

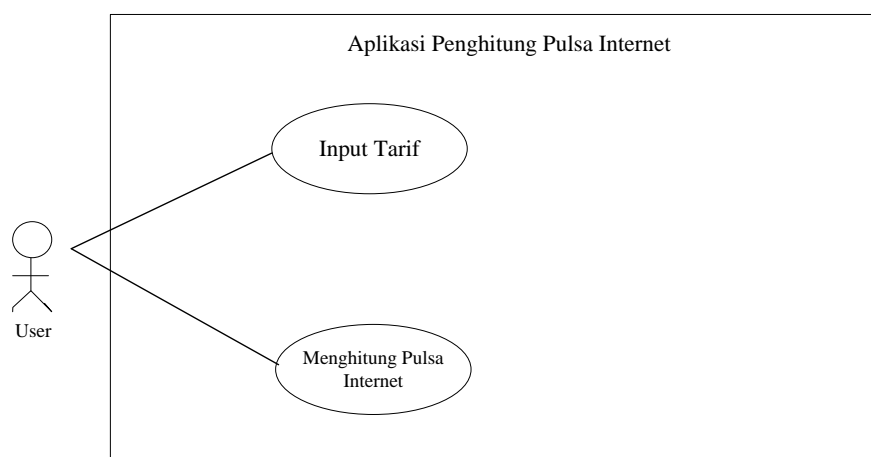
## BAB III

### PERANCANGAN DAN PEMBUATAN APLIKASI

#### 3.1 PERANCANGAN DIAGRAM *USE CASE*

Diagram *use case* menggambarkan aktifitas yang dilakukan oleh suatu sistem dan lebih membahas apa yang dilakukan sistem, bukan bagaimana melakukannya. Diagram *use case* ini menjelaskan kebutuhan yang dikehendaki pengguna, serta sangat berguna dalam menentukan struktur organisasi dan model dari pada sebuah sistem

Adapun *use case* diagram aplikasi penghitung pulsa internet ini seperti pada Gambar 3.1.

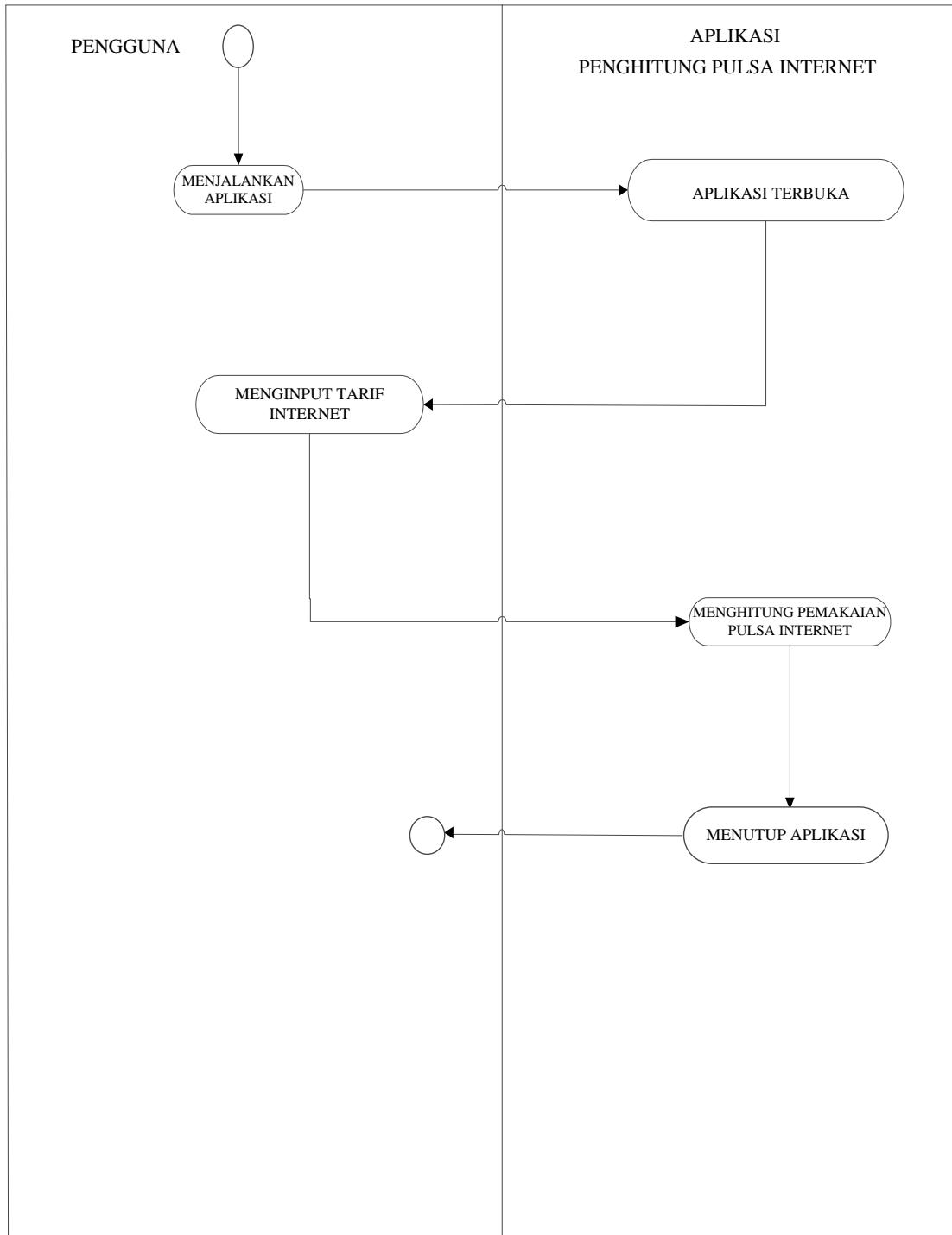


Gambar 3.1 Diagram *Use Case*

#### 3.2 PERANCANGAN DIAGRAM *ACTIVITY*

Diagram *activity* memperlihatkan aliran urutan aktifitas dari sebuah aplikasi yang menunjukkan bagaimana aktifitas-aktifitas satu sama lain.

