**DPPL-05**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Yang Penting Diskon!

untuk:

Masyarakat Bandung dan sekitarnya

Dipersiapkan oleh:

Kelompok 5

Taufik Fathurahman (1301160790)

Brenda Irena (1301164641)

Husnul Khotimah Farid (1301162742)

Siti Fitria Yonalia (1301162744)

Yan Roberto (1301154715)

Program Studi Teknik Informatika

FIF – Telkom University

Jl. Telekomunikasi, Bandung 40257

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TU-logo-primer-memusat | **Prodi S1- Teknik Informatika**  **Universitas Telkom** | Nomor Dokumen | | Halaman |
| *DPPL-05* | | *75* |
| Revisi |  | *Tgl: 26/11/2018* |

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A | Matriks kerunutan (4) |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX  TGL | - | A | B | C | D | E | F | G |
| Ditulis oleh |  | Kelompok 5 |  |  |  |  |  |  |
| Diperiksa oleh |  | ASPRAK APPL |  |  |  |  |  |  |
| Disetujui oleh |  | ASPRAK APPL |  |  |  |  |  |  |

Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
| 72 | Matriks kerunutan (4) |  |  |

Daftar Isi

1. Pendahuluan 7

1.1 Tujuan Penulisan Dokumen 7

1.2 Lingkup Masalah 7

1.3 Definisi dan Istilah 8

1.4 Aturan Penamaan dan Penomoran 9

1.5 Referensi 9

1.6 Ikhtisar Dokumen 10

2 Deskripsi Perancangan Global 11

2.1 Rancangan Lingkungan Implementasi 11

2.2 Deskripsi Arsitektural 11

2.3 Deskripsi Komponen 12

3 Perancangan Rinci 13

3.1 Realisasi Use Case 13

3.1.1 Use Case Kelola Kategori 13

3.1.1.1 Identifikasi Kelas 13

3.1.1.2 Sequence Diagram 14

3.1.1.3 Diagram Kelas 15

3.1.2 Use Case Kelola Produk 15

3.1.2.1 Identifikasi Kelas 15

3.1.2.2 Sequence Diagram 16

3.1.2.3 Diagram Kelas 18

3.1.3 Use Case Kelola Toko 18

3.1.3.1 Identifikasi Kelas 18

3.1.3.2 Sequence Diagram 19

3.1.3.3 Diagram Kelas 21

3.1.4 Use Case Kelola Akun 21

3.1.4.1 Identifikasi Kelas 21

3.1.4.2 Sequence Diagram 22

3.1.4.3 Diagram Kelas 23

3.1.5 Use Case Login 23

3.1.5.1 Identifikasi Kelas 23

3.1.5.2 Sequence Diagram 24

3.1.5.3 Diagram Kelas 25

3.1.6 Use Case Registrasi 26

3.1.6.1 Identifikasi Kelas 26

3.1.6.2 Sequence Diagram 26

3.1.6.3 Diagram Kelas 27

3.1.7 Use Case Pilih Kategori 28

3.1.7.1 Identifikasi Kelas 28

3.1.7.2 Sequence Diagram 28

3.1.7.3 Diagram Kelas 29

3.1.8 Use Case Pilih Barang 29

3.1.8.1 Identifikasi Kelas 29

3.1.8.2 Sequence Diagram 30

3.1.8.3 Diagram Kelas 30

3.2 Perancangan Detil Kelas 31

3.2.1 Kelas Akun 32

3.2.2 Kelas Toko 32

3.2.3 Kelas Kategori 33

3.2.4 Kelas Aplikasi 34

3.2.5 Kelas Shop Cart 34

3.2.6 Kelas View Kelola Produk 35

3.2.7 Kelas View Kelola Kategori 35

3.2.8 Kelas Controller Kelola Kategori 36

3.2.9 Kelas Controller Kelola Akun 37

3.2.10 Kelas View Kelola Account 37

3.2.11 Kelas View Produk 38

3.2.12 Kelas View Kategori 38

3.2.13 Kelas Controller Login 38

3.2.14 Kelas View Login 39

3.2.15 Kelas User 39

3.2.16 Kelas Produk 40

3.2.17 Kelas Database 41

3.2.18 Kelas Controller Kelola Produk 42

3.2.19 Kelas Controller Kelola Toko 42

3.2.20 Kelas View Kelola Toko 43

3.2.21 Kelas View Registrasi 43

3.2.22 Kelas Controller Registrasi 44

3.2.23 Kelas Controller Kategori 44

3.2.24 Kelas Controller Cart 45

3.2.25 Kelas Controller produk 45

3.2.26 Kelas View Cart 46

3.3 Diagram Kelas Keseluruhan 47

3.4 Algoritma/Query 48

3.5 Diagram Statechart 56

3.6 Perancangan Antarmuka 57

3.7 Perancangan Representasi Persistensi Kelas 71

4 Matriks Kerunutan 72

5 Lampiran Lain 74

5.1 Use Case 74

5.2 Deployment Diagram 75

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini akan berisikan penjelasan Deskripsi Perancangan Perangkat Lunak (DPPL) untuk Aplikasi Yang Penting Diskon (AYEDIS). Dokumen ini akan ditujukan sebagai deskripsi rinci dari perangkat lunak yang akan dibangun, acuan pada proses pembangunan dan evaluasi selama pembangunan perangkat lunak. Dokumen ini akan ditujukan kepada tim pengembang perangkat lunak dan *stakeholders* yang terlibat didalamnya.

