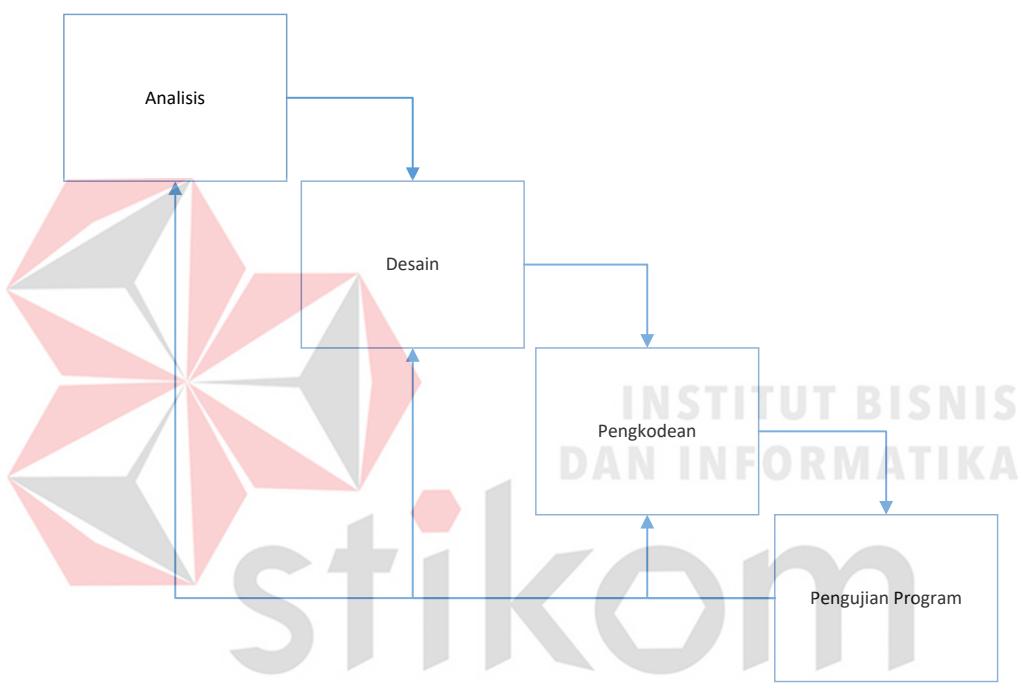


## **BAB III**

### **ANALIS DAN PERANCANGAN SISTEM**

Pada tahap analisis dan perancangan sistem ini akan dijelaskan tahap-tahap dalam penggerjaan penelitian ini, sehingga penelitian dapat berjalan dengan baik. Berikut tahap penelitian akan ditunjukkan melalui gambar 3.1 sebagai berikut.



Gambar 3.1 Tahapan Penelitian Tugas Akhir

#### **3.1 Analisis Sistem**

Tahap analisis sistem merupakan tahap yang sangat penting dalam membangun suatu perangkat lunak yang baik dan benar. Pada tahap ini yang dilakukan adalah mengidentifikasi, mengevaluasi, dan mempelajari apakah sistem yang ada sudah berjalan dengan baik. Dalam pengumpulan data sebagai bahan pendukung penyusunan tugas akhir, maka dibutuhkan adanya pemahaman terhadap data dan informasi terkait. Data dan informasi didapat melalui studi pustaka yang dilakukan pada penelitian terdahulu.

Selain itu, dalam penelitian ini juga membutuhkan beberapa informasi sebagai berikut:

- a. Informasi tentang *Software Development Life Cycle* (SDLC).
- b. Informasi terkait distribusi dan estimasi *effort*.
- c. Informasi terkait metode *Use Case Point* (UCP).
- d. Informasi terkait Pajak Pertambahan Nilai (PPn)

### **3.1.1 Identifikasi Masalah**

Dalam penentuan harga perkiraan sendiri untuk proyek pengembangan perangkat lunak kepemerintahan selama ini belum memiliki acuan. Dalam penentuan HPS tersebut PPK sebagai penentu besarnya nilai HPS 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.

Berdasarkan survey yang dilakukan pada penelitian yang sudah dilakukan sebelumnya, peneliti menemukan beberapa penelitian untuk membantu PPK dalam menentukan besarnya nilai HPS. Akan tetapi penelitian tersebut hanya sampai pada pembuatan model saja. Berikut adalah hasil survey penelitian yang sudah dilakukan:

- a. Nilai Distribusi *Effort*

Menentukan aktivitas-aktivitas yang terjadi dalam pembuatan proyek perangkat lunak kepemerintahan, dimana setiap aktivitas memiliki *man hours*,

*man days* yaitu butuh berapa waktu dalam melakukan satu aktivitas dan berapa orang yang mengerjakan.

*Effort* (usaha) dari sebuah proyek pengembangan perangkat lunak dapat didefinisikan sebagai waktu yang dikonsumsi oleh proyek yang dinyatakan dengan hitungan orang dalam jam, hari, bulan, atau tahun tergantung pada ukuran proyek, sebagai contoh adalah  $effort = people * time$  (Chatters, 1999) dalam (Haapio, 2011).

a. Nilai *Use Case Point*

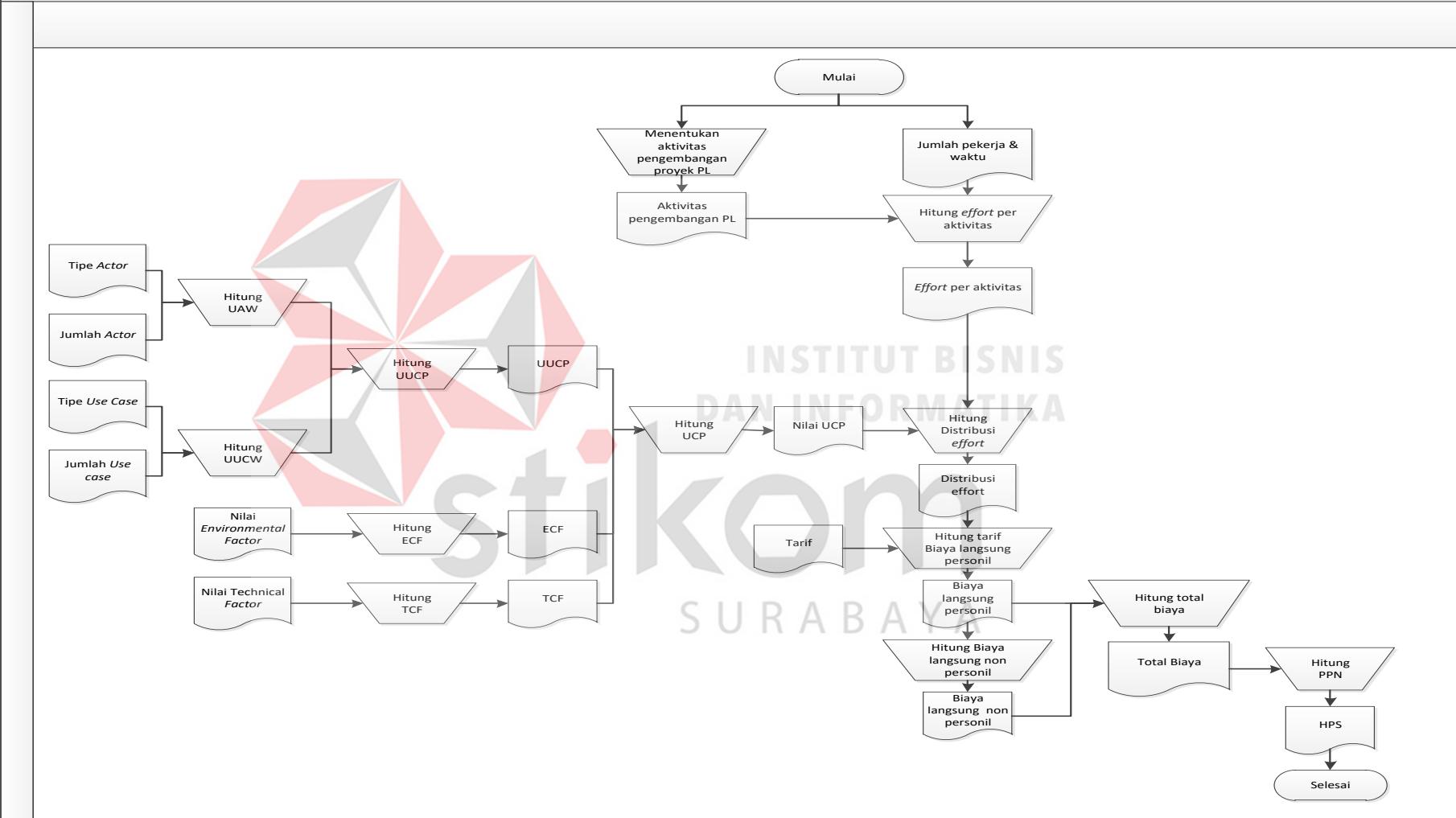
Nilai yang dihasilkan dari perhitungan metode use case point yaitu dapat memberikan estimasi yang hampir mendekati estimasi sebenarnya yang dihasilkan dari pengalaman pembuatan atau pengembangan software

b. Nilai estimasi biaya

Dengan diketahuinya nilai dari estimasi effort tersebut, maka dapat dilanjutkan untuk perhitungan selanjutnya, yaitu perhitungan biaya proyek pengembangan perangkat lunak.

Berikut adalah docflow penentuan harga perkiraan sendiri untuk proyek pengembangan perangkat lunak yang akan ditunjukkan pada gambar 3.2.

Docflow Penetuan Harga Perkiraan Sendiri Proyek Pengembangan perangkat lunak



Gambar 3.2 Document Flow Penentuan Harga Perkiraan Sendiri

Berdasarkan permasalahan di atas maka dibutuhkan suatu aplikasi untuk menentukan nilai HPS yang dapat digunakan oleh pemerintah yang ingin menggunakannya. Aplikasi penentuan HPS akan dirancang berbasis web dengan menggunakan metode UCP. Kemudian aplikasi tersebut akan digunakan oleh PPK dalam menetapkan HPS untuk proyek perangkat lunak kepemerintahan dimasa mendatang.

### **3.1.2 Prosedur menentukan HPS**

Prosedur penyusunan HPS dimulai dari mengidentifikasi besarnya pagu dana dari Daftar Isian Pelaksanaan Anggaran (DIPA) dan atau Petunjuk Operasional (PO). Setelah diketahui jumlah dana, PPK membuat rencana spesifikasi kebutuhan perangkat lunak (SKPL) yang akan dibangun. Dokumen SKPL digunakan sebagai masukan untuk menghasilkan estimasi HPS dengan aplikasi penentuan HPS. Komponen-komponen terdiri dari tiga aktivitas biaya yaitu:

1. Menghitung biaya langsung personil
2. Menghitung biaya langsung non personil
3. Dan menghitung besarnya pajak.

### **3.1.3 Analisis Kebutuhan**

Berdasarkan analisis yang dilakukan bahwa dalam pengadaan perangkat lunak pemerintah, PPK sebagai penentu besarnya anggaran membutuhkan suatu acuan dalam menentukan nilai HPS untuk proyek pengembangan perangkat lunak kepemerintahan. Informasi yang dibutuhkan oleh PPK dalam menentukan besarnya HPS meliputi:

- a. Informasi mengenai total biaya langsung personil

- b. Informasi mengenai total biaya langsung non personil
- c. Informasi mengenai keuntungan yang wajar
- d. Informasi mengenai total biaya pajak
- e. Informasi mengenai total HPS

### 3.2 Perancangan Sistem

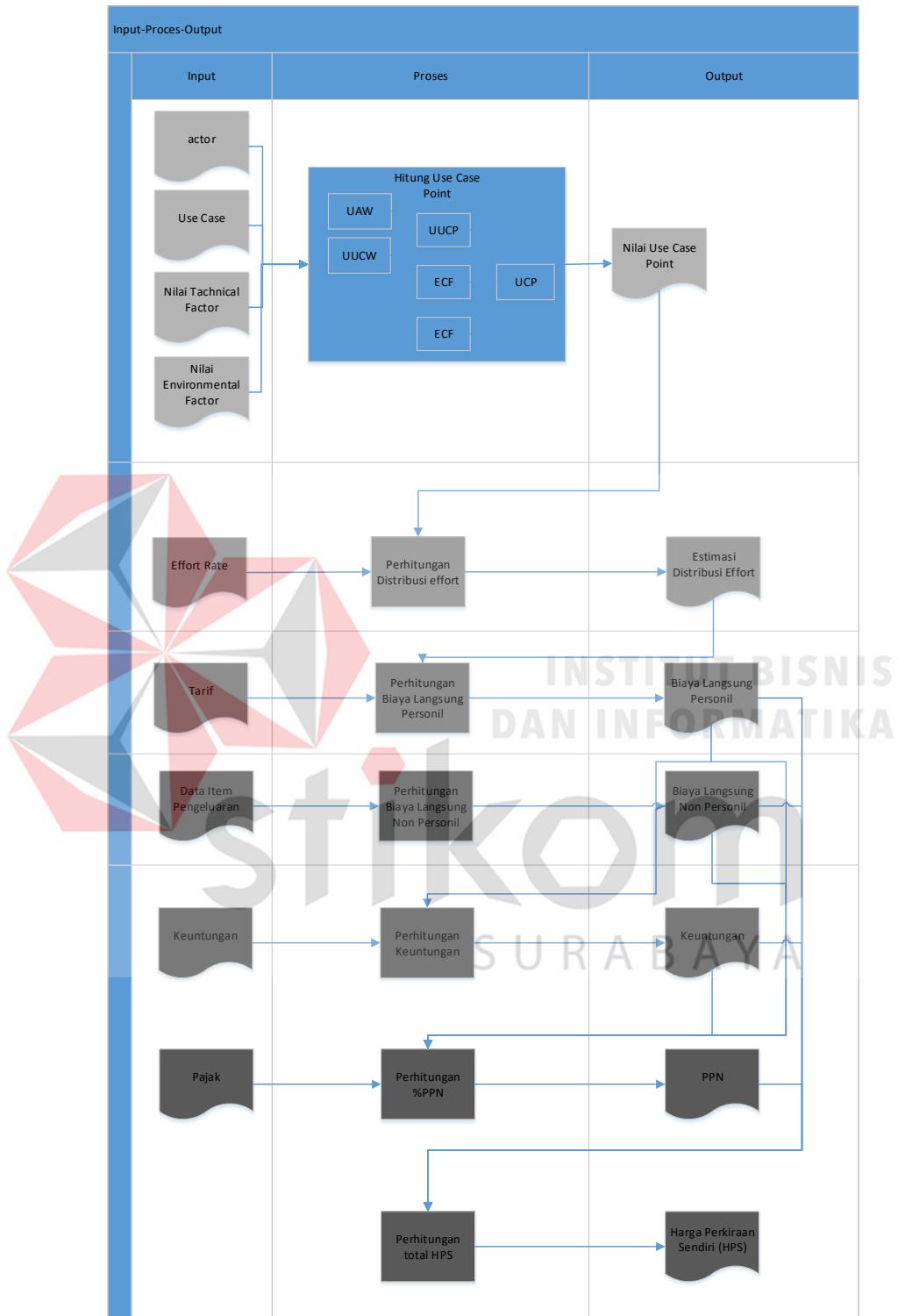
Desain sistem merupakan tahap persiapan implementasi sistem. Pada tahap ini akan dilakukan perancangan tentang bagaimana aplikasi akan dibuat dengan menyesuaikan solusi permasalahan yang sudah dijabarkan sebelumnya.

