

**APLIKASI PENENTUAN HARGA PERKIRAAN SENDIRI UNTUK PROYEK PENGEMBANGAN PERANGKAT LUNAK KEPEMERINTAHAN BERBASIS WEB****Dony Purwanto**<sup>1)</sup> **Arifin Puji Widodo**<sup>2)</sup> **Teguh Sutanto**<sup>3)</sup>

S1 / Jurusan Komputerisasi Akuntansi

Institut Bisnis dan Informatika Stikom Surabaya

Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email: 1) [Donypurwan@gmail.com](mailto:Donypurwan@gmail.com), 2) [Arifin@stikom.edu](mailto:Arifin@stikom.edu), 3) [Teguh@stikom.edu](mailto:Teguh@stikom.edu)

*Abstract* : Owner Estimate (OE) is the price of goods or services what is calculated based on the expertise and data that can be accounted for. Determination of OE for software projects until now is guided by the Indonesia Presidential Decree No. 70 of 2012, which only describes OE for the procurement of goods, construction projects, consulting services, and procurement of other services. In that regulation does not specifically explained the determination of OE for the software project. So in this case Committing Officer (CO) as a constituent of OE have adversity to determining OE for software development projects.

Based on these problems then built the application of determining owner estimates (OE) price for the web-based software development projects, which the method calculation used for effort estimation is Use Case Points (UCP). Built application will be able to perform the calculation of Unadjusted Actor Weight (UAW), unadjusted Use Case Weight (UUCW), Technical Complexity Factor (TCF), and the Environmental Complexity Factor (ECF). Then from these values, the application will calculate the effort estimation that continued with the process of calculating the direct costs of personnel. Further, the application can also calculate the direct non-personnel costs, profits, taxes, up to the cost estimation.

From the testing and evaluation conducted, the application can generate value Unadjusted Actor Weight (UAW), Unadjusted Use Case Weight (UUCW), Technical Complexity Factor (TCF), Environmental Complexity Factor (ECF), effort estimation, direct cost of personnel, direct cost non-personnel, profits, taxes, and cost estimation. So that can be concluded that application can be used to calculate the price of Owner Estimate (OE) for software development projects of small and medium-scale governance.

**Keywords:** Information Systems, Owner Estimates, Software Project, Use Case Point (UCP)

Harga Perkiraan Sendiri (HPS) atau *Owner Estimate* (OE) merupakan perkiraan harga barang atau jasa yang dikalkulasi secara keahlian dan berdasarkan data yang dapat dipertanggungjawabkan. Dalam penyusunannya, HPS didasarkan pada hasil survei harga pasar setempat yang dilaksanakan menjelang pengadaan dengan mempertimbangkan informasi yang ada. Jika penentuan nilai HPS terjadi kesalahan maka pelaksanaannya tidak akan berjalan dengan baik sehingga menyebabkan kegagalan proyek. Kesalahan dalam menentukan HPS sering terjadi. Hal ini disebabkan karena tidak adanya aplikasi untuk membantu pihak PPK dalam menentukan nilai HPS yang tepat dan akurat, padahal banyak sekali masalah yang sering terjadi salah satunya yaitu adanya aktivitas yang tidak atau belum dimasukkan kedalam komponen penentuan HPS, dimana aktivitas tersebut sangat berpengaruh terhadap nilai HPS yang berakibat ketika barang sudah diperoleh, barang tidak bisa dioperasikan karena HPS yang disusun tidak mencakup kegiatan yang mendukung beroperasinya barang tersebut. Oleh karena itu, PPK harus menyusun HPS berdasarkan sumber data yang dapat dipertanggungjawabkan sehingga menghasilkan nilai yang valid serta dapat menjadi acuan dalam

penentuan HPS untuk pengadaan barang/jasa khususnya pengadaan perangkat lunak (Fatimah, 2014).

Selama ini penentuan HPS untuk proyek perangkat lunak masih berpedoman pada Perpres No 70 tahun 2012, dimana hanya menjelaskan HPS untuk pengadaan barang, proyek konstruksi, jasa konsultasi, dan pengadaan jasa lainnya. Didalam Perpres tersebut tidak menjelaskan mengenai penentuan HPS untuk proyek perangkat lunak secara lebih khusus. Sehingga nilai HPS yang diperoleh kurang tepat. Dengan adanya aplikasi ini nantinya diharapkan dapat menyelesaikan masalah dalam penentuan harga perkiraan sendiri proyek perangkat lunak. Aplikasi akan di bangun dengan menggunakan metode UCP. UCP adalah metode yang mampu memberikan estimasi effort yang dapat digunakan untuk membuat suatu proyek berdasarkan jumlah dan kompleksitas *use case* yang ada pada proyek perangkat lunak tersebut (Karner, 1993). Berikut kegiatan penelitian mengenai UCP yang pernah dilakukan sebelumnya, seperti berikut :

1. Nilai estimasi *effort* dengan upaya yang sebenarnya menggunakan metode UCP memiliki deviasi sebesar 19%, sementara

estimasi para ahli memiliki deviasi sebesar 20% (Anda, 2002).

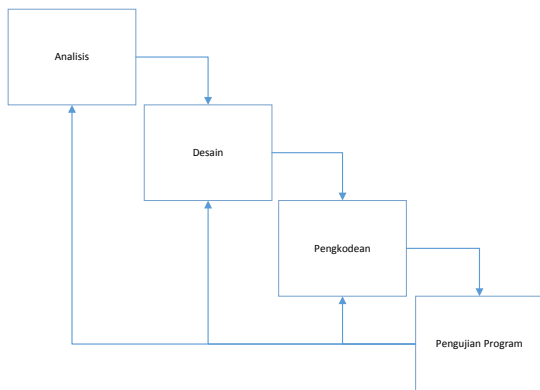
2. Penelitian yang lain menunjukkan bahwa terjadi deviasi sebesar 6% (Nageswaran, 2001).
3. Pendapat terakhir menunjukkan terjadi deviasi sebesar 9% (Carroll, 2005).

Dari beberapa penelitian di atas membuktikan bahwa metode UCP lebih baik dari perkiraan para ahli dan merupakan metode yang tepat untuk melakukan estimasi effort.