Adapun diagram *activity* aplikasi penghitung pulsa internet ini seperti pada Gambar 3.2.

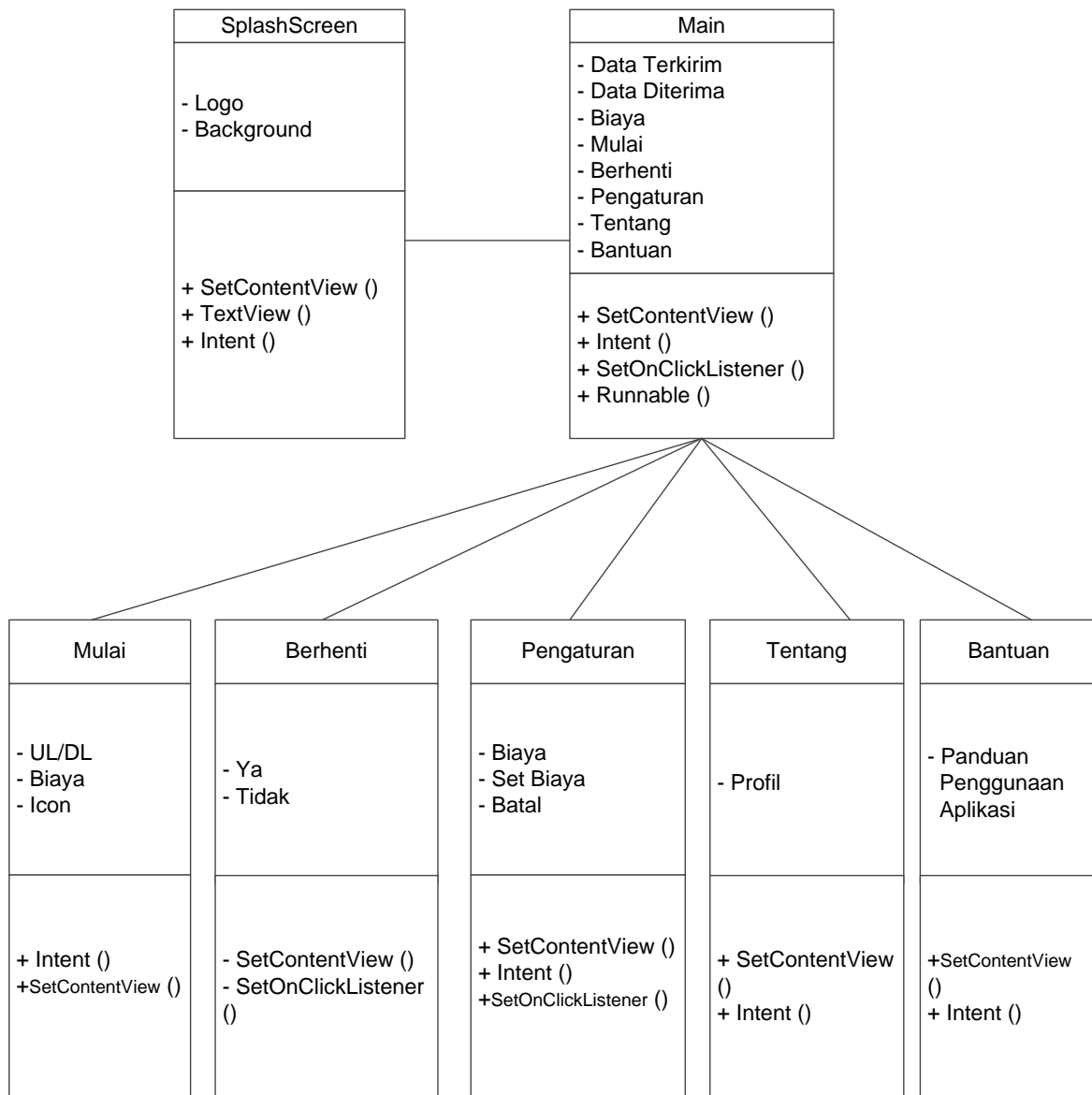


Gambar 3.2 Diagram Activity

### 3.3 PERANCANGAN DIAGRAM CLASS

Diagram *Class* menggambarkan hubungan apa yang terjadi, bukan apa yang terjadi jika kelas-kelas berhubungan.

Adapun diagram *class* dari aplikasi penghitung pulsa internet seperti pada Gambar 3.3.

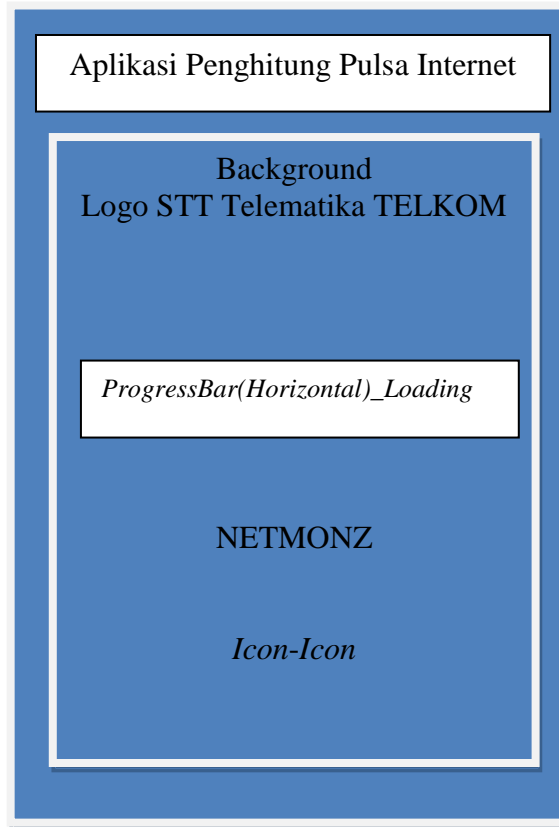


Gambar 3.3 Diagram *Class*

### 3.4 PERANCANGAN DESAIN

Aplikasi ini dibuat menggunakan Eclipse Indigo, yang terdiri dari beberapa *activity* pendukung aplikasi penghitung pulsa internet. Adapun desain rancangan dari tampilan *layout activity* tersebut antara lain:

