

## DESKRIPSI PERANCANGAN PERANGKAT LUNAK

### Elisitasi Aplikasi Self Service

#### Kartu Perdana Telyu-sel

untuk:

Customer Telyu-sel

Dipersiapkan oleh:

1301194292 Gunawan Nur Ahmad

1301190352 Jehua Kusuma Dewa

1301194216 Rafly Athalla

1301190156 Dinda Atikah Wulandari

(Kelompok 3 | IF-43-01)

Program Studi Informatika - Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

	<b>Prodi S1- Informatika</b> <b>Universitas Telkom</b>	Nomor Dokumen		Halaman
		<i>DPPL-xx &lt;xx:no grp&gt;</i>		<i>&lt;#&gt;/&lt;jml #</i>
		Revisi	<i>&lt;nomor revisi&gt;</i>	<i>Tgl: &lt;isi tanggal&gt;</i>

---

**DAFTAR PERUBAHAN**

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX	-	A	B	C	D	E	F	G
-------	---	---	---	---	---	---	---	---

TGL								
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi
---------	--------	---------	--------

--	--	--	--

## **DAFTAR ISI**

<b>1. Pendahuluan</b>	<b>5</b>
Tujuan Penulisan Dokumen	6
Lingkup Masalah	6
Definisi dan Istilah	6
Referensi	6
Sistematika Pembahasan	6
<b>Deskripsi Perancangan Global</b>	<b>6</b>
Rancangan Lingkungan Implementasi	7
Deskripsi Arsitektural	7
Deskripsi Komponen	7
<b>Perancangan Rinci</b>	<b>8</b>
Realisasi Use Case	8
Use Case <nama use case 1>	8
Identifikasi Kelas	8
Sequence Diagram	8
Diagram Kelas	8
Perancangan Detail Kelas	8
Kelas <nama kelas>	8
Kelas <nama kelas>	9
Diagram Kelas Keseluruhan	9
Algoritma/Query	9
Diagram Statechart	9
Perancangan Antarmuka	9
Perancangan Representasi Persistensi Kelas	10
<b>Matriks Kerunutan</b>	<b>10</b>

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

## **1. Pendahuluan**

### **1.1 Tujuan Penulisan Dokumen**

Dokumen ini berisi Deskripsi Perancangan Perangkat Lunak (DPPL) untuk sistem Self Service Telyu-sel .Tujuan dari penulisan dokumen ini adalah untuk memberikan penjelasan mengenai perangkat lunak yang akan dibangun baik berupa gambaran umum maupun penjelasan detail dan menyeluruh.

Pengguna dari dokumen ini adalah pengembang perangkat lunak sistem Telyu-sel dan pengguna (user) dari perangkat lunak atau personil-personil yang terlibat dalam sistem. Dokumen ini akan digunakan sebagai bahan acuan dalam proses pengembangan dan sebagai bahan evaluasi pada saat proses pengembangan perangkat lunak maupun di akhir pengembangannya. Dengan adanya dokumen DPPL ini diharapkan pengembangan perangkat lunak akan lebih terarah dan lebih terfokus serta tidak menimbulkan ambiguitas terutama bagi pengembang perangkat lunak sistem Self Service Telyu-sel.

### **1.2 Lingkup Masalah**

Perangkat lunak yang akan dikembangkan adalah perangkat lunak Self Service Telyu-sel, yaitu merupakan perangkat lunak yang digunakan untuk user pengguna kartu perdana telyu-sel dimana dapat memberikan informasi mengenai paket-paket internet, pulsa, sms, telpon, dll dan masa tenggang kartu maupun masa paket berakhir. Telyu-sel dapat melakukan hal-hal berikut ini : Dengan adanya Telyu-sel ini diharapkan, pengguna/user dapat dengan mudah mendapatkan akses untuk membeli paket internet dll tanpa harus keluar dari rumah, serta dapat melihat informasi kapan berakhirnya paket yang digunakan.

### **1.3 Definisi dan Istilah**

No	Definisi	Keterangan
----	----------	------------

1.	DPPL	Dokumen untuk mendokumentasikan deskripsi perancangan perangkat lunak.
2.	DBMS	Suatu sistem yang dirancang untuk mengelola suatu basis data dan menjalankan operasi terhadap data.
3.	VSCODE	Code editor yang bisa dijalankan di perangkat desktop berbasis Windows, Linux, dan MacOS.

#### 1.4 Referensi

- *Software Engineering, Aparctitioner's Approach 5th edition, Roger S Pressman, McGraw Hill, 2001*
- *IEEE Std 830-1993, IEEE Recommended Parctice for Software Requirement Specifications.*
- *Arry Ekananta, Spesifikasi kebutuhan perangkat lunak (SKPL). Jurusan ilmu computer institute pertanian bogor*

#### 1.5 Sistematika Pembahasan

Bab 1. Berbicara tentang kenapa Aplikasi ini dibuat

Bab 2. Berbicara tentang deskripsi dari Perancangan Aplikasi secara global

Bab 3. Berbicara tentang deskripsi dari Perancangan Aplikasi secara terperinci

Bab 4. Berbicara tentang pemetaan dari Use case dan kelas yang terkait

## 2 Deskripsi Perancangan Global

## 2.1 Rancangan Lingkungan Implementasi

Operating system : Android/IOS

DBMS : MySQL

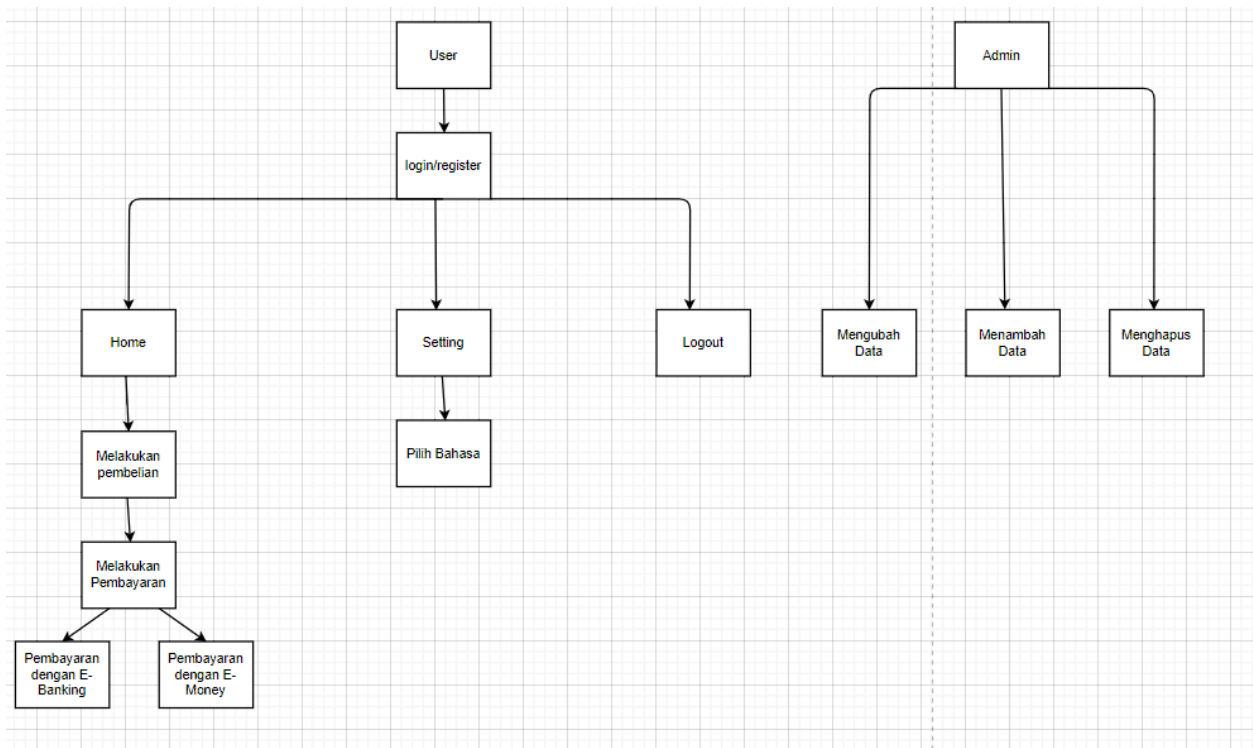
Development tools : VSCode

Bahasa pemrograman : Dart

Framework : Flutter

## 2.2 Deskripsi Arsitektural

*Berikan penjelasan singkat tentang arsitektur /L yang akan dibangun. Gambarkan dalam bentuk diagram komponen.*



Aplikasi yang kita buat memakai model Arsitektur Client / Server dimana user nanti akan mendownload aplikasi dan dapat mengakses segala fitur yang ada di aplikasi dengan persetujuan dari server / terkoneksi dengan server.