## Lingkup Masalah

AYEDIS (Aplikasi Yang Penting Diskon) merupakan sebuah produk perangkat lunak dapat membantu proses pemberian informasi dari produsen kepada konsumen. Informasi pada perangkat lunak ini akan berfokus pada produk-produk yang sedang pada masa diskon untuk wilayah Bandung dan sekitarnya. AYEDIS akan mampu menolong para pencari barang diskon yang berada di wilayah Bandung dan sekitarnya dengan mudah, praktis, akurat, dan ter-*update*. Dan bagi para produsen yang ingin memasarkan, atau mengiklankan produk-produknya yang sedang dalam masa diskon dapat menjadikan perangkat lunak ini sebagai wadah promosi dan pemasaran. Para produsen atau akan kami sebut sebagai *user seller* pada perangkat lunak ini dapat melakukan pembuatan toko, penambahan, pembaharuan, dan penghapusan produk setiap saat dengan mudah dan cepat, sehingga keakuratan dan keterbaharuan dari informasi yang ada pada perangkat lunak selalu terjaga dengan baik.

## Definisi dan Istilah

|  |  |  |
| --- | --- | --- |
| NO | Kata-Kata Sukar | Penjelasan |
| 1 | SKPL | 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. |
| 2 | Use Case Diagram | Gambaran grafis dari beberapa atau semua *actor, use-case,* dan interaksi antara komponen-komponen tersebut yang memperkenalkan satu sistem yang dibangun. |
| 3 | AYEDIS | Merupakan singkatan dari perangkat lunak yang sedang dikembangkan yaitu “Yang Penting Diskon!” |
| 4 | API | (*Application Programming Interface)* adalah sekumpulan instruksi program dan protokol yang digunakan untuk membangun aplikasi perangkat lunak. |
| 5 | Smartphone | Telepon genggam yang mempunyai kemampuan tingkat tinggi, kadang-kadang dengan fungsi yang menyerupai komputer. |
| 6 | DPPL | Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) merupakan dokumen deskripsi dari perancangan perangkat lunak yang akan dikembangkan dan bertujuan untuk memberikan landasan yang diperlukan dalam proses pengkodean aplikasi. |
| 7 | Android | Sistem operasi berbasis *Linux* yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. |
| 8 | Jellybean | Salah satu jenis sistem operasi pada Android. |
| 9 | Oracle | Salah satu produk yang bisa digunakan untuk mengolah data atau mengumpulkan data. |
| 10 | Quadcore | Salah satu jenis prosesor. |
| 11 | RAM | Suatu memori tempat penyimpanan data sementara. |
| 12 | DBMS | Sistem pengorganisasian dan sistem pengolahan *database* pada komputer. |
| 13 | User Customer | Pengguna yang mendapatkan informasi. |
| 14 | User Seller | Pengguna yang mempromosikan produk (toko). |
| 15 | Admin | Pengguna yang mengelola sistem. |
| 16 | Database | Basis data. |

## Aturan Penamaan dan Penomoran

Dalam dokumen DPPL ini akan ada dua aturan dalam penamaan dan penomoran yaitu antara lain:

1. Penomoran algoritma menggunakan ”Algo-”, yang diikuti dengan tiga angka secara beruntun yang diawali dengan 001.
2. Penomoran query akan menggunakan” Query – y”, yang akan ikuti dengan kombinasi tiga angka secara beruntung dengan diawali oleh 001.

## Referensi

Referensi yang digunakan dalam pembuatan dokumen DPPL ini adalah sebagai berikut:

1. *Template* Pengisian Deskripsi Perancangan Perangkat Lunak (DPPL) Berorientasi Objek.
2. Dokumen SKPL Sistem Informasi Inventaris.
3. Dari link : <http://sisil.dosen.ittelkom-pwt.ac.id/wp-content/uploads/sites/3/2017/10/Contoh-DPPL.pdf> . Diakses pada 20/11/2018
4. Dari link : <https://www.academia.edu/32875999/DESKRIPSI_PERANCANGAN_PERANGKAT_LUNAK_DPPL_HALEN_BOOKSTORE>. Diakses pada 20/11/2018

## Ikhtisar Dokumen

Dokumen DPPL ini berisikan deskripsi rancangan perangkat lunak sistem informasi barang diskon yang kami beri nama “Yang Penting Diskon!” atau disingkat AYEDIS yang akan dikembangkan berdasarkan dokumen SKPL. Pada dokumen DPPL ini akan dijelaskan rincian dari rancangan perangkat lunak sehingga dapat diimplementasikan. Dokumen ini secara garis besar teridiri dari lima bab dengan perincian sebagai berikut:

1. Pendahuluan

Pendahuluan berisi penjelasan tentang dokumen DPPL yang mencakup tujuan pembuatan dokumen ini, lingkup masalah yang diselesaikan oleh perangkat lunak yang dikembangkan, definisi, aturan penamaan dan penomoran, referensi, dan ikhtisar dokumen.

1. Deskripsi Perancangan Global

Deskripsi perancangan global berisi tentang rancangan dari perangkat lunak yang akan dibangun meliputi, rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen.

1. Perancangan Rinci

Perancangan rinci pada dokumen ini berisi tentang realisasi use case, perancangan detil kelas, deskripsi diagram kelas, algoritma/query, diagram statechart, perancangan antarmuka, dan perancangan representasi persistensi kelas.

1. Matriks Kerunutan

Matriks kerunutan berisi tentang hal fungsional yang terdapat pada dokumen SKPL.

1. Lampiran Lainnya

Dalam lampiran lainnya ini akan berisikan lampiran-lampiran yang dirasa penting untuk ada pada dokumen DPPL*.*

# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

Berikut ini merupakan lingkup operasi dari aplikasi AYEDIS (Aplikasi Yang Penting Diskon) :