### 1. Activity Splash Screen



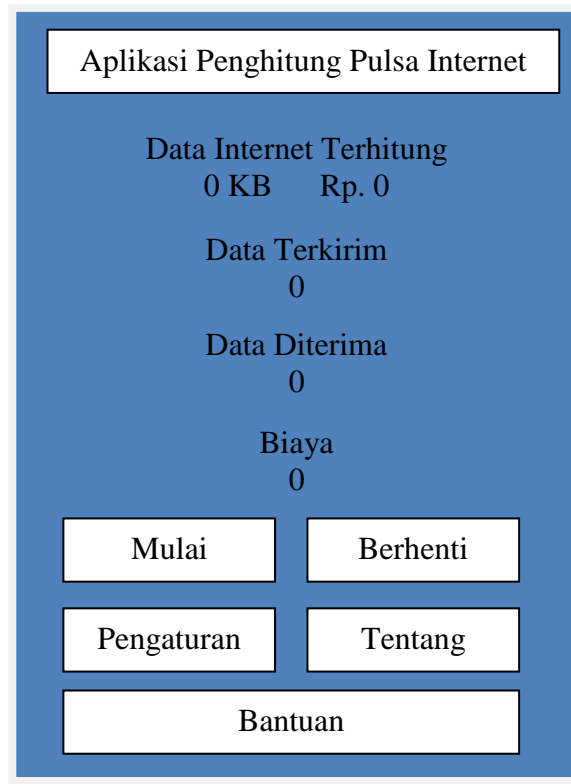
Gambar 3.4 Desain *Layout Activity Splash Screen*

*Activity* ini berupa halaman utama untuk memulai aplikasi penghitung pulsa internet yang berupa *splash screen*. Adapun obyek penyusun dari *activity splash screen* seperti pada Tabel 3.1.

Tabel 3.1 Obyek Penyusun *Activity Splash Screen*

<i>Value</i>	<i>Type Data</i>	<i>Size</i>	<i>Pallete &amp; Id</i>	<b>Keterangan</b>
<i>Background</i> : - Logo - Judul Aplikasi - <i>Icon</i>	-	-	<i>Images&amp;Media</i> : <i>ImageView</i>	<i>ContentDescription</i> : <i>@drawable/splash</i> . Mengacu pada <i>background</i>
<i>Loading</i>	<i>String</i>	-	<i>Form Widgets</i> : <i>Textview</i>	Mengacu pada keterangan <i>progressbar</i>
<i>ProgressBar</i>	-	-	- <i>Form Widgets</i> : <i>ProgressBar</i> ( <i>Horizontal</i> ) - <i>progressBar1</i>	Mengacu pada pembukaan aplikasi, waktu yang dibutuhkan untuk membuka aplikasi

## 2. Activity Main



Gambar 3.5 Desain *Layout Activity Main*

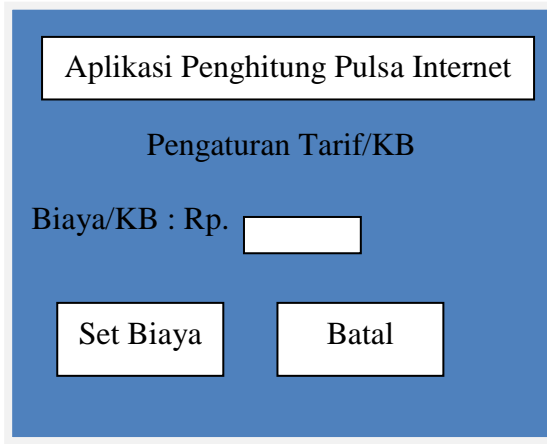
*Activity* ini berupa *layout* untuk menampilkan data internet yang terhitung dan biaya. Pada *activity* ini juga terdapat empat *button* (menu), untuk penggunaan aplikasi lebih lanjut. Adapun obyek penyusun dari *activity main* seperti pada Tabel 3.2.

Tabel 3.2 Obyek Penyusun *Activity Main*

<i>Value</i>	<i>Type Data</i>	<i>Size</i>	<i>Pallete &amp; Id</i>	<b>Keterangan</b>
Data Internet Terhitung	<i>String</i>	18	- <i>Form Widgets</i> : <i>Textview</i> - Judul	Judul
Data Terkirim	<i>String</i>	16	<i>Form Widgets</i> : <i>Textview</i>	Judul
0	<i>String</i>	16	- <i>Form Widgets</i> : <i>Textview</i> - TX	Menampilkan <i>byte</i> yang berjalan setiap KB-nya

<i>Value</i>	<i>Type Data</i>	<i>Size</i>	<i>Pallete &amp; Id</i>	<b>Keterangan</b>
Data Diterima	<i>String</i>	16	<i>Form Widgets : Textview</i>	Judul
0	<i>String</i>	16	- <i>Form Widgets : Textview</i> - RX	Menampilkan <i>byte</i> yang berjalan setiap KB-nya
Biaya	<i>String</i>	16	<i>Form Widgets : Textview</i>	Judul
0	<i>String</i>	16	- <i>Form Widgets : Textview</i> - <i>Cost</i>	Menampilkan biaya yang berjalan setiap KB-nya dalam satuan Rp
Mulai	<i>String</i>	15	- <i>Form Widgets : ButtonSmall</i> - <i>btn_start</i>	Memulai penghitungan dan muncul sebagai notifikasi
Berhenti	<i>String</i>	15	- <i>Form Widgets : ButtonSmall</i> - <i>btn_stop</i>	Menutup aplikasi dan memberhentikan penghitungan
Pengaturan	<i>String</i>	15	- <i>Form Widgets : ButtonSmall</i> - <i>btn_setting</i>	Mengatur tarif internet dari suatu operator per KB-nya.
Tentang	<i>String</i>	15	- <i>Form Widgets : ButtonSmall</i> - <i>btn_tentang</i>	Menampilkan judul aplikasi, logo, pembuat, email
Bantuan	<i>String</i>	15	- <i>Form Widgets : ButtonSmall</i> - <i>btn_help</i>	Menampilkan petunjuk/panduan untuk menggunakan aplikasi.