Pada tahap desain sistem ini dituangkan ke dalam beberapa tahapan, yaitu meliputi:

1. Diagram *Input Process Output* (IPO)
2. *Context Diagram*
3. *Data Flow Diagram*
4. *Conceptual Data Model*, dan
5. *Physical Data Model*.

Tahapan-tahapan tersebut akan berfungsi untuk melihat keterkaitan dan kesesuaian antara data masukan dan proses pengolahan data dengan keluaran berupa hasil penelitian.

### 3.2.1 Model Pengembangan Sistem



Gambar 3.3 Desain IPO Pembuatan Aplikasi Penentuan HPS untuk Proyek

Pengembangan Perangkat Lunak Kepemerintahan.

Berikut ini merupakan penjelasan mengenai *Input, Process, Output* yang ada pada gambar 3.3 diatas

#### A. *Input*

##### 1) *Actor*

*Actor* merupakan pengguna atau *user* yang akan menggunakan aplikasi penentuan HPS perangkat lunak kepemerintahan. Data *actor* dibutuhkan untuk mengklasifikasikan *actor* kedalam kategori *simple, medium, complex* berdasarkan media interaksi pengguna. Dimana data ini akan digunakan untuk menghitung nilai *Unadjusted Actor Weight* dalam penentuan nilai *Use Case Point*. Contoh data *actor* adalah sebagai berikut:

Tipe <i>Actor</i>	:	<i>Complex</i>
Bobot	:	3
Deskripsi	:	Berinteraksi dengan GUI atau Web Page
Jumlah <i>Actor</i>	:	4

##### 2) *Use Case*

*Use case* merupakan kegiatan atau transaksi yang ada di dalam aplikasi. Data *use case* sama dengan data *actor* dibutuhkan untuk mengklasifikasikan *use case* kedalam kategori *simple, medium, complex*. Yang kemudian data tersebut digunakan untuk menghitung *Unadjusted Use Case Weight* pada penentuan nilai *Use Case Point*. Berikut adalah contoh data *use case*:

Tipe <i>Use Case</i>	:	<i>Complex</i>
Bobot	:	15
Deskripsi	:	Menggunakan > 7 transaksi
Jumlah <i>Use Case</i>	:	8

### 3) Nilai *Technical Factor*

Merupakan faktor teknis yang mempengaruhi dalam pembuatan perangkat lunak atau aplikasi. Data ini dibutuhkan untuk menghitung nilai *Technical Complexity Factor* dalam menentukan nilai *Use Case Point*. Contoh data *Technical Factor* adalah sebagai berikut:

Faktor teknis : Distribusi kebutuhan system

Bobot : 2

Nilai : 3

### 4) Nilai *Environmental Factor*

Merupakan faktor lingkungan yang mempengaruhi pembuatan perangkat lunak atau aplikasi. Data ini dibutuhkan untuk menghitung nilai *Environmental Complexity Factor* dalam menentukan nilai *Use Case Point*. Contoh data *Environmental Factor* adalah sebagai berikut:

Faktor Lingkungan : Keakraban dengan Proyek yang akan dikerjakan

Bobot : 1,5

Nilai : 3

### 5) *Effort Rate*

*Effort Rate* merupakan hasil dari penelitian yang dilakukan sebelumnya. ER merupakan nilai usaha (staff-hour) yang dibutuhkan tiap satu UCP. Contoh data effort rate adalah sebagai berikut:

Nilai Effort Rate : 20

## 6) Tarif

Data tarif dibutuhkan sebagai acuan atau pedoman harga dalam menentukan besarnya biaya yang dikeluarkan. Dimana untuk tariff dasar menggunakan pedoman minimum yang diterbitkan oleh INKINDO. Contoh data tarif adalah sebagai berikut:

Kota : Jakarta

Jenis Tarif : Maksimum

Nama Profesi : Project Manager

Billing rate : Rp 943. 688

## 7) Data item pengeluaran

Data item pengeluaran merupakan biaya pengeluaran baik pembelian perlengkapan maupun peralatan yang dilakukan selama proyek pengembangan perangkat lunak dilakukan. Contoh data item pengeluaran adalah sebagai berikut:

Jenis Biaya : Fixed Unit Rate

Item Pengeluaran : Biaya Komunikasi

Kuantitas : 3

Harga Satuan : 2.000.000

Sub Total : 6.000.000

## 8) Persentase Keuntungan

Merupakan jumlah prosentase keuntungan yang ingin diperoleh dari pembuatan proyek aplikasi. Contoh data persentase keuntungan adalah sebagai berikut:

Persentase Keuntungan : 10%

## 9) Pajak

Pajak yang dikenakan dalam proyek pengembangan perangkat lunak ini adalah pajak pertambahan nilai (PPN) yaitu sebesar 10%.

Nama Pajak : Pajak Pertambahan Nilai (PPN)

Persentase Pajak : 10%

## B. Process

### 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). Dimana dalam menentukan nilai UCP ada beberapa sub perhitungan yang harus dilakukan yaitu 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:

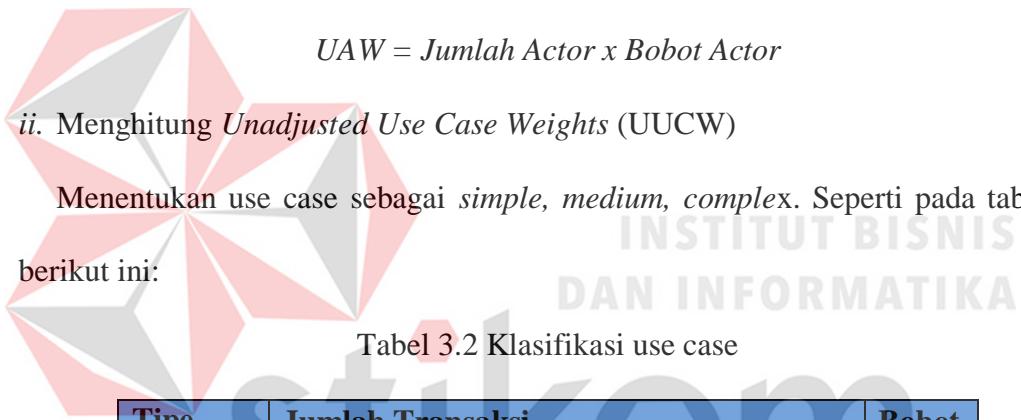
##### i. Menghitung *Unadjusted Actor Weights* (UAW)

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

Tabel 3.1 Klasifikasi aktor

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

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



Tipe	Jumlah Transaksi	Bobot
Simple	Didefinisikan dengan API	5
Average	Berinteraksi melalui Protokol, seperti TCP/IP	10
Complex	Berinteraksi dengan GUI atau Web Page	15

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

$$UUCW = Jumlah Use Case \times Bobot Use Case$$

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 faktor teknis yang mempengaruhi dalam pembuatan proyek perangkat lunak kepemerintahan.

Tabel 3.3 *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 3 4Tabel 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 3.5 Environmental Factor

<b>Environmental Factor</b>		<b>Bobot</b>
1.	Familiar dengan proses yang digunakan	1.5
2.	Pengalaman aplikasi	0.5
3.	Pengalaman tim terhadap Object Oriented	1
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 3.6 Parameter penilaian

<b>Parameter</b>		<b>Nilai</b>
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 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). Clemons 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 kepemerintahan. 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:

$$\text{Distribusi Effort} = \text{Hours of Effort} \times \text{Effort per Aktivitas}$$

#### 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 kepemerintahan. 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:

$$\text{Biaya langsung personil} = \text{Estimasi Effort} \times \text{Tarif}$$

#### 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

kepemerintahan. 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 kepemerintahan yang ditetapkan oleh PPK. Rumus untuk menentukan besarnya keuntungan yaitu:

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

## 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 mengkalikan jumlah total biaya yang didapat dengan besarnya persentase pajak yaitu sebesar 10%. Berikut rumusnya:

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

## 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:

$$\begin{aligned}
 HPS = & Biaya langsung personil + Biaya langsung non personil + Keuntungan \\
 & + Pajak
 \end{aligned}$$

### C. Output

#### 1) Nilai Use Case Point

Nilai *use case point* merupakan nilai yang dibutuhkan dalam menentukan besarnya estimasi nilai distribusi *effort*. Dalam proses perhitungan nilai *use case point* ini akan ditampilkan nama proyek, nilai UUCP, nilai TCF, nilai ECF, dan nilai UCP. Contoh nilai *use case point* adalah sebagai berikut:

Tabel 3.7 Tampilan Nilai *Use Case Point*

<b>Nilai <i>Use Case Point</i></b>			
Nama Proyek : aplikasi desain untuk UI Digital Information Display Pemkot Bekasi			
<b>Nilai UUCP</b>	<b>Nilai TCF</b>	<b>Nilai ECF</b>	<b>Nilai UCP</b>
435	1,015	0,89	392

#### 2) Estimasi Distribusi *Effort*

Nilai estimasi distribusi *effort* merupakan nilai yang dibutuhkan untuk menentukan estimasi biaya setelah dikali dengan tarif biaya yang terdiri dari kelompok aktivitas, sub aktivitas, dan persentase *effort* untuk masing-masing aktivitas. Contoh estimasi distribusi *effort* adalah sebagai berikut:

Tabel 3.8 Laporan Estimasi Distribusi Effort

<b>Estimasi Distribusi Effort</b>		
<b>Proyek Pengembangan Perangkat lunak Kepemerintahan</b>		
<b>No</b>	<b>Kelompok Aktivitas</b>	<b>% Effort</b>
1.	Fase Pengembangan	
a.	Penggalian Kebutuhan	7,5%
b.	Analisis Spesifikasi Kebutuhan	17,5%
<b>Total</b>		25%

### 3) Biaya Langsung Personil

Laporan biaya langsung personil merupakan laporan biaya tenaga kerja langsung yang dikeluarkan dalam proyek pengembangan perangkat lunak kepemerintahan. Laporan biaya langsung personil terdiri dari posisi, jumlah orang, lama kerja, billing rate, sub total, dan total biaya langsung personil.

Tabel 3.9 Laporan Biaya Langsung Personil

<b>Biaya Langsung Personil</b>				
Proyek pembuatan aplikasi desain untuk UI Digital Information Display				
Pemkot Bekasi				
Wilayah Proyek: Bekasi				
No	Posisi	Estimasi Effort	Tarif	Subtotal
1.	Manajer Proyek	1.017	266.000	Rp 270.488.556
2.	Sistem Analis	821	136.000	Rp 111.648.302
3.	Programmer	2.955	89.400	Rp 264.177.973
<b>Total Biaya Langsung personil</b>				Rp 646.314.831

### 4) Keuntungan

Keuntungan merupakan keuntungan yang diambil dari pengembangan proyek perangkat lunak kepemerintahan yang ditetapkan oleh PPK.

Tabel 3.10 Laporan Keuntungan

<b>Biaya Langsung Personil</b>				
Proyek pembuatan aplikasi desain untuk UI Digital Information Display				
Pemkot Bekasi				
Wilayah Proyek: Bekasi				
No	Posisi	Estimasi Effort	Tarif	Subtotal
1.	Manajer Proyek	1.017	266.000	Rp 270.488.556
2.	Sistem Analis	821	136.000	Rp 111.648.302
3.	Programmer	2.955	89.400	Rp 264.177.973
<b>Total Biaya Langsung personil</b>				Rp 646.314.831
<b>Keuntungan (%)</b>				10%
<b>Total Keuntungan</b>				Rp 64.631.483

### 5) Biaya Langsung Non Personil

Biaya langsung non personil merupakan laporan biaya selain tenaga kerja langsung. Biaya langsung non personil terdiri dari informasi item pengeluaran, volume, kuantitas, Harga/satuan, sub total dan total biaya langsung non personil.

Tabel 3.11 Laporan langsung non personil

<b>Biaya Langsung Non Personil</b>				
Proyek pembuatan aplikasi desain untuk UI Digital Information Display Pemkot Bekasi				
No	Item Pengeluaran	Kuantitas	Harga/Satuan	Sub Total
1.	Alat tulis habis pakai	3	Rp 2.480.550	Rp 7.441.650
2	Biaya Komunikasi	3	Rp 2.013.200	Rp 6.039.600
<b>Total Biaya Langsung Non Personil</b>				Rp 13.481.250

### 6) Laporan HPS

Laporan HPS merupakan laporan akhir yang dihasilkan dari aplikasi penentuan harga perkiraan sendiri ini. Berikut adalah contoh output dari laporan HPS yang akan ditunjukkan pada tabel 3.12, 3.13, 3.14 yaitu:

Tabel 3.12 Laporan Detil Estimasi Biaya

No	Aktivitas	Project Manager			SA/SD			Programmer			Estimasi Biaya	
		Tarif Rp 266.000			Tarif Rp 136.000			Tariff Rp 89.400				
		%Effort	Estimasi effort	Sub jumlah	%Effort	Estimasi effort	Sub jumlah	%Effort	Estimasi Effort	Sub jumlah		
1.	<b>Pengalian Kebutuhan</b>											
a.	Survey ke SKPD terkait	0.3%	19	5.027.046	0.2%	11	1.435.308				6.462.354	
2.	<b>Analisis Spesifikasi Kebutuhan</b>											
a	Rapat hasil analisis dengan tim pengembang	0.4%	27	7.266.050	0.4%	27	3.714.228	0.9%	55	4.934.383	15.914.661	
b	Analisis proses bisnis aplikasi berdasarkan kebutuhan	0.1%	8	2.185.089	0.9%	55	7.402.848				9.587.937	
3.	<b>Perancangan</b>											
a.	Sosialisasi hasil analisis kebutuhan ke internal	0.2%	14	3.633.025							3.633.025	
4.	<b>Implementasi</b>											
a.	Ekseskuksi kode program							5.0%	310	27.673.980	27.673.980	
5.	<b>Testing dan Integrasi</b>											
a.	Pengujian dan Intergrasi dengan metode <i>whitebox</i> dan <i>blackbox</i>	0.4%	27	7.266.050	0.4%	27	3.714.228	1.3%	82	7.388.385	18.368.663	
5.	<b>Pnyerahan dan Implementasi</b>											
a.	Rapat penerimaan kesiapan aplikasi dengan Stakeholder	0.7%	44	11.615.518	0.7%	44	5.442.049	1.3%	82	7.388.385	24.445.952	
b.	Serah terima aplikasi dan database	0.4%	27	7.266.050	0.4%	27	3.714.228				10.980.278	
<b>Total</b>		2.5%	166	44.258.828	3,0%	191	25.422.889	8.5%	529	47.385.133	117.066.850	