Berdasarkan permasalahan yang sudah di uraikan maka akan dibuat suatu aplikasi penentuan harga perkiraan sendiri untuk proyek perangkat lunak dengan menggunakan metode UCP, dimana aplikasi ini akan membantu PPK dalam menentukan estimasi biaya untuk proyek perangkat lunak pemerintahan. Di harapkan dengan adanya aplikasi ini proses penentuan harga perkiraan sendiri yang dilakukan PPK menjadi lebih tepat dan akurat.

**METODE**

Metode penelitian yang digunakan pada penelitian ini adalah metode SDLC (*Software Development Life Cycle*) dengan model *Waterfall*.



Gambar 1. Tahapan Penelitian Tugas Akhir

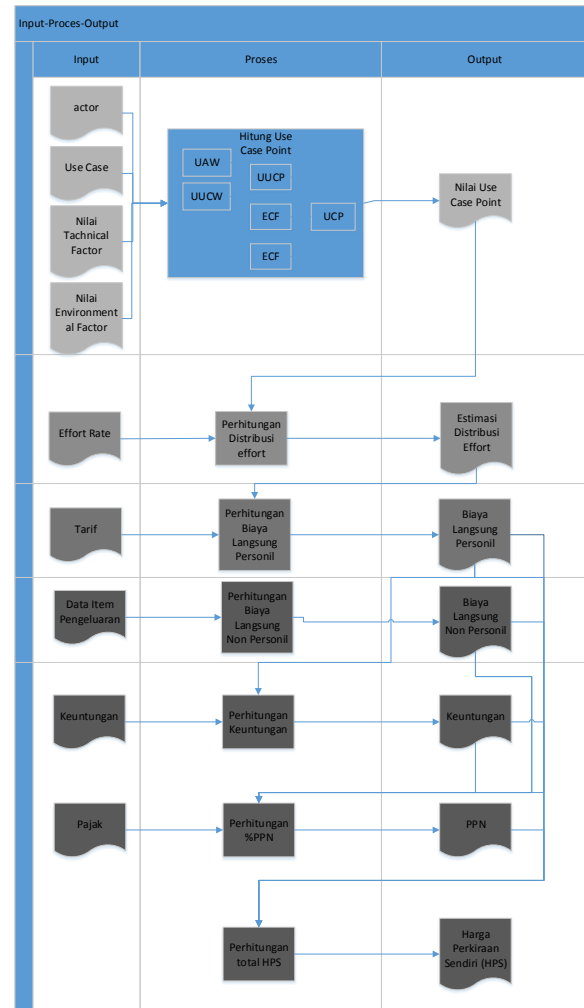
**Analisis Sistem**

Pada tahap analisis sistem akan dilakukan identifikasi, evaluasi, dan mempelajari sistem yang sudah berjalan dengan baik. Tujuan dilakukan analisis sistem ini adalah agar desain sistem yang dirancang sesuai dengan kebutuhan pengguna atau *user* serta mampu memberikan solusi terhadap masalah tersebut.

**Desain Sistem**

Setelah dilakukan analisa sistem dan diketahui kebutuhan sistem yang akan dibuat, maka tahap selanjutnya yaitu melakukan desain sistem. Berikut ini adalah model pengembangan sistem dari

aplikasi penentuan harga perkiraan sendiri proyek perangkat lunak yang akan di buat :



Gambar 2. Desain IPO Pembuatan Aplikasi Penentuan HPS untuk Proyek Pengembangan Perangkat Lunak Pemerintahan.

Berikut adalah proses-proses yang dilakukan dalam menentukan nilai harga perkiraan sendiri proyek perangkat pemerintahan.

1. Hitung *Use Case Point* (UCP)

*Use Case Point* (UCP) adalah metode yang mempunyai kemampuan untuk memberikan estimasi effort yang diperlukan untuk membuat suatu proyek berdasarkan jumlah dan kompleksitas use case yang dimiliki oleh perangkat lunak tersebut (Karner, 1993). Perhitungan UCP sebagai berikut:

1.a Hitung *Unadjusted Use Case Point* (UUCP)

Untuk mendapatkan nilai UUCP, maka perlu dilakukan perhitungan terkait kompleksitas *actor* dan

use case. Terdapat dua langkah yang dilakukan untuk menghitung UUCP, antara lain sebagai berikut:

i. Menghitung *Unadjusted Actor Weights (UAW)*

Pada proses perhitungan UAW ini hal pertama yang dilakukan adalah mengklasifikasikan aktor dalam kategori *simple*, *medium*, dan *complex*. Seperti pada tabel berikut ini:

Tabel 1. Klasifikasi aktor

Klasifikasi Aktor	Tipe Aktor	Bobot
<i>Simple</i>	Didefinisikan dengan API	1
<i>Average</i>	Berinteraksi melalui Protokol, seperti TCP/IP	2
<i>Complex</i>	Berinteraksi dengan GUI atau Web Page	3

Total nilai UAW dapat diperoleh dengan menghitung berapa banyak jumlah aktor untuk masing-masing kategori yang kemudian di kali dengan bobot masing-masing aktor sesuai dengan tabel. Rumus UAW yaitu:

$$UAW = \text{Jumlah Aktor} \times \text{Bobot Aktor}$$

ii. Menghitung *Unadjusted Use Case Weights (UUCW)*

Menentukan use case sebagai *simple*, *medium*, *complex*. Seperti pada tabel 2 berikut ini:

Tabel 2. Klasifikasi use case

Tipe	Jumlah Transaksi	Bobot
<i>Simple</i>	1 sampai 3 transaksi	5
<i>Average</i>	4 sampai 7 transaksi	10
<i>Complex</i>	8 transaksi atau lebih	15

Total nilai UUCW dapat diperoleh dengan menghitung berapa banyak jumlah use case untuk masing-masing tipe use case yang kemudian di kali dengan bobot masing-masing use case sesuai dengan tabel. Rumus UUCW yaitu:

$$UUCW = \text{Jumlah Use Case} \times \text{Bobot Use Case}$$

Setelah diketahui nilai UAW dan nilai UUCW maka langkah selanjutnya yaitu dilakukan perhitungan UUCP. Rumus perhitungan UUCP yaitu sebagai berikut:

$$UUCP = UAW + UUCW$$

1.b Hitung *Technical Complexity Factor (TCF)*

Setelah diketahui nilai UUCP langkah selanjutnya yaitu menentukan nilai TCF. TCF merupakan factor teknis yang mempengaruhi dalam pembuatan proyek perangkat lunak pemerintahan.