### 3. Activity Setting



Gambar 3.6 Desain *Layout Activity Setting*

*Activity* ini berupa *layout* untuk menampilkan pengaturan tarif/KB yang telah ditetapkan oleh operator Indosat, berapa rupiah/KB-nya. Adapun obyek penyusun dari *activity setting* seperti pada Tabel 3.3.

Tabel 3.3 Obyek Penyusun *Activity Setting*

<i>Value</i>	<i>Type Data</i>	<i>Size</i>	<i>Pallete &amp; Id</i>	<b>Keterangan</b>
Pengaturan Tarif/KB	<i>String</i>	16	- <i>Form Widgets</i> : <i>Textview</i> - <i>textview1</i>	Judul
Biaya/KB	<i>String</i>	15	- <i>Form Widgets</i> : <i>Textview</i> - <i>txt_setting</i>	Judul
Biaya	<i>Number</i>	12	- <i>Text Fields</i> : <i>Number</i> - <i>txb_biaya</i>	<i>Textbox</i> untuk <i>setting</i> biaya/KB Contoh : 1.0
Set Biaya	<i>String</i>	15	- <i>Form Widgets</i> : <i>Button</i> - <i>btn_ok</i>	<i>Button</i> untuk menetapkan pengaturan biaya/KB (Ok)
Batal	<i>String</i>	15	- <i>Form Widgets</i> : <i>Button</i> - <i>btn_cancel</i>	<i>Button</i> untuk membatalkan pengaturan biaya

#### 4. Activity About



Gambar 3.7 Desain *Layout Activity About*

*Activity* ini berupa *layout* untuk menampilkan profil berupa judul aplikasi, logo, pembuat dan email. Adapun obyek penyusun dari *activity about* seperti pada Tabel 3.4.

Tabel 3.4 Obyek Penyusun *Activity About*

<b>Value</b>	<b>Type Data</b>	<b>Size</b>	<b>Pallete &amp; Id</b>	<b>Keterangan</b>
Netmonz	<i>String</i>	18	- <i>Form Widgets:</i> <i>Textview</i> <i>-namaaplikasi</i>	Nama aplikasi
Aplikasi Penghitung Pulsa Internet Berbasis Volume Based Pada Handphone	<i>String</i>	15	- <i>Form Widgets :</i> <i>Textview</i> <i>- judul</i>	Judul aplikasi
Logo STT Telematika TELKOM	<i>String</i>	-	- <i>Form Widgets :</i> <i>ImageView</i> <i>- imageView1</i>	Logo STT Telematika TELKOM



<i>Value</i>	<i>Type Data</i>	<i>Size</i>	<i>Pallete &amp; Id</i>	<b>Keterangan</b>
Oleh	<i>String</i>	12	- <i>Form Widgets</i> : <i>Textview</i>	Judul
Sri Maya Sari Nainggolan	<i>String</i>	15	- <i>Form Widgets</i> : <i>Textview</i>	Pembuat
Email	<i>String</i>	12	- <i>Form Widgets</i> : <i>Textview</i>	Judul
Srymayasari nainggolan@yahoo.com	<i>String</i>	15	- <i>Form Widgets</i> : <i>Textview</i>	Email pembuat
Kembali	<i>String</i>	15	- <i>Form Widgets</i> : <i>Textview</i> - <i>btn_back</i>	<i>Button</i> untuk kembali ke menu awal

### 5. Activity Help

Aplikasi Penghitung Pulsa Internet

Panduan Untuk Menggunakan Aplikasi Penghitung Pulsa Internet

1. Pastikan anda sudah mengisi paket internet
2. Pilih menu pengaturan
3. Isikan Biaya/KB
4. Klik Set Biaya
5. Klik Menu Mulai, maka notifikasi akan tampil
6. Jika ingin melihat profil pembuat aplikasi, Klik Menu Tentang
7. Untuk menghentikan aplikasi, Klik Menu Berhenti

Kembali

Gambar 3.8 Tampilan *Activity Help*

*Activity* ini berupa *layout* untuk menampilkan profil berupa panduan untuk menggunakan aplikasi penghitung pulsa internet seperti pada Tabel 3.5.

Tabel 3.5 Obyek Penyusun *Activity Help*

<i>Value</i>	<i>Type Data</i>	<i>Size</i>	<i>Pallete &amp; Id</i>	<b>Keterangan</b>
Panduan untuk menggunakan aplikasi penghitung pulsa internet	<i>String</i>	10	- <i>Form Widgets</i> : <i>Textview</i>	Langkah-langkah menggunakan aplikasi

### 3.5 PERANCANGAN DATA

Perancangan data terdiri dari rancangan data aplikasi penghitung pulsa internet meliputi *input* data dan *output* data, antara lain :

#### 1. *Input* Data

*Input* data dari aplikasi penghitung pulsa internet berupa *input* tarif/KB yang ditetapkan oleh provider IM3.

#### 2. *Output* Data

*Output* data yang dihasilkan berupa data internet yang terhitung berupa *byte* yang keluar dan masuk dan biaya data yang dikenakan setiap KB-nya.