## 2.3 Deskripsi Komponen



Diisi dengan daftar modul. Daftar modul bisa dalam bentuk tabel berikut:

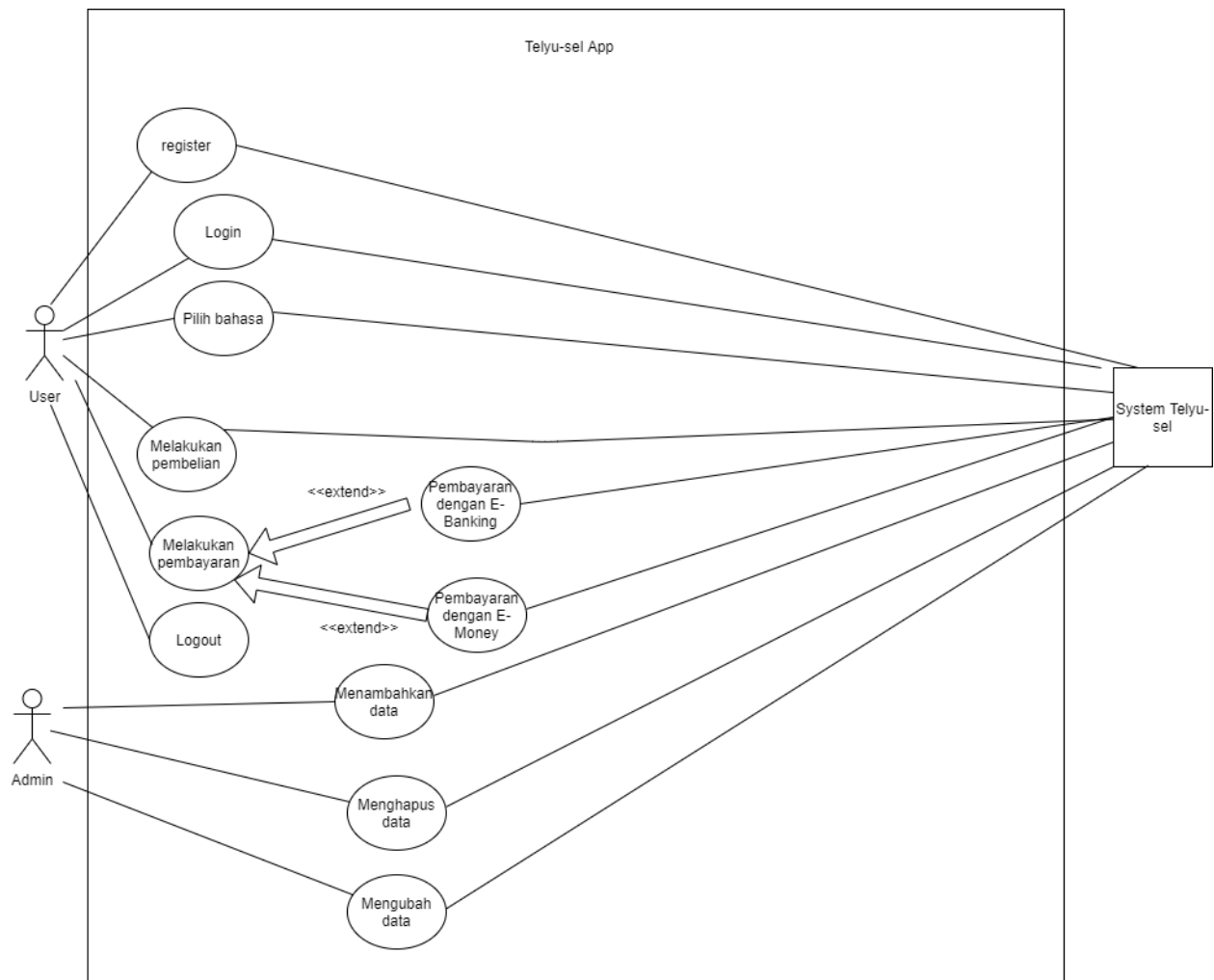
[illegible]

### 3 Perancangan Rinci

#### 3.1 Realisasi Use Case

##### 3.1.1 Use Case Diagram

Jika use case ini akan direalisasikan dalam bentuk aplikasi berbasis web, maka subbab yang terkait dengan perancangan elemen aplikasi berbasis web harus diisi.



### 3.1.1.1 Identifikasi Kelas

*Identifikasi kelas yang terkait dengan use case tersebut. Kelas di tahap perancangan dapat berbeda dengan dengan kelas di tahap analisis. Dapat menggunakan tabel di bawah:*

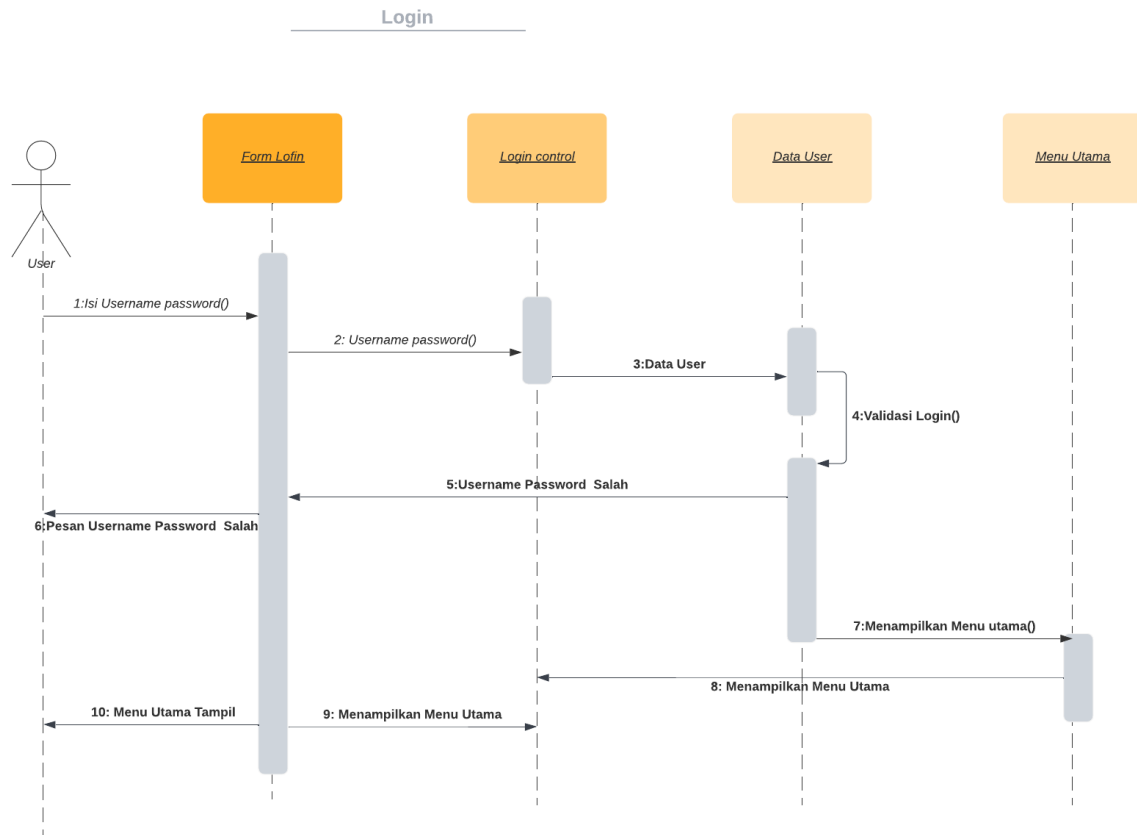
<i>No</i>	<i>Nama Kelas Perancangan</i>	<i>Tipe Kelas</i>
<i>1</i>	<i>Form login</i>	<i>interface</i>
<i>2</i>	<i>Login control</i>	<i>controller</i>
<i>3</i>	<i>Data User</i>	<i>Database</i>

