dd

**DPPL-xx**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

SMART PARENT

untuk:

Orang Tua

Dipersiapkan oleh:

HADI SABILILHAQ 1301174674

FAKHRAN FADHLUR JAMALUDIN 1301172729

MUHAMMAD ROIHAN FADHLULLAH 1301174649

HOWARD JEREMY 1301174638

Program Studi Informatika

Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Prodi S1- Informatika**  **Universitas Telkom** | Nomor Dokumen | | Halaman |
| *DPPL-xx <xx:no grp>* | |  |
| Revisi | *<nomor revisi>* | *Tgl: 21-04-2019* |

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A |  |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX  TGL | - | A | B | C | D | E | F | G |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
|  |  |  |  |

Daftar Isi

[**1. Pendahuluan**](#_gjdgxs) **5**

[Tujuan Penulisan Dokumen](#_30j0zll) 6

[Lingkup Masalah](#_1fob9te) 6

[Definisi dan Istilah](#_3znysh7) 6

[Referensi](#_tyjcwt) 6

[Sistematika Pembahasan](#_3dy6vkm) 6

[**Deskripsi Perancangan Global**](#_1t3h5sf) **6**

[Rancangan Lingkungan Implementasi](#_4d34og8) 7

[Deskripsi Arsitektural](#_2s8eyo1) 7

[Deskripsi Komponen](#_17dp8vu) 7

[**Perancangan Rinci**](#_26in1rg) **8**

[Realisasi Use Case](#_lnxbz9) 8

[Use Case <nama use case 1>](#_35nkun2) 8

[Identifikasi Kelas](#_1ksv4uv) 8

[Sequence Diagram](#_44sinio) 8

[Diagram Kelas](#_2jxsxqh) 8

[Perancangan Detil Kelas](#_1y810tw) 8

[Kelas <nama kelas>](#_4i7ojhp) 8

[Kelas <nama kelas>](#_2xcytpi) 9

[Diagram Kelas Keseluruhan](#_1ci93xb) 9

[Algoritma/Query](#_3whwml4) 9

[Diagram Statechart](#_2bn6wsx) 9

[Perancangan Antarmuka](#_qsh70q) 9

[Perancangan Representasi Persistensi Kelas](#_3as4poj) 10

[**Matriks Kerunutan**](#_1pxezwc) **10**

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Tujuan dibuatnya dokemen ini adalah untuk mendokumentasikan segala aktivitas yang dilakukan selama yang dilakukan selama pengembangan dan pembuatan proyek perangkat lunak aplikasi smart parent.Selain itu juga penullisan dokumen ini digunakan sebagai salah satu acuan nantinya dalam pengimplementasian nantinya.Adapun aplikasi smart parent ini ditujukan untuk para orang tua yang mempunyai anak dibawah umur 10 tahun agar orang tua selalu mengetahui dan waspada akan kebaradaan sang anak.

## Lingkup Masalah

Aplikasi smart parent ini merupakan sebuah aplikasi yang dapat memberikan kemudahan bagi para orang tua terkhusus bagi orang tua yang terkadang kurang waspada dan tidak selalu disamping sang anak agar selalu mengetahui keadaan dan posisi sang anak.

## Definisi dan Istilah

* SKPL adalah Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggrisnya sering juga disebut sebagai Software Requirements Spesification (SRS), dan merupakan spesifikasi dari perangkat lunak yang akan dikembangkan.
* ERD adalah Entity Relationship Diagram, diagram dan notasi yang digunakan untuk merepresentasikan struktur data statis pada perangkat lunak.
* SP adalah singkatan dari Smart Parent
* NFR merupakan singkatan dari Non Functional Requirements
* KF singkatan dari Kebutuhan Fungsional
* KNF singkatan dari Kebutuhan Non Fungsional

## Referensi

Yang kami jadikan sebagai Referensi adalah SKPL dan fitur tracking location pada handphone

## Sistematika Pembahasan

a. Bab 1 Pendahuluan merupakan Tujuan Penulisan Dokumen, Lingkup Masalah, Definisi dan Istilah , Aturan Penamaan dan Penomoran, referensi, dan Sistematika Dokumen.

b. Bab 2 Deskripsi Perancangan Global yang berisi tentang rancangan lingkungan implementasi, deskripsi arsitektual, dan deskripsi komponen.

c. Bab 3 Perancangan Rinci berisi tentang realisasi use case, perancangan antar muka, identifikasi object / kelas baru, sequence diagram, perancangan detil kelas, diagram keseluruhan , perancangan representasi database dan perancangan algoritma dan query.

# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

Aplikasi smart parent ini bisa di implementasi dalam ios ataupun android.Aplikasi ini dibuat dengan bahasa pemrograman java.