### 3.6 PERANCANGAN *HARDWARE*

Aplikasi penghitung pulsa internet ini terdiri dari beberapa perangkat keras untuk mendukung kinerja program. Berikut spesifikasi perangkat keras yang digunakan dalam mengembangkan aplikasi penghitung pulsa internet ini, yakni :

#### 1. *Personal Computer (PC)*

Spesifikasi minimum *personal computer* yang dapat mendukung kinerja perangkat lunak Eclipse Indigo. Adapun spesifikasi tersebut seperti pada Tabel 3.6

Tabel 3.6 Spesifikasi Perangkat Komputer

Jenis	Sistem Minimum
<i>Processor</i>	Intel Core 2 Duo
RAM	2 GB
VGA	1 GB

Jenis	Sistem Minimum
<i>Operating System</i>	Win XP (32/64 bit), Linux, Win 7 (32/64 bit)
<i>Harddisk</i>	80 GB

## 2. *Handphone*

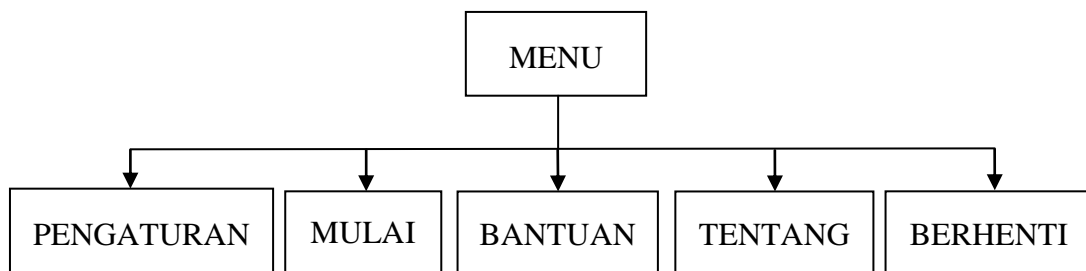
Spesifikasi *handphone* yang dapat digunakan untuk menjalankan aplikasi penghitung pulsa internet seperti pada Tabel 3.7.

Tabel 3.7 Spesifikasi *Handphone*

Jenis	Sistem Minimum
<i>Handphone</i>	<i>Handphone</i> lain yang mendukung sistem operasi Android 2.3.3 (Gingerbread), Samsung Galaxy Young GT-S5360 (yang digunakan).
<i>Operating System</i>	Android 2.3.3 (Gingerbread) dan Android 4.0.3 (IceCreamSandwich)
API	Level 10 dan 15

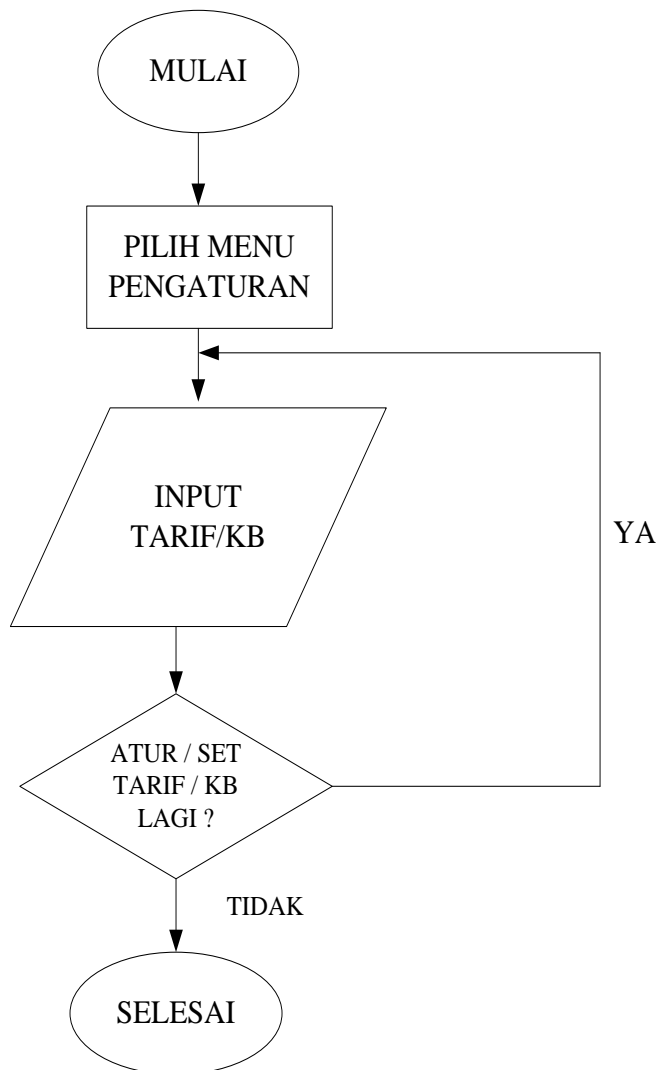
### 3.7 PEMBUATAN STRUKTUR APLIKASI

Tahap pembuatan struktur aplikasi merupakan langkah untuk menggambarkan secara umum bentuk program yang direncanakan dan telah berlangsung. Blok diagram struktur aplikasi penghitung pulsa internet seperti pada Gambar 3.8.



Gambar 3.9 Blok Diagram Struktur Aplikasi

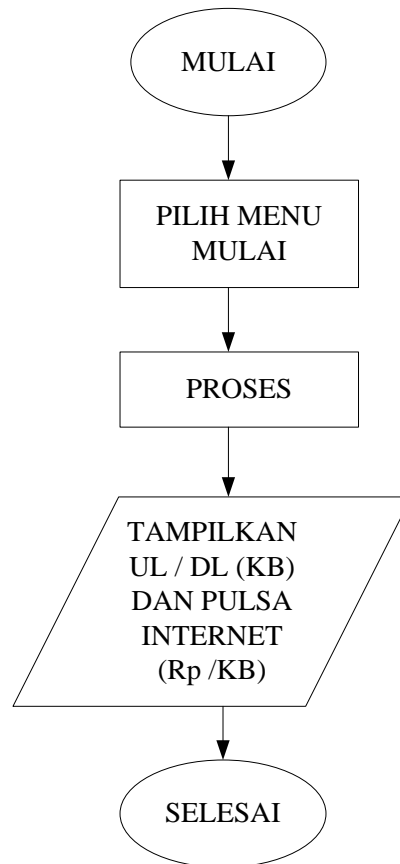
## 1. Blok Diagram Pengaturan



Gambar 3.10 Blok Diagram Menu Pengaturan

Pada Gambar 3.10 dapat dijelaskan bahwa pada menu pengaturan ini terdapat *input*. *Input* berupa tarif internet/KB yang disediakan oleh operator Indosat. Di mana skema perhitungan internet untuk operator Indosat khususnya IM3 adalah Rp 1/KB. Pengaturan untuk tarif internet dapat di-*input* secara manual sesuai dengan skema perhitungan internet yang telah ditetapkan oleh operator Indosat. Pada menu pengaturan pengguna pertama sekali memasukkan biaya/KB kemudian memilih *button* set biaya.

