet) |  |          |          |          |          |          |          |  |  |
| <b>T2</b>     | <b>Distributed Data Processing</b>   | Menggambarkan sejauh mana aplikasi dapat mentransfer data antar komponen-komponen fisik aplikasi | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |  |  |
|               | Pemilihan Nilai untuk T2:<br>0. Data tidak ditransfer atau diproses pada komponen lain diluar sistem (pemrosesan dan input output berada di tempat yang sama yaitu aplikasi stand-alone)<br>1. Data disiapkan untuk di transfer, kemudian diproses pada komponen diluar sistem oleh pengguna (misal: aplikasi mendownload data ke perangkat pengguna, sehingga pengguna dapat menggunakan excel atau alat pelaporan lainnya untuk menyiapkan grafik dan melakukan analisis lainnya)<br>2. Data disiapkan untuk di transfer, kemudian diproses pada komponen lain dari sistem (misal: aplikasi mengirim data ke client atau server. Kemudian data tersebut diolah atau digunakan untuk menghasilkan laporan, dan lain-lain)   |  |          |          |          |          |          |          |  |  |



|           |   |  |          |          |          |          |          |          |
|-----------|---|--|----------|----------|----------|----------|----------|----------|
|           | <p>3. Pengolahan terdistribusi data yang online dan dalam satu arah saja (data dikirim ke client atau server dalam 1 arah saja. Kemudian data diproses atau digunakan untuk menghasilkan laporan dan lain-lain oleh aplikasi penerima. Data ini mencakup transaksi yang mengupdate ILF pada client atau server)</p> <p>4. Pengolahan terdistribusi dan transfer data yang online dan di kedua arah (data dikirim ke client dan server dalam 2 arah. Kemudian data diproses atau digunakan untuk menghasilkan laporan dan lain-lain oleh aplikasi penerima. Data ini mencakup transaksi yang mengupdate ILF pada client atau server)</p> <p>5. Fungsi pengolahan secara dinamis dilakukan pada komponen yang paling tepat dari sistem (aplikasi berjalan dibawah sistem operasi yang secara otomatis menangani dynamic allocation* antar komponen dan pengguna sistem operasi dapat mempengaruhi desain dan implementasi aplikasi)</p> |  |          |          |          |          |          |          |
| <b>T3</b> | <b>Performance</b>  | Menggambarkan sejauh mana response time dan pertimbangan performance mempengaruhi pengembangan aplikasi. | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|           | <p>Pemilihan Nilai untuk T3:</p> <p>0. Tidak ada kebutuhan performance khusus yang dinyatakan oleh pengguna</p> <p>1. Performance dan desain aplikasi dinyatakan namun tidak memerlukan tindakan khusus</p> <p>2. Response time sangat penting selama jam kerja. Batas waktu pengolahan sampai hari kerja berikutnya</p> <p>3. Response time sangat penting selama jam kerja. Adanya batas waktu pengolahan menghambat pemrosesan data</p> <p>4. Selain nilai diatas, kebutuhan performance sistem cukup ketat sehingga memerlukan analisis performance pada tahap perancangan</p> <p>5. Selain nilai diatas, tools yang digunakan untuk menganalisis performace juga digunakan pada fase desain, pengembangan dan/atau implementasi untuk memenuhi kebutuhan performance yang dikemukakan oleh pengguna</p>  |  |          |          |          |          |          |          |
| <b>T4</b> | <b>Heavily Used Configuration</b>   | Menggambarkan sejauh mana pengaruh pembatasan sumber daya komputer terhadap fase pengembangan aplikasi.  | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|           | <p>Pemilihan Nilai untuk T1:</p> <p>0. Tidak ada pembatasan operasional eksplisit atau implisit</p> <p>1. Pembatasan operasional memang ada, tetapi kurang ketat dibandingkan aplikasi lain</p>   |  |          |          |          |          |          |          |