## Deskripsi Komponen

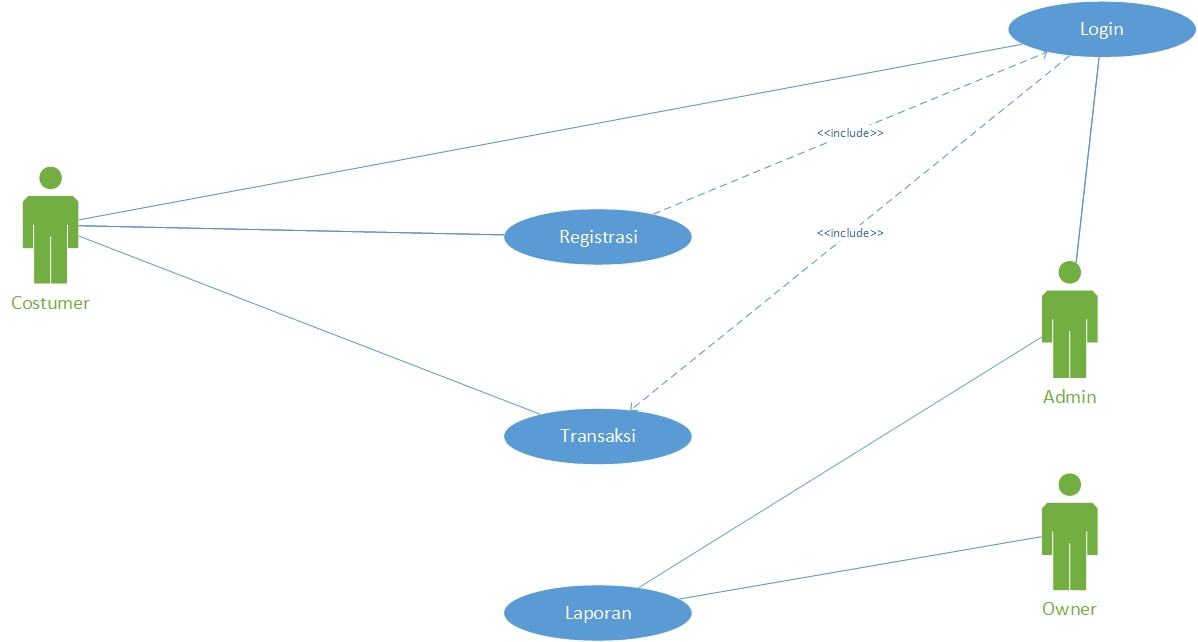
*Diisi dengan daftar modul. Daftar modul bisa dalam bentuk tabel berikut:*

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Komponen** | **Keterangan** |
| 1. | Handphone | Untuk Aplikasinya |
| 2. | Internet | Untuk mengakses aplikasinya |
| 3. | Operating System | Untuk wadah aplikasi |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Perancangan Rinci

## Realisasi Use Case

### Use Case



#### 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:*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| *1* | *Registrasi* | *Public* |
| *2* | *registrasiController* | *Public* |
| *3* | *registrasiView* | *Public* |

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

#### Usecase Scenario

Use Case Name : Login

Input : Data Akun

Output : Data Login

Actor : Customer

Precondition : Jika ingin login customer harus menginputkan data akun pribadinya yang sudah terdaftar.

Post Condition : customer telah menginputkan data akunnya dan system berhasil meng-accept agar customer dapat login.

Description : untuk masuk dalam aplikasi

Typical course of Event :

|  |  |
| --- | --- |
| Actor | System |
| 1. Buka menu login |  |
|  | 1. Menampilkan menu login |
| 1. Masukkan data akun |  |
|  | 1. Mengkonfirmasi data akun |
| 1. Sudah masuk dalam aplikasi |  |

Use case name : Transaksi

Input : data layanan yang ingin dibeli

Output : info total harga, data pemesanan

Actor : Customer, Admin

Precondition : Jika ingin memesan layanan aplikasi , customer telah login untuk menggunakan menu layanan yang diinginkan

Post condition : customer telah memilih layanan

Description : untuk memilih layanan, dan melakukan pemesanan

Typical course of event :

|  |  |  |
| --- | --- | --- |
| Customer | Admin | System |
| 1. Memilih menu layanan |  |  |
|  |  | 1. Menampilkan men u layanan |
| 1. Memilih layanan yang mau dipesan |  |  |
|  |  | 1. Mengkonfirmasi layanan tersebut 2. Menampilkan harga total |
| 1. Checkout pembayaran |  |  |
|  | 1. Menerima info pemesanan |  |

Use case name : Pembayaran

Input : pembayaran aplikasi dan pembayaran alat

Output :

Actor : Customer dan Admin

Precondition : Jika customer ingin melakukan pembayaran, customer yang telah melakukan check out melakukan pembayaran di ATM ataupun M-Banking sesuai yang telah dipilih sebelum check out

Postcondition : customer telah melakukan permbayaran

Description : untuk melakukan pembayaran

Typical course of event :

|  |  |  |
| --- | --- | --- |
| Customer | Admin | System |
| 1. Melakukan pembayaran |  |  |
|  |  | 1. Mengkonfirmasi hasil pembayaran |
| 1. Menerima konfirmasi pembayaran |  |  |
|  | 1. Menerima bukti pembayaran dari customer |  |

Use case name : Pengiriman

Input : Barang dan aplikasi yang telah dibayar

Output : Pengiriman barang dan layanan aplikasi

Actor : Customer, Admin

Precondition : Customer yang telah melakukan pembayaran barang dan aplikasi, akan dikirimkan barangnya.

Post condition : barang telah dikirimkan ke customer

Description : untuk melakukan pengiriman barang ke customer

Typical course of event :

|  |  |  |
| --- | --- | --- |
| Customer | Admin | System |
|  | 1. Mengirimkan barang melalui kurir |  |
|  |  | 1. Memberi info barang telah dikirim |
| 1. Menerima info barang telah dikirim |  |  |

Use case name : Laporan

Input : data yang terjadi ditransaksi

Output : info transaksi

Actor : Admin, owner

Precondition : Jika ingin membuat laporan admin menuliskan data-data dan info yang terjadi selama proses transaksi