Tabel 3. *Technical Factor*

	Technical Factor	Bobot
1.	Kebutuhan Sysetm terdistribusi	2
2.	Waktu respon	1
3.	Efisiensi pengguna	1
4.	Kompleksitas proses internal	1
5.	Penggunaan kode dari hasil daur ulang	1
6.	Kemudahan untuk instal	0.5
7.	Kemudahan untuk digunakan	0.5
8.	Mudah dipakai di berbagai platform	2
9.	Maintenance System	1
10.	Proses paralel	1
11.	Fitur keamanan	1
12.	Akses pihak ke-3	1
13.	Pelatihan pengguna	1

Berdasarkan faktor teknis tersebut pengguna akan memberikan nilai pada setiap faktor teknis. Dimana pada setiap faktor memiliki parameter penilaian.

Tabel 4. Parameter penilaian

	Parameter	Nilai
1	Sangat tidak setuju	1
2	Tidak setuju	2
3	Cukup	3
4	Sutuju	4
5	Sangat setuju	5

Nilai-nilai yang di berikan pengguna terhadap masing-masing faktor akan dikalikan dengan bobot nilai masing-masing faktor. Yang kemudian dihitung nilai TCF dengan rumus:

$$TCF = 0.6 + (0.01 \times TF)$$

1.c Hitung *Environmental Complexity Factor (ECF)*

Tabel 5. *Environmental Factor*

	Environmental Factor	Bobot
1.	Familiar dengan proses yang digunakan	1.5
2.	Pengalaman aplikasi	0.5
3.	Pengalaman tim terhadap Object	1

Environmental Factor		Bobot
	Oriented	
4.	Kemampuan memimpin analisis	0.5
5.	Motivasi tim	1
6.	Stabilitas kebutuhan	2
7.	Pekerja yang paru waktu	-1
8.	Tingkat kesulitan bahasa pemrograman	-1

Berdasarkan *environmental factor* tersebut pengguna akan memberikan nilai pada setiap *environmental factor*. Dimana pada setiap faktor memiliki parameter penilaian.

Tabel 6. Parameter penilaian

Parameter		Nilai
1	Sangat tidak setuju	1
2	Tidak setuju	2
3	Cukup	3
4	Setujui	4
5	Sangat setuju	5

Nilai-nilai yang di berikan pengguna terhadap masing-masing faktor akan dikalikan dengan bobot nilai masing-masing faktor. Yang kemudian dihitung nilai ECF dengan rumus:

$$ECF = 1.4 + (-0.03 \times EF)$$

Setelah diketahui nilai UUCP, nilai TCF, dan nilai ECF, kemudian dapat dilakukan perhitungan nilai UCP dengan rumus sebagai berikut:

$$UCP = UUCP + TCF + ECF$$

## 2. Perhitungan *Hours of Effort*

Nilai *Hours of Effort* didapatkan dari hasil perkalian antara nilai UCP dengan *Effort Rate* (ER). ER merupakan nilai usaha (staff-hour) yang dibutuhkan tiap satu UCP (Stewart, 2002). Nilai *Effort Rate* (ER) memiliki beberapa variasi berdasarkan penelitian yang dilakukan oleh para ahli. Karner mengusulkan nilai ER 20 man-hours dengan menggunakan tiga data proyek pengembangan perangkat lunak. Schneider mengusulkan nilai ER sebesar 20, 28 dan 36 man-hours menggunakan dasar kompleksitas proyek dengan mengacu pada Technical Complexity Factor (TCF). Clemmons mengusulkan nilai ER sebesar 18 man-hours dengan menggunakan dasar kualitas personil tim dan data historis. Ochodek mendapatkan nilai ER berkisar antara 4 sampai 35 man-hours yang dihitung dari proyek-proyek yang telah dilakukan. Rumus perhitungan nilai *hours of effort* sebagai berikut:

$$Hours\ of\ Effort = UCP \times ER$$

## 3. Perhitungan Estimasi *Distribusi Effort*

Setelah di ketahui nilai *Hours of Effort* maka proses selanjutnya yaitu dilakukan perhitungan estimasi distribusi effort, dimana nilai *Hours of Effort* akan menjadi inputan pada proses ini. Perhitungan estimasi distribusi *effort* bertujuan untuk menghasilkan nilai estimasi *effort* berdasarkan tiap aktivitas dalam pembuatan proyek perangkat lunak pemerintahan. Inputan yang dibutuhkan dalam proses penentuan distribusi effort ini adalah effort per aktivitas yang didapat dari penelitian yang sudah dilakukan sebelumnya dan nilai *Hours of Effort*. Nilai estimasi distribusi effort di peroleh dengan rumus:

$$Distribusi\ Effort = Hours\ of\ Effort \times Effort\ per\ Aktivitas$$

## 4. Perhitungan Biaya Langsung Personil

Berdasarkan estimasi nilai distribusi effort yang sudah diperoleh, langkah selanjutnya dilakukan perhitungan biaya langsung personil. Proses ini bertujuan untuk mendapatkan biaya tenaga kerja langsung yang dibutuhkan dalam pembuatan proyek pengembangan perangkat lunak pemerintahan. Inputan yang dibutuhkan dalam proses perhitungan biaya langsung personil ini adalah nilai estimasi distribusi effort dan tarif biaya yang mengacu pada *Indonesia Salary Guide* yang dikeluarkan oleh kelly service dan INKINDO 2014. Yang kemudian tarif biaya tersebut dikalikan dengan nilai estimasi distribusi effort untuk menghasilkan nilai biaya langsung personil. Perhitungan Biaya Langsung Personil (BLP) dilakukan sebagai berikut:

$$Biaya\ langsung\ personil = Estimasi\ Effort \times Tarif$$

## 5. Perhitungan Biaya Langsung Non Personil

Setelah dilakukan perhitungan biaya langsung personil, selanjutnya akan dilakukan perhitungan biaya langsung non personil. Biaya langsung non personil merupakan biaya-biaya selain tenaga kerja langsung yang dikeluarkan untuk menunjang pelaksanaan kegiatan proyek pengembangan perangkat lunak pemerintahan. Inputan yang dibutuhkan dalam proses ini adalah data pengeluaran biaya dari proyek perangkat lunak yang dibuat. Yang kemudian dari data pengeluaran tersebut jumlah kuantitas item dikalikan dengan harga satuan untuk mendapatkan sub total. Dari subtotal tersebut dijumlahkan semuanya untuk mendapatkan total biaya langsung non personil.