|           |   |  |          |          |          |          |          |          |
|-----------|---|--|----------|----------|----------|----------|----------|----------|
|           | <ol style="list-style-type: none"> <li>2. Pembatasan operasional memang ada, tapi bersifat khusus atau hanya untuk aplikasi ini</li> <li>3. Pembatasan operasional yang ditetapkan membutuhkan batasan khusus pada satu bagian saja (pada prosesor pusat atau prosesor khusus)</li> <li>4. Pembatasan operasional yang ditetapkan membutuhkan batasan khusus pada seluruh aplikasi (pada prosesor pusat dan prosesor khusus)</li> <li>5. Selain nilai-nilai diatas, kendala khusus ada pada aplikasi yang memiliki sistem terdistribusi</li> </ol>  |  |          |          |          |          |          |          |
| <b>T5</b> | <b>Transaction Rate</b>   | Mendeskripsikan sejauh mana tingkat transaksi mempengaruhi perkembangan aplikasi. Contoh: tingkat transaksi tinggi dapat mempengaruhi desain, pengembangan, instalasi pada aplikasi. | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|           | <ol style="list-style-type: none"> <li>0. Tidak ada periode transaksi</li> <li>1. Tingkat transaksi yang rendah memiliki efek yang sedikit pada fase desain, pengembangan dan instalasi aplikasi</li> <li>2. Tingkat transaksi rata-rata memiliki beberapa efek pada fase desain, pengembangan dan instalasi aplikasi</li> <li>3. Tingkat transaksi yang tinggi mempengaruhi fase desain, pengembangan dan/atau instalasi aplikasi</li> <li>4. Tingkat transaksi yang tinggi dinyatakan oleh pengguna dalam analisa kebutuhan atau adanya perjanjian bahwa tingkat layanan cukup tinggi, disamping itu memerlukan tahap perancangan, pengembangan dan instalasi aplikasi</li> <li>5. Tingkat transaksi yang tinggi dinyatakan oleh pengguna dalam analisa kebutuhan atau adanya perjanjian bahwa tingkat layanan cukup tinggi, disamping itu memerlukan penggunaan alat bantu untuk menganalisis kinerja dalam fase perancangan, pengembangan dan instalasi aplikasi</li> </ol> |  |          |          |          |          |          |          |
| <b>T6</b> | <b>Online Data Entry</b>  | Mendeskripsikan seberapa banyak prosentase informasi yang dimasukkan secara OnLine.  | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|           | <ol style="list-style-type: none"> <li>0. Tidak ada informasi yang dimasukkan secara On-Line</li> <li>1. 1% sampai 7% dari total transaksi yang dimasukkan secara On-Line</li> <li>2. 8% sampai 15% dari total transaksi yang dimasukkan secara On-Line</li> <li>3. 16% sampai 23% dari total transaksi yang dimasukkan secara On-Line</li> <li>4. 24% sampai 30% dari total transaksi yang dimasukkan secara On-Line</li> <li>5. Lebih dari 30% dari total transaksi yang dimasukkan secara On-Line</li> </ol>   |  |          |          |          |          |          |          |
| <b>T7</b> | <b>End-User Efficiency</b>  | Menggambarkan tingkat efisiensi pengguna atau user friendly dan  | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |

|           |   |  |          |          |          |          |          |          |
|-----------|---|--|----------|----------|----------|----------|----------|----------|
|           |   | mengukur kemudahan penggunaan bagi pengguna aplikasi. Faktor ini menekankan pada desain untuk efisiensi pengguna meliputi: Tombol navigasi (misal: hyperlink), menu, drop down menu, pop-up windows, support penggunaan 2 bahasa atau lebih, dan lain-lain.  |          |          |          |          |          |          |
|           | <ul style="list-style-type: none"> <li>0. Tidak ada</li> <li>1. Satu sampai tiga</li> <li>2. Empat sampai lima</li> <li>3. Enam atau lebih, tetapi tidak ada kebutuhan pengguna tertentu yang berkaitan dengan efisiensi pengguna</li> <li>4. Enam atau lebih, kebutuhan akan efisiensi pengguna cukup kuat untuk membuat desain aplikasi yang bersifat user friendly</li> <li>5. Enam atau lebih, menunjukkan bahwa kebutuhan untuk efisiensi pengguna cukup kuat untuk menunjukkan bahwa tujuan telah tercapai</li> </ul>   |  |          |          |          |          |          |          |
| <b>T8</b> | <b>Online Update</b>  | Mendeskripsikan sejauh mana Internal Logical File (ILF)* diupdate secara online.   | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|           | <ul style="list-style-type: none"> <li>0. Tidak ada</li> <li>1. Update online 1-3 file</li> <li>2. Update online 4 atau lebih file</li> <li>3. Update online lebih dari 4 file, menyertakan ILF dalam update online</li> <li>4. Update online lebih dari 4 file, selain itu perlindungan terhadap kehilangan data penting dilakukan yaitu dengan merancang khusus serta diprogram dalam sistem</li> <li>5. Update online lebih dari 4 file, selain itu perlindungan terhadap kehilangan data dilakukan dengan pemulihan otomatis dengan meminimalkan intervensi dari manusia</li> </ul> |  |          |          |          |          |          |          |
| <b>T9</b> | <b>Complex Processing</b>   | Mendeskripsikan sejauh mana proses logika mempengaruhi perkembangan aplikasi misalnya: sensitif control*, pengolahan input/output menggunakan proses matematika, proses logika yang bersifat ekstensif, terjadi banyak sekali exception processing*, sehingga menghasilkan transaksi yang tidak lengkap dan harus diolah kembali, dan lain-lain. | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |

|            |  |  |          |          |          |          |          |          |
|------------|--|--|----------|----------|----------|----------|----------|----------|
|            | <p>Nilai:</p> <ol style="list-style-type: none"> <li>0. Tidak ada</li> <li>1. Salah satu</li> <li>2. Dua hal</li> <li>3. Tiga hal</li> <li>4. Empat hal</li> <li>5. Lima hal</li> </ol>  |  |          |          |          |          |          |          |
| <b>T10</b> | <b>Reusability</b>   | Mendesripsikan sejauh mana aplikasi dan kode program dirancang, dikembangkan dan didukung untuk dapat digunakan dalam aplikasi lain.   | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|            | <ol style="list-style-type: none"> <li>0. Tidak ada kode yang dapat digunakan kembali</li> <li>1. Kode dapat digunakan kembali dalam aplikasi lain</li> <li>2. Kurang dari 10% dari seluruh kode program yang dikembangkan yang dapat digunakan di lebih dari satu aplikasi</li> <li>3. 10% atau lebih dari seluruh kode program yang dikembangkan yang dapat digunakan di lebih dari satu aplkasi</li> <li>4. Aplikasi ini dikemas secara khusus dan/atau didokumentasikan agar memudahkan proses penggunaan ulang kode program dan aplikasi disesuaikan dengan level source code</li> <li>5. Aplikasi ini dikemas secara khusus dan/atau didokumentasikan agar memudahkan proses penggunaan ulang kode program dan aplikasi disesuaikan untuk penggunaan dengan menggunakan parameter dari pengguna</li> </ol> |  |          |          |          |          |          |          |
| <b>T11</b> | <b>Installation Ease</b>   | Menggambarkan sejauh mana konversi dari lingkungan sebelumnya ke lingkungan yang baru mempengaruhi pengembangan sistem. Kemudahan konversi dan proses instalasi merupakan karakteristik aplikasi yaitu dilihat dari proses konversi dan proses instalasi serta alat bantu yang disediakan untuk membantu proses konversi dan membantu selama tahap uji sistem dilakukan. | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|            | <ol style="list-style-type: none"> <li>0. Tidak ada pertimbangan khusus yang dinyatakan oleh pengguna serta tidak ada persiapan khusus yang diperlukan untuk proses instalasi aplikasi</li> <li>1. Tidak ada pertimbangan khusus yang dinyatakan oleh pengguna serta ada persiapan khusus yang diperlukan untuk proses instalasi aplikasi</li> </ol>   |  |          |          |          |          |          |          |