*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

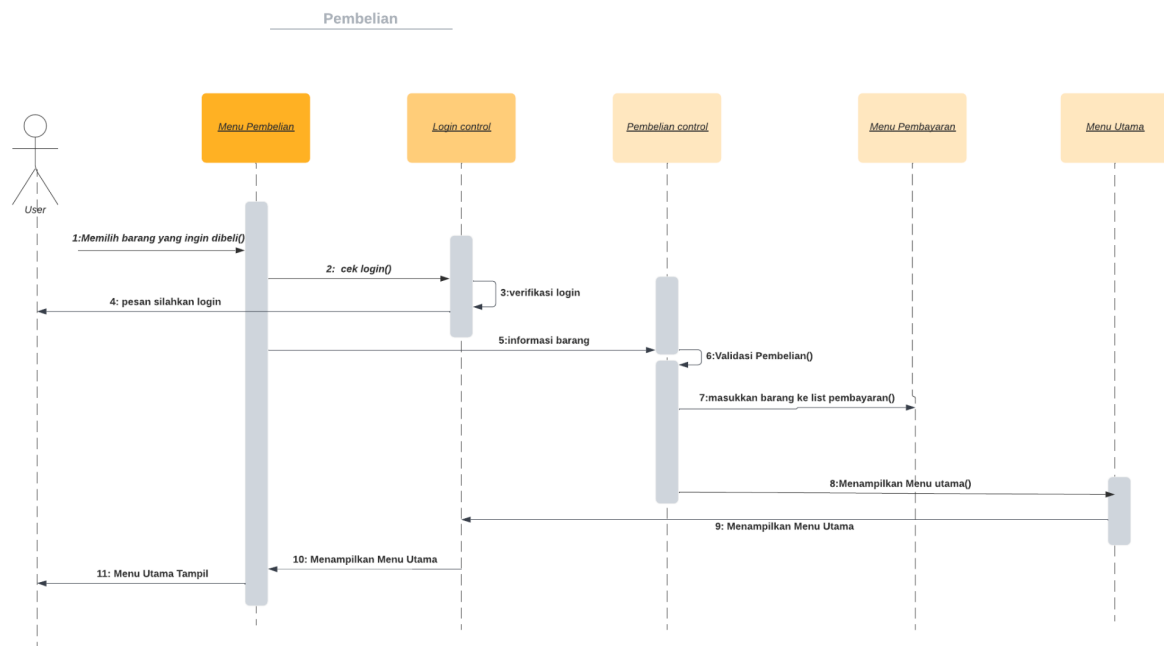
### 3.1.1.2 Sequence Diagram

*Buatlah diagram sequence untuk setiap skenario use case. Skenario melibatkan kelas-kelas yang sudah diidentifikasi.*

*Use Case 1 : Login*



*Use Case 2 : Pembelian*



Use Case 3 :

### 3.1.1.3 Diagram Kelas

Buatlah diagram kelas untuk use case tersebut. buat class diagram **BUKAN KESELURUHAN**, tapi **PER USE CASE**

## 3.1.2 Use Case 1

Nama Use Case:

Login

**Tujuan:**

Untuk bisa masuk aplikasi dan mengakses semua fitur aplikasi

**Input:**

Email,password dan nomor yang terdaftar

**Output:**

User dapat masuk kedalam aplikasi dan menggunakan segala fitur yang ada dalam aplikasi

**Skenario Utama:**

Use case ini berjalan Ketika user membuka aplikasi dan ingin melakukan login

**Prakondisi:**

Sistem harus dalam keadaan siap menerima perintah dan aplikasi harus versi terbaru

**Pascakondisi:**

Berhasil masuk aplikasi dan dapat memilih bahasa

**Skenario Normal:**

Actor	Reaksi sistem
-------	---------------

User/Costumer yang terdaftar akun-nya, memasukkan username dan password	Sistem memverifikasi akun
---	---------------------------

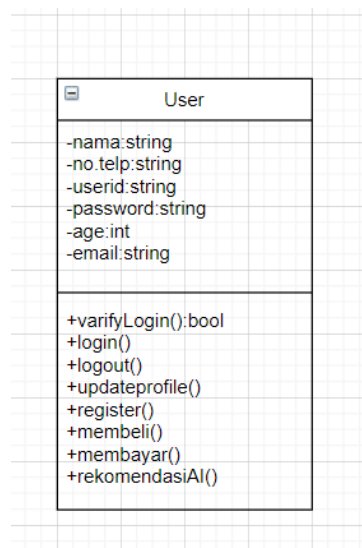
### Skenario Alternative:

Verifikasi akun gagal, tampilkan pesan, username atau password yang anda masukkan tidak benar

### Skenario Eksepsional:

Koneksi putus saat alternative flow, relaunch aplikasi ketika koneksi sudah tersambung

## 3.1.2.1 Diagram Kelas



## 3.1.3 Use Case 2

**Nama Use Case:**

Pembelian

**Tujuan:**

Untuk membeli paket data internet, telepon, atau SMS yang disediakan oleh sistem

**Input:**

Produk yang ingin dibeli

**Output:**

Produk yang akan dibeli masuk ke dalam list pembayaran

**Skenario Utama:**

Use case dimulai ketika user memilih salah satu dari produk yang ditawarkan oleh sistem. Lalu sistem memeriksa validitas user dari proses login yang sudah dilakukan, jika user adalah user yang valid maka user akan diminta melakukan verifikasi pada pembeliannya, jika user tidak menyetujui pembelian maka user akan dibawa kembali ke halaman utama, namun jika user menyetujui pembelian maka user akan dibawa ke halaman pembayaran, Jika user adalah user yang tidak valid maka akan dikembalikan ke halaman login.

**Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk menerima pembelian



**Pascakondisi:**

User dibawa pada halaman pembayaran untuk melakukan pembayaran

**Skenario Normal:**

Actor	Reaksi sistem
User mengklik tombol beli pada salah satu produk yang ditawarkan oleh sistem	Sistem mengecek validasi user
User dibawa pada halaman verifikasi pembelian	Sistem memeriksa apakah user menyetujui pembelian atau tidak
	Sistem membawa user ke halaman pembayaran

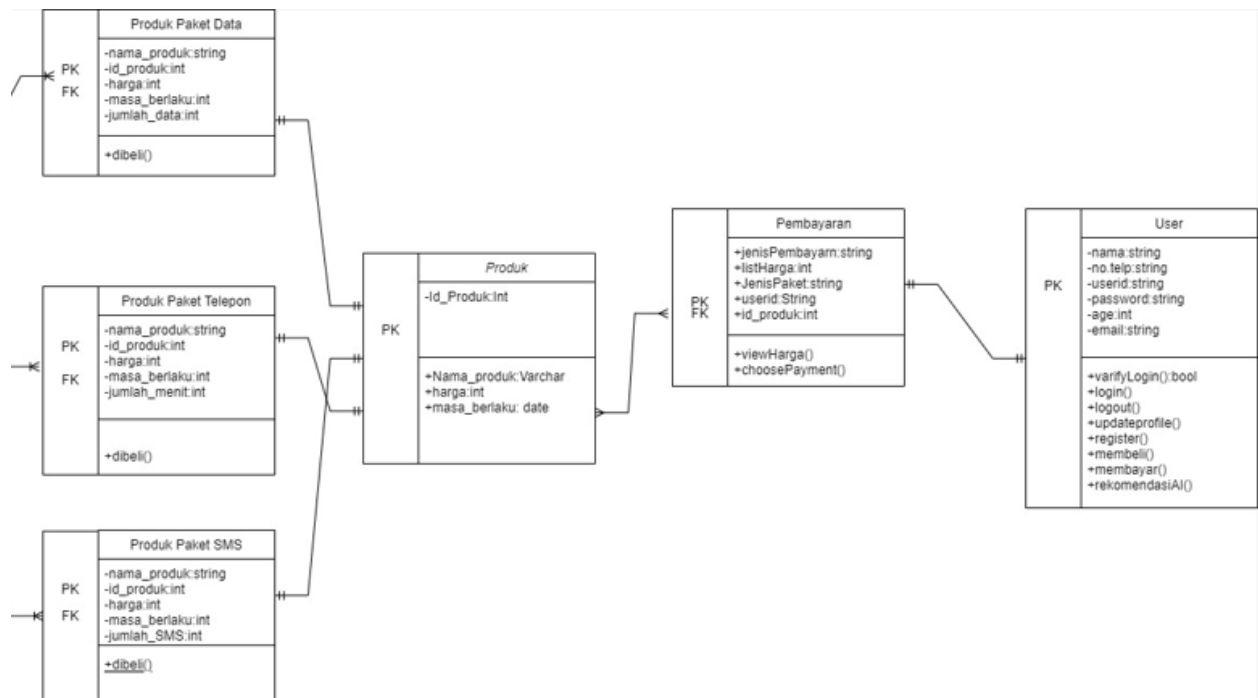
**Skenario alternative:**

- Validasi user gagal maka gagalkan pembelian dan kembali ke halaman login
- User tidak menyetujui verifikasi maka gagalkan pembelian dan kembali ke halaman utama

**Skenario eksepsional:**

Terjadi kesalahan sistem pada proses pembelian maka batalkan pembelian

**3.1.3.2 Diagram Kelas**



### 3.1.4 Use Case 3

#### Nama Use Case:

Pembayaran

#### Tujuan:

Untuk membayar paket data internet, telepon, atau SMS yang disediakan system

#### Input:

Mengklik tombol pembayaran yang sudah disiapkan oleh sistem

#### Output:

Fitur akan mengeluarkan pesan pembayaran berhasil, atau kembali ke menu utama bila pembayaran tidak disetujui oleh user.

### **Skenario Utama:**

Use case dimulai ketika user mengklik tombol pembayaran yang disediakan oleh sistem. Lalu sistem memeriksa verifikasi pembayaran, jika user menyetujui pembayarannya, maka akan tampil pesan pembayaran berhasil. apabila user tidak menyetujui pembayaran, maka user akan dibawa ke halaman utama.

### **Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk menerima pembayaran

### **Pascakondisi:**

Sistem harus sedang dalam keadaan siap untuk menerima pembayaran

### **Skenario alternative**

- User tidak menyetujui verifikasi maka gagalkan pembayaran dan kembali ke halaman utama

### **Skenario Normal Flow:**

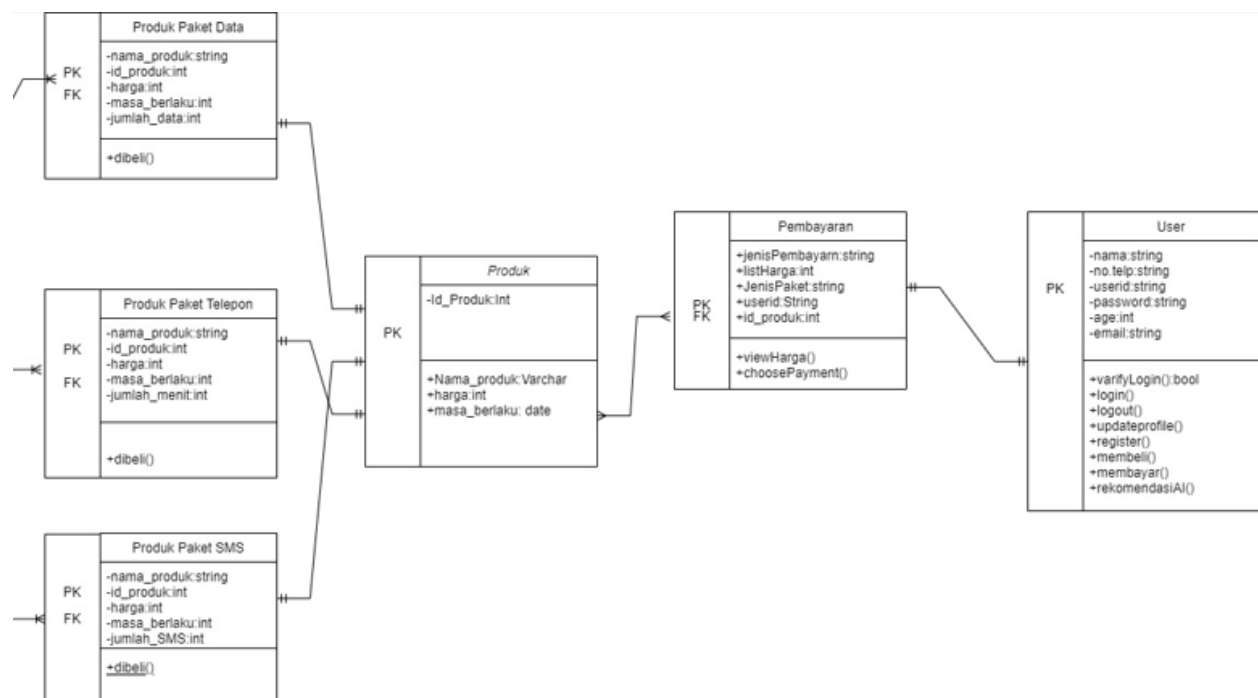
Actor	Reaksi sistem
-------	---------------

User memilih metode pembayaran yang disediakan oleh sistem	User dibawa pada halaman verifikasi pembayaran
	Sistem memeriksa apakah user menyetujui pembayaran atau tidak
	Sistem membawa user ke halaman verifikasi untuk menunggu

### Skenario eksepsional:

Terjadi dibawa pada halaman verifikasi untuk menunggu pembayaran berhasil

### 3.1.4.3 Diagram Kelas



### **3.1.5 Use Case 4**

**Nama Use Case:**

Logout

**Tujuan:**

Keluar akun pengguna

**Input:**

Fitur keluar akun pengguna

**Output:**

Memilih keluar akun dengan syarat sudah login akun

**Skenario Utama:**

Use case dimulai ketika user memilih logout akun yang disediakan oleh sistem. Sistem akan memvalidasi pilihan user apabila user sudah login akun. Setelah di validasi oleh sistem, akun pengguna akan keluar tanpa menghapus akun

**Prakondisi:**

Sistem harus sedang dalam keadaan siap atau dalam keadaan sudah login akun

**Pascakondisi:**

User dibawa pada halaman verifikasi untuk logout akun

**Skenario Normal Flow:**

Actor	Reaksi sistem
User memilih logout yang disediakan oleh sistem	Sistem mengecek validasi user
User dibawa pada halaman verifikasi logout akun pengguna	Sistem memeriksa apakah user menyetujui logout saat user menyetujui logout akun
	Sistem membawa user ke halaman login saat user menyetujui logout akun