Tabel 3.13 Sub Laporan estimasi Biaya per aktivitas

No	Aktivitas	Estimasi Biaya
<b>1.</b>	<b>Pengalian Kebutuhan</b>	
a.	Survey ke SKPD terkait	6.462.354
<b>2.</b>	<b>Analisis Spesifikasi Kebutuhan</b>	
a.	Rapat hasil analisis dengan tim pengembang	15.914.661
b.	Analisis proses bisnis aplikasi berdasarkan kebutuhan	9.587.937
<b>3.</b>	<b>Perancangan</b>	
a.	Sosialisasi hasil analisis kebutuhan ke internal	3.633.025
<b>3.</b>	<b>Implementasi</b>	
a.	Ekskususi kode program	27.673.980
<b>4.</b>	<b>Testing dan Integrasi</b>	
a.	Pengujian dan Intergrasi dengan metode <i>whitebox</i> dan <i>blackbox</i>	18.368.663
<b>5.</b>	<b>Penyerahan dan Implementasi</b>	
a.	Rapat penerimaan kesiapan aplikasi dengan Stakeholder	24.445.952
b.	Serah terima aplikasi dan database	10.980.278
Total		117.066.850

Tabel 3.14 Sub Laporan estimasi biaya per jabatan

No	Jabatan	Estimasi Biaya
1	Project Manager	44.258.828
2	SD/SA	25.422.889
3	Programmer	47.385.133
Total Biaya		117.066.850

Berdasarkan laporan estimasi biaya tersebut, kemudian dapat dihitung biaya langsung personil kedalam penentuan HPS. Laporan HPS di dapat dari total

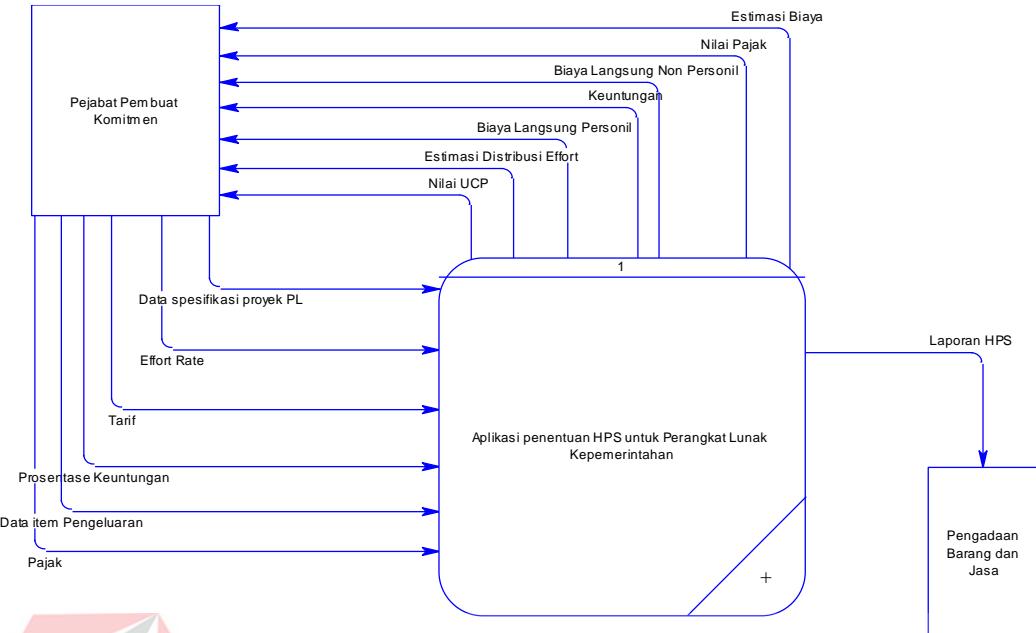
seluruh biaya, antara lain: biaya langsung personil, biaya langsung non personil, keuntungan, dan pajak.

Tabel 3.15 Laporan Harga Perkiraan Sendiri

<b>Harga Perkiraan Sendiri</b>					
Proyek pembuatan aplikasi desain untuk UI Digital Information Display Pemkot Bekasi					
<b>I. Biaya Langsung Personil</b>					
<b>No</b>	<b>Posisi</b>				
1.	Manajer Proyek				
2.	Sistem Analis				
3.	Programmer				
Sub Total Biaya Langsung Personil					Rp 117.066.850
Keuntungan					Rp 11.706.685
Total Biaya Langsung Personil					Rp 128.773.535
<b>II. Biaya Langsung Non Personil</b>					
<b>No</b>	<b>Item Pengeluaran</b>	<b>Volume</b>	<b>Kuantitas</b>	<b>Harga/Satuan</b>	<b>Sub Total</b>
1.	Alat Tulis Habis Pakai	1	3	Rp 2.480.550	Rp 7.441.650
2.	Biaya Komunikasi	1	3	Rp 2.013.200	Rp 6.039.600
Total Biaya Langsung Non Personil					Rp 13.481.250
Biaya Total					Rp 142.254.785
PPN 10%					Rp 14.225.479
Total HPS					Rp 156.480.264

### 3.2.2 Context Diagram

Context Diagram dari aplikasi ini melibatkan 2 entity yaitu Pejabat Pembuat Komitmen (PPK) dan Pengadaan barang/Jasa. Seperti yang digambarkan pada gambar 3.4 berikut ini:

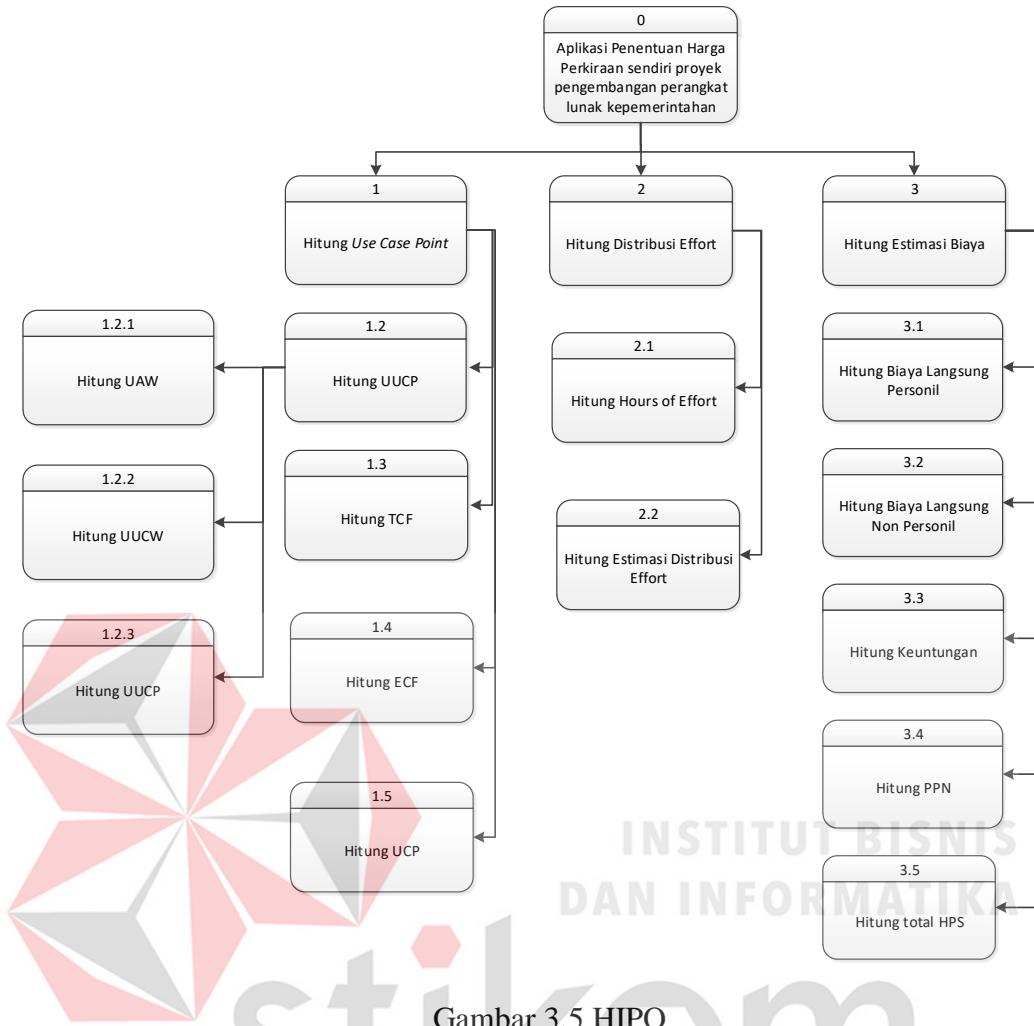


Gambar 3.4 Context Diagram

### 3.2.3 Hierarchy Plus Input Process Output

Gambar 3.5 Berikut ini adalah HIPO dari aplikasi penentuan HPS untuk proyek perangkat lunak kepemerintahan.

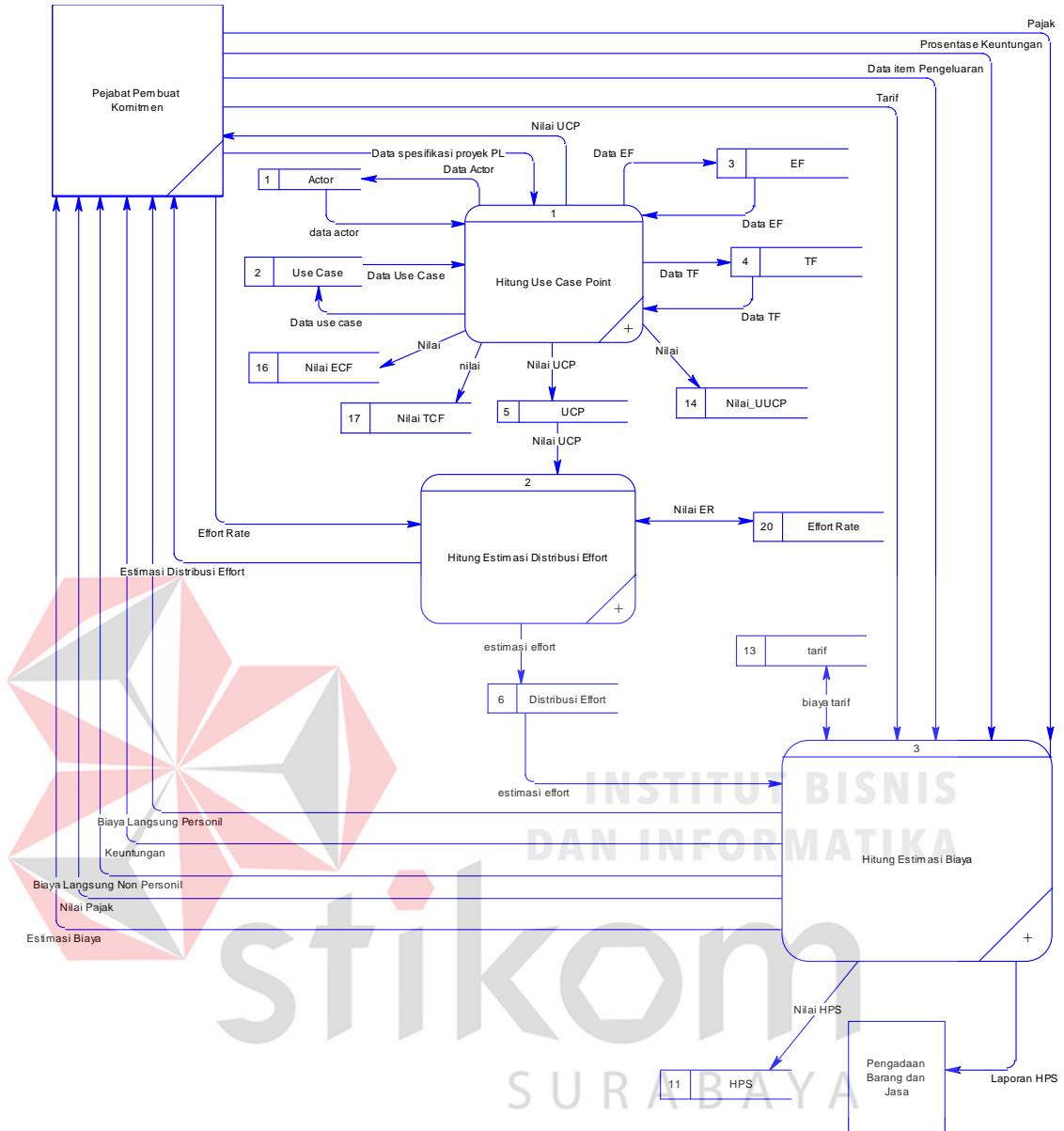




Gambar 3.5 HIPO

### 3.2.4 Data Flow Diagram Level 0

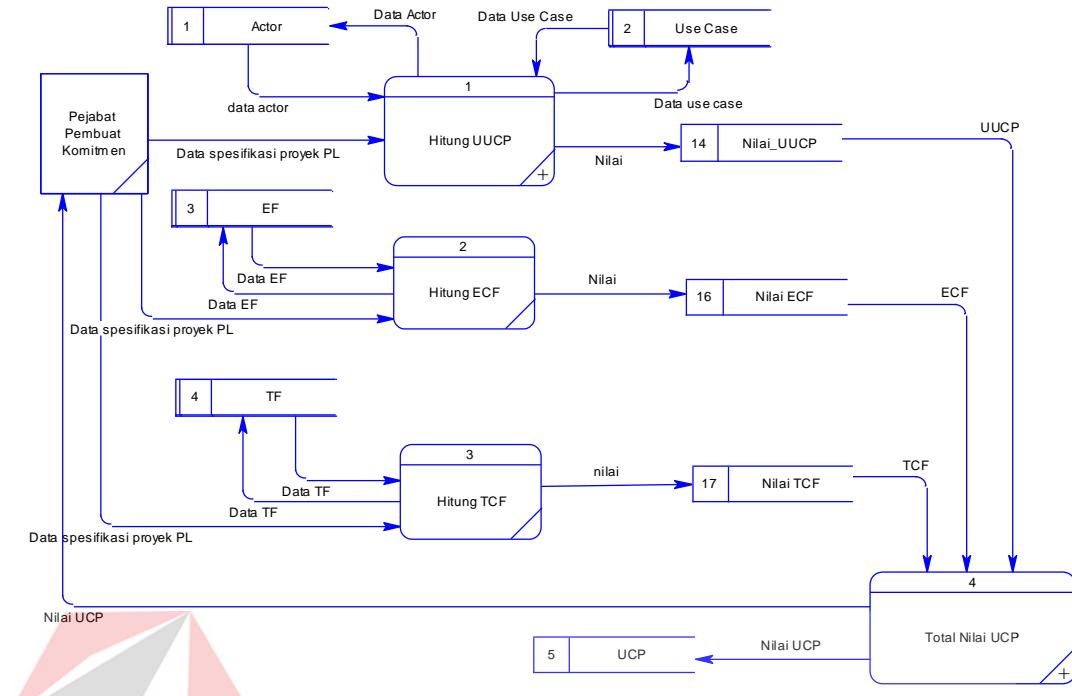
*Data Flow Diagram level 0 merupakan diagram yang lebih rinci dari Context Diagram. Berikut akan ditunjukkan pada gambar 3.6.*