|            |                         |  |          |          |          |          |          |          |  |
|------------|-------------------------|--|----------|----------|----------|----------|----------|----------|--|
|            |                         | <p>2. Kebutuhan konversi dan instalasi dinyatakan oleh pengguna dan panduan konversi serta instalasi disediakan dan diuji. Dampak dari adanya konversi pada aplikasi tidak dianggap penting</p> <p>3. Kebutuhan konversi dan instalasi dinyatakan oleh pengguna dan panduan konversi serta instalasi disediakan dan diuji. Dampak dari adanya konversi pada aplikasi dianggap penting</p> <p>4. Selain nilai 2 dan 3, alat bantu konversi dan instalasi otomatis disediakan dan diuji</p> <p>5. Selain nilai 2, 3 dan 4, alat bantu konversi dan instalasi otomatis disediakan dan diuji</p>   |          |          |          |          |          |          |  |
| <b>T12</b> | <b>Operational Ease</b> | Menggambarkan seberapa efektif penerapan aspek operasional seperti start-up, back-up dan recovery terhadap aplikasi yang dikembangkan.   | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |  |
|            |                         | <p>0. Tidak ada operasional khusus selain prosedur back-up normal yang dinyatakan oleh pengguna 1-4Satu, beberapa atau semua item berikut berlaku untuk aplikasi yang dikembangkan.</p> <p>1. Prosedur start-up, back-up dan recovery disediakan namun membutuhkan intervensi dari pengguna</p> <p>2. Prosedur start-up, back-up dan recovery disediakan namun tidak membutuhkan intervensi dari pengguna</p> <p>3. Aplikasi ini meminimalkan kebutuhan instalasi dan/atau remote data access yang memerlukan intervensi dari pengguna</p> <p>4. Aplikasi ini meminimalkan kebutuhan penggunaan kertas</p> <p>5. Aplikasi ini dirancang untuk melakukan prosedur startup, back-up dan recovery tanpa pengawasan, yang artinya tidak ada intervensi dari pengguna yang diperlukan untuk mengoperasikan sistem</p> |          |          |          |          |          |          |  |
| <b>T13</b> | <b>Multiple Sites</b>   | Menggambarkan sejauh mana aplikasi dikembangkan untuk perangkat keras dan perangkat lunak yang berbeda   | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |  |
|            |                         | <p>0. Kebutuhan instalasi hanya di 1 jenis perangkat</p> <p>1. Kebutuhan instalasi lebih dari 1 tempat instalasi dan aplikasi dirancang hanya dapat beroperasi di perangkat keras dan perangkat lunak yang jenisnya bersifat identik</p> <p>2. Kebutuhan instalasi lebih dari 1 tempat instalasi dan aplikasi dirancang hanya dapat beroperasi di perangkat keras dan perangkat lunak yang jenisnya serupa</p>   |          |          |          |          |          |          |  |

|            |                         |  |          |          |          |          |          |          |
|------------|-------------------------|--|----------|----------|----------|----------|----------|----------|
|            |                         | <p>3. Kebutuhan instalasi lebih dari 1 tempat instalasi dan aplikasi dirancang hanya dapat beroperasi di perangkat keras dan perangkat lunak yang jenisnya berbeda</p> <p>4. Adanya dokumentasi dan support plan untuk mendukung aplikasi di beberapa tempat instalasi dan aplikasi dijelaskan seperti pada nilai 2</p> <p>5. Adanya dokumentasi dan support plan untuk mendukung aplikasi di beberapa tempat instalasi dan aplikasi dijelaskan seperti pada nilai 3</p>   |          |          |          |          |          |          |
| <b>T14</b> | <b>Fasiliate Change</b> | <p>Mendeskripsikan sejauh mana aplikasi telah dikembangkan untuk memudahkan proses modifikasi struktur data. Karakteristik berikut dapat digunakan pada aplikasi yang akan dikembangkan:</p> <p><b>Flexible Query*</b></p> <ul style="list-style-type: none"> <li>• Tersedia flexible query dan fasilitas report yang dapat menangani request dengan kompleksitas simple (sederhana) (terhitung 1 item)</li> <li>• Tersedia flexible query dan fasilitas report yang dapat menangani request dengan kompleksitas average (rata-rata) (terhitung 2 items)</li> <li>• Tersedia flexible query dan fasilitas report yang dapat menangani request dengan kompleksitas complex (kompleks) (terhitung 3 items)</li> </ul> <p><b>Business Control Data*</b></p> <ul style="list-style-type: none"> <li>• Business Control Data disimpan dalam tabel yang dikelola oleh pengguna dengan proses interaktif online namun perubahan hanya berlaku pada siklus berikutnya (terhitung 1 item)</li> <li>• Business Control Data disimpan dalam tabel yang dikelola oleh pengguna dengan proses interaktif online namun perubahan tersebut</li> </ul> | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |

|  |  |                                      |  |  |  |  |  |  |
|--|--|--------------------------------------|--|--|--|--|--|--|
|  |  | langsung berlaku (terhitung 2 items) |  |  |  |  |  |  |
|  | 0. Bukan dari salah satu item diatas<br>1. Sebanyak 1 item<br>2. Sebanyak 2 items<br>3. Sebanyak 3 items<br>4. Sebanyak 4 items<br>5. Sebanyak 5 items |                                      |  |  |  |  |  |  |