Post condition : admin telah membuat laporan

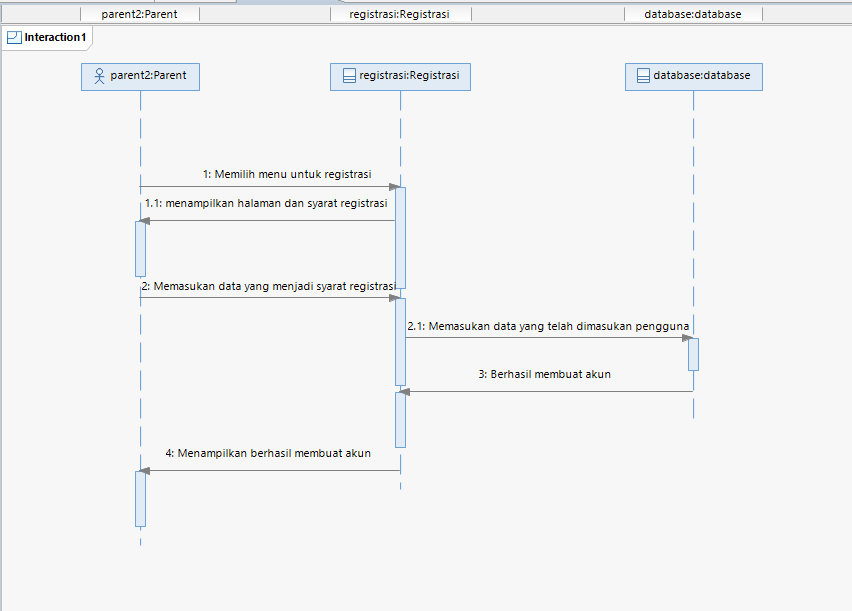
Description : untuk membuat laporan dan memberikan laporan kepada owner

Typical course of event :

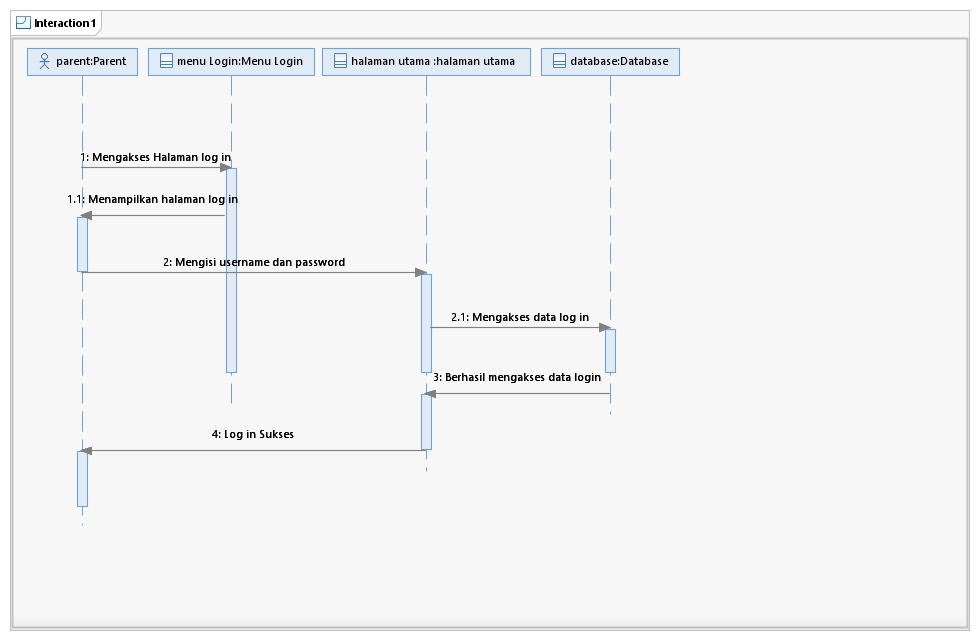
|  |  |
| --- | --- |
| Admin | Owner |
| 1. Membuat Laporan 2. Menigirm laporan |  |
|  | 1. Menerima laporan |