6. Perhitungan Keuntungan

Merupakan proses perhitungan keuntungan yang ingin diperoleh dari pengembangan proyek perangkat lunak pemerintahan yang ditetapkan oleh PPK. Rumus untuk menentukan besarnya keuntungan yaitu:

$$Keuntungan = \% \text{ Keuntungan} \times \text{Total biaya langsung personil}$$

7. Perhitungan Pajak Pertambahan Nilai (PPN)

Pajak yang dikenakan untuk pengadaan proyek pengembangan perangkat lunak yaitu pajak pertambahan nilai sebesar 10% dari total biaya. Cara perhitungan untuk mendapatkan nilai besarnya pajak yaitu dengan mengalikan jumlah total biaya yang didapat dengan besarnya persentase pajak yaitu sebesar 10%. Berikut rumusnya:

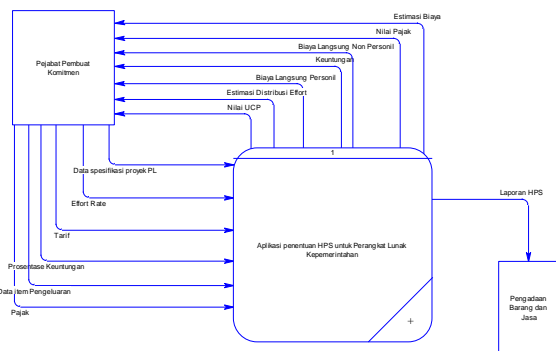
$$PPN = 10 \% \times \text{Total Biaya}$$

8. Perhitungan Total HPS

Setelah diketahui nilai jumlah biaya langsung personil, biaya langsung non personil, keuntungan dan besarnya pajak yang dikenakan. Untuk mendapatkan nilai HPS, Langkah selanjutnya yaitu menjumlahkan seluruh biaya tersebut dengan rumus sebagai berikut:

$$HPS = \text{Biaya langsung personil} + \text{Biaya langsung non personil} + \text{Keuntungan} + \text{Pajak}$$

Dari proses yang sudah di jelaskan tersebut, dibuatlah rancangan context diagram. Seperti yang digambarkan pada gambar 3 berikut ini:

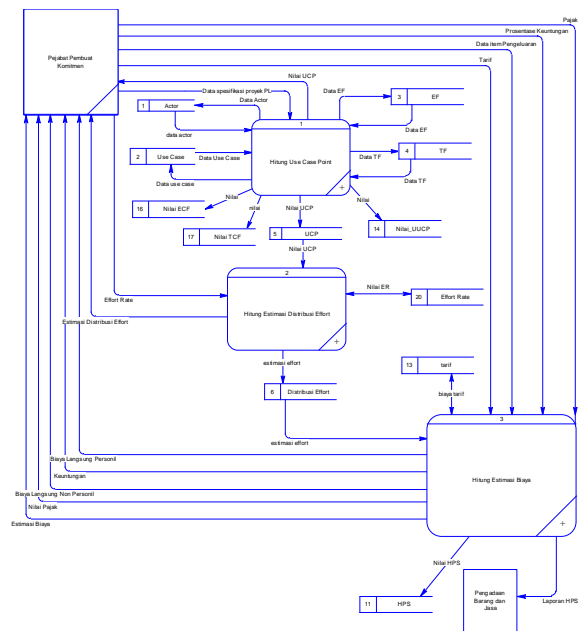


Gambar 3. Context Diagram

Context Diagram dari aplikasi ini melibatkan 2 entity yaitu Pejabat Pembuat Komitmen (PPK) dan Bagian pengadaan barang dan jasa. Pada sistem ini, PPK input data yang kemudian dari data tersebut akan diolah sehingga menghasilkan

informasi, seperti nilai UCP, estimasi effort, biaya langsung personil, keuntungan, biaya langsung non personil, pajak, dan estimasi biaya.

Context diagram memiliki tingkatan atau level yang disebut dengan Data Flow Diagram (DFD) yang menjelaskan lebih detail desain sistem berdasarkan proses dari context diagram. Berikut merupakan DFD level nol dari aplikasi penentuan harga perkiraan sendiri proyek perangkat lunak, dapat dilihat pada gambar 4.



Gambar 4. Data Flow Diagram Level 0

Pengkodean

Aplikasi ini di bangun berbasis web dengan menggunakan bahasa pemrograman Hypertext Preprocessor (PHP) dan tools yang digunakan yaitu Notepad++, MySQL, dan XAMPP

Pengujian Sistem

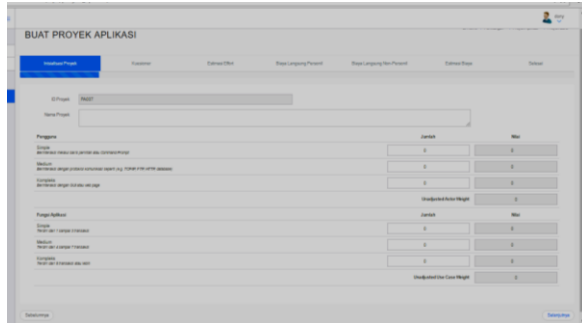
Untuk mengetahui aplikasi sudah berjalan sesuai yang diharapkan, maka perlu adanya pengujian aplikasi. Dalam pengujian aplikasi penentuan harga perkiraan sendiri proyek perangkat lunak ini, uji coba dilakukan secara *blackbox testing*.

HASIL DAN PEMBAHASAN

Aplikasi dibangun untuk memberikan solusi terhadap permasalahan yang ada yaitu aplikasi yang mampu menentukan nilai harga perkiraan sendiri proyek perangkat lunak pemerintahan. Komponen penentuan harga perkiraan sendiri tersebut meliputi biaya langsung personil, biaya langsung non personil, keuntungan, dan pajak.

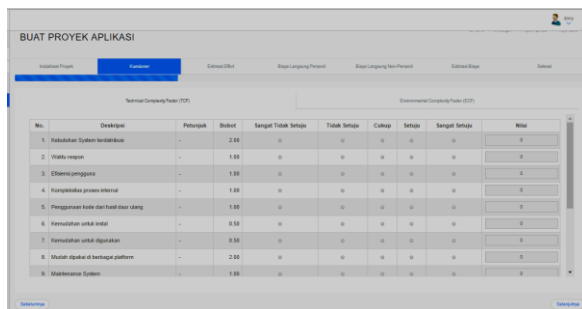


Langkah awal yang harus dilakukan dalam menghitung nilai harga perkiraan sendiri ini yakni melakukan inialisasi kebutuhan data proyek aplikasi yang akan dibuat. Berikut halaman inialisasi proyek dapat dilihat pada gambar 5.