## LAMPIRAN HASIL PENILAIAN COMPLEXITY FACTOR CEKLOK

### Technical Complexity Factor pada Metode Use Case Point

| No. | Technical Factor                     | Answer |    |    |      |      |        |    |    | Average | Weight | Result |
|-----|--------------------------------------|--------|----|----|------|------|--------|----|----|---------|--------|--------|
|     |                                      | TL     | SA | BE | FE 1 | FE 2 | Mobile | QA | PO |         |        |        |
| T1  | Distributed System Required          | 4      | 5  | 4  | 4    | 4    | 4      | 3  | 4  | 4       | 2      | 8      |
| T2  | Response Time                        | 5      | 4  | 4  | 4    | 4    | 4      | 2  | 4  | 3,875   | 2      | 7,75   |
| T3  | End User Efficiency                  | 4      | 3  | 4  | 4    | 5    | 3      | 1  | 3  | 3,375   | 1      | 3,375  |
| T4  | Complex Internal Processing Required | 5      | 4  | 3  | 3    | 2    | 3      | 4  | 3  | 3,375   | 1      | 3,375  |
| T5  | Reusable Code                        | 5      | 4  | 4  | 3    | 4    | 4      | 3  | 4  | 3,875   | 1      | 3,875  |
| T6  | Easy to Install                      | 3      | 3  | 3  | 4    | 3    | 5      | 1  | 3  | 3,125   | 0,5    | 1,5625 |
| T7  | Easy to Use                          | 4      | 3  | 5  | 4    | 5    | 5      | 5  | 4  | 4,375   | 0,5    | 2,1875 |
| T8  | Portable                             | 4      | 5  | 4  | 4    | 5    | 2      | 5  | 3  | 4       | 2      | 8      |
| T9  | Easy to Change                       | 3      | 5  | 3  | 3    | 5    | 4      | 3  | 3  | 3,625   | 1      | 3,625  |
| T10 | Concurrent                           | 4      | 4  | 3  | 3    | 3    | 4      | 2  | 3  | 3,25    | 1      | 3,25   |
| T11 | Security Features                    | 4      | 5  | 4  | 3    | 4    | 5      | 4  | 4  | 4,125   | 1      | 4,125  |
| T12 | Access for Third Parties             | 3      | 2  | 3  | 3    | 3    | 4      | 3  | 3  | 3       | 1      | 3      |
| T13 | Special Training Required            | 4      | 3  | 3  | 3    | 4    | 1      | 1  | 2  | 2,625   | 1      | 2,625  |
|     |                                      |        |    |    |      |      |        |    |    |         |        | 54,75  |



**Environment Complexity Factor pada Metode Use Case Point**

| No. | Tech                           | Answer    |    |    |      |      |        |    |    | Average | Weight | Result |
|-----|--------------------------------|-----------|----|----|------|------|--------|----|----|---------|--------|--------|
|     |                                | Team Lead | SA | BE | FE 1 | FE 2 | Mobile | QA | PO |         |        |        |
| E1  | Familiarity With The Project   | 4         | 3  | 3  | 4    | 3    | 2      | 2  | 4  | 3,125   | 1,5    | 4,6875 |
| E2  | Application Experience         | 3         | 3  | 2  | 4    | 5    | 3      | 5  | 4  | 3,625   | 0,5    | 1,8125 |
| E3  | OO Programming Experience      | 4         | 3  | 3  | 4    | 4    | 3      | 3  | 4  | 3,5     | 1      | 3,5    |
| E4  | Lead Analyst Capability        | 3         | 3  | 3  | 4    | 4    | 3      | 4  | 4  | 3,5     | 0,5    | 1,75   |
| E5  | Motivation                     | 4         | 4  | 4  | 4    | 3    | 3      | 4  | 4  | 3,75    | 1      | 3,75   |
| E6  | Stable Requirements            | 4         | 3  | 3  | 3    | 5    | 4      | 3  | 4  | 3,625   | 2      | 7,25   |
| E7  | Part Time Staff                | 3         | 2  | 0  | 3    | 0    | 3      | 0  | 2  | 1,625   | -1     | -1,625 |
| E8  | Difficult Programming Language | 3         | 1  | 0  | 3    | 2    | 0      | 5  | 1  | 1,875   | -1     | -1,875 |
|     |                                |           |    |    |      |      |        |    |    |         |        | 19,25  |