Gambar 3.6 DFD Level 0

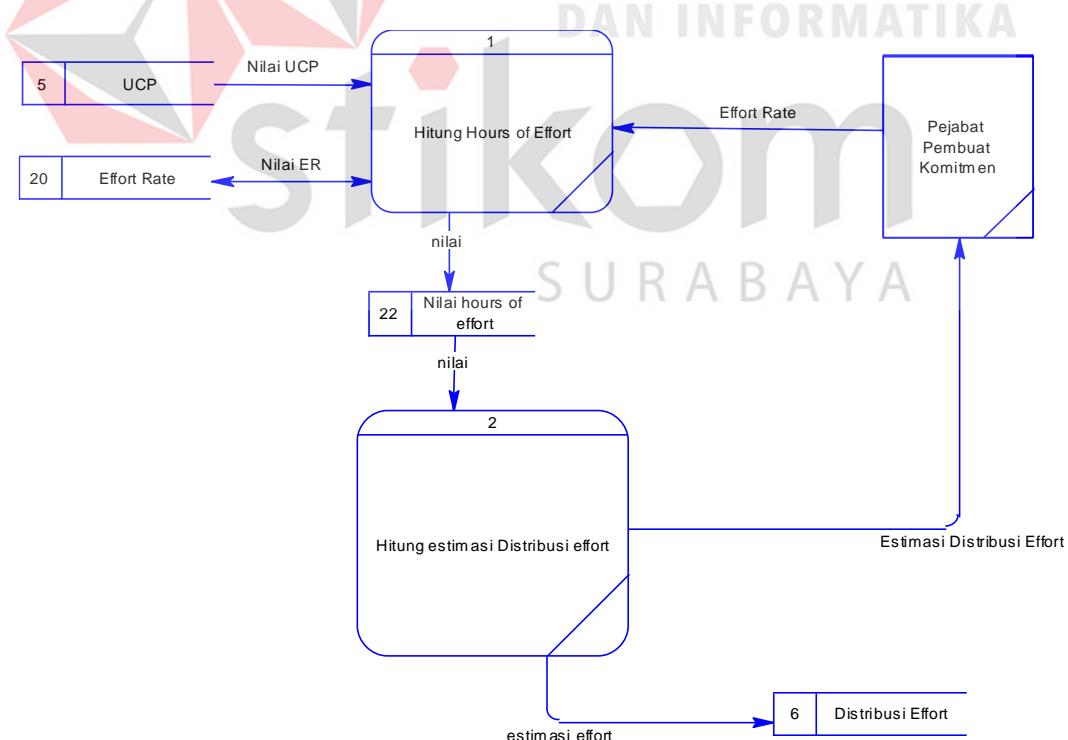
### 3.2.5 Data Flow Diagram Level 1 Use Case Point

Gambar 3.7 adalah sub-proses perhitungan *Use Case Point* yang ada pada DFD Level 0. DFD Level 1 ini terdiri dari proses menghitung *Unadjusted Use Case Point* dan Perhitungan *Use Case Point*.



Gambar 3.7 DFD Level 1 UCP

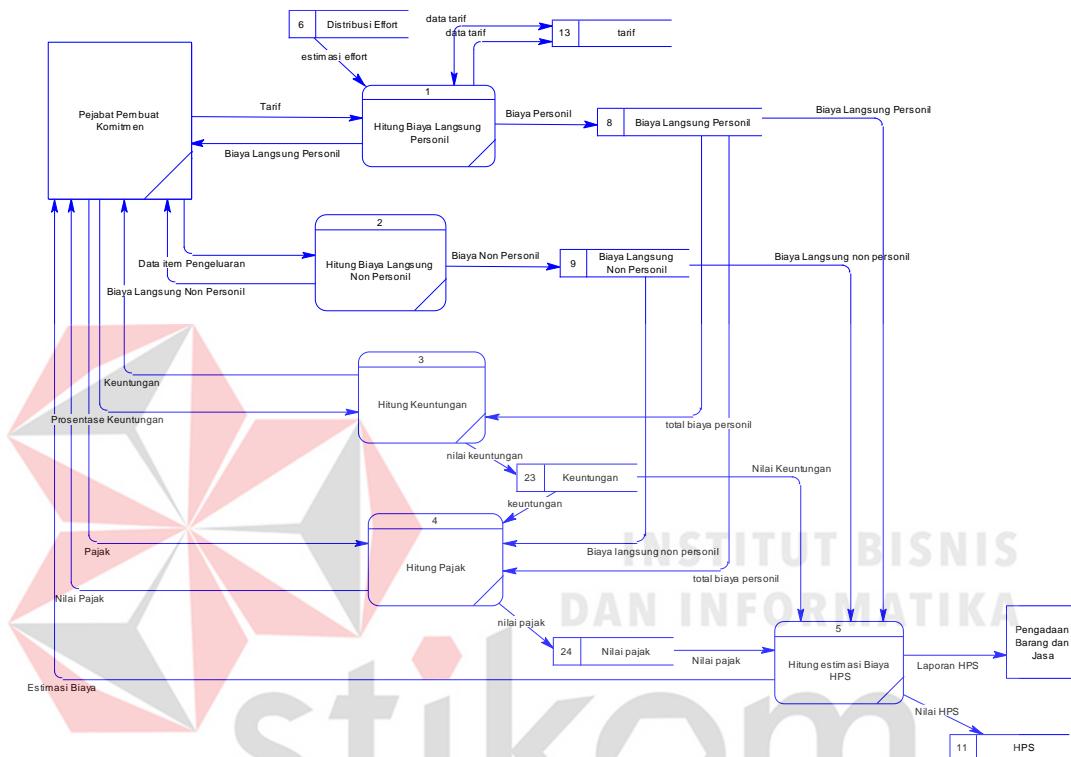
### 3.2.6 Data Flow Diagram Level 1 Estimasi Distribusi Effort



Gambar 3.8 DFD Level 1 Estimasi Distribusi Effort

### 3.2.7 Data Flow Diagram Level 1 Estimasi Biaya

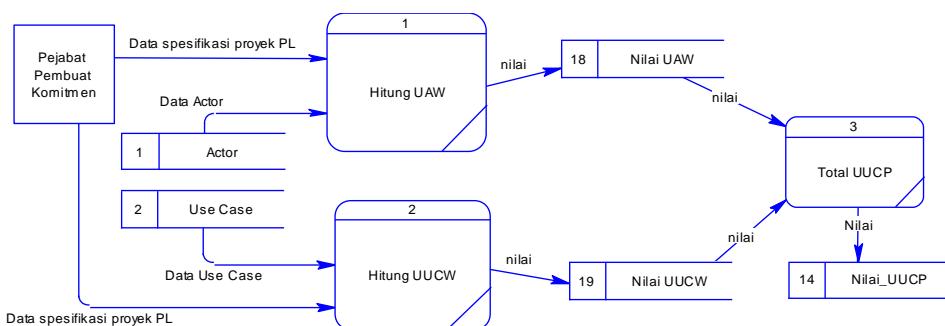
Gambar 3.8 adalah sub-proses perhitungan estimasi biaya yang ada pada DFD Level 0. DFD Level 1 ini terdiri dari proses menghitung biaya langsung personil, biaya langsung non personil, keuntungan, dan pajak.



Gambar 3.9 DFD Level 1 Estimasi Biaya

### 3.2.8 Data Flow Diagram Level 2 UUCP

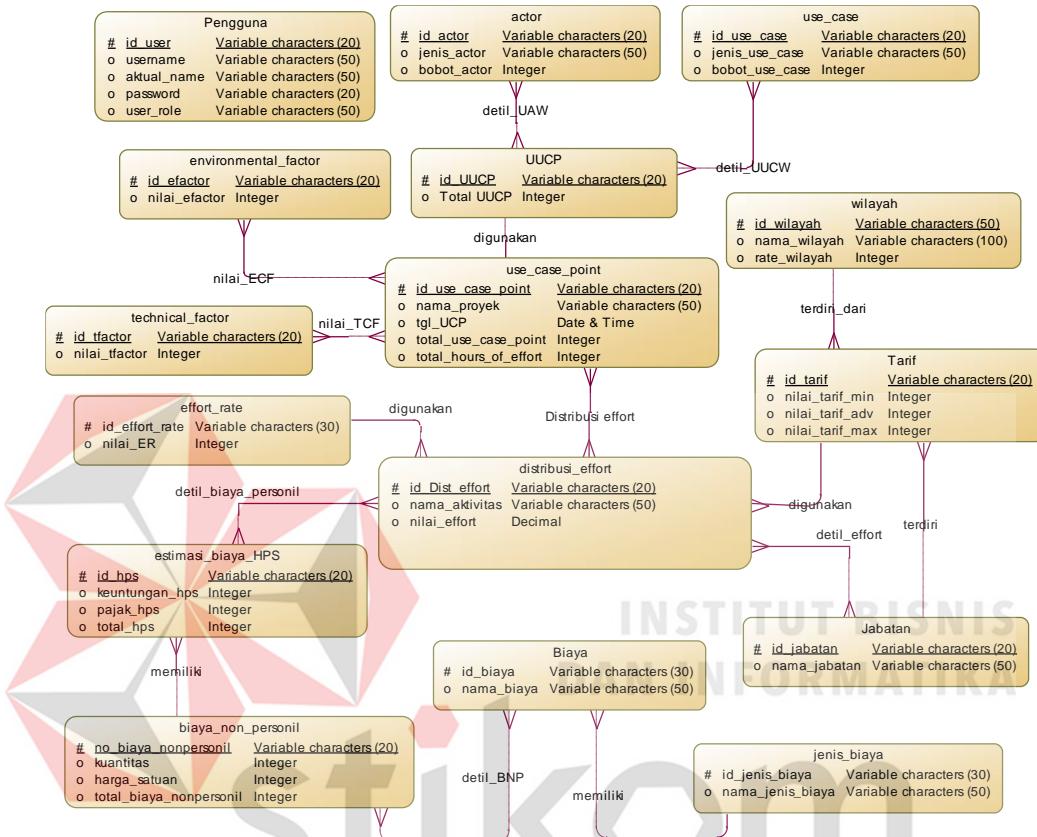
DFD level 2 ini terdiri dari proses menghitung *Unadjusted Actor Weight*, dan *Unadjusted Use Case Weight*.



Gambar 3.10 DFD Level 2 UUCP

### 3.2.9 Conceptual Data Model

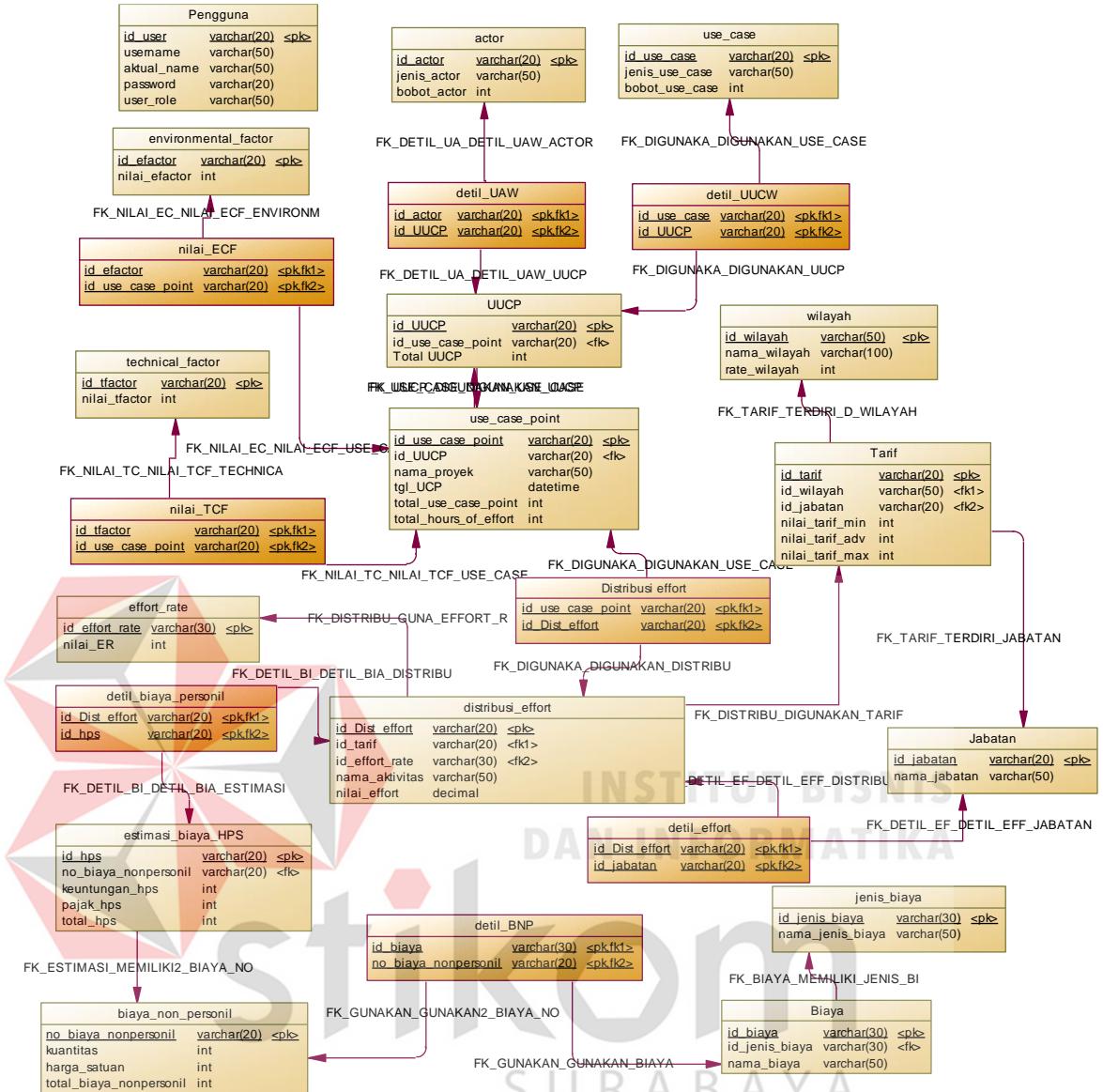
CDM merupakan gambaran konsep struktur basis data yang dirancang untuk suatu program atau aplikasi. CDM dapat dilihat pada Gambar 3.11.



Gambar 3.11 Conceptual Data Model

### 3.2.10 Physical Data Model

PDM menggambarkan secara detail konsep rancangan basis data yang dirancang untuk suatu program aplikasi. PDM merupakan hasil generate dari CDM. PDM dapat dilihat pada Gambar 3.12.



Gambar 3.12 Physical Data Model

### 3.2.11 Struktur Tabel

Berikut struktur tabel yang digunakan dalam pembuatan Aplikasi Penentuan Harga Perkiraan Sendiri untuk Proyek Perangkat Lunak Kepemerintahan.

#### a. Tabel Pengguna

Tabel pengguna berfungsi untuk menyimpan data pengguna yang akan mengakses aplikasi ini.