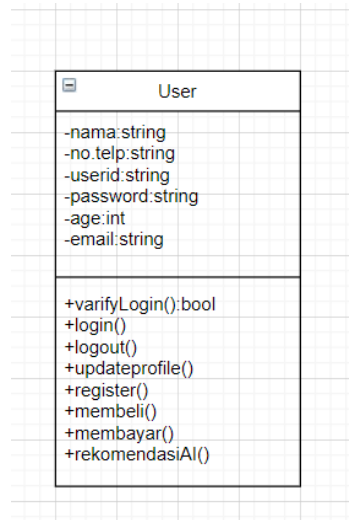
**Skenario alternative:**

Terjadi kesalahan sistem pada proses logout

**Skenario eksepsional:**

Terjadi kesalahan sistem pada proses logout maka pembatalan logout akun

**3.1.5.4 Diagram Kelas**



### 3.1.6 Use Case 5

#### Nama Use Case:

Pembayaran dengan E-Money

#### Tujuan:

Salah satu fitur untuk membayar paket kuota,sms,telepon dan data roaming

#### Input:

jenis e-money yang akan digunakan

#### Output:

e-money yang sudah dipilih akan digunakan untuk membayar

**Skenario Utama:**

Use case dimulai ketika user memilih salah satu fitur e-money yang di sediakan oleh sistem. Lalu sistem memeriksa validitas user dari proses e-money yang sudah dilakukan, jika user memiliki saldo yang cukup , maka pembayaran akan dapat berhasil, dan apabila saldo user tidak cukup, maka pembayaran tidak dapat berhasil atau gagal.

**Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk menerima pembayaran

**Pascakondisi:**

User dibawa pada halaman verifikasi untuk menunggu pembayaran berhasil

**1.1.8. Skenario Normal Flow:**

<b>Actor</b>	<b>Reaksi sistem</b>
User memilih fitur e-money yang di sediakan oleh system	Sistem mengecek validasi user
User dibawa pada halaman verifikasi pembayaran	Sistem memeriksa apakah user memiliki saldo cukup atau tidak



	Sistem membawa user ke halaman verifikasi untuk menunggu pembayaran berhasil
--	--

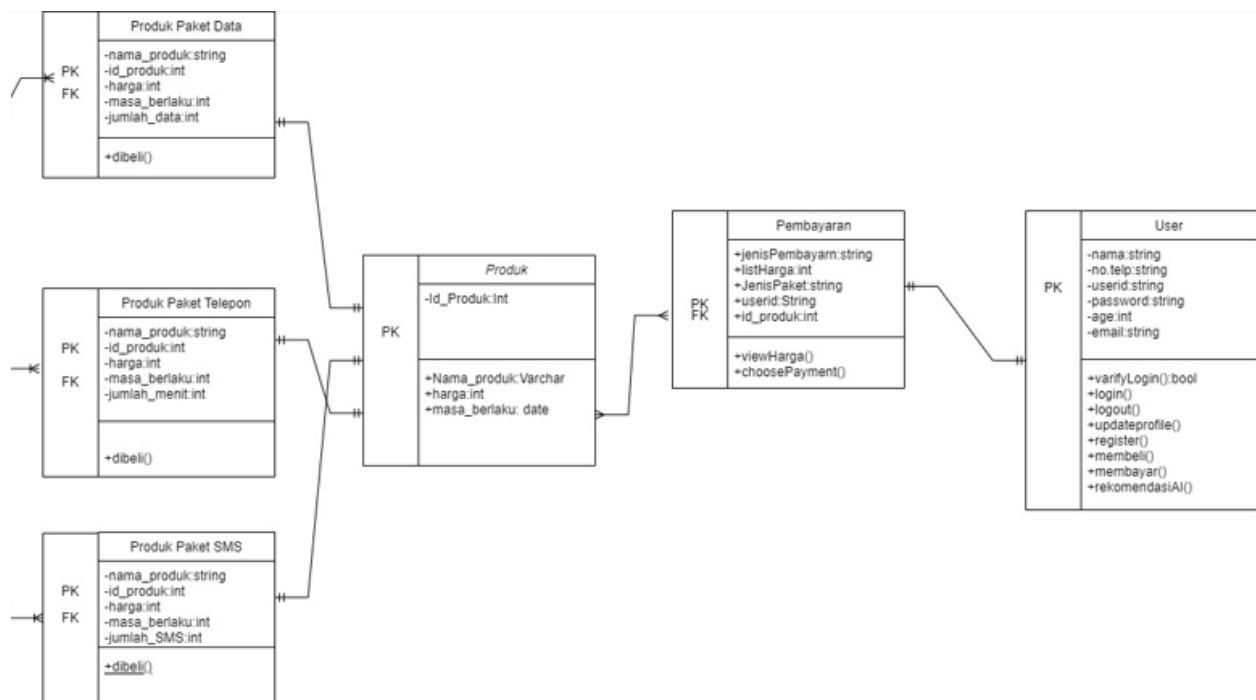
### Skenario alternative:

- Validasi user gagal maka gagalkan pembayaran dan kembali ke halaman login
- User tidak menyetujui verifikasi maka gagalkan pembayaran dan kembali ke halaman utama

### Skenario eksepsional:

Terjadi kesalahan sistem pada proses pembayaran maka batalkan pembayaran

## 3.1.6.5 Diagram Kelas



### **3.1.7 Use Case 6**

**Nama Use Case:**

Pembayaran dengan E-Banking

**Tujuan:**

Salah satu fitur untuk membayar paket kuota, sms, telepon, dan data roaming

**Input:**

Memilih E-banking apa yang akan kita pilih

**Output:**

Fitur akan mengeluarkan kode virtual account untuk dapat proses pembayaran

**Skenario Utama:**

Use case dimulai ketika user memilih salah satu fitur e-banking yang di sediakan oleh sistem. Lalu sistem memeriksa validitas user dari proses e-banking yang sudah dilakukan, jika user salah menginput kode virtual account maka pembayaran akan gagal, dan jika user menginput kode virtual account yang benar maka pembayaran berhasil

**Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk menerima pembayaran

**Pascakondisi:**

User dibawa pada halaman verifikasi untuk menunggu pembayaran berhasil

**Skenario Normal Flow:**

<b>Actor</b>	<b>Reaksi sistem</b>
User memilih fitur e-banking yang disediakan oleh system	Sistem mengecek validasi user
User dibawa pada halaman verifikasi pembayaran	Sistem memeriksa apakah user menginput kode virtual account benar atau salah
	Sistem membawa user ke halaman verifikasi untuk menunggu pembayaran berhasil

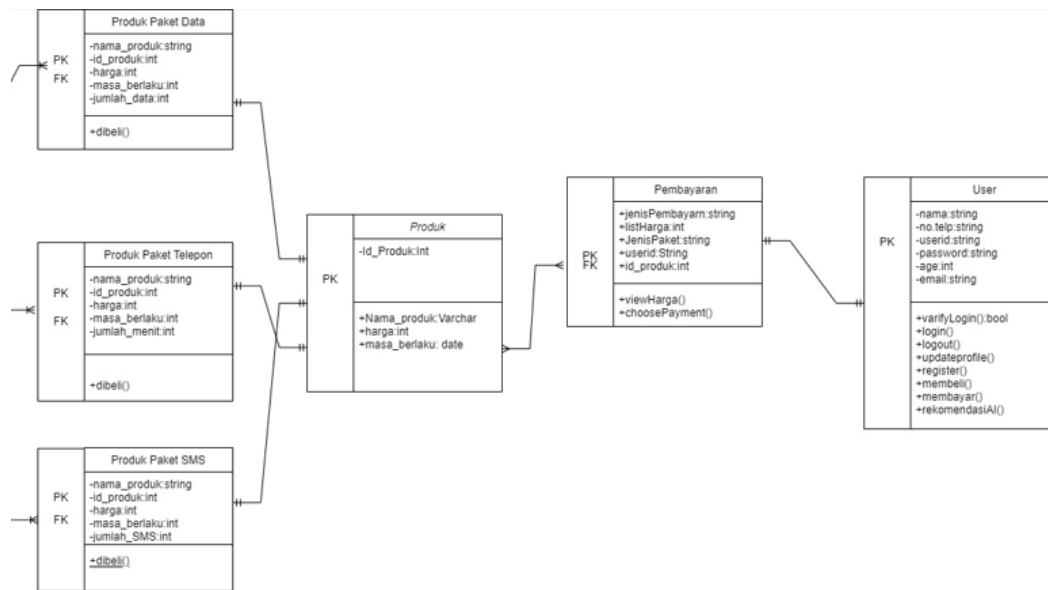
**Skenario alternative:**

- Validasi user gagal maka gagalkan pembayaran dan kembali ke halaman login
- User tidak menyetujui verifikasi maka gagalkan pembayaran dan kembali ke halaman utama