**Technical Complexity Factor pada Metode Function Point**

| No. | Tech                        | Answer    |    |    |      |      |        |    |    | Average |
|-----|-----------------------------|-----------|----|----|------|------|--------|----|----|---------|
|     |                             | Team Lead | SA | BE | FE 1 | FE 2 | Mobile | QA | PO |         |
| T1  | Data Communications         | 4         | 3  | 5  | 5    | 4    | 5      | 3  | 3  | 4,00    |
| T2  | Distributed Data Processing | 5         | 0  | 4  | 4    | 3    | 5      | 4  | 5  | 3,75    |
| T3  | Performance                 | 4         | 5  | 2  | 2    | 2    | 5      | 4  | 5  | 3,63    |
| T4  | Heavily Used Configuration  | 4         | 0  | 4  | 4    | 3    | 2      | 3  | 3  | 2,88    |
| T5  | Transaction Rate            | 5         | 3  | 3  | 3    | 4    | 2      | 4  | 4  | 3,50    |
| T6  | Online Data Entry           | 5         | 5  | 5  | 5    | 3    | 5      | 5  | 5  | 4,75    |
| T7  | End-User Efficiency         | 5         | 4  | 5  | 5    | 5    | 2      | 4  | 4  | 4,25    |
| T8  | Online Update               | 3         | 5  | 1  | 1    | 3    | 2      | 4  | 3  | 2,75    |
| T9  | Complex Processing          | 4         | 5  | 3  | 3    | 3    | 4      | 2  | 4  | 3,50    |
| T10 | Reusability                 | 5         | 5  | 3  | 3    | 3    | 3      | 5  | 5  | 4,00    |
| T11 | Installation Ease           | 3         | 3  | 2  | 2    | 3    | 3      | 3  | 3  | 2,75    |
| T12 | Operation Ease              | 5         | 4  | 1  | 1    | 3    | 0      | 3  | 4  | 2,63    |
| T13 | Multiple Sites              | 5         | 5  | 3  | 3    | 3    | 4      | 3  | 5  | 3,88    |
| T14 | Facilitate Change           | 4         | 5  | 1  | 1    | 3    | 4      | 0  | 4  | 2,75    |
|     |                             |           |    |    |      |      |        |    |    | 49,00   |

## LAMPIRAN HASIL PENILAIAN COMPLEXITY FACTOR COMMJUNCTION

### Technical Complexity Factor pada Metode Use Case Point

| No. | Tech                                 | Answer    |    |    |    |    |    | Average | Weight | Result |
|-----|--------------------------------------|-----------|----|----|----|----|----|---------|--------|--------|
|     |                                      | Team Lead | SA | BE | FE | QA | PO |         |        |        |
| T1  | Distributed System Required          | 4         | 5  | 5  | 4  | 3  | 4  | 4,167   | 2      | 8,333  |
| T2  | Response Time                        | 5         | 4  | 5  | 4  | 2  | 4  | 4,000   | 2      | 8,000  |
| T3  | End User Efficiency                  | 4         | 3  | 5  | 3  | 1  | 3  | 3,167   | 1      | 3,167  |
| T4  | Complex Internal Processing Required | 5         | 4  | 3  | 4  | 4  | 3  | 3,833   | 1      | 3,833  |
| T5  | Reusable Code                        | 5         | 4  | 5  | 4  | 3  | 4  | 4,167   | 1      | 4,167  |
| T6  | Easy to Install                      | 3         | 3  | 5  | 1  | 3  | 3  | 3,000   | 0,5    | 1,500  |
| T7  | Easy to Use                          | 4         | 3  | 4  | 5  | 5  | 4  | 4,167   | 0,5    | 2,083  |
| T8  | Portable                             | 4         | 5  | 5  | 3  | 5  | 3  | 4,167   | 2      | 8,333  |
| T9  | Easy to Change                       | 3         | 5  | 5  | 4  | 3  | 3  | 3,833   | 1      | 3,833  |
| T10 | Concurrent                           | 4         | 4  | 3  | 3  | 2  | 3  | 3,167   | 1      | 3,167  |
| T11 | Security Features                    | 4         | 5  | 5  | 4  | 4  | 4  | 4,333   | 1      | 4,333  |
| T12 | Access for Third Parties             | 3         | 2  | 1  | 4  | 3  | 3  | 2,667   | 1      | 2,667  |
| T13 | Special Training Required            | 4         | 3  | 3  | 4  | 1  | 2  | 2,833   | 1      | 2,833  |
|     |                                      |           |    |    |    |    |    |         |        | 56,250 |