Tabel 3.16 Pengguna

Nomor	Field	Type	Length	Key
1.	ID_USER	Varchar	20	Primary Key
2.	USERNAME	Varchar	50	
3.	AKTUAL NAME	Varchar	50	
4.	PASSWORD	Varchar	50	
5.	USER ROLE	Varchar	50	

#### b. Tabel Actor

Merupakan tabel yang berfungsi untuk menyimpan data *actor* yang digunakan untuk menentukan nilai HPS.

Tabel 3.17 Actor

Nomor	Field	Type	Length	Key
1.	ID_ACTOR	Varchar	20	Primary Key
2.	INTERAKSI	Varchar	50	
3.	JENIS_ACTOR	Varchar	50	
4.	BOBOT_ACTOR	Integer		

#### c. Tabel detil\_UAW

Tabel 3.18 Detil\_UAW

Nomor	Field	Type	Length	Key
1.	ID_ACTOR	Varchar	20	Primary key
2.	ID_UUCP	Varchar	20	Primary key
3.	TOTAL_UAW	Integer		

#### d. Tabel Use case

Tabel use case berfungsi untuk menyimpan data *use case* yang ada dalam aplikasi penentuan HPS ini.

Tabel 3.19 Use Case

Nomor	Field	Type	Length	Key
1.	ID_USE_CASE	Varchar	20	Primary Key
2.	DESKRIPSI	Varchar	50	
3.	JENIS_USE_CASE	Varchar	50	
4.	BOBOT_USE_CASE	Integer		

#### e. Tabel detil\_UUCW

Tabel detil UUCW berfungsi untuk menyimpan data *Unadjusted Use Case Weight* pada aplikasi penentuan HPS ini.

Tabel 3.20 Detil\_UUCW

Nomor	Field	Type	Length	Key
1.	ID_USE_CASE	Varchar	20	Primary Key
2.	ID_UUCP	Varchar	20	Primary Key
3.	TOTAL_UUCW	Integer		

## f. Tabel UUCP

Tabel ini berfungsi menyimpan data *use case point* pada aplikasi penentuan HPS ini.

Tabel 3.21 UUCP

Nomor	Field	Type	Length	Key
1.	ID_UUCP	Varchar	20	Primary Key
2.	ID_USE_CASE	Varchar	20	Primary Key
3.	TOTAL_UUCP	Integer		

g. Tabel *Environmental factor*

Tabel ini berfungsi untuk menyimpan data faktor-faktor lingkungan yang mempengaruhi pembuatan proyek perangkat lunak.

Tabel 3.22 Environmental Factor

Nomor	Field	Type	Length	Key
1.	ID_EFACTOR	Varchar	20	Primary Key
2.	NILAI_EFACTOR	Integer		

## h. Tabel Detil\_Nilai\_ECF

Tabel 3.23 Detil\_Nilai\_ECF

Nomor	Field	Type	Length	Key
1.	ID_EFACTOR	Varchar	20	Primary Key
2.	NILAI_USE_CASE_POINT	Integer		
3.	NILAI_ECF	Integer		

i. Tabel *Technical factor*

Tabel ini berfungsi untuk menyimpan data faktor-faktor teknis yang mempengaruhi proyek pembuatan perangkat lunak kepemerintahan.

Tabel 3.24 Technical factor

Nomor	Field	Type	Length	Key
1.	ID_TFACTOR	Varchar	20	Primary Key
2.	NILAI_TFACTOR	Integer		

j. Tabel DETIL\_NILAI\_TCF

Tabel 3.25 Detil\_Nilai\_TCF

Nomor	Field	Type	Length	Key
1.	ID_TFACTOR	Varchar	20	Primary Key
2.	NILAI_USE_CASE_POINT	Integer		
3.	NILAI_TCF	Integer		

k. Tabel *Effort Rate*

Tabel 3.26 Effort Rate

Nomor	Field	Type	Length	Key
1.	ID EFFORT RATE	Varchar	20	Primary Key
2.	NILAI_ER	Integer		

l. Tabel *Use Case Point*

Tabel *use case point* berfungsi untuk menyimpan data nilai *use case point* yang didapat dari perhitungan.

Tabel 3.27 Use Case Point

Nomor	Field	Type	Length	Key
1.	ID_USE_CASE_POINT	Varchar	20	Primary Key
2.	ID_UUCP	Varchar	20	Foreign Key
3.	ID_EFACTOR	Varchar	20	Foreign Key
4.	ID_TFAKTOR	Varchar	20	Foreign Key
5.	TOTAL_USE_CASE_POINT	Integer		

m. Tabel Detil distribusi effort

Tabel 3.28 Detil distribusi effort

Nomor	Field	Type	Length	Key
1.	ID_USE_CASE_POINT	Varchar	20	Primary Key
2.	ID_DIST EFFORT	Varchar	20	Primary Key

n. Tabel tarif

Tabel yang berfungsi untuk menyimpan data tarif.

Tabel 3.29 Tarif

Nomor	Field	Type	Length	Key
1.	ID_TARIF	Varchar	20	Primary Key
2.	ID_JABATAN	Varchar	50	Foreign Key
3.	ID_WILAYAH	Varchar	50	Foreign Key
4.	TARIF_MIN	Integer		
5.	TARIF_ADV	Integer		
6.	TARIF_MAX	Integer		

o. Tabel Jabatan

Tabel 3.30 Jabatan

Nomor	Field	Type	Length	Key
1.	ID_JABATAN	Varchar	20	Primary Key
2.	NAMA_JABATAN	Varchar	50	

p. Tabel Wilayah

Tabel 3.31 Wilayah

Nomor	Field	Type	Length	Key
1.	ID_WILAYAH	Varchar	20	Primary Key
2.	NAMA_WILAYAH	Varchar	50	
3.	RATE_WILAYAH			

q. Tabel *distribusi effort*

Tabel ini berfungsi untuk menyimpan data distribusi effort yang terdiri data aktivitas dan masing-masing effortnya.

Tabel 3.32 Distribusi effort

Nomor	Field	Type	Length	Key
1.	NO_DIST EFFORT	Varchar	20	Primary Key
3.	NAMA_AKTIVITAS	Varchar	20	
4.	ESTIMASI EFFORT	Decimal	50	

r. Tabel detil\_effort

Tabel 3.33 Detil\_effort

Nomor	Field	Type	Length	Key
1.	ID_JABATAN	Varchar	20	Primary Key
2.	ID_DIST EFFORT	Varchar	20	

s. Tabel Jenis Biaya

Tabel 3.34 Jenis Biaya

Nomor	Field	Type	Length	Key
1.	ID_JENIS_BIAYA	Varchar	20	Primary Key
2.	NAMA_JENIS_BIAYA	Varchar	50	

t. Tabel Biaya

Tabel 3.35 Biaya

Nomor	Field	Type	Length	Key
1.	ID_JENIS_BIAYA	Varchar	20	Primary Key
2.	NAMA_JENIS_BIAYA	Varchar	50	

u. Tabel biaya langsung non personil

Berfungsi untuk menyimpan data biaya langsung non personil

Tabel 3.36 Biaya langsung Non Personil

Nomor	Field	Type	Length	Key
1.	NO_BIAYA_NONPERSONIL	Varchar	20	Primary Key
2.	KUANTITAS	Integer		
3.	HARGA_SATUAN	Integer		
4.	TOTAL_BIAYA_NONPERSONIL	Integer		

v. Tabel HPS

Tabel ini berfungsi untuk menyimpan hasil dari pembuatan aplikasi ini yaitu laporan HPS

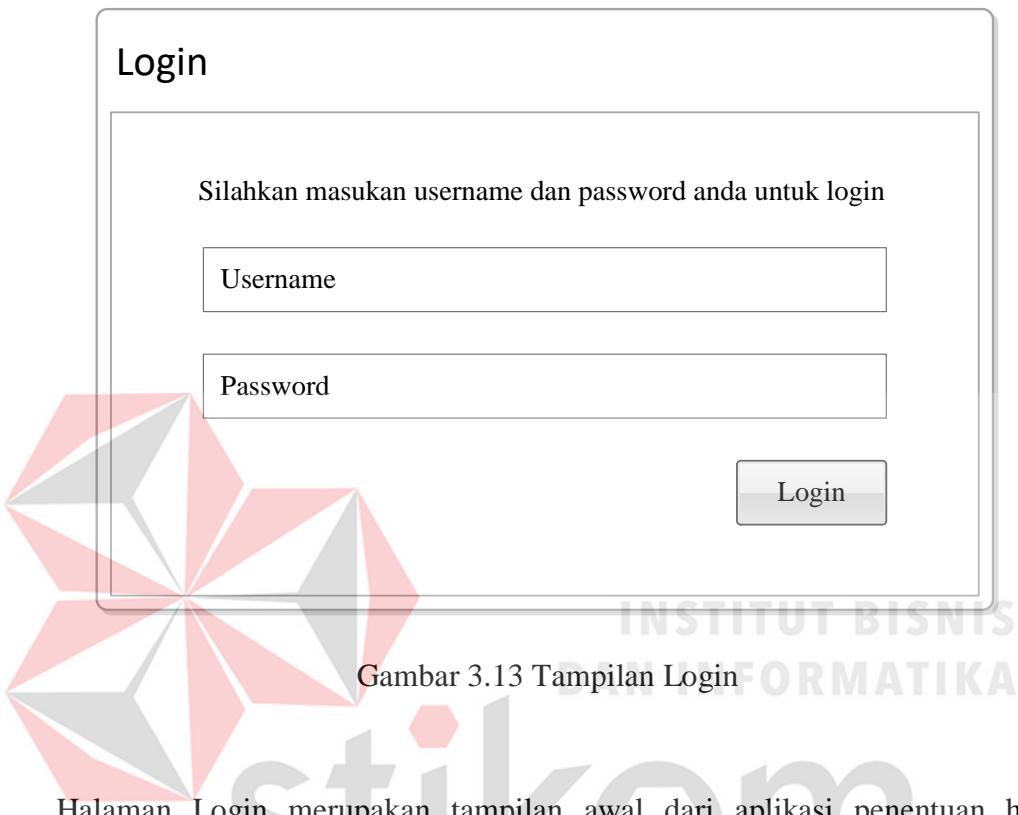
Tabel 3.37 Harga Perkiraan Sendiri

Nomor	Field	Type	Length	Key
1.	ID_HPS	Varchar	20	Primary Key
2.	NO_BIAYA_NONPERSONIL	Varchar	20	
3.	PAJAK_HPS	Integer		
4.	KEUNTUNGAN_HPS	Integer		
5.	TOTAL_HPS	Integer		

### 3.2.12 Desain Input/ Output

#### A. Desain Input

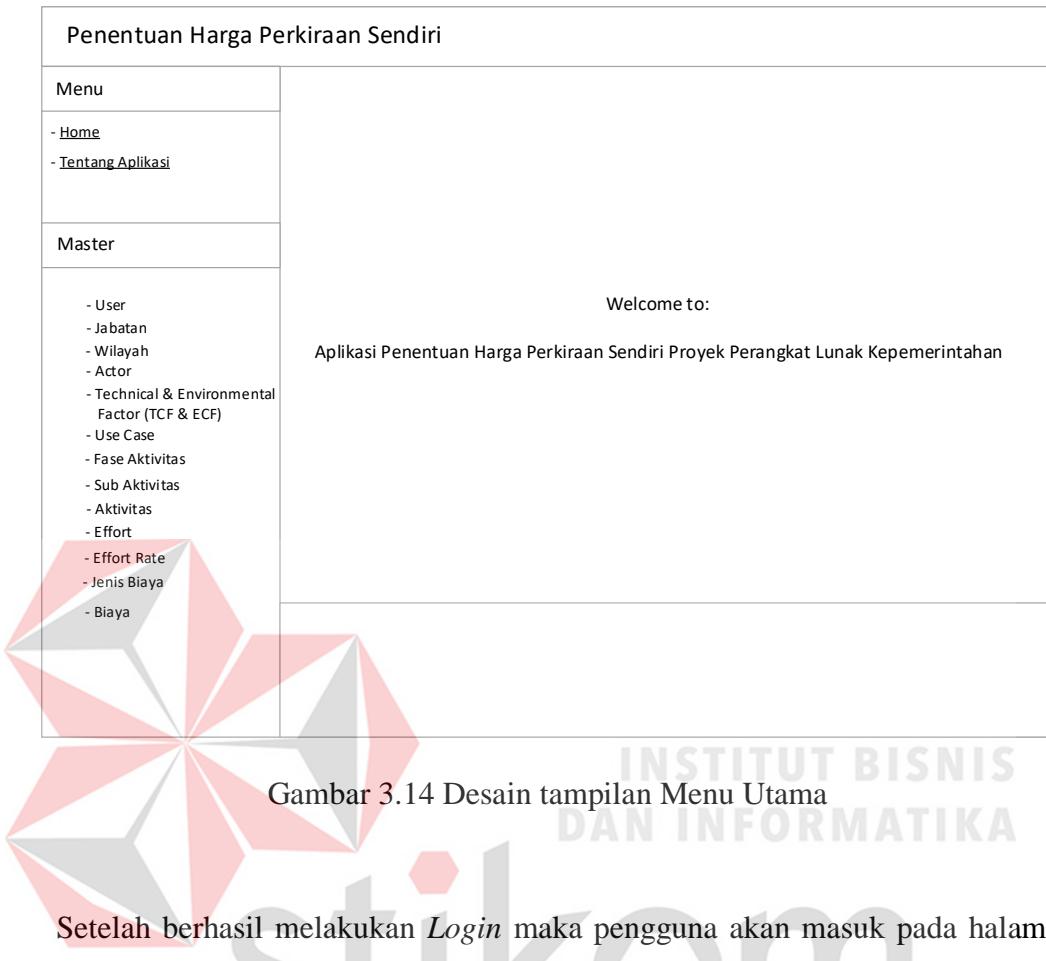
##### a. Menu Login



Gambar 3.13 Tampilan Login

Halaman Login merupakan tampilan awal dari aplikasi penentuan harga perkiraan sendiri proyek perangkat lunak ini. Login berfungsi sebagai sistem keamanan untuk membatasi pengguna dalam mengakses aplikasi. Di dalam form login ini terdapat *Username* dan *Password* yang harus diisi oleh pengguna. Dimana masing-masing pengguna memiliki *Username* dan *Password* yang berbeda-beda. Pada form login ini juga terdapat tombol *Login* yang berfungsi untuk melanjutkan ke halaman berikutnya.

## b. Halaman Menu Utama



Gambar 3.14 Desain tampilan Menu Utama

Setelah berhasil melakukan *Login* maka pengguna akan masuk pada halaman menu utama. Pada halaman ini pengguna akan dihubungkan ke beberapa fungsi yang dimiliki oleh aplikasi ini. Fungsi-fungsi aplikasi bisa dilihat di sebelah kiri pada halaman ini.