**Skenario eksepsional**

Terjadi kesalahan sistem pada proses pembayaran maka batalkan pembayaran

### **3.1.7.6 Diagram Kelas**



### 3.1.8 Use Case 7

#### Nama Use Case:

Menambah data

#### Tujuan:

Untuk menambah paket kuota, paket telepon, dan paket sms

#### Input:

data yang ingin ditambahkan ke aplikasi

#### Output:

data yang ditambahkan masuk kedalam aplikasi.

### **Skenario Utama:**

Use case dimulai ketika admin melakukan penambahan data seperti paket kuota, paket sms, paket telepon kepada list menu produk

### **Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk ditambahkan datanya

### **Pascakondisi:**

Admin akan menambahkan data apa saja yang diperlukan oleh customer

### **Skenario Normal Flow:**

<b>Actor</b>	<b>Reaksi sistem</b>
Admin menambahkan beberapa data	Sistem akan memunculkan data yang telah di tambahkan oleh admin

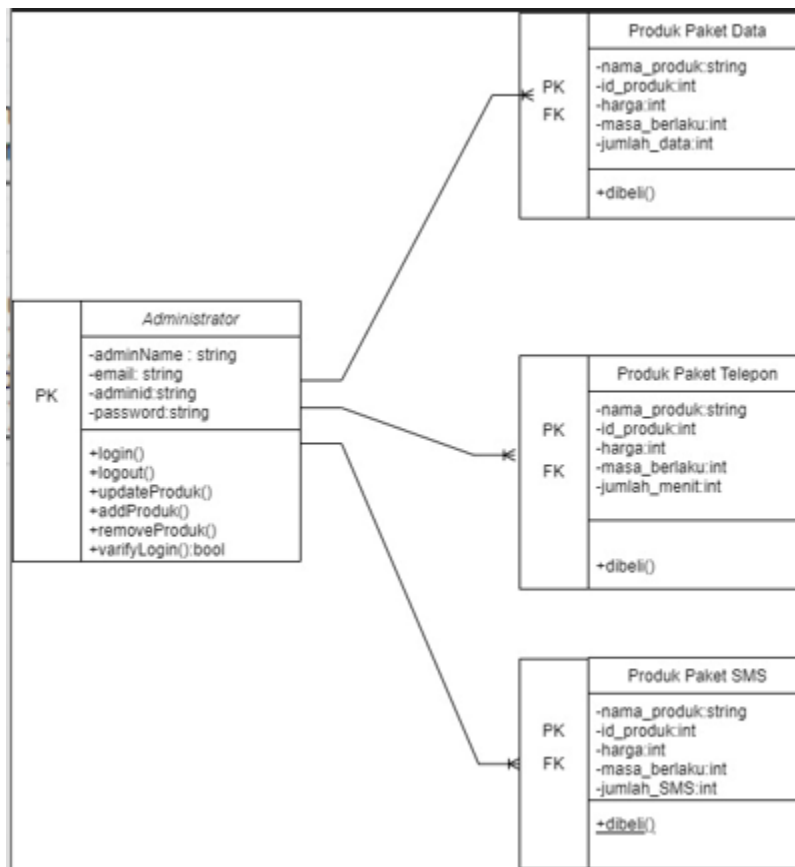
### **Skenario alternative:**

Validasi admin gagal, maka data yang telah ditambahkan akan tidak muncul di layar para customer.

### Skenario eksepsional:

Sistem akan tidak bisa dipakai pada saat admin menambahkan data

### 3.1.8.7 Diagram Kelas



### 3.1.9 Use Case 8

#### Nama Use Case:

Mengubah data

#### Tujuan:

Mengubah data produk

**Input:**

memasukkan perintah data mana yang ingin di ubah dan aspek apa yang akan di ubah dari data tersebut.

**Output:**

Data yang telah berhasil diubah, menggantikan data yang sebelumnya.

**Skenario Utama:**

Use case dimulai ketika admin memilih salah satu produk yang akan diubah. Lalu sistem akan mengecek validitas data admin, jika admin adalah admin yang valid maka admin akan dibawa pada halaman edit produk, jika admin tidak valid maka akan dibawa ke halaman login. Pada halaman edit, admin akan mengisi form mengenai data yang akan diubah. Ketika admin sudah selesai mengisi form, lalu admin perlu untuk meng klik tombol update untuk untuk konfirmasi update produk. Setelah itu sistem akan mengecek data input yang diisi oleh admin, jika data valid maka data produk akan terupdate jika data tidak valid maka admin perlu mengisi ulang di halaman edit.

**Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk menerima perintah update

**Pascakondisi:**

User dibawa ke halaman beranda setelah update produk selesai

**Skenario Normal Flow:**

Actor	Reaksi sistem
Admin meng klik tombol edit pada salah satu produk	Sistem mengecek validasi admin
Admin dibawa pada halaman edit	Sistem melakukan validasi data yang di input admin
Admin mengisi form pada halaman edit	

**Skenario alternative:**

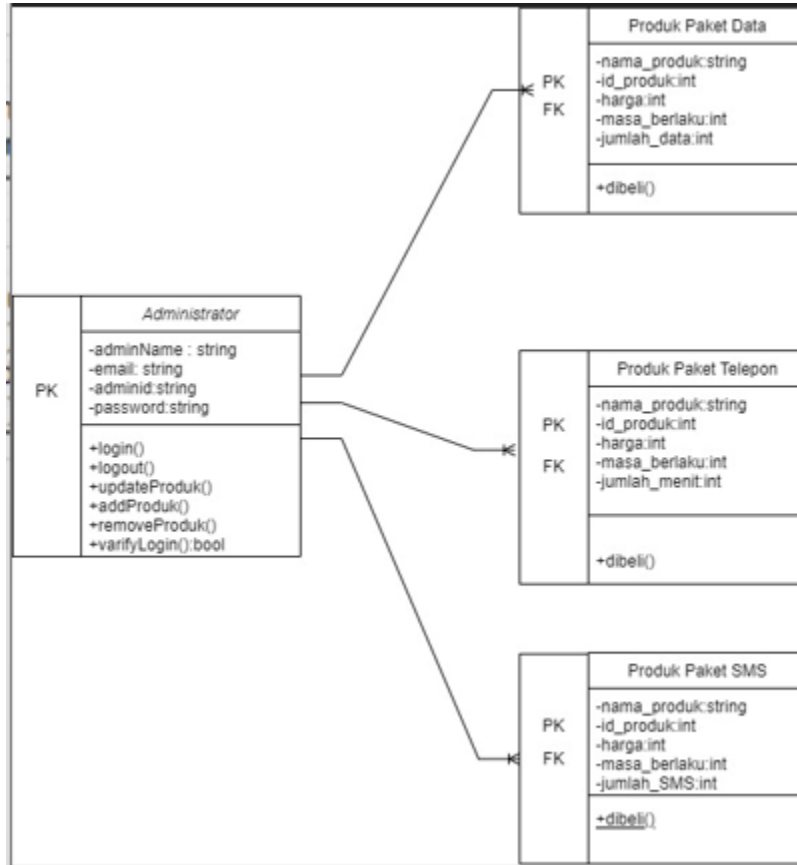
- Validasi admin gagal maka gagalkan update dan kembali ke halaman login
- Validasi data input pada halaman edit gagal maka admin akan dibawa kembali ke halaman edit

**Skenario eksepsional:**

Terjadi kesalahan sistem pada proses update produk maka batalkan update produk



### 3.1.9.8 Diagram Kelas



### 3.1.10 Use Case 9

**Nama Use Case:**

Menghapus data

**Tujuan:**

Untuk menghapus paket kuota, paket telepon, paket sms

**Input:**

Memasukkan perintah data mana yang akan di hapus

**Output:**

Data yang telah di hapus, tidak akan ditampilkan oleh aplikasi lagi dan tidak dapat di akses user.