#### Sequence Diagram

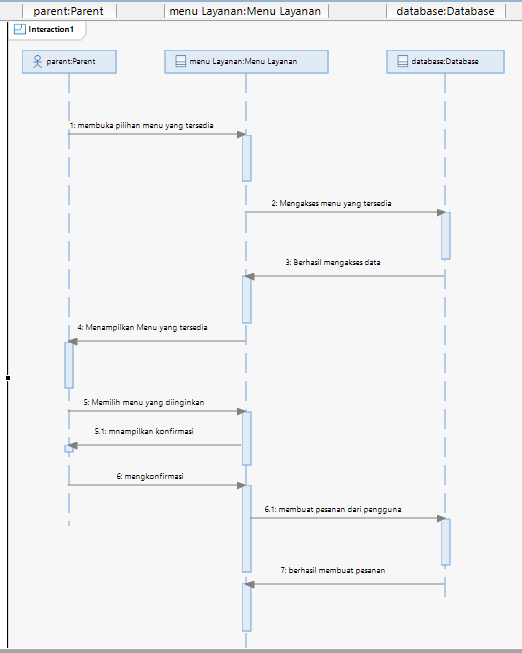
Registrasi



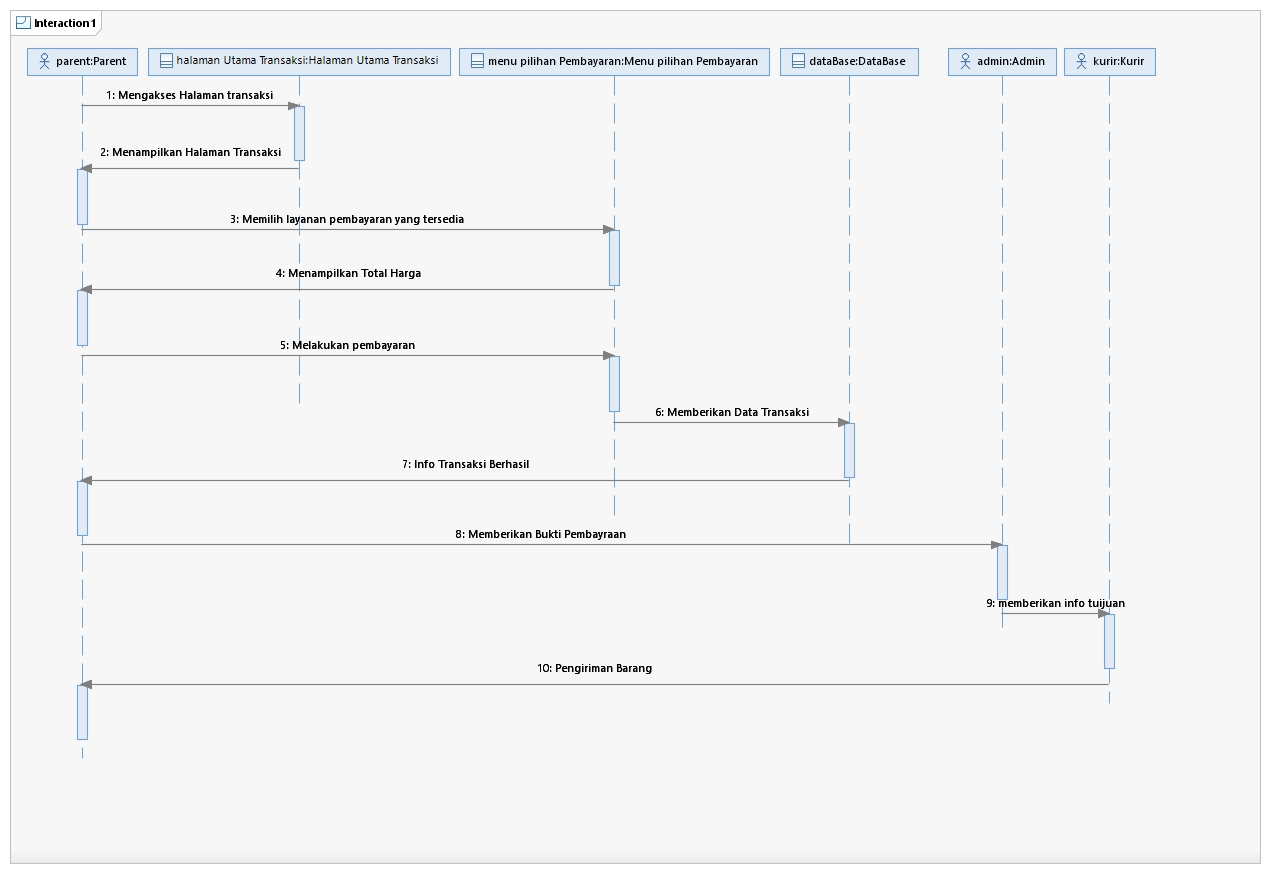
Login



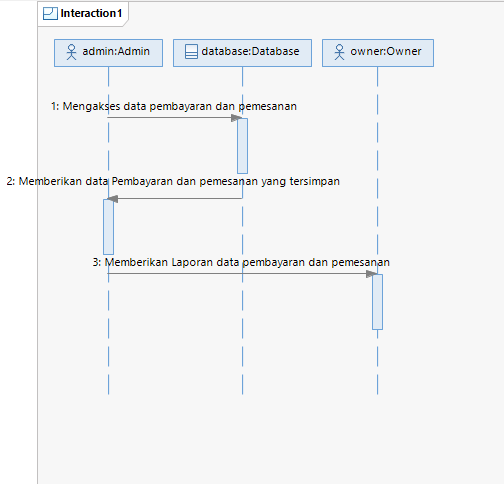
Menu layanan



Transaksi

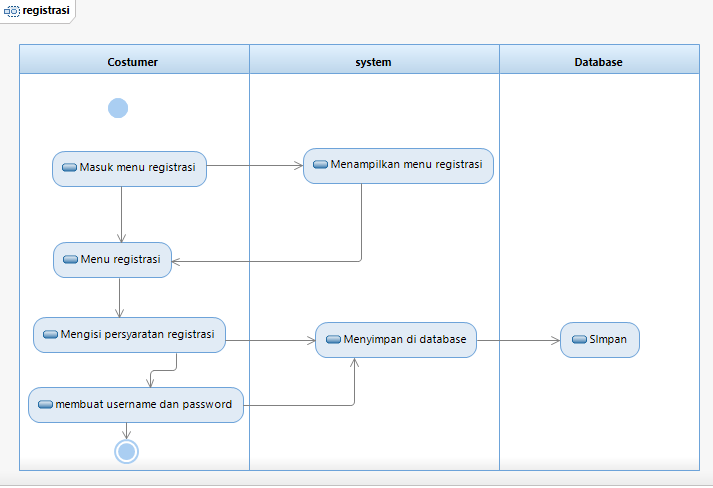


Laporan

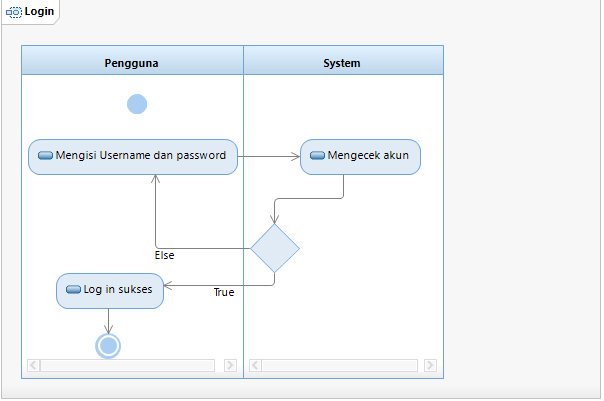


**.1.1.4 *Deployment* Diagram**

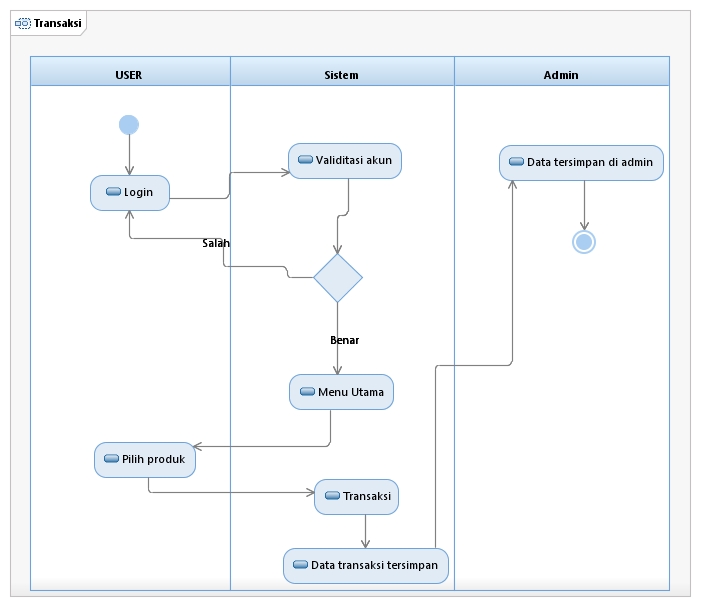