## 2. Blok Diagram Mulai



Gambar 3.11 Blok Diagram Menu Mulai

Pada Gambar 3.11 dapat dijelaskan bahwa pada menu mulai ini akan berisi perhitungan pulsa internet yang berupa pulsa internet, tampilan *upload* (UL) dan *download* (DL). Pada saat pengguna memilih menu mulai maka otomatis proses dan akan menampilkan notifikasi berupa biaya dan UL/DL, di mana sebelumnya pengguna memasukkan nilai Rp 1/KB pada menu pengaturan.

Pada saat menu mulai dipilih dan akan secara langsung program akan membaca dan menampilkan *upload*, *download* dan pulsa internet yang dikenakan pada pengguna, sesuai dengan kebutuhan layanan internet yang digunakan pada masing-masing pengguna.

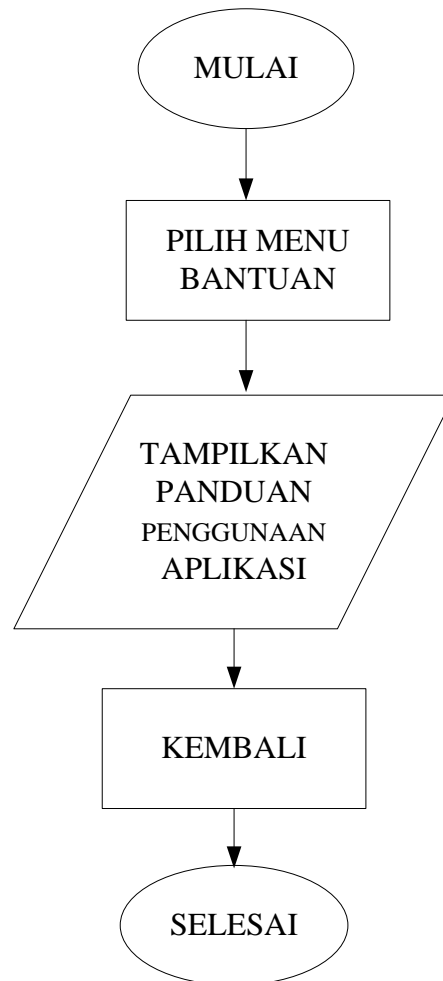
Adapun algoritma perhitungan pulsa internet, jika pada saat menu mulai di pilih yaitu :

Jumlah pemakaian = Tarif internet x Total jumlah data *download* dan *upload*

Maka,

Jumlah pemakaian = Rp 1/KB X (200 KB + 100KB) = Rp 300/KB.

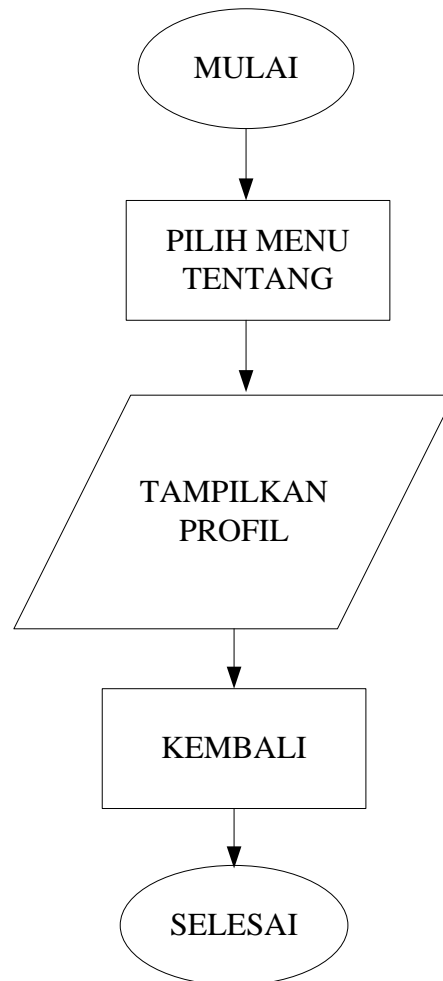
### 3. Blok Diagram Bantuan



Gambar 3.12 Blok Diagram Menu Bantuan

Pada Gambar 3.12 dapat dijelaskan bahwa pada saat pengguna memilih menu bantuan, maka akan menampilkan panduan untuk menggunakan aplikasi, berupa langkah-langkah untuk menggunakan aplikasi penghitung pulsa internet ini. Jika pengguna ingin kembali ke menu awal, pengguna dapat memilih *button* kembali.