### c. Halaman *Maintenance Data Actor*

Penentuan Harga Perkiraan Sendiri																																		
<p><b>Menu</b></p> <ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul> <p><b>Master</b></p> <ul style="list-style-type: none"> <li>- User</li> <li>- <b>Jabatan</b></li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<p><b>Data Actor</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">Insert Data</th> <th colspan="2">Daftar Actor</th> </tr> <tr> <th>ID Actor</th> <th><input type="text"/></th> <th>ID Actor</th> <th>Nama Jenis Aktor</th> </tr> <tr> <th>Nama Jenis Actor</th> <th><input type="text"/></th> <th>Deskripsi</th> <th>Bobot</th> </tr> </thead> <tbody> <tr> <td>Deskripsi Jenis Aktor</td> <td><input type="text"/></td> <td></td> <td></td> </tr> <tr> <td>Bobot</td> <td><input type="text"/></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;"><input type="button" value="Insert"/></td> <td></td> <td></td> </tr> </tbody> </table>	Insert Data		Daftar Actor		ID Actor	<input type="text"/>	ID Actor	Nama Jenis Aktor	Nama Jenis Actor	<input type="text"/>	Deskripsi	Bobot	Deskripsi Jenis Aktor	<input type="text"/>			Bobot	<input type="text"/>				<input type="button" value="Insert"/>			<p><b>Daftar Actor</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID Actor</th> <th>Nama Jenis Aktor</th> <th>Deskripsi</th> <th>Bobot</th> </tr> </thead> <tbody> <tr> <td colspan="4"></td> </tr> </tbody> </table>	ID Actor	Nama Jenis Aktor	Deskripsi	Bobot				
Insert Data		Daftar Actor																																
ID Actor	<input type="text"/>	ID Actor	Nama Jenis Aktor																															
Nama Jenis Actor	<input type="text"/>	Deskripsi	Bobot																															
Deskripsi Jenis Aktor	<input type="text"/>																																	
Bobot	<input type="text"/>																																	
	<input type="button" value="Insert"/>																																	
ID Actor	Nama Jenis Aktor	Deskripsi	Bobot																															

Gambar 3.15 Desain tampilan Insert Data Actor

Halaman *maintenance* data *actor* merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data *actor*. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID *Actor*, Nama Jenis Actor, Deskripsi Jenis Actor, dan Bobot. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data *actor* yang sudah disimpan akan tampil pada tabel daftar *actor* yang berada di sebelahnya.

#### d. Halaman *Maintenance Data Use Case*

Penentuan Harga Perkiraan Sendiri													
Menu	Data Use Case												
<ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul> <b>Master</b> <ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<table border="1"> <thead> <tr> <th colspan="4">Daftar Use case</th> </tr> <tr> <th>ID Use case</th> <th>Deskripsi</th> <th>Jenis Use Case</th> <th>Bobot</th> </tr> </thead> <tbody> <tr> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </tbody> </table> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p><b>Insert Data</b></p> <p>ID Use Case <input type="text"/>      Nama Use Case <input type="text"/>      Deskripsi <input type="text"/>      Bobot <input type="text"/></p> <p style="text-align: right;"><input type="button" value="Insert"/></p> </div>	Daftar Use case				ID Use case	Deskripsi	Jenis Use Case	Bobot	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Daftar Use case													
ID Use case	Deskripsi	Jenis Use Case	Bobot										
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>										

Gambar 3.16 Desain tampilan insert data Use Case

Halaman maintenance data *use case* merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data *use case*. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID *Use Case*, Nama *Use Case*, Deskripsi, dan Bobot. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data *use case* yang sudah di simpan akan tampil pada tabel daftar *use case* yang berada di sebelahnya.

e. Halaman *Maintenance* Data Jabatan

Penentuan Harga Perkiraan Sendiri																					
<b>Menu</b> - Home - Tentang Aplikasi  <b>Master</b> - User - Jabatan - Wilayah - Actor - Technical & Environmental Factor (TCF & ECF) - Use Case - Fase Aktivitas - Sub Aktivitas - Aktivitas - Effort <b>- Effort Rate</b> <b>- Jenis Biaya</b> - Biaya	<div style="background-color: #f0f0f0; padding: 10px;"> <b>Data Jabatan</b>  <b>Insert Data</b> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">ID Jabatan</td> <td style="width: 70%;"><input type="text"/></td> </tr> <tr> <td>Nama Jabatan</td> <td><input type="text"/></td> </tr> <tr> <td>Tarif Minimal</td> <td><input type="text"/></td> </tr> <tr> <td>Tarif Adverage</td> <td><input type="text"/></td> </tr> <tr> <td>Tarif Maksimal</td> <td><input type="text"/></td> </tr> </table> <div style="text-align: right; margin-top: 10px;"> <input type="button" value="Insert"/> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <b>Daftar Jabatan</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID Jab</th> <th>Nama Jab</th> <th>Tarif M</th> <th>Tarif Adv</th> <th>Tarif Maksimal</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> </div> </div>	ID Jabatan	<input type="text"/>	Nama Jabatan	<input type="text"/>	Tarif Minimal	<input type="text"/>	Tarif Adverage	<input type="text"/>	Tarif Maksimal	<input type="text"/>	ID Jab	Nama Jab	Tarif M	Tarif Adv	Tarif Maksimal					
ID Jabatan	<input type="text"/>																				
Nama Jabatan	<input type="text"/>																				
Tarif Minimal	<input type="text"/>																				
Tarif Adverage	<input type="text"/>																				
Tarif Maksimal	<input type="text"/>																				
ID Jab	Nama Jab	Tarif M	Tarif Adv	Tarif Maksimal																	

Gambar 3.17 Desain tampilan insert data Jabatan

Halaman *maintenance* data jabatan merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data jabatan. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID Jabatan, Nama Jabatan, Tarif Minimal, Tarif Adverage, dan Tarif Maksimal. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data jabatan yang sudah disimpan akan tampil pada tabel daftar jabatan yang berada di sebelahnya.

#### f. Halaman Maintenance Wilayah Proyek

Penentuan Harga Perkiraan Sendiri																									
<b>Menu</b> <ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul> <b>Master</b> <ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<p><b>Data Wilayah Proyek</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="3">Insert Data</th> </tr> </thead> <tbody> <tr> <td>ID Wilayah</td> <td colspan="2"><input type="text"/></td> </tr> <tr> <td>Nama Wilayah</td> <td colspan="2"><input type="text"/> Enter Text</td> </tr> <tr> <td>Rate Wilayah</td> <td colspan="2"><input type="text"/> Enter Text</td> </tr> <tr> <td colspan="3" style="text-align: right;"><input type="button" value="Insert"/></td> </tr> </tbody> </table> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="3">Daftar Wilayah</th> </tr> <tr> <th>ID Wilayah</th> <th>Nama Wilayah</th> <th>Rate Wilayah</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Insert Data			ID Wilayah	<input type="text"/>		Nama Wilayah	<input type="text"/> Enter Text		Rate Wilayah	<input type="text"/> Enter Text		<input type="button" value="Insert"/>			Daftar Wilayah			ID Wilayah	Nama Wilayah	Rate Wilayah			
Insert Data																									
ID Wilayah	<input type="text"/>																								
Nama Wilayah	<input type="text"/> Enter Text																								
Rate Wilayah	<input type="text"/> Enter Text																								
<input type="button" value="Insert"/>																									
Daftar Wilayah																									
ID Wilayah	Nama Wilayah	Rate Wilayah																							

Gambar 3.18 Tampilan maintenance data wilayah proyek

Halaman *maintenance* data wilayah proyek merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data wilayah proyek. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID Wilayah, Nama Wilayah Proyek, dan Rate Wilayah. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data jabatan yang sudah di simpan akan tampil pada tabel daftar wilayah yang berada di sebelahnya.

g. Halaman *Maintenance Data User*

Penentuan Harga Perkiraan Sendiri																									
<b>Menu</b> - <a href="#">Home</a> - <a href="#">Tentang Aplikasi</a>  <b>Master</b> - User - Jabatan - Wilayah - Actor - Technical & Environmental Factor (TCF & ECF) - Use Case - Fase Aktivitas - Sub Aktivitas - Aktivitas - Effort - Effort Rate - Jenis Biaya - Biaya	<div style="background-color: #f0f0f0; padding: 10px;"> <p style="margin: 0;"><b>Data User</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">Insert Data</th> </tr> </thead> <tbody> <tr> <td>ID User</td> <td><input style="width: 150px; height: 25px;" type="text"/></td> </tr> <tr> <td>Username</td> <td><input style="width: 150px; height: 25px;" type="text"/></td> </tr> <tr> <td>User Actual Name</td> <td><input style="width: 150px; height: 25px;" type="text"/></td> </tr> <tr> <td>User Password</td> <td><input style="width: 150px; height: 25px;" type="text"/></td> </tr> <tr> <td>User Role</td> <td><input style="width: 150px; height: 25px;" type="text"/></td> </tr> <tr> <td colspan="2" style="text-align: right; padding-top: 10px;"> <input style="width: 80px; height: 30px;" type="button" value="Insert"/> </td> </tr> </tbody> </table> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <b>Daftar User</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID User</th> <th>User</th> <th>User A</th> <th>User Pass</th> <th>User Role</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> </div> </div>	Insert Data		ID User	<input style="width: 150px; height: 25px;" type="text"/>	Username	<input style="width: 150px; height: 25px;" type="text"/>	User Actual Name	<input style="width: 150px; height: 25px;" type="text"/>	User Password	<input style="width: 150px; height: 25px;" type="text"/>	User Role	<input style="width: 150px; height: 25px;" type="text"/>	<input style="width: 80px; height: 30px;" type="button" value="Insert"/>		ID User	User	User A	User Pass	User Role					
Insert Data																									
ID User	<input style="width: 150px; height: 25px;" type="text"/>																								
Username	<input style="width: 150px; height: 25px;" type="text"/>																								
User Actual Name	<input style="width: 150px; height: 25px;" type="text"/>																								
User Password	<input style="width: 150px; height: 25px;" type="text"/>																								
User Role	<input style="width: 150px; height: 25px;" type="text"/>																								
<input style="width: 80px; height: 30px;" type="button" value="Insert"/>																									
ID User	User	User A	User Pass	User Role																					

Gambar 3.19 Desain tampilan insert data User

Halaman *maintenance* data *user* merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data *user*. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID *User*, *Username*, *User Actual Name*, *User Password*, dan *User Role*. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data *user* yang sudah disimpan akan tampil pada tabel daftar *user* yang berada di sebelahnya.

#### h. Halaman *Maintenance* Data Fase Aktivitas

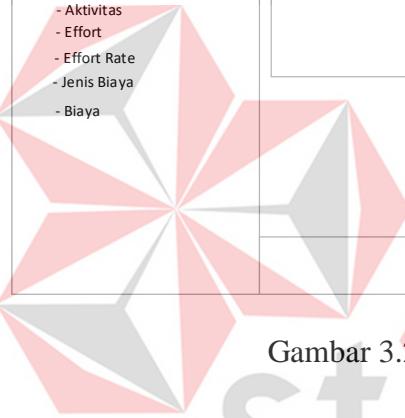
Penentuan Harga Perkiraan Sendiri									
Menu	Data Fase Aktivitas								
<ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul>	<table border="1"> <thead> <tr> <th colspan="2">Insert Data</th> </tr> </thead> <tbody> <tr> <td>ID Fase Aktivitas</td> <td><input type="text"/></td> </tr> <tr> <td>Nama Fase Aktivitas</td> <td><input type="text"/></td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Simpan"/></td> </tr> </tbody> </table>	Insert Data		ID Fase Aktivitas	<input type="text"/>	Nama Fase Aktivitas	<input type="text"/>	<input type="button" value="Simpan"/>	
Insert Data									
ID Fase Aktivitas	<input type="text"/>								
Nama Fase Aktivitas	<input type="text"/>								
<input type="button" value="Simpan"/>									
<b>Master</b> <ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<table border="1"> <thead> <tr> <th colspan="2">Daftar Fase Aktivitas</th> </tr> <tr> <th>ID Fase Aktivitas</th> <th>Nama Fase Aktivitas</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Daftar Fase Aktivitas		ID Fase Aktivitas	Nama Fase Aktivitas				
Daftar Fase Aktivitas									
ID Fase Aktivitas	Nama Fase Aktivitas								

Gambar 3.20 Desain tampilan insert data Fase Aktivitas

Halaman *maintenance* data fase aktivitas merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data fase aktivitas. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID Fase Aktivitas, dan Nama Fase Aktivitas. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data fase aktivitas yang sudah di simpan akan tampil pada tabel daftar fase aktivitas yang berada di sebelahnya.

### i. Halaman *Maintenance* Data Sub Aktivitas

**Penentuan Harga Perkiraan Sendiri**

<b>Menu</b> <ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul> <b>Master</b> <ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<div style="background-color: #f0f0f0; padding: 5px;"> <b>Data Sub Aktivitas</b> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <b>Insert Data</b> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">ID Sub Aktivitas</td> <td style="width: 70%;"><input type="text"/></td> </tr> <tr> <td>Fase Aktivitas</td> <td><input type="text"/></td> </tr> <tr> <td>Nama Sub Aktivitas</td> <td><input type="text"/> Enter Text</td> </tr> </table> <div style="text-align: right; margin-top: 5px;"> <input type="button" value="Insert"/> </div> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;"> <b>Daftar Sub Aktivitas</b> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID Sub Aktivitas</th> <th>Fase Aktiv</th> <th>Nama Sub Aktivitas</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table> </div> </div>	ID Sub Aktivitas	<input type="text"/>	Fase Aktivitas	<input type="text"/>	Nama Sub Aktivitas	<input type="text"/> Enter Text	ID Sub Aktivitas	Fase Aktiv	Nama Sub Aktivitas				
ID Sub Aktivitas	<input type="text"/>													
Fase Aktivitas	<input type="text"/>													
Nama Sub Aktivitas	<input type="text"/> Enter Text													
ID Sub Aktivitas	Fase Aktiv	Nama Sub Aktivitas												

Gambar 3.21 Tampilan data Sub Aktivitas

Halaman *maintenance* data sub aktivitas merupakan halaman yang berfungsi untuk menyimpan, merubah, dan menghapus data sub aktivitas. Pada halaman ini pengguna harus mengisi ID Sub Aktivitas, Fase Aktivitas, dan Nama Sub Aktivitas. Untuk melanjutkan ke proses selanjutnya pengguna dapat mengklik tombol *insert*.

j. Halaman *Maintenance Data Aktivitas*

Penentuan Harga Perkiraan Sendiri													
<b>Menu</b> - <a href="#">Home</a> - <a href="#">Tentang Aplikasi</a>  <b>Master</b> <ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<b>Data Aktivitas</b> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <b>Insert Data</b> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">ID Aktivitas</td> <td style="width: 85%;"><input type="text"/></td> </tr> <tr> <td>Sub Aktivitas</td> <td><input type="text"/> <span style="border: 1px solid #ccc; padding: 2px;">▼</span></td> </tr> <tr> <td>Nama Aktivitas</td> <td><input type="text"/> Enter Text</td> </tr> </table> <div style="text-align: right; margin-top: 5px;"> <input type="button" value="Insert"/> </div> </div> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> <b>Daftar Aktivitas</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID Aktivitas</th> <th>Sub Aktivitas</th> <th>Nama Aktivitas</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table> </div>	ID Aktivitas	<input type="text"/>	Sub Aktivitas	<input type="text"/> <span style="border: 1px solid #ccc; padding: 2px;">▼</span>	Nama Aktivitas	<input type="text"/> Enter Text	ID Aktivitas	Sub Aktivitas	Nama Aktivitas			
ID Aktivitas	<input type="text"/>												
Sub Aktivitas	<input type="text"/> <span style="border: 1px solid #ccc; padding: 2px;">▼</span>												
Nama Aktivitas	<input type="text"/> Enter Text												
ID Aktivitas	Sub Aktivitas	Nama Aktivitas											

Gambar 3.22 Desain tampilan insert data Aktivitas

Halaman *maintenance* data aktivitas merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data aktivitas. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID Aktivitas, Sub Aktivitas, dan Nama Aktivitas. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data aktivitas yang sudah di simpan akan tampil pada tabel daftar aktivitas yang berada di sebelahnya.

k. Halaman *Maintenance Data Effort*

Gambar 3.23 Desain tampilan insert data Effort

Halaman *maintenance* data *effort* merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data *effort*. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu Aktivitas, Jabatan, dan Estimasi *Effort*. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data *effort* yang sudah disimpan akan tampil pada tabel daftar *effort* yang berada di sebelahnya.