**Registrasi**

****

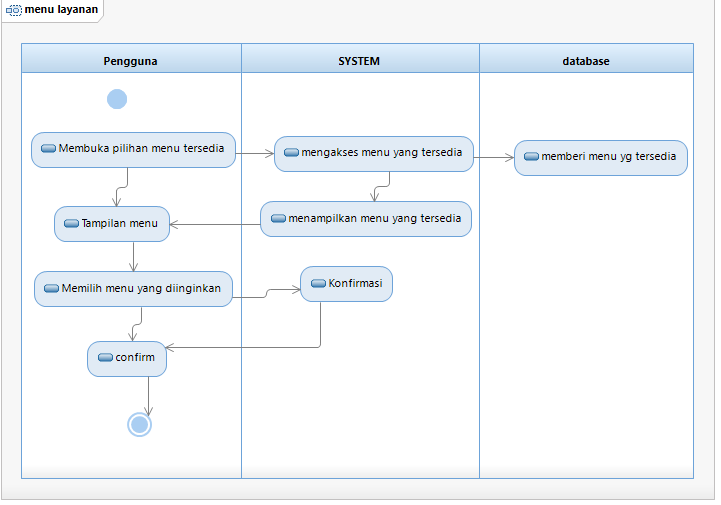
**Login**

****

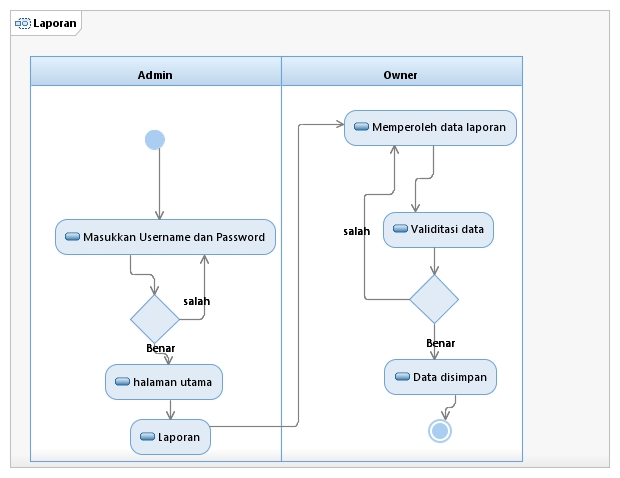
**Transaksi**

****

**Menu Layanan**

****

**Laporan**

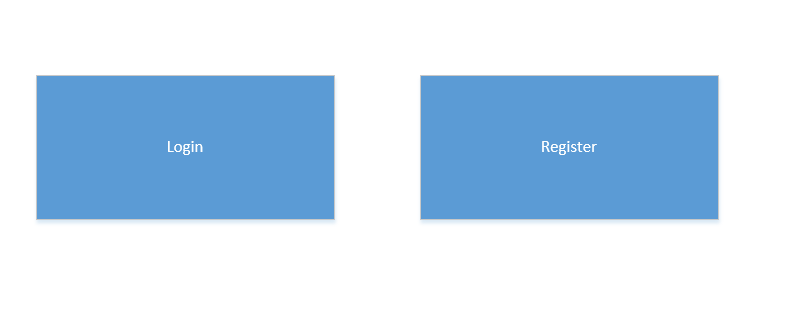
****

#### Diagram Kelas

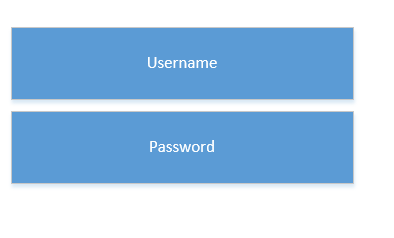
*Buatlah diagram kelas untuk use case tersebut. buat class diagram* ***BUKAN KESELURUHAN, tapi PER USE CASE***