**Environment Complexity Factor pada Metode Use Case Point**

| No. | Tech                           | Answer    |    |    |    |    |    | Average | Weight | Result |
|-----|--------------------------------|-----------|----|----|----|----|----|---------|--------|--------|
|     |                                | Team Lead | SA | BE | FE | QA | PO |         |        |        |
| E1  | Familiarity With The Project   | 4         | 3  | 4  | 4  | 3  | 4  | 3,67    | 1,5    | 5,5    |
| E2  | Application Experience         | 3         | 3  | 3  | 3  | 5  | 3  | 3,33    | 0,5    | 1,67   |
| E3  | OO Programming Experience      | 4         | 3  | 2  | 3  | 3  | 3  | 3       | 1      | 3      |
| E4  | Lead Analyst Capability        | 3         | 3  | 4  | 4  | 4  | 4  | 3,67    | 0,5    | 1,83   |
| E5  | Motivation                     | 4         | 4  | 4  | 4  | 4  | 4  | 4       | 1      | 4      |
| E6  | Stable Requirements            | 4         | 3  | 4  | 3  | 3  | 4  | 3,5     | 2      | 7      |
| E7  | Part Time Staff                | 0         | 0  | 0  | 0  | 0  | 0  | 0       | -1     | 0      |
| E8  | Difficult Programming Language | 2         | 2  | 2  | 2  | 2  | 2  | 2       | -1     | -2     |
|     |                                |           |    |    |    |    |    |         |        | 21     |

## **Technical Complexity Factor pada Metode Function Point**

**LAMPIRAN PERHITUNGAN SPRINT RATE DAN DAY RATE  
CEKLOK**

| <b>No</b> | <b>Position</b>  | <b>Sprint Rate</b> |
|-----------|------------------|--------------------|
| 1         | BE               | Rp978.260,87       |
| 2         | FE               | Rp978.260,87       |
| 3         | Mobile           | Rp978.260,87       |
| 4         | UI/UX            | Rp869.565,22       |
| 5         | DB               | Rp869.565,22       |
| 6         | QA               | Rp869.565,22       |
| 7         | PO               | Rp978.260,87       |
| 8         | IoT/Tech Lead/SA | Rp1.086.956,52     |
| 9         | Team Lead        | Rp1.304.347,83     |
|           | <b>TOTAL</b>     | <b>Rp8.913.043</b> |

**LAMPIRAN PERHITUNGAN SPRINT RATE DAN DAY RATE  
COMMJUNCTION**

| <b>No</b> | <b>Position</b> | <b>Sprint Rate</b> |
|-----------|-----------------|--------------------|
| 1         | BE              | Rp978.260,87       |
| 2         | FE              | Rp978.260,87       |
| 3         | Mobile          | Rp978.260,87       |
| 4         | QA              | Rp869.565,22       |
| 5         | PO              | Rp978.260,87       |
| 6         | SA              | Rp1.086.956,52     |
| 7         | Team Lead       | Rp1.304.347,83     |
|           | <b>TOTAL</b>    | <b>Rp7.173.913</b> |

## BIODATA PENULIS

Penulis bernama lengkap Lia Ninda Safitri yang lahir di Malang, 26 Februari 1995. Merupakan lulusan S1 Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya Malang. Dan telah menyelesaikan studi S2 di Magister Manajemen Teknologi, Program Studi Magister Teknologi Informasi, Fakultas Desain Kreatif dan Bisnis Digital, Institut Teknologi Sepuluh Nopember Surabaya.



Terhitung sejak pengerjaan skripsi S1, penulis tertarik pada bidang desain sistem dan proses bisnis. Semasa menyelesaikan S2, penulis telah menyelesaikan tiga penelitian dan dipublikasikan. Dua penelitian berkenaan dengan proses bisnis, dan satu lain berkenaan dengan manajemen proyek pengembangan perangkat lunak. Penelitian terakhir ini berhubungan langsung dengan pekerjaan penulis saat itu yaitu sebagai Product Owner.

Kedepannya, penulis sangat terbuka apabila terdapat diskusi lebih lanjut di bidang manajemen pengembangan perangkat lunak, SCRUM dan proses bisnis. Untuk informasi lebih lanjut mengenai penelitian ini dapat menghubungi via email.