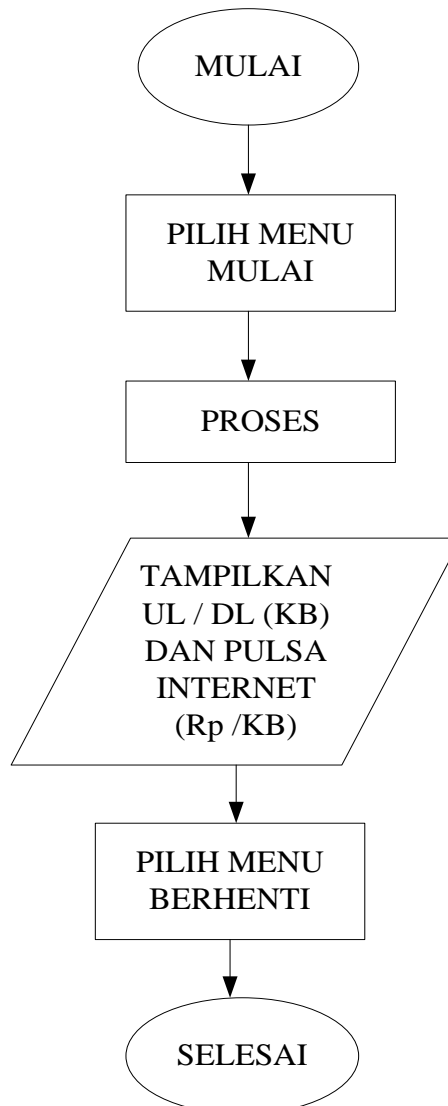
#### 4. Blok Diagram Tentang



Gambar 3.13 Blok Diagram Menu Tentang

Pada Gambar 3.13 dapat dijelaskan bahwa pada saat pengguna memilih menu tentang ini akan menampilkan profil yang berupa nama aplikasi yaitu Netmonz, judul tugas akhir yaitu Aplikasi Penghitung Pulsa Internet Berbasis Volume Based, logo STT Telematika TELKOM, pembuat, email dari pembuat aplikasi dan *button* kembali yang digunakan untuk kembali ke menu sebelumnya.

## 5. Blok Diagram Berhenti



Gambar 3.14 Blok Diagram Menu Berhenti

Pada Gambar 3.14 dapat dijelaskan bahwa pada menu berhenti ini instruksi yang dimasukkan yaitu untuk menghentikan aplikasi untuk mendeteksi kecepatan internet baik *upload* dan *download* pada saat melakukan *browsing*, *streaming* ataupun pada saat akses internet pertama kali. Pada menu berhenti ini juga akan menghentikan aplikasi penghitung pulsa internet tersebut.

Pada saat pengguna memilih menu berhenti ini maka akan muncul *alert* yang berisikan “Keluar dari aplikasi dan hentikan penghitungan”, maka pengguna bebas memilih “Ya” atau “Tidak”.



### 3.8 PEMBUATAN APLIKASI

#### 1. Tahap Implementasi Program

Pada tahap implementasi program mencakup :

##### a. Pengkodean

Tahap pengkodean yang dilakukan adalah dengan menuliskan kode-kode program (*coding*) pada Eclipse Indigo, di mana *list program* terlampir pada lampiran.

##### b. Pengkoneksian

Pada tahap pengkoneksian, yang dilakukan adalah mengkoneksikan aplikasi dengan internet agar *byte* dapat di-*capture* berupa *byte* untuk data terkirim dan *byte* untuk data diterima. Pengkoneksian dilakukan dengan menuliskan *coding* yang akhirnya aplikasi dapat mendeteksi *byte* yang keluar dan masuk.

##### c. Pemasukan Data

Setelah tahap pengkoneksian antara aplikasi dengan internet selesai, maka tahap berikutnya adalah melakukan pemasukan data berupa tarif/KB yang disediakan oleh operator Indosat khususnya IM3. Pada tahap *input* dilakukan dengan cara memasukkan tarif/KB yang sudah ditetapkan oleh pihak operator, yang terdapat pada bagian menu pengaturan.

#### 2. Langkah-Langkah Pembuatan Aplikasi

Dalam pembuatan program Aplikasi Penghitung Pulsa Internet terdapat langkah – langkah yang dilakukan antara lain :

##### a. Pembuatan *Project* pada Eclipse Indigo

a) Memilih *start* lalu memilih *eclipse.exe*.

b) Pada *menu bar* memilih menu *file* lalu memilih *New* kemudian memilih *Android Application Project*.

c) Pada kotak dialog *New Android Application* terdapat beberapa *field* yang harus diisi yaitu :

1) *Application Name* : *NetMon*

2) *Project Name* : *NetMon*

3) *Package Name* : *com.aplikasi.netmon*

- 4) *Minimum Required SDK* : API 10, Android 2.3.3 (Gingerbread)
- 5) *Target SDK* : API 15, Android 4.0.3  
(IceCreamSandwich)
- 6) *Compile With* : API 15, Android 4.0.3  
(IceCreamSandwich)
- 7) *Theme* : *None*

d) Pada kotak dialog *Configure Launcher Icon* memilih *tab clipart* pada *foreground*. *Clipart* disesuaikan dengan *icon* yang ada, di mana *icon* nantinya akan tampil pada aplikasi.

e) Pada kotak dialog *Blank Activity* ada beberapa *field* yang diisi berupa :

- 1) *Activity Name* : *Main*
- 2) *Layout Name* : *activity\_main*
- 3) *Navigation Type* : *None*

f) Setelah selesai kemudian memilih *finish*.

b. Pembuatan *Layout*

Pada tahap pembuatan *layout* ini jika pada saat membuat *project* baru maka secara otomatis di dalam *project* tersebut sudah terdapat satu *layout activity (default)*. Adapun langkah-langkah pembuatan *layout* baru dan mendesain *layout* seperti berikut :

- a) Memilih *folder layout* lalu memilih *2x layout activity main.xml*.
- b) Memilih *layout* untuk *activity* dengan cara, pada *activity* yang kosong berikan *layout LinearLayout (Vertical)*, kemudian mendesain dengan menambahkan *pallette*.
- c) Mendesain menu *button* dan *textview* melalui *pallette* dengan cara :
  - 1) Menarik dan menahan *pallette button* ke *layout activity main*.
  - 2) Mendesain dan mengatur sesuai perancangan desain aplikasi.
- d) Mengganti tulisan *value button* dan *textview* dengan cara :
  - 1) Pada *button*, dengan cara memilih *button* kemudian menekan tombol kanan pada *button* lalu memilih *edit text*.
  - 2) Mengganti nama *button* dengan Mulai, maka pada *properties text* akan berubah menjadi Mulai.