#### Antarmuka Pengguna

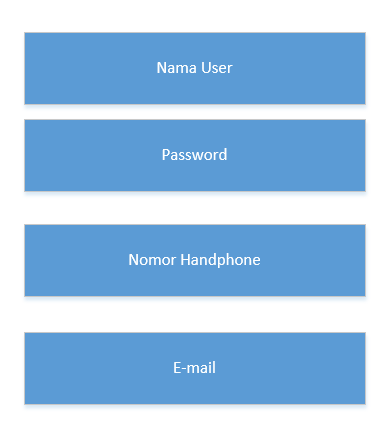
* Menu Utama

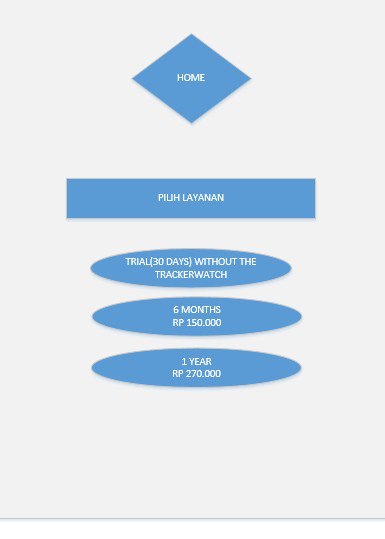


* *Login*



* *Register*



* Halaman uta

## Perancangan Detil Kelas

*Bagian ini diisi dengan daftar seluruh kelas dalam tabel berikut:*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *1* | *DB\_ORDER* | *-* |
| *2* | *RegistrasiController* | *Registrasi* |
| *3* | *Registrasiview* | *Registrasi* |
| *4* | *Login* | *User* |
| *5* | *LoginView* | *user* |
| *6* | *Login Controller* | *-* |
| *7* | *View Order* | *-* |
| *8* | *Controller Order* | *Transaksi* |
| *9* | *DB\_Order* | *-* |
| *10* | *View tagihan transaksi* | *Transaksi* |
| *11* | *View notifikasi pembayaran* | *Transaksi* |
| *12* | *DB\_ORder* | *-* |
| *13* | *View delivery* | *Kurir* |
| *14* | *DB\_ORDER* | *-* |
| ***15*** | ***View laporan*** | ***Laporan*** |
| *16* | *Controller laporan* | *Laporan* |
| *17* | *DB\_Order* | *-* |
|  |  |  |

*Untuk setiap kelas:*

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

### Kelas <nama kelas>

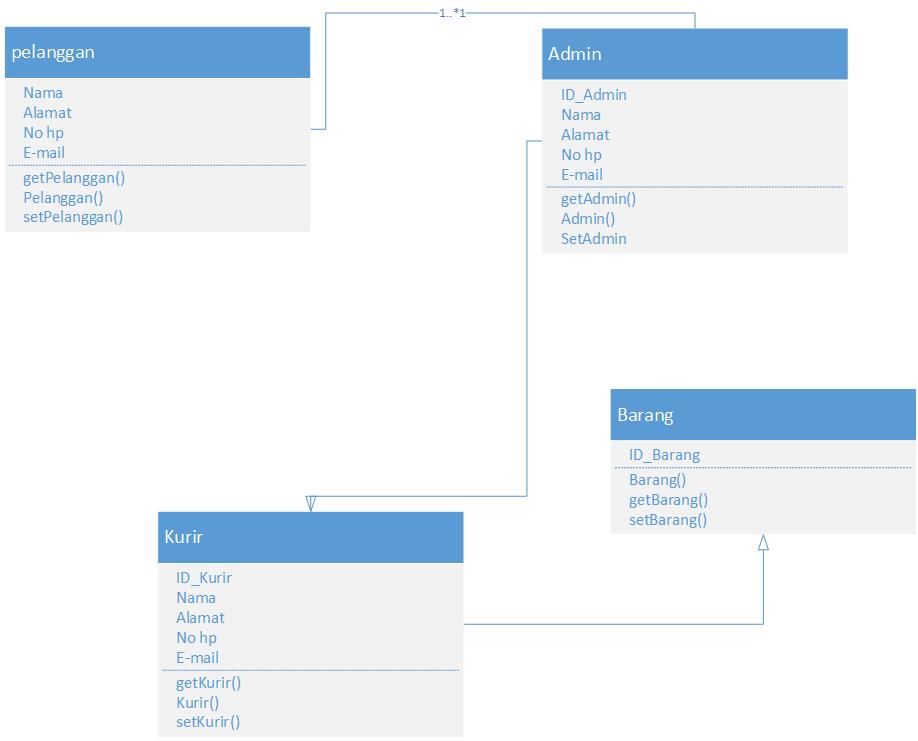
Bagian ini diisi dengan daftar operasi dan atribut Buat untuk setiap kelas.