Gambar 5. Tampilan halaman inialisasi data proyek

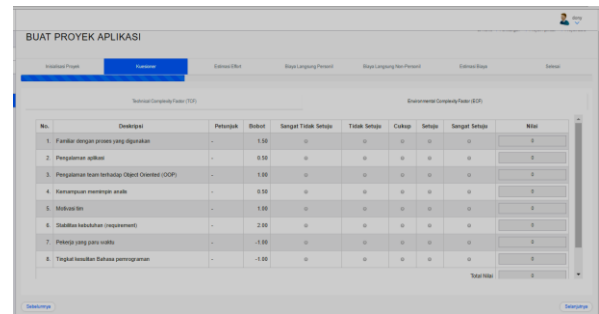
Setelah melakukan inialisasi data proyek perangkat lunak yang akan dibuat dengan menginputkan spesifikasi kebutuhan proyek perangkat lunak yang akan dibuat tahap selanjutnya akan dilakukan penilaian terhadap faktor-faktor teknis yang mempengaruhi proyek pembuatan aplikasi. pengguna memberikan skor pada setiap faktor teknis pembuatan aplikasi. Dimana setiap faktor teknis tersebut memiliki parameter masing-masing yakni sangat tidak setuju, tidak setuju, cukup, setuju, sangat setuju. Setelah pemberian skor dilakukan maka diperoleh nilai dari *technical complexity factor*. Berikut halaman koesioner *technical complexity factor* dapat dilihat pada gambar 6.



Gambar 6. Tampilan halaman kuesioner *technical complexity factor*

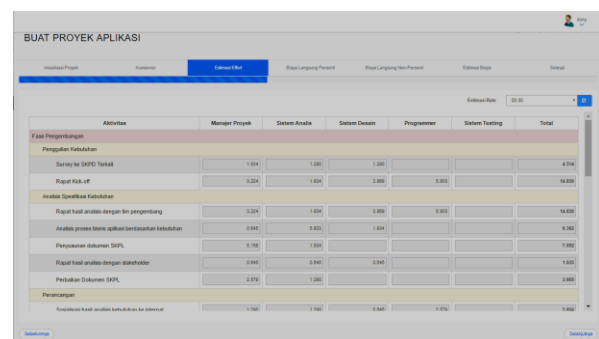
Sama halnya dengan halaman kuesioner *technical complexity factor*, pada halaman kuesioner perhitungan *environmental complexity factor* ini akan dilakukan penilaian terhadap faktor-faktor lingkungan yang mempengaruhi proyek pembuatan aplikasi. pengguna memberikan skor pada setiap faktor lingkungan pembuatan aplikasi. Dimana setiap faktor lingkungan tersebut memiliki parameter

masing-masing yakni sangat tidak setuju, tidak setuju, cukup, setuju, sangat setuju. Setelah pemberian skor dilakukan maka diperoleh nilai dari *environmental complexity factor*. Berikut halaman koesioner *environmental complexity factor* dapat dilihat pada gambar 7.



Gambar 7. Tampilan halaman kuesioner *environmental complexity factor*

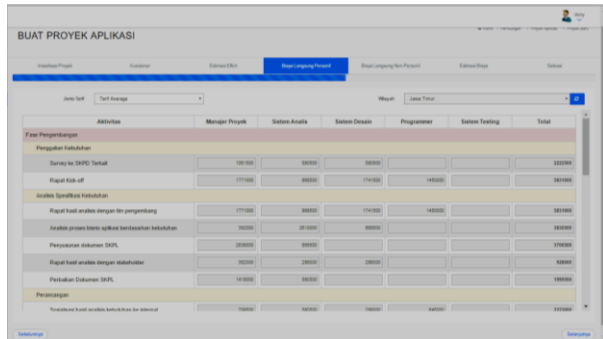
Tahap selanjutnya akan dilakukan perhitungan estimasi distribusi effort yang nantinya akan menghasilkan nilai effort yang dibutuhkan untuk membuat suatu proyek perangkat lunak. Untuk menghasilkan nilai estimasi distribusi effort ini data yang dibutuhkan yaitu nilai UCP yang didapat dari halaman inialisasi data proyek dan *Effort Rate (ER)* yang didapat dari penelitian sebelumnya. Kedua data tersebut akan dikalikan untuk menghasilkan nilai *hours of effort* yang kemudian nilai tersebut akan dikalikan dengan *effort* per aktivitas untuk menghasilkan nilai estimasi distribusi *effort*. Berikut halaman estimasi distribusi *effort* dapat dilihat pada gambar 8.



Gambar 8. Tampilan halaman estimasi distribusi *effort*

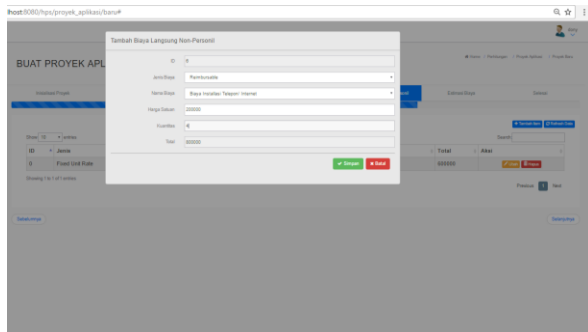
Setelah diketahui nilai estimasi *effort* peraktivitas dan per jabatan maka dapat dilakukan perhitungan estimasi biaya dengan mengkalikan nilai estimasi *effort* dengan standar gaji yang sudah ditentukan.

Standar gaji yang digunakan mengacu pada *Indonesia Salary Guide* yang dikeluarkan oleh Kelly Service dan Pedoman Standar Minimal oleh Ikatan Nasional Konsultan Indonesia (INKINDO). Berikut halaman perhitungan biaya langsung personil dapat dilihat pada gambar 9.