- 3) Pada *properties* Id menjadi `@+id/button1`.
  - 4) Hal yang sama juga dilakukan untuk memberi/mengganti *value* pada *textview*.
  - e) Memberikan fungsi dari setiap *button* dengan cara memilih *folder* *src* lalu memilih *package* *com.aplikasi.netmon*.
  - f) Pada setiap *button* yang didesain maka otomatis pada *text editor* akan tampil untuk tempat memberikan fungsi-fungsi.
  - g) Penambahan *layout* baru seperti *activity* *splash screen*, *setting* dan *about* dilakukan dengan langkah-langkah berikut:
    - 1) Memilih *folder layout* yang masih berada pada satu *project*.
    - 2) Menekan tombol kanan *folder layout* lalu memilih *New* kemudian memilih *Android XML File*.
    - 3) Pada kotak dialog *New Android XML File* mengisikan *field File* dengan isi *activity\_splash\_screen*.
    - 4) Hal yang sama juga dilakukan untuk menambahkan *layout activity* baru seperti *activity about* dan *setting*.
  - h) Pada *folder layout* nantinya akan terdapat empat *layout* dengan masing-masing *activity* sebagai berikut :
    - 1) *Layout Activity Splash Screen*
    - 2) *Layout Activity Main*
    - 3) *Layout Activity Setting*
    - 4) *Layout Activity About*
    - 5) *Layout Activity Help*
- c. Pembuatan *Package*
- Dalam pembuatan *project* baru secara otomatis atau *default* sudah terdapat satu *package* di dalam *project* dengan nama yang diberikan pada saat pembuatan *project* pertama sekali. Fungsi untuk menambahkan *package* yang baru agar masing-masing *class* tempatnya teratur sehingga tidak disatukan dalam satu *package* bersamaan. Adapun langkah-langkah yang dilakukan untuk pembuatan *package* baru yaitu sebagai berikut :
- a) Pada *project* *Netmon* yang berada pada *folder* *src*, terdapat dua tambahan *package* baru yaitu *com.android.counter.holder* dan *com.authorwjf*.

- b) *Package default* yaitu `com.android.counter.apps` tidak perlu ditambahkan karena sudah *default* ada di dalam pada saat pembuatan *project*.
- c) *Package* baru dapat ditambahkan dengan cara menekan tombol kanan *folder* `src` lalu memilih *New* kemudian memilih *Package*.
- d) Pada kotak dialog *New Java Package* mengisikan *field name* yaitu `com.android.counter.holder`.
- e) Setelah selesai membuat *package*, *package* masih berwarna abu-abu, agar *package* berwarna coklat maka dilakukan dengan cara :
  - 1) Menekan tombol kanan pada *package* `com.android.counter.holder` lalu memilih *New* kemudian memilih *Class*.
  - 2) Pada kotak dialog *Java Class* mengisikan *field name* seperti *SetingActivity*.
  - 3) Setelah selesai maka secara otomatis *package* berubah menjadi warna coklat dan sudah berisikan *class* yang baru dengan ekstensi `.java` yaitu *SetingActivity.java*.
  - 4) Hal yang sama dilakukan untuk menambahkan *package* yang baru.

d. Pembuatan AVD Manager (*Android Virtual Device*)

Pembuatan AVD ini dilakukan agar pada saat proses *running*, *emulator device* sesuai dengan *handphone* yang digunakan yaitu *Android 2.3.3*. Adapun cara pembuatan AVD sebagai berikut :

- a) Ada dua cara menuju AVD yang pertama dengan memilih menu *window* → *Android Virtual Device Manager*, kedua dengan memilih *icon* seperti *handphone* dengan gambar *Android*.
- b) Pada kotak dialog kemudian memilih *new*, lalu mengisikan masing-masing *field* pada kotak dialog tersebut. AVD yang dibuat pada aplikasi ini yaitu :
  - 1) *AVD Name* : `NetMonz`
  - 2) *Device* : `2.7" QVGA slider (240 x 320: 1dpi)`
  - 3) *Target* : `Android 2.3.3 – API Level 10`
  - 4) *CPU / ABI* : `ARM (armeabi)`
- c) Kemudian memilih *OK* → *start*, maka *emulator* dengan *device* *Android 2.3.3* akan tampil.

### 3.9 PERENCANAAN PENGUJIAN APLIKASI

Pengujian dilakukan pada beberapa *layout* seperti pada *layout activity main*, yang diuji untuk menampilkan *byte* yang keluar dan yang masuk apakah berhasil atau tidak dan menampilkan biaya apakah sesuai dengan jumlah *byte* yang masuk dan keluar. Pengujian *layout activity main* dapat dilihat pada *listing* program berikut :

```
private final Runnable mRunnable = new Runnable() {
    public void run() {

        RX = (TextView) findViewById(R.id.RX);
        TX = (TextView) findViewById(R.id.TX);
        Cost = (TextView) findViewById(R.id.Cost);
        TotalBiaya = (TextView) findViewById(R.id.totalBiaya);
        TotalData = (TextView) findViewById(R.id.totalData);

        // RX //
        long rxBytes = TrafficStats.getTotalRxBytes() - mStartRX;
        float rxBytes_float = (float) rxBytes / 1000;
        String rx_result = String.format("%.02f", rxBytes_float);
        RX.setText(rx_result + " KB/s");

        // TX //
        long txBytes = TrafficStats.getTotalTxBytes() - mStartTX;
        float txBytes_float = (float) txBytes / 1000;
        String tx_result = String.format("%.02f", txBytes_float);
        TX.setText(tx_result + " KB/s");

        // Biaya //
        float cost = (txBytes_float + rxBytes_float) *
            ValueHolder.getBiaya();
        String cost_result = String.format("%.02f", cost);
        Cost.setText("Rp. " + cost_result);

        //float total_data = txBytes_float + rxBytes_float;
        //String data_result = String.format("%.02f", total_data);
        //totalData = data_result;

        totalRP = cost_result;

        mTotalData = txBytes_float + rxBytes_float;
        String stotalData = String.format("%.02f", mTotalData);
        TotalData.setText(stotalData + " KB,");

        TotalBiaya.setText("Rp. " + cost_result);
    }
}
```

Sedangkan untuk *layout* lain, meliputi *layout* tentang, berhenti, dan pengaturan.