**Skenario Utama:**

Use case dimulai ketika admin ingin melakukan penghapusan seperti paket kuota, paket sms, paket telepon kepada list menu produk .

**Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk ditambahkan datanya

**Pascakondisi**

produk yang dihapus oleh admin tidak akan tampil lagi di aplikasi

**Skenario Normal Flow**

Normal Flow events		
	Actor Actions	Sistem Action
	Admin menghapus suatu produk	Sistem akan menghapuskan objek tersebut dari aplikasi

**Skenario alternative**

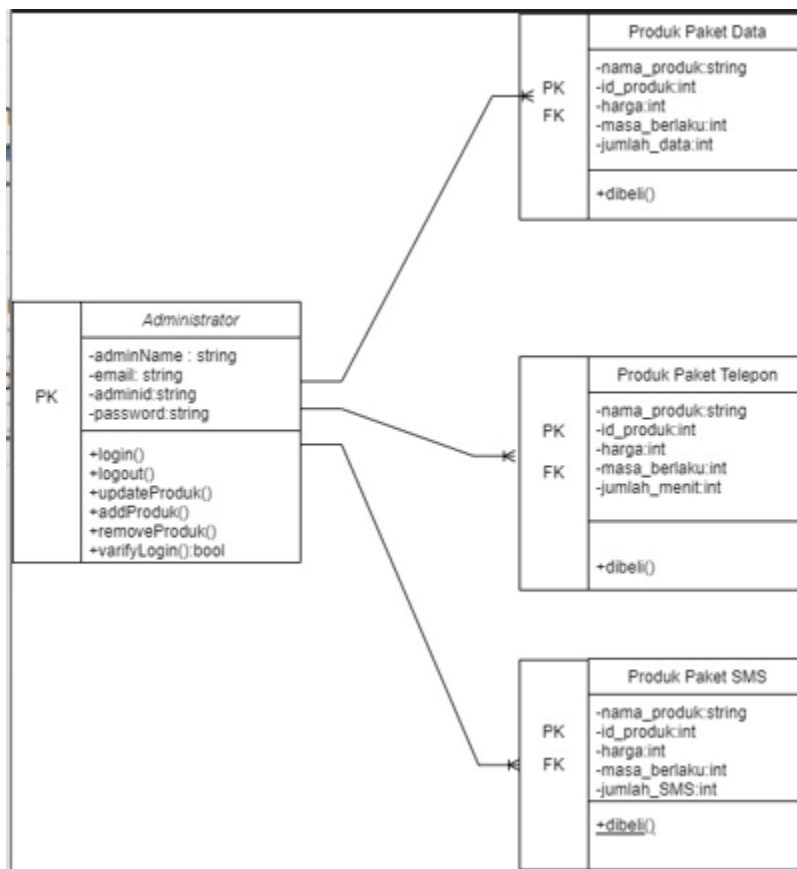
sistem tidak dapat menemukan produk yang akan di hapus, menampilkan pesan produk yang akan dihapus tidak ada.

### Skenario eksepsional

Koneksi putus saat akan menghapus produk,

maka produk yang akan dihapus gagal, admin harus mengulangi menghapus ketika sudah tersambung internet.

#### 3.1.10.9 Diagram Kelas



#### 3.1.11 Use Case 10

**Nama Use Case:**

Pilih bahasa

**Tujuan:**

Untuk memilih Bahasa yang akan dipakai dalam mengakses aplikasi

**Input:**

Memasukan pilihan Bahasa yang akan dipilih.

**Output:**

Aplikasi akan berjalan dengan Bahasa yang dipilih.

**Skenario Utama:**

Use case ini berjalan Ketika user melakukan login pertama kali dalam aplikasi, kemudian user memilih Bahasa dengan fitur pilih Bahasa yang akan mempengaruhi Bahasa apa yang akan digunakan dalam aplikasi tersebut.

**Prakondisi:**

- Sistem harus dalam keadaan siap menerima perintah

- Aplikasi harus versi terbaru
- Sudah melakukan login

**Pascakondisi:**

Memilih Bahasa yang digunakan: inggris atau Indonesia.

**Skenario Normal Flow:**

Actor	Reaksi sistem
User/customer Memilih Bahasa Indonesia atau inggris	Sistem akan memverifikasi Bahasa yang dipilih dan mensinkronkannya kedalam segala fitur dan operasional aplikasi

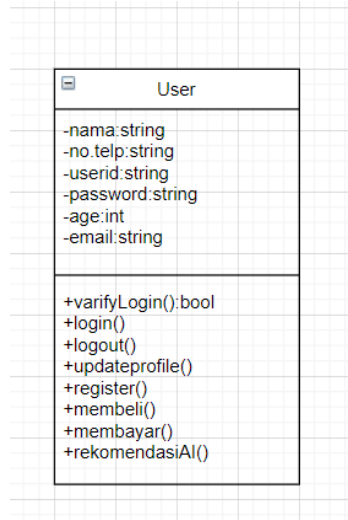
**Skenario alternative:**

Terjadi kesalahan sistem pada proses pengubahan bahasa

**Skenario eksepsional:**

Terjadi kesalahan sistem pada proses mengubah bahasa maka batalkan pengubahan bahasa

### 3.1.11.10 Diagram Kelas



### 3.1.12 Use Case 11

#### Nama Use Case:

register

#### Tujuan:

Email, password, dan nomer yang belum terdaftar

#### Input:

Memasukkan perintah data mana yang akan di hapus

#### Output:

Email, password, dan nomer terdaftar dalam aplikasi

#### Skenario Utama:

Use case dimulai ketika user mengklik tombol register dalam halaman login, lalu diminta nama pengguna, password, email dan nomor handphone dengan provider telyu-sel. Setelah itu user menunggu verifikasi apakah akun yang dibuatnya berhasil

apa tidak, apabila berhasil, user akan dibawa ke menu beranda dan apabila gagal user akan dibawa ke menu login Kembali.