*Nama Kelas : ……..*

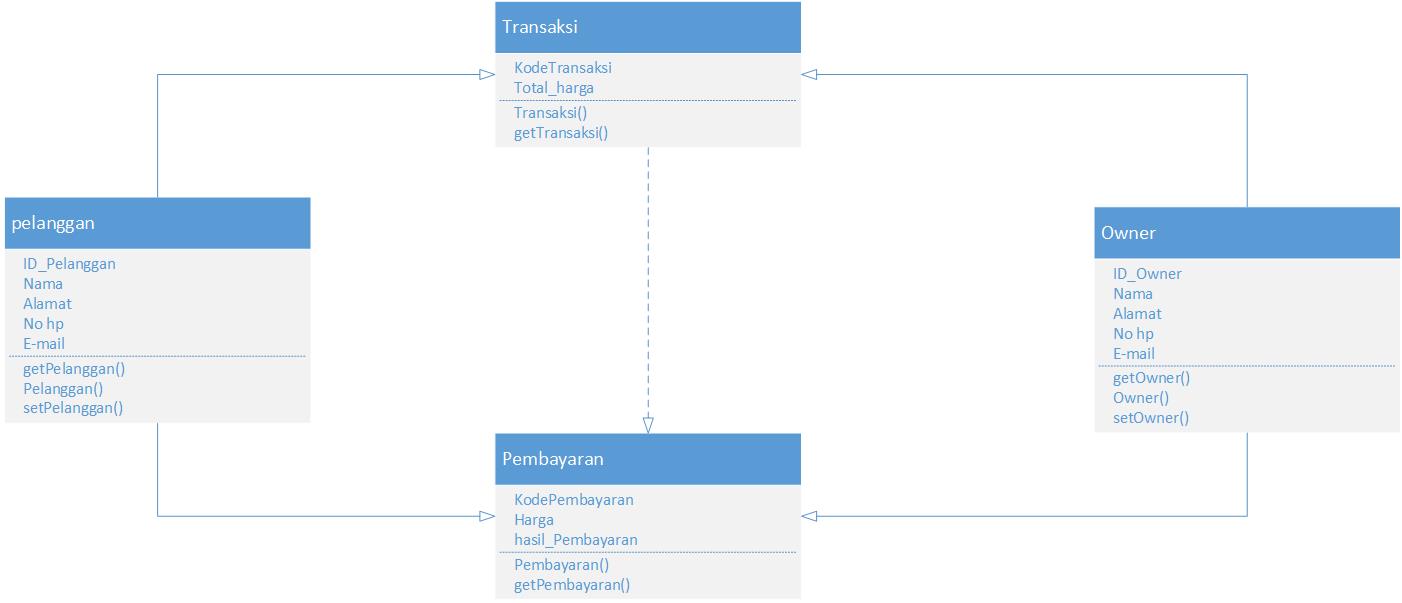
|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| *Diisi dengan signature operasi* |  |  |
|  |  |  |
|  |  |  |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Diisi dengan nama atribut* |  | *Tuliskan tipenya sesuai dengan yang dikenal pada bahasa pemrograman yang digunakan* |
|  |  |  |
|  |  |  |