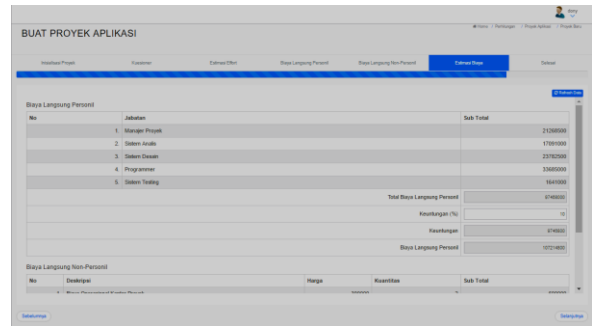
Gambar 9. Tampilan halaman perhitungan biaya langsung personil

Pada tahap ini dilakukan perhitungan biaya-biaya non personil yang dikeluarkan untuk menunjang pembuatan proyek perangkat lunak. Biaya langsung non personil terdiri dari 3 (tiga) komponen yaitu *Reimbursable*, *Fixed Unit Rate*, dan *Lump Sum*. Berikut halaman estimasi distribusi *effort* dapat dilihat pada gambar 10.



Gambar 10. Tampilan halaman perhitungan biaya langsung non personil

Setelah diketahui seluruh komponen biaya penentuan HPS. Tahap selanjutnya yaitu menghitung estimasi biaya untuk mengetahui nilai HPS. Pada halaman ini semua biaya akan di tampilkan, seperti biaya langsung personil, biaya langsung non personil, dan pajak. untuk dapat melihat total keuntungan pengguna harus memasukan persentase keuntungan yang ingin diperoleh dari persentase keuntungan yang dimasukan kemudian diperoleh nilai keuntungan dari pembuatan proyek perangkat lunak. Berikut halaman estimasi distribusi *effort* dapat dilihat pada gambar 11.



Gambar 11. Halaman perhitungan estimasi biaya

Laporan akhir dari penentuan harga perkiraan sendiri proyek perangkat lunak pemerintahan ini terdiri dari informasi total biaya langsung personil, keuntungan, biaya langsung non personil, pajak, dan total nilai HPS. Berikut adalah laporan harga perkiraan sendiri, dapat dilihat pada gambar 12.

No.	Jabatan	Biaya
1	Manajer Proyek	Rp23.941.999,00
2	Sistem Analis	Rp41.800.000,00
3	Sistem Desain	Rp45.480.000,00
4	Programmer	Rp53.861.000,00
5	Sistem Testing	Rp11.107.999,00
Total		Rp166.231.998,00
Keuntungan (%)		30,00%
Keuntungan		Rp49.871.999,00
Biaya Langsung Personil		Rp116.359.997,00

No	Deskripsi	Harga	Qty	Sub Total
1	Biaya Komsumsi	Rp500.000,00	4	Rp2.000.000,00
2	Biaya Perjalanan Dapur	Rp1.800.000,00	1	Rp1.800.000,00
Total Biaya Langsung Non-Personil				Rp3.800.000,00
Total Biaya				Rp120.159.997,00
PPN (10%)				Rp12.015.999,00
GRAND TOTAL				Rp132.175.996,00

Gambar 12. Laporan harga perkiraan sendiri

**Evaluasi**

Berdasarkan hasil uji coba penelitian pada aplikasi penentuan harga perkiraan sendiri proyek perangkat lunak pemerintahan, maka pada tahap ini akan dilakukan evaluasi hasil uji coba penelitian dengan membandingkan antara hasil sebenarnya yang dihasilkan aplikasi dengan hasil yang diharapkan berdasarkan perhitungan manual.

**Inisialisasi data proyek**

Pada uji coba halaman inisialisasi data proyek yang dilakukan berikut cara perhitungan manualnya.

**Perhitungan UAW:**

Tabel 7. Perhitungan nilai *Unadjusted Actor Weight*

Perhitungan <i>Unadjusted Actor Weight</i>			
Typ	Bobot	Jumlah Actor	Bobot x Jumlah Actor
Simple	1	1	1 x 1 = 1
Medium	2	2	2 x 2 = 4
Complex	3	2	3 x 2 = 6

Perhitungan <i>Unadjusted Actor Weight</i>			
Tipe Actor	Bobot	Jumlah Actor	Bobot x Jumlah Actor
Total Nilai UAW			1 + 4 + 6 = 11

**Perhitungan UUCW:**

Tabel 8. Perhitungan nilai *Unadjusted Use Case Weight*

Perhitungan <i>Unadjusted Use Case Weight</i>			
Tipe Use Case	Bobot	Jumlah Use Case	Bobot x Jumlah Use Case
Simple	5	2	5 x 2 = 10
Medium	10	1	10 x 1 = 10
Complex	15	3	15 x 3 = 45
Total Nilai UUCW			10 + 10 + 45 = 65

Uji coba memberikan hasil yang positif. Pada skenario uji coba yang sudah ditentukan output yang diharapkan untuk nilai UAW sebesar 11 dan nilai UUCW sebesar 65 dan berdasarkan hasil sebenarnya menunjukkan hasil yang sesuai dengan skenario yang dibuat. Sehingga dapat disimpulkan bahwa halaman inialisasi data proyek mampu menghasilkan nilai *Unadjusted Actor Weight* (UAW) dan nilai *Unadjusted Use Case Point* (UUCP) sesuai dengan output yang diharapkan yang dibuktikan dengan perhitungan manual.

**Kuesioner *Technical Complexity Factor* (TCF) dan *Environmental Complexity Factor* (ECF)**

Berdasarkan hasil uji coba penelitian pada perhitungan nilai *Technical Complexity Factor* (TCF) dan *Environmental Complexity Factor* (ECF), hasil menunjukkan kesamaan antara output yang sebenarnya dengan output yang diharapkan.