## 1. Halaman Maintenance Data Technical Factor

**Penentuan Harga Perkiraan Sendiri**

Menu	Hitung Technical Complexity Factor (TCF)																			
- Home - Tentang Aplikasi	<input type="radio"/> Technical Complexity Factor <input checked="" type="radio"/> Environmental Complexity Factor																			
<b>Master</b>	<b>Insert Data</b> <table border="1"> <tr> <td>ID TCF</td> <td><input type="text"/></td> </tr> <tr> <td>Keterangan</td> <td><input type="text"/></td> </tr> <tr> <td>Petunjuk</td> <td><input type="text"/></td> </tr> <tr> <td>Bobot</td> <td><input type="text"/></td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Simpan"/></td> </tr> </table> <b>Daftar TCF</b> <table border="1"> <thead> <tr> <th>ID TCF</th> <th>Keterangan</th> <th>Petunjuk</th> <th>Bobot</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		ID TCF	<input type="text"/>	Keterangan	<input type="text"/>	Petunjuk	<input type="text"/>	Bobot	<input type="text"/>	<input type="button" value="Simpan"/>		ID TCF	Keterangan	Petunjuk	Bobot				
ID TCF	<input type="text"/>																			
Keterangan	<input type="text"/>																			
Petunjuk	<input type="text"/>																			
Bobot	<input type="text"/>																			
<input type="button" value="Simpan"/>																				
ID TCF	Keterangan	Petunjuk	Bobot																	
- User - Jabatan - Wilayah - Actor - Technical & Environmental Factor (TCF & ECF) - Use Case - Fase Aktivitas - Sub Aktivitas - Aktivitas - Effort - Effort Rate - Jenis Biaya - Biaya																				

Gambar 3.24 Desain tampilan insert data Technical Factor

Halaman *maintenance* data *technical factor* merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data *technical factor*. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID *Technical Factor*, Keterangan, Petunjuk, dan Bobot. Setelah dilakukan pengsisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data *technical factor* yang sudah di simpan akan tampil pada tabel daftar *technical factor* yang berada di sebelahnya.

m. Halaman *Maintenance Data Environmental Factor*

Penentuan Harga Perkiraan Sendiri															
<b>Menu</b> <ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul> <b>Master</b> <ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<h3>Hitung Environmental Complexity Factor (ECF)</h3> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="background-color: #cccccc;">Technical Complexity Factor</th> <th style="background-color: #cccccc;">Environmental Complexity Factor</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center; padding: 10px;"> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <b>Insert Data</b> </div> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="flex: 1;"> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">ID ECF</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Keterangan</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Petunjuk</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Bobot</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="text-align: right; margin-top: 10px;"><input type="button" value="Simpan"/></div> </div> </div></td> </tr> <tr> <td colspan="2" style="text-align: center; padding: 10px;"> <b>Daftar ECF</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID ECF</th> <th>Keterangan</th> <th>Petunjuk</th> <th>Bobot</th> </tr> </thead> <tbody> <tr> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </tbody> </table> </td> </tr> </tbody> </table>	Technical Complexity Factor	Environmental Complexity Factor	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <b>Insert Data</b> </div> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="flex: 1;"> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">ID ECF</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Keterangan</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Petunjuk</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Bobot</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="text-align: right; margin-top: 10px;"><input type="button" value="Simpan"/></div> </div> </div>		<b>Daftar ECF</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID ECF</th> <th>Keterangan</th> <th>Petunjuk</th> <th>Bobot</th> </tr> </thead> <tbody> <tr> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </tbody> </table>		ID ECF	Keterangan	Petunjuk	Bobot	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Technical Complexity Factor	Environmental Complexity Factor														
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <b>Insert Data</b> </div> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="flex: 1;"> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">ID ECF</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Keterangan</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Petunjuk</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="display: flex; justify-content: space-between; width: 100%;"> <div style="width: 45%;">Bobot</div> <div style="width: 45%;"><input type="text"/></div> </div> <div style="text-align: right; margin-top: 10px;"><input type="button" value="Simpan"/></div> </div> </div>															
<b>Daftar ECF</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID ECF</th> <th>Keterangan</th> <th>Petunjuk</th> <th>Bobot</th> </tr> </thead> <tbody> <tr> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </tbody> </table>		ID ECF	Keterangan	Petunjuk	Bobot	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>						
ID ECF	Keterangan	Petunjuk	Bobot												
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>												

Gambar 3.25 Desain tampilan insert data Technical Factor

Halaman *maintenance data environmental factor* merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data *environmental factor*. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID *Environmental Factor*, Keterangan, Petunjuk, dan Bobot. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data *environmental factor* yang sudah di simpan akan tampil pada tabel daftar *environmental factor* yang berada di sebelahnya.

n. Halaman *Maintenance Data Effort Rate*

Penentuan Harga Perkiraan Sendiri									
Menu	Data Rate Effort								
<ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul>	<table border="1"> <thead> <tr> <th colspan="2">Insert Data</th> </tr> </thead> <tbody> <tr> <td>ID Rate Effort</td> <td><input type="text"/></td> </tr> <tr> <td>Nilai</td> <td><input type="text"/> Enter Text</td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Insert"/></td> </tr> </tbody> </table>	Insert Data		ID Rate Effort	<input type="text"/>	Nilai	<input type="text"/> Enter Text	<input type="button" value="Insert"/>	
Insert Data									
ID Rate Effort	<input type="text"/>								
Nilai	<input type="text"/> Enter Text								
<input type="button" value="Insert"/>									
Master	<table border="1"> <thead> <tr> <th colspan="2">Daftar Rate Effort</th> </tr> <tr> <th>ID Rate Effort</th> <th>Nilai</th> </tr> </thead> <tbody> <tr> <td colspan="2"></td> </tr> </tbody> </table>	Daftar Rate Effort		ID Rate Effort	Nilai				
Daftar Rate Effort									
ID Rate Effort	Nilai								
<ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>									

Gambar 3.26 Desain tampilan insert data Effort Rate

Halaman *maintenance* data *effort rate* merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data *effort rate*. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID *Effort Rate*, dan Nilai. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data *effort rate* yang sudah di simpan akan tampil pada tabel daftar *effort rate* yang berada di sebelahnya.

### o. Halaman *Maintenance* Data Jenis Biaya

Penentuan Harga Perkiraan Sendiri															
<b>Menu</b> <ul style="list-style-type: none"> <li>- <a href="#">Home</a></li> <li>- <a href="#">Tentang Aplikasi</a></li> </ul> <b>Master</b> <ul style="list-style-type: none"> <li>- User</li> <li>- Jabatan</li> <li>- Wilayah</li> <li>- Actor</li> <li>- Technical &amp; Environmental Factor (TCF &amp; ECF)</li> <li>- Use Case</li> <li>- Fase Aktivitas</li> <li>- Sub Aktivitas</li> <li>- Aktivitas</li> <li>- Effort</li> <li>- Effort Rate</li> <li>- Jenis Biaya</li> <li>- Biaya</li> </ul>	<b>Data Jenis Biaya</b> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2">Insert Data</th> </tr> </thead> <tbody> <tr> <td style="width: 15%;">ID Jenis Biaya</td> <td><input type="text"/></td> </tr> <tr> <td>Jenis Biaya</td> <td><input type="text" value="Enter Text"/></td> </tr> <tr> <td colspan="2" style="text-align: right;"><input type="button" value="Insert"/></td> </tr> </tbody> </table> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th colspan="2">Daftar Jenis Biaya</th> </tr> <tr> <th>ID Jenis Biaya</th> <th>Jenis Biaya</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="height: 50px;"></td> </tr> </tbody> </table>	Insert Data		ID Jenis Biaya	<input type="text"/>	Jenis Biaya	<input type="text" value="Enter Text"/>	<input type="button" value="Insert"/>		Daftar Jenis Biaya		ID Jenis Biaya	Jenis Biaya		
Insert Data															
ID Jenis Biaya	<input type="text"/>														
Jenis Biaya	<input type="text" value="Enter Text"/>														
<input type="button" value="Insert"/>															
Daftar Jenis Biaya															
ID Jenis Biaya	Jenis Biaya														

Gambar 3.27 Desain tampilan insert data Jenis Biaya

Halaman *maintenance* data jenis biaya merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data jenis biaya. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID Jenis Biaya, dan Jenis Biaya. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data jenis biaya yang sudah di simpan akan tampil pada tabel daftar jenis biaya yang berada di sebelahnya.

p. Halaman *Maintenance Data Biaya*

Penentuan Harga Perkiraan Sendiri								
<b>Menu</b> - <a href="#">Home</a> - <a href="#">Tentang Aplikasi</a>  <b>Master</b> - User - Jabatan - Wilayah - Actor - Technical & Environmental Factor (TCF & ECF) - Use Case - Fase Aktivitas - Sub Aktivitas - Aktivitas - Effort - Effort Rate - Jenis Biaya - Biaya	<b>Data Biaya</b> <div style="display: flex; justify-content: space-between;"> <div style="flex: 1;"> <b>Insert Data</b> <hr/>           ID Biaya <input type="text"/>            Jenis Biaya <input type="text"/>            Nama Biaya <input type="text" value="Enter Text"/> </div> <div style="flex: 1; text-align: right;"> <b>Daftar Biaya</b> <hr/> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ID Biaya</th> <th>Jenis Biaya</th> <th>Nama Biaya</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table> </div> </div>	ID Biaya	Jenis Biaya	Nama Biaya				
ID Biaya	Jenis Biaya	Nama Biaya						

Gambar 3.28 Desain tampilan insert data Biaya

Halaman *maintenance* data Biaya merupakan halaman yang berfungsi untuk menambahkan, mengubah, dan menghapus data Biaya. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID Biaya, Jenis Biaya, Nama Biaya. Setelah dilakukan pengisian kemudian diklik tombol *insert* untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data biaya yang sudah di simpan akan tampil pada tabel daftar biaya yang berada di sebelahnya.

q. Halaman Inisialisasi Data Proyek

Pengguna	Jumlah	Nilai
Simple	<input type="text"/>	<input type="text"/>
Medium	<input type="text"/>	<input type="text"/>
Kompleks	<input type="text"/>	<input type="text"/>
		Unadjusted Actor Weight <input type="text"/>

Fungsi Aplikasi	Jumlah	Nilai
Simple	<input type="text"/>	<input type="text"/>
Medium	<input type="text"/>	<input type="text"/>
Kompleks	<input type="text"/>	<input type="text"/>
		Unadjusted Use Case Weight <input type="text"/>

**Selanjutnya**

Gambar 3.29 Tampilan inisialisasi data proyek

Halaman inisialisasi data proyek berisikan data proyek yang akan dihitung atau ditentukan nilai estimasi biayanya. Data yang harus di inputkan pada halaman ini adalah ID Proyek, Nama Proyek, Data jumlah pengguna yang dibagi menjadi 3 (tiga) yaitu Simpel, Medium, dan Kompleks berdasarkan interaksinya. Dan jumlah fungsi aplikasi yang juga dibagi menjadi 3 (tiga) yaitu Simpel, Medium, dan Kompleks berdasarkan banyaknya transaksi. Untuk melanjutkan ke halaman berikutnya, pengguna dapat mengklik tombol selanjutnya.

r. Halaman Kuesioner *Technical Complexity Factor*

Gambar 3.30 Tampilan kuesioner Technical Complexity Factor

Halaman ini bertujuan untuk menghitung nilai *Technical Complexity Factor*.

Untuk mendapatkan nilai tersebut pengguna harus memberikan skor pada setiap faktor teknis yang mempengaruhi pembuatan proyek aplikasi. Parameter skor yang digunakan pada halaman ini yaitu sangat tidak setuju, tidak setuju, cukup, setuju, dan sangat setuju. Dimana parameter-parameter tersebut sudah memiliki nilai masing-masing. Setelah melakukan skoring pada setiap faktor teknis maka akan didapat nilai *Teknical Complexity Factor*. Untuk melanjutkan ke halaman selanjutnya pengguna dapat mengklik tombol selanjutnya. Dan apabila ingin ke halaman sebelumnya pengguna dapat mengklik tombol sebelumnya.

s. Halaman Kuesioner *Environmental Complexity Factor*

Gambar 3.31 Tampilan kuesioner Environmental Complexity Factor

Halaman ini bertujuan untuk menghitung nilai *Environmental Complexity Factor*. Untuk mendapatkan nilai tersebut pengguna harus memberikan skor pada setiap faktor lingkungan yang mempengaruhi pembuatan proyek aplikasi. Parameter skor yang digunakan pada halaman ini yaitu sangat tidak setuju, tidak setuju, cukup, setuju, dan sangat setuju. Dimana parameter-parameter tersebut sudah memiliki nilai masing-masing. Setelah melakukan skoring pada setiap faktor lingkungan maka akan didapat nilai *Environmental Complexity Factor*. Untuk melanjutkan ke halaman selanjutnya pengguna dapat mengklik tombol selanjutnya. Dan apabila ingin ke halaman sebelumnya pengguna dapat mengklik tombol sebelumnya.