### **Prakondisi:**

Sistem harus sedang dalam keadaan siap untuk menerima registrasi akun baru

### **Pascakondisi**

User dibawa pada halaman verifikasi untuk menunggu apakah akunnya berhasil dibuat atau tidak

### **Skenario Normal Flow**

Normal Flow events		
	Actor Actions	Sistem Action
	User mengklik tombol registrasi	Sistem mengecek validasi user
	User mengisi nama pengguna,password,no mor handphone,email	Sistem memeriksa apakah user mengisi registrasi dengan lengkap apa tidak
	User dibawa pada halaman verifikasi untuk menunggu hasil registrasi	Sistem membawa user ke halaman verifikasi untuk menunggu registrasi berhasil

--	--

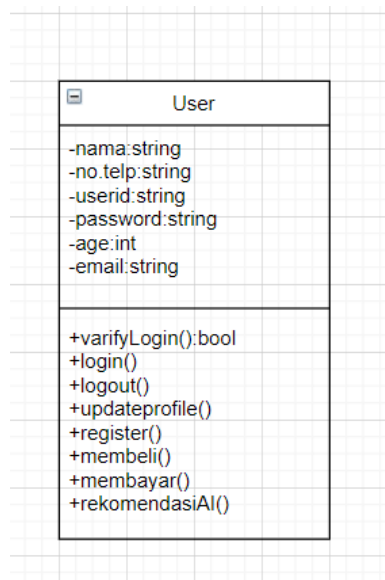
### Skenario alternative

- Validasi user gagal maka gagalkan registrasi dan kembali ke halaman login
- User tidak menyetujui verifikasi maka gagalkan registrasi dan kembali ke halaman utama

### Skenario eksepsional

Terjadi kesalahan sistem pada proses registrasi maka batalkan registrasi

## 3.1.12.11 Diagram Kelas



## 3.2 Perancangan Detail Kelas

Bagian ini diisi dengan daftar seluruh kelas dalam tabel berikut:



<i>No</i>	<i>Nama Kelas Perancangan</i>	<i>Nama Kelas Analisis Terkait</i>

*Untuk setiap kelas:*

- *identifikasi operasi (mengacu pada tanggung-jawab kelas), termasuk visibility-nya*
- *identifikasi atribut, termasuk visibility-nya*

### **3.2.1 Kelas <nama kelas>**

Bagian ini diisi dengan daftar operasi dan atribut Buat untuk setiap kelas.

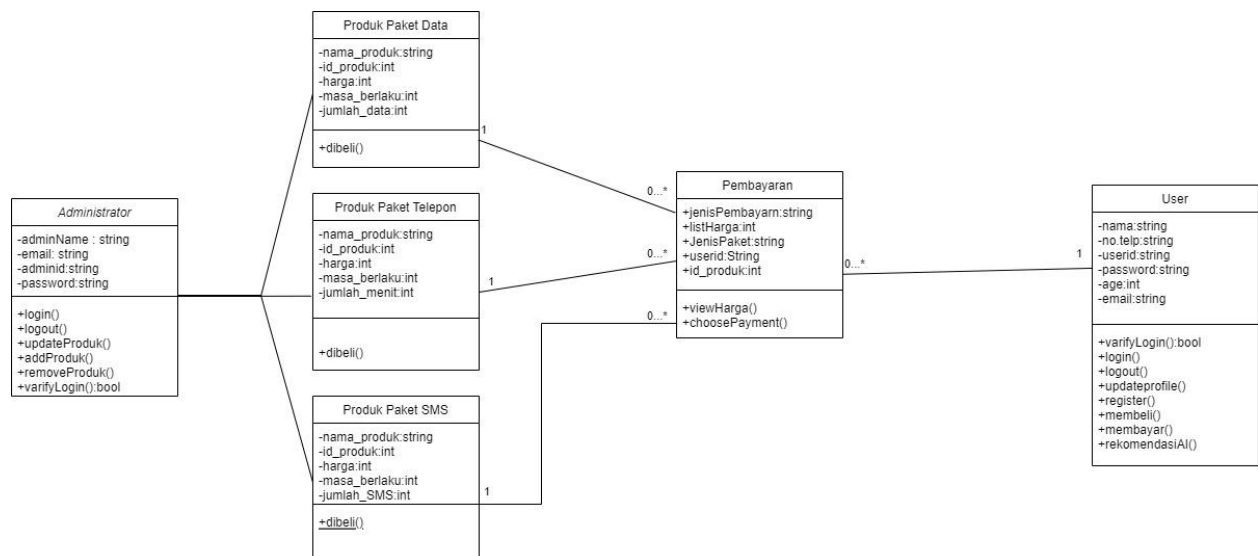
*Nama Kelas : .....*

<i><b>Nama Operasi</b></i>	<i><b>Visibility</b></i> <i><b>(private, public)</b></i>	<i><b>Keterangan</b></i>
<i>Diisi dengan signature operasi</i>		

<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
<i>Diisi dengan nama atribut</i>		<i>Tuliskan tipenya sesuai dengan yang dikenal pada bahasa pemrograman yang digunakan</i>

### 3.2.2 Kelas <nama kelas>

### 3.3 Diagram Kelas Keseluruhan



### 3.4 Algoritma/Query

Bagian ini hanya diisi untuk kerangka algoritma untuk **method-method dari Class** yang dianggap cukup penting. Implementasi skeleton code juga sudah dapat dilakukan untuk kelas-kelas yang terdefinisi pada bahasa pemrograman tertentu. Boleh dibuat subbab per kelas.

Contoh:

Nama Kelas :

Nama Operasi :

Algoritma : (Algo-xxx)

{Jika mengacu query tertentu, lengkapi tabel query di bawah}

Query :

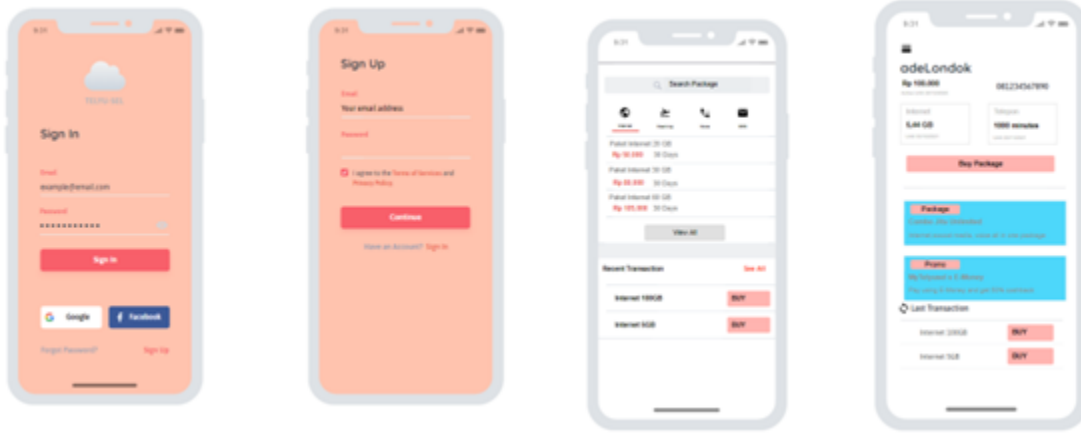
No Query	Query	Keterangan
Q-xxx		Tuliskan fungsi dari querynya

### 3.5 Perancangan Antarmuka

Bagian ini diisi dengan versi awal prototipe antarmuka.

Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya, misalnya seperti di bawah ini

Antarmuka : {diisi dengan no. layar atau no gambar rancangan antarmuka}



Id_Objek	Jenis	Nama	Keterangan
		Diisi dengan string yang tampil dilayar	Diisi dengan penjelasan reaksi sistem, misalnya membuka layar apa, link kemana. Jika menyangkut suatu kode yang cukup rumit, acu algoritma yang telah diuraikan di atas.
Button1	Button	OK	Jika di klik, akan mengaktifkan Proses AlgoXXX.
RTF1	RTF Box		Isi Teks yang disimpan pada File xxx

*Jika objek dikaitkan ke File lain (misalnya file gambar, file teks), berikan nama file terkait dan deskripsi ringkas dalam kolom keterangan*

### **3.6 Perancangan Representasi Persistensi Kelas**

*Bagian ini diisi dengan rancangan skema basis data dan traceability-nya terhadap kelas entity.  
(PEMBUATAN SKEMA RELASI)*

## **4 Matriks Kerumutan**

*Mapping use case dengan kelas-kelas terkait*

<b>Requirement</b>	<b>Usecase Terkait</b>	<b>Kelas</b>
FR-01		
FR-02		

---

Pindahkan ke lampiran