Ketentuan:

Tabel 9. Kategori penilaian *technical factor*

Kategori	Nilai
Sangat Tidak Setuju	1
Tidak Setuju	2
Cukup	3
Setuju	4
Sangat Setuju	5

**Perhitungan *Technical Complexity Factor* (TCF):**

Tabel 10. Hitung nilai setiap *technical factor*

Nama Faktor Teknis	Bobot	Nilai	Bobot x Nilai
Kebutuhan System terdistribusi	2	3	2 x 3 = 6
Waktu respon	1	4	1 x 4 = 4
Efisiensi pengguna	1	5	1 x 5 = 5
Kompleksitas proses internal	1	4	1 x 4 = 4
Penggunaan kode dari hasil daur ulang	1	3	1 x 3 = 3
Kemudahan untuk instal	0.5	5	0.5 x 5 = 2.5
Kemudahan untuk digunakan	0.5	5	0.5 x 5 = 2.5
Mudah dipakai di berbagai platform	2	4	2 x 4 = 8
<i>Maintenance System</i>	1	4	1 x 4 = 4
Proses paralel	1	4	1 x 4 = 4
Fitur keamanan	1	5	1 x 5 = 5
Akses pihak ke-3	1	3	1 x 3 = 3
Pelatihan pengguna	1	4	1 x 4 = 4
Nilai TF			55
TCF=0.6+(0.01xTF)			0.6+(0.01*55)= 1.15

**Perhitungan *Environmental Complexity Factor*:**

Ketentuan:

Tabel 11. Hitung nilai setiap *environmental factor*

Nama Faktor	Bobot	Nilai	Bobot x Nilai
Familiar dengan proses yang digunakan	1.5	4	1.5 x 4 = 6
Pengalaman aplikasi	0.5	4	0.5 x 4 = 2
Pengalaman team terhadap Object Oriented (OOP)	1	3	1 x 3 = 3
Kemampuan memimpin analisis	0.5	3	0.5 x 3 = 1.5
Motivasi tim	1	5	1 x 5 = 5
Stabilitas	2	4	2 x 4 = 8



Nama Faktor	Bobot	Nilai	Bobot x Nilai
kebutuhan (requirement)			
Pekerja yang paru waktu	-1	3	-1 x 3 = -3
Tingkat kesulitan Bahasa pemrograman	-1	3	-1 x 3 = -3
			19,5
ECF = 1.4 + (-0, 03*EF)			1.4+(0.03*19.5) = 0.815

Output yang sebenarnya menunjukkan bahwa nilai TCF sebesar 1,15 dan ECF sebesar 8,15. Dimana berdasarkan perhitungan manual diperoleh output yang sama. Dengan demikian pada halaman ini hasil sesuai dengan skenario yang sudah ditetapkan

**Estimasi Distribusi Effort**

Hasil uji coba perhitungan estimasi distribusi effort menunjukkan kesamaan antara hasil yang sebenarnya dengan hasil yang diharapkan. Sebagai contoh dimana manajer proyek memiliki estimasi sebesar 28,9 sesuai dengan out yang dihasilkan melalui perhitungan manual.

**Perhitungan Distribusi Effort**

Effort Rate (ER) yang digunakan berdasarkan usulan karner yaitu sebesar 20.

$$UCP = UUCP \times TCF \times ECF = 76 \times 1.15 \times 0.815 = 71,231$$

$$\text{Hours of effort} = ER \times UCP = 20 \times 71,231 = 1.424,62$$

Tabel 12. Hitung estimasi distribusi effort

Aktivitas House of Effort = 1424,62		Fase Pengembangan	Fase Sedang Berlangsung	Quality dan Testing	Total
Manajer proyek	Effort	8,8%	9,8%	1,7%	20,3%
	Estimasi Effort	125,37	139,61	24,22	289,2

Aktivitas House of Effort = 1424,62		Fase Pengembangan	Fase Sedang Berlangsung	Quality dan Testing	Total
Sistem Analis	Effort	10,4%	3,8%	2,7%	16,9%
	Estimasi Effort	148,16	54,14	38,46	240,8
Sistem Desain	Effort	10,5%	4,9%	2,2%	17,6%
	Estimasi Effort	149,59	69,81	31,34	250,7
Programmer	Effort	23,8%	10,6%	5,6%	40%
	Estimasi Effort	339,06	151,01	79,78	569,9
Sistem Testing	Effort	4,3%	0,9%		5,2%
	Estimasi Effort	61,2587	12,82		74,1
Total Effort		57,8%	30%	12,2%	100%
Total Estimasi Effort		823,43	427,39	173,8	1.424,6

**Biaya Langsung Personil**

Pada uji coba halaman inialisasi data proyek yang dilakukan, uji coba memberikan hasil yang positif. Semua fungsi dan rumus perhitungan yang terdapat pada halaman inialisasi data proyek berjalan sesuai skenario yang telah ditetapkan. Aplikasi mampu menghasilkan nilai estimasi biaya langsung personil sesuai dengan output yang diharapkan.

**Perhitungan Biaya Langsung Personil**

Tabel 4.14 Perhitungan Nilai Tarif

Jabatan	Tarif Maksimum	Rate Jawa timur = 0,787
Manajer Proyek	Rp 600.000	Rp 472.200
Sistem Analis	Rp 500.000	Rp 393.500
Sistem Desain	Rp 500.000	Rp 393.500
Programmer	Rp 300.000	Rp 236.100
Sistem Testing	Rp 200.000	Rp 157.400

Tabel 4.15 Hitung biaya langsung personal

Aktivitas House of Effort = 1424,62		Fase Pengembangan	Fase Sedang Bertanggung	Quality dan Testing	Total
Tarif 472.200	Manajer proyek Σ Effort	125,37	139,61	24,22	289,2
	Biaya	59.198.090	65.925.145	11.435.995	136.559.229
Tarif 393.500	Sistem Analisis Σ Effort	148,16	54,14	38,46	240,8
	Biaya	Rp58.301.149	Rp21.302.343	15.135.875	94.739.367
Tarif 393.500	Sistem Desain Σ Effort	149,59	69,81	31,34	250,7
	Biaya	Rp58.861.737	Rp27.468.811	12.332.935	98.663.483
Tarif 236.100	Programmer Σ Effort	339,06	151,01	79,78	569,9
	Biaya	Rp80.051.962	Rp35.653.395	18.835.756	134.541.113
Tarif 157.400	Sistem Testing Σ Effort	61,2587	12,82		74,1
	Biaya	Rp9.642.113	Rp2.018.117		11.660.230
Total Estimasi Effort		823,43	427,39	173,8	1.424,6
Total Biaya		Rp266.055.051	Rp152.367.810	57.740.561	476.149.153

Hasil uji coba perhitungan biaya langsung personal menunjukkan kesamaan antara hasil yang sebenarnya dengan hasil yang diharapkan yaitu sebesar 476.149.153, sesuai dengan output yang dihasilkan melalui perhitungan manual.