## Diagram Kelas Keseluruhan

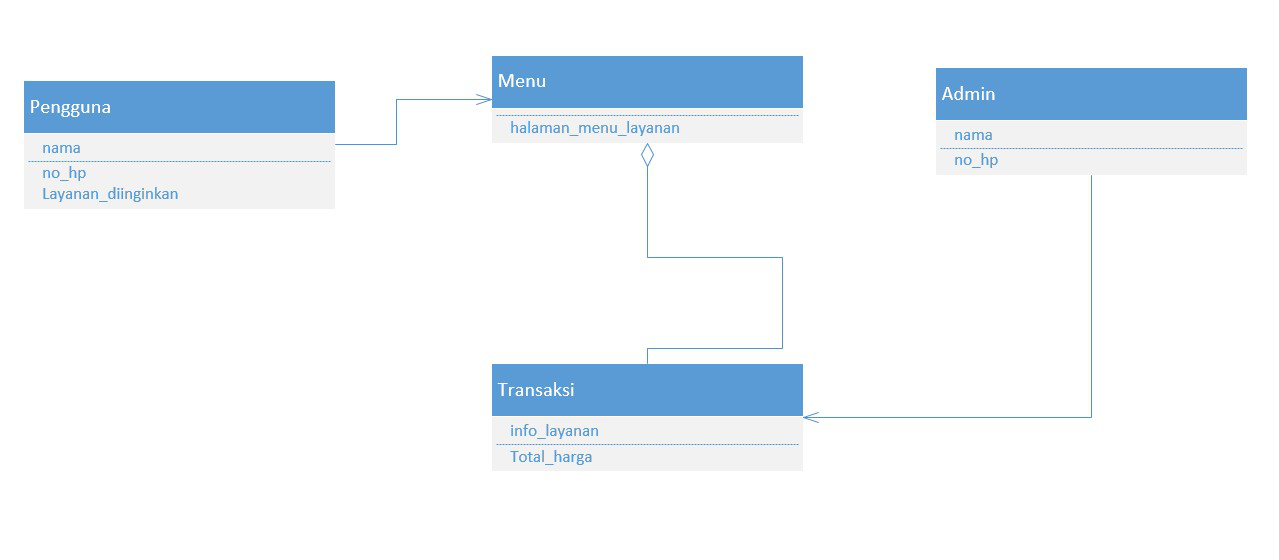
### Kelas <Pengiriman >



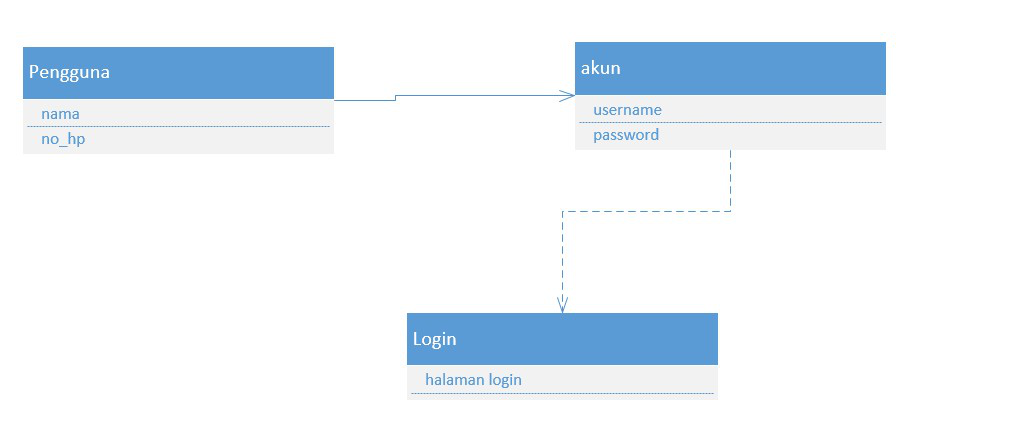
**Kelas <Pembayaran>**



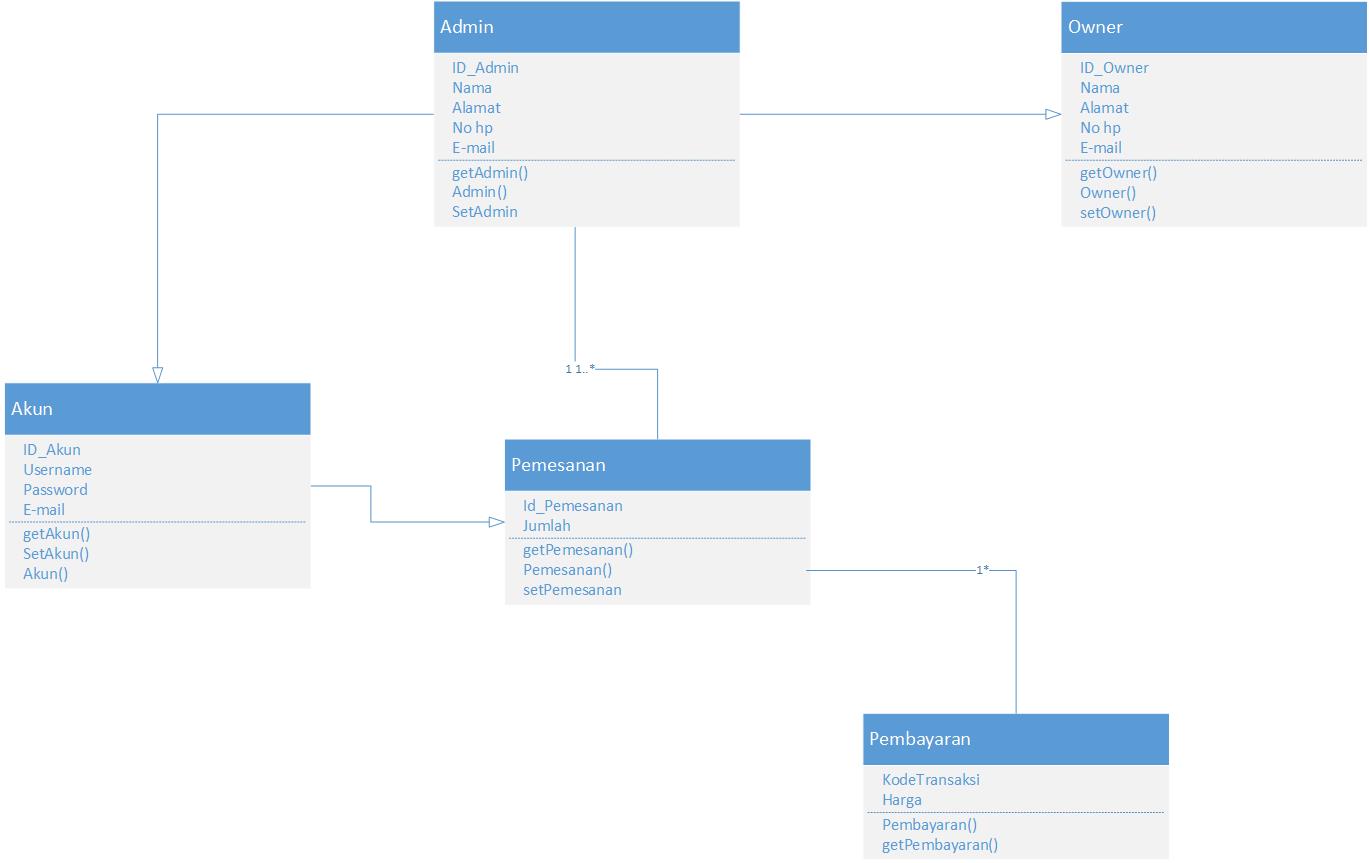
**Kelas <Transaksi >**



**Kelas <Login >**



**Kelas <Laporan >**



## Algoritma/Query

|  |  |  |
| --- | --- | --- |
| No Query | Query | Keterangan |
| Q-001 | Registrasi | Membuat akun pelanggan dengan cara registrasi menggunakan alamat email dan nomor telepon agar bias membeli aplikasi |
| Q-002 | Login | Registrasi berhasil, masuk menu Login , Login berhasil, masuk menu utama dan di menu utama ada pilihan trial, berlangganan selama 6 bulan dan berlangganan selama 1 tahun. Setelah memilih maka akan masuk ke menu transaksi antar bank kecuali yang meggunakan trial. |
| Q-003 | Transaksi | Menampilkan alamat utama, memberi pilihan bank untuk bertransaksi, memberikan pilihan jenis transaksi. Setelah itu akan dikasih no transaksi yang harus di bayar. |
| Q-004 | Pengiriman | Setelah transaksi berhasil, admin akan mengecek kembali alamat dan durasi pembelian aplikasi tersebut. Setelah semua sukses maka admin akan mengirimkan lewat kurir. |

## 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 yg tampil pd layar* | *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 diklik, 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*

## Perancangan Representasi Persistensi Kelas

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

# Matriks Kerunutan

*Mapping use case dengan kelas-kelas terkait*

|  |  |
| --- | --- |
| **Requirement** | **Usecase Terkait** |
| FR-01 | Registrasi |
| FR-02 | Login |
| FR-03 | Transaksi |
| FR-04 | Status |
| FR-05 | Delivery |
| FR-06 | Layanan |