## B. Desain Output

### a. Halaman Estimasi Effort

Gambar 3.32 Desain tampilan estimasi effort

Halaman ini berfungsi untuk menampilkan data estimasi *effort* yang dibutuhkan untuk membuat proyek perangkat lunak berdasarkan data proyek perangkat lunak yang akan dibuat. Untuk menampilkan data estimasi effort pada halaman ini, pengguna harus memilih *effort rate* yang akan digunakan terlebih dahulu. Setelah *effort rate* di tentukan maka data estimasi effort akan tampil berdasarkan setiap aktivitas dan personil. Untuk melanjutkan ke halaman berikutnya pengguna dapat mengklik tombol selanjutnya dan jika pengguna ingin kembali ke halaman sebelumnya maka pengguna dapat mengklik tombol sebelumnya.

b. Halaman Biaya Langsung Personil

# Penentuan Harga Perkiraan Sendiri

**Menu**

- [Home](#)
- [Tentang Aplikasi](#)

**Perhitungan**

- [Proyek Aplikasi](#)

### Buat Proyek Aplikasi

Inisialisasi Proyek
Kuesioner
Estimasi Effort
Biaya Langsung Personil
Biaya Langsung Non Personil
Estimasi Biaya
Selesaikan

Tarif

Wilayah

Proses

Aktivitas	Manajer Proyek	Sistem Analis	Sistem Desain	Programmer	Total
<b>Total</b>					

Sebelumnya
Selanjutnya

Gambar 3.33 Tampilan Biaya Langsung Personil

Halaman ini bertujuan untuk menghasilkan biaya langsung personil, untuk menghasilkan biaya langsung personil pengguna harus memilih tarif yang akan digunakan serta di wilayah mana proyek aplikasi akan dibuat. Setelah memilih tarif dan wilayah pada halaman ini maka data biaya langsung personil akan tampil secara detail baik berdasarkan setiap aktivitas maupun setiap personil. Untuk melanjutkan ke halaman berikutnya pengguna dapat mengklik tombol selanjutnya dan jika pengguna ingin kembali ke halaman sebelumnya maka pengguna dapat mengklik tombol sebelumnya.

### c. Halaman Biaya Langsung Non Personil

Gambar 3.34 Tampilan detail estimasi effort

Halaman biaya langsung non personil merupakan halaman yang berfungsi untuk menambahkan biaya langsung non personil yang dikeluarkan pada saat pelaksanaan proyek pengembangan perangkat lunak. Pada halaman ini terdapat data yang harus diisi oleh pengguna yaitu ID Biaya non personil, Jenis Biaya, Nama biaya, Kuantitas, Harga, dan Jumlah. Setelah dilakukan pengisian kemudian diklik tombol simpan untuk melakukan penyimpanan data pada *database*. Jika data sudah tersimpan di *database* maka data biaya langsung non personil yang sudah di simpan akan tampil pada tabel daftar biaya non personil yang berada di bawahnya. Untuk melanjutkan ke halaman berikutnya pengguna dapat mengklik tombol selanjutnya dan jika pengguna ingin kembali ke halaman sebelumnya maka pengguna dapat mengklik tombol sebelumnya.

#### d. Halaman Estimasi Biaya

Penentuan Harga Perkiraan Sendiri

Buat Proyek Aplikasi

Menu

- Home
- Tentang Aplikasi

Perhitungan

- Proyek Aplikasi

Biaya Langsung Personil

No	Jabatan	Jumlah

Keuntungan (%)

Keuntungan (Rp)

Total Biaya Langsung Personil

Biaya Langsung Non Personil

No	Nama Biaya	Kuantitas	Jumlah	Harga	Subtotal

Total Biaya Langsung Non Personil

Total Biaya   
PPN   
Grand Total Biaya

Sebelumnya  Selanjutnya

Gambar 3.35 Halaman Estimasi Biaya

Halaman estimasi biaya ini menampilkan hasil akhir dari perhitungan yang dilakukan pengguna. Untuk menjalankan halaman ini pengguna hanya cukup *refresh* data dengan mengklik tombol *refresh* data. Setelah memilih data akan tampil di bawahnya, yaitu yang berisikan informasi biaya langsung personil dan biaya langsung non personil. Pada halaman ini juga menyediakan keuntungan yang harus diisi oleh pengguna. Untuk melanjutkan ke halaman berikutnya pengguna dapat mengklik tombol selanjutnya dan jika pengguna ingin kembali ke halaman sebelumnya maka pengguna dapat mengklik tombol sebelumnya.

### e. Halaman Akhir Proyek Aplikasi

Penentuan Harga Perkiraan Sendiri

Menu

- [Home](#)
- [Tentang Aplikasi](#)

Perhitungan

- [Proyek Aplikasi](#)

Buat Proyek Aplikasi

Inisialisasi Proyek | Kuesioner | Estimasi Effort | Biaya Langsung Personil | Biaya Langsung Non Personil | Estimasi Biaya | Selesai

Kembali ke Daftar Proyek Aplikasi | Cetak

Sebelumnya

Gambar 3.36 Halaman Akhir Proyek Aplikasi

Halaman ini merupakan halaman akhir dari proses perhitungan estimasi biaya pembuatan proyek perangkat lunak. Pada halaman ini terdapat 2 (dua) tombol pilihan yaitu kembali ke daftar proyek aplikasi yang akan membawa pengguna kembali ke halaman awal perhitungan proyek aplikasi dan tombol cetak yang berfungsi untuk mencetak hasil dari estimasi biaya yang telah dihitung.

## f. Laporan Harga Perkiraan Sendiri

**Penentuan Harga Perkiraan Sendiri**

Menu	Laporan Harga Perkiraan Sendiri																																										
- Home																																											
- Tentang Aplikasi																																											
Perhitungan	<p style="text-align: center;">Laporan Harga Perkiraan Sendiri Pembuatan Aplikasi Keuangan pemerintah kabupaten Sidoarjo</p> <p><b>Biaya Langsung Personil</b></p> <table border="1"> <thead> <tr> <th>No</th> <th>Jabatan</th> <th>Jumlah</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> </tbody> </table> <p><b>Sub Total Biaya Langsung Personil</b></p> <table border="1"> <thead> <tr> <th>Keuntungan</th> <th></th> </tr> </thead> <tbody> <tr><td> </td><td> </td></tr> <tr><td><b>Total Biaya Langsung Personil</b></td><td> </td></tr> </tbody> </table> <p><b>Biaya Langsung Non Personil</b></p> <table border="1"> <thead> <tr> <th>No</th> <th>Nama Biaya</th> <th>Kuantitas</th> <th>Jumlah</th> <th>Harga</th> <th>Subtotal</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td colspan="6"><b>Total Biaya Langsung Non Personil</b></td></tr> </tbody> </table> <p style="text-align: right;">Total Biaya _____ PPN _____ Grand Total _____</p>	No	Jabatan	Jumlah										Keuntungan				<b>Total Biaya Langsung Personil</b>		No	Nama Biaya	Kuantitas	Jumlah	Harga	Subtotal													<b>Total Biaya Langsung Non Personil</b>					
No	Jabatan	Jumlah																																									
Keuntungan																																											
<b>Total Biaya Langsung Personil</b>																																											
No	Nama Biaya	Kuantitas	Jumlah	Harga	Subtotal																																						
<b>Total Biaya Langsung Non Personil</b>																																											
- Proyek Aplikasi																																											

Gambar 3.37 Tampilan Laporan harga perkiraan sendiri

Halaman ini menampilkan laporan hasil perhitungan seluruh estimasi biaya yang telah dilakukan. Laporan ini berisikan informasi mengenai jumlah biaya langsung personil, jumlah biaya langsung non personil, keuntungan, pajak, dan total seluruh estimasi biaya.

### 3.3 Membuat Coding

Aplikasi akan dirancang berbasis web, Bahasa yang digunakan yaitu PHP.

Tools yang digunakan yaitu sebagai berikut:

- Notepad++*, Digunakan untuk membuat aplikasi
- MySQL, Sebagai database sistem
- XAMPP, Sebagai web server

### 3.4 Pengujian Program

Uji coba ini dilakukan untuk memastikan beberapa hasil penelitian telah dibuat dengan benar mengikuti tujuan dan kebutuhan yang diharapkan. Uji coba dilakukan dengan cara pengujian *black box testing*. Dari hasil uji coba memungkinkan ditemukan kekurangan yang dapat dijadikan ulasan pada tahapan perbaikan aplikasi. Berikut merupakan tabel sekenario pengujian penelitian.

Tabel 3.38 Desain Uji Coba Form Login

Object Pengujian		Form Login	
No	Tujuan	Proses	Output yang diharapkan
1.	<b>Uji Coba Halaman Login Aplikasi</b>	Mengetahui tampilan dan fungsi form login dapat berjalan dan menghasilkan <i>output</i> yang diharapkan	
		<i>Login</i> Aplikasi Bagian Admin	Pengguna bagian Admin dapat mengakses aplikasi sesuai hak aksesnya
		<i>Login</i> Aplikasi Bagian Pejabat Pembuat Komitmen (PPK)	Pengguna bagian PPK dapat mengakses aplikasi sesuai hak aksesnya
		Validasi jika Username dan Password yang Di-input Salah	Muncul pemberitahuan bahwa username dan password yang dimasukkan salah

Tabel 3.39 Desain Uji Coba Form Inisialisasi Data Proyek

<b>Object Pengujian</b>		Form Inisialisai Data Proyek	
<b>Keterangan</b>		Mengetahui tampilan dan fungsi form inisialisasi data proyek dapat berjalan dan menghasilkan output sesuai dengan output yang diharapkan	
<b>NO</b>	<b>Tujuan</b>	<b>Proses</b>	<b>Output yang diharapkan</b>
2.	<b>Uji Coba Inisialisasi Proyek</b>	Uji nilai perhitungan <i>unadjusted actor weight</i> per jenis actor	Nilai <i>unadjusted actor weight</i> sesuai dengan rumus perhitungan per jenis <i>actor</i>
		Uji nilai total nilai <i>unadjusted actor weight</i>	nilai total nilai <i>unadjusted actor weight</i>
		Uji nilai perhitungan <i>unadjusted use case weight</i> per jenis use case	Nilai <i>unadjusted use case weight</i> sesuai dengan rumus perhitungan per jenis <i>use case</i>
		Uji nilai total nilai <i>unadjusted use case weight</i>	nilai total nilai <i>unadjusted use case weight</i>

Tabel 3.40 Desain Uji Coba Form Kuesioner Technical Complexity Factor

<b>Object Pengujian</b>		Form Kuesioner Technical Complexity Factor	
<b>Keterangan</b>		Melihat tampilan dan fungsi form kusioner <i>technical complexity factor</i> dapat berjalan dan output yang dihasilkan sesuai dengan yang di harapkan.	
No	Nama Tes	Proses	Output yang diharapkan
3.	<b>Uji Coba Kuesioner <i>Technical Complexity Factor</i></b>	Uji Coba Nilai per Faktor	Nilai per Faktor
		Uji Coba Total Nilai <i>Technical Factor</i>	Total nilai <i>Technical Factor</i>
		Uji nilai <i>technical complexity factor</i>	Nilai <i>technical complexity factor</i> sesuai dengan rumus perhitungan

Tabel 3.41 Desain Uji Coba Form Kuesioner Environmental Complexity Factor

<b>Object Pengujian</b>		Form Kuesioner Environmental Factor	
<b>Keterangan</b>		Melihat tampilan dan fungsi form kusisioner <i>environmental complexity factor</i> dapat berjalan dan output yang dihasilkan sesuai dengan yang di harapkan.	
No	Tujuan	Proses	Output yang diharapkan
4.	<b>Uji Coba Kuesioner <i>Environmental Complexity Factor</i></b>	Uji Coba Nilai per Faktor	Nilai per Faktor
		Uji Coba Total Nilai <i>Environmental Factor</i>	Total nilai <i>Environmental Factor</i>
		Uji nilai <i>environmental complexity factor</i>	Nilai <i>environmental complexity factor</i> sesuai dengan rumus perhitungan

Tabel 3.42 Desain Uji Coba Perhitungan Distribusi Effort

<b>Object Pengujian</b>	Form Distribusi Effort		
<b>Keterangan</b>	Melihat tampilan dan fungsi form distribusi effort dapat berjalan dan output yang dihasilkan sesuai dengan yang di harapkan.		
No	Tujuan	Proses	Output yang diharapkan
5.	<b>Uji Coba Perhitungan Estimasi Effort</b>	Uji Perhitungan Nilai <i>Effort</i>	Nilai estimasi <i>effort</i> sesuai dengan rumus perhitungan

Tabel 3.43 Desain Uji Coba Perhitungan Biaya Langsung Personil

<b>Object Pengujian</b>	Form Biaya Langsung Personil		
<b>Keterangan</b>	Melihat tampilan dan fungsi form biaya langsung personil dapat berjalan dan output yang dihasilkan sesuai dengan yang di harapkan.		
No	Tujuan	Proses	Output yang diharapkan
6.	<b>Uji Coba Perhitungan Biaya Langsung Personil</b>	Uji Perhitungan Biaya Langsung Personil	Nilai Biaya langsung personil sesuai dengan rumus perhitungan

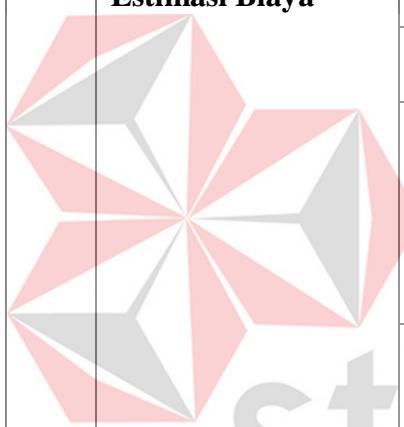
Tabel 3.44 Desain Uji Coba Perhitungan Biaya Langsung Non Personil

<b>Object Pengujian</b>		Form Biaya Langsung Non Personil	
<b>Keterangan</b>		Melihat tampilan dan fungsi form biaya langsung non personil dapat berjalan dan output yang dihasilkan sesuai dengan yang diharapkan	
No	Tujuan	Proses	Output yang diharapkan
7.	<b>Uji Coba Perhitungan Biaya Langsung Non Personil</b>	Uji coba tambah biaya langsung non personil Uji Perhitungan Biaya Langsung Non Personil	Menampilkan form tambah biaya langsung non personil Nilai Biaya langsung non personil sesuai dengan rumus perhitungan

Tabel 3.45 Desain Uji Coba Perhitungan Estimasi Biaya

<b>Object Pengujian</b>		Form Estimasi Biaya	
<b>Keterangan</b>		Melihat tampilan dan fungsi form estimasi biaya dapat berjalan dan output yang dihasilkan sesuai dengan yang di harapkan.	
No	Tujuan	Proses	Output yang diharapkan
8.	<b>Uji Coba</b>	Uji total biaya	Nilai total biaya langsung

<b>Object Pengujian</b>		Form Estimasi Biaya	
<b>Keterangan</b>		Melihat tampilan dan fungsi form estimasi biaya dapat berjalan dan output yang dihasilkan sesuai dengan yang di harapkan.	
No	Tujuan	Proses	Output yang diharapkan
	<b>Perhitungan Estimasi Biaya</b>	langsung personil	personil
	Uji total keuntungan		Nilai total keuntungan
	Uji total biaya langsung non personil		Nilai total biaya langsung non personil
	Uji total Pajak		Nilai Pajak



**INSTITUT BISNIS  
DAN INFORMATIKA**  
**stikom**  
**SURABAYA**

Tabel 3.46 Desain Uji Coba Laporan Harga Perkiraan Sendiri

<b>Object Pengujian</b>	Form Laporan Harga Perkiraan sendiri		
<b>Keterangan</b>	Melihat tampilan dan fungsi form laporan harga perkiraan sendiri dapat berjalan dan output yang dihasilkan sesuai dengan yang di harapkan.		
No	Tujuan	Proses	Output yang diharapkan
9.	<b>Uji Coba Cetak Laporan</b>	Uji Cetak Laporan	Laporan Hasil Perhitungan Harga Perkiraan Sendiri Proyek Perangkat Lunak