**Biaya Langsung Non personal**

Hasil uji coba perhitungan biaya langsung non personal menunjukkan kesamaan antara hasil yang sebenarnya dengan hasil yang diharapkan. Hasil sebenarnya menunjukkan total nilai biaya langsung non personal sebesar Rp 3.800.000, sedangkan output yang diharapkan sebesar Rp 3.800.000. Sehingga

dapat disimpulkan bahwa aplikasi mampu memberikan informasi total biaya langsung non personal sesuai dengan skenario yang sudah ditetapkan.

**Perhitungan**

Tabel 16. Perhitungan biaya langsung non personal

Nama Biaya	Harga	Kuantitas	Jumlah
Biaya Komunikasi	500.000	4	2.000.000
Biaya Perjalanan Darat	300.000	6	1.800.000
Total Biaya Langsung Personal			3.800.000

**Estimasi Biaya**

Berdasarkan hasil uji coba penelitian pada perhitungan estimasi biaya, hasil menunjukkan kesamaan antara output yang sebenarnya dengan output yang diharapkan. Output yang sebenarnya menunjukkan bahwa aplikasi dapat menampilkan nilai total biaya langsung personal, biaya langsung non personal, keuntungan, dan pajak sesuai dengan skenario yang sudah ditetapkan. Dimana hasil sebenarnya pada biaya langsung personal menunjukkan nilai sebesar Rp476.149.153, keuntungan yang diperoleh sebesar 10% dengan nilai Rp47.614.915 biaya langsung non personal sebesar Rp 3.800.000, pajak pertambahan nilai sebesar 10% dengan nilai Rp52.756.406, hingga hasil akhir diperoleh total nilai harga perkiraan sendiri sebesar Rp580.320.474 memiliki kesesuaian dengan perhitungan manual yang dilakukan.

**Perhitungan Estimasi Biaya**

Biaya Langsung Personal	Rp476.149.153
Keuntungan sebesar 10%	Rp47.614.915
Total Biaya Langsung Personal	<b>Rp523.764.068</b>
Total Biaya Langsung Non Personal	<b>Rp3.800.000 +</b>
Total Biaya PPN 10%	<b>Rp527.564.068</b>
	<b>Rp52.756.406 +</b>
<b>GRAND TOTAL</b>	<b>Rp580.320.474</b>

**Laporan HPS**

Pada form laporan HPS ini hasil uji coba diperoleh kesesuaian antara output yang sebenarnya dengan output yang diharapkan. Aplikasi mampu mencetak laporan dan informasi yang terdapat dalam laporan sesuai dengan yang diharapkan.

## KESIMPULAN

Dari kegiatan uji coba dan evaluasi yang dilakukan pada 3 (tiga) data proyek pengembangan perangkat lunak, maka dapat disimpulkan bahwa aplikasi mampu menghasilkan nilai *Unadjusted Actor Weight* (UAW) untuk masing-masing proyek pengembangan perangkat lunak yaitu proyek A sebesar 12, Proyek B sebesar 12, dan proyek C sebesar 9. Sedangkan nilai *Unadjusted Use Case Weight* (UUCW) yaitu proyek A sebesar 325, Proyek B sebesar 290, dan proyek C sebesar 185. Aplikasi juga mampu menghasilkan nilai *Technical Complexity Factor* (TCF) proyek A sebesar 1,045 proyek B sebesar 1,065 dan proyek C sebesar 1,045. Nilai *Environmental Complexity Factor* (ECF) proyek A sebesar 0,875 proyek B sebesar 0,815 dan Proyek C sebesar 0,650. Kemudian dari nilai-nilai tersebut aplikasi mampu menghasilkan estimasi *effort* untuk masing-masing proyek pengembangan perangkat lunak untuk proyek A 6160, proyek B 5240, dan proyek C 3760. Dari nilai estimasi *effort* yang telah diketahui kemudian aplikasi dapat menghitung biaya langsung personil. Selain itu aplikasi juga mampu menghitung biaya langsung non personil, keuntungan dan Pajak Pertambahan Nilai (PPN). Sehingga dari komponen-komponen nilai tersebut nilai HPS dapat diperoleh.

## SARAN

Saran yang dapat disampaikan untuk pengembangan aplikasi penentuan harga perkiraan sendiri ini, antara lain:

1. Tampilan dari sistem dapat dibuat lebih memudahkan pengguna atau *user friendly*.
2. Untuk *contents* dari aktivitas distribusi *effort* masih menggunakan pengembangan perangkat lunak skala kecil menengah, ke depannya diharapkan dapat menyediakan seluruh skala dari kecil sampai besar.
3. Penelitian mengenai distribusi *effort* menggunakan model masih sedikit, diharapkan melakukan uji coba dengan model lainnya, sehingga ke depannya dapat mendukung aplikasi ini.

## RUJUKAN

- Carroll, Edward R. 2005. *Estimating Software Based on Use Case Points*. 2005 Object-Oriented, Programming, Systems, Languages, and Object Oriented Programming Systems Languages and Applications (OOPSLA) Conference, San Diego, CA, pp.257–265.
- Clemmons, Roy K. 2006. *Project Estimation With Use Case Point*. Diversified Technical Service Inc.

- Fatimah. (2014). *Penyusunan Harga Perkiraan Sendiri (HPS) dengan Menggunakan Sistem Activity Based Costing (ABC)*: Widya Swara Balai Diklat Keuangan Malang. Online : [www.bppk.depkeu.go.id](http://www.bppk.depkeu.go.id). Diakses pada tanggal 22 februari 2016
- Ikatan Nasional Konsultasi Indonesia (INKINDO). 2014. *Pedoman standar Minimal 2014 Biaya Langsung Personil dan Biaya Langsung Non Personil untuk Kegiatan Jasa Konsultasi*. Online : <http://www.inkindo.org>. Diakses pada tanggal 08 Juli 2015
- Karner, Gustav. 1993. *Resource Estimation for Objectory Projects*. Objective Systems SF AB.
- Kelly. 2016. *2016 Salary Guide*. Kelly Service Indonesia
- Nageswaran, Suresh. 2001. *Test Effort Estimation Using Use Case Points*. June 2001 <[www.cognizant.com/cogcommunity](http://www.cognizant.com/cogcommunity)
- Peraturan Presiden Republik Indonesia No 70 Tahun 2012. Online : <http://www.lkpp.go.id/v2/files/content/file/09082012110709BT%20Perpres%200702012.pdf> Diakses pada tanggal 15 Juli 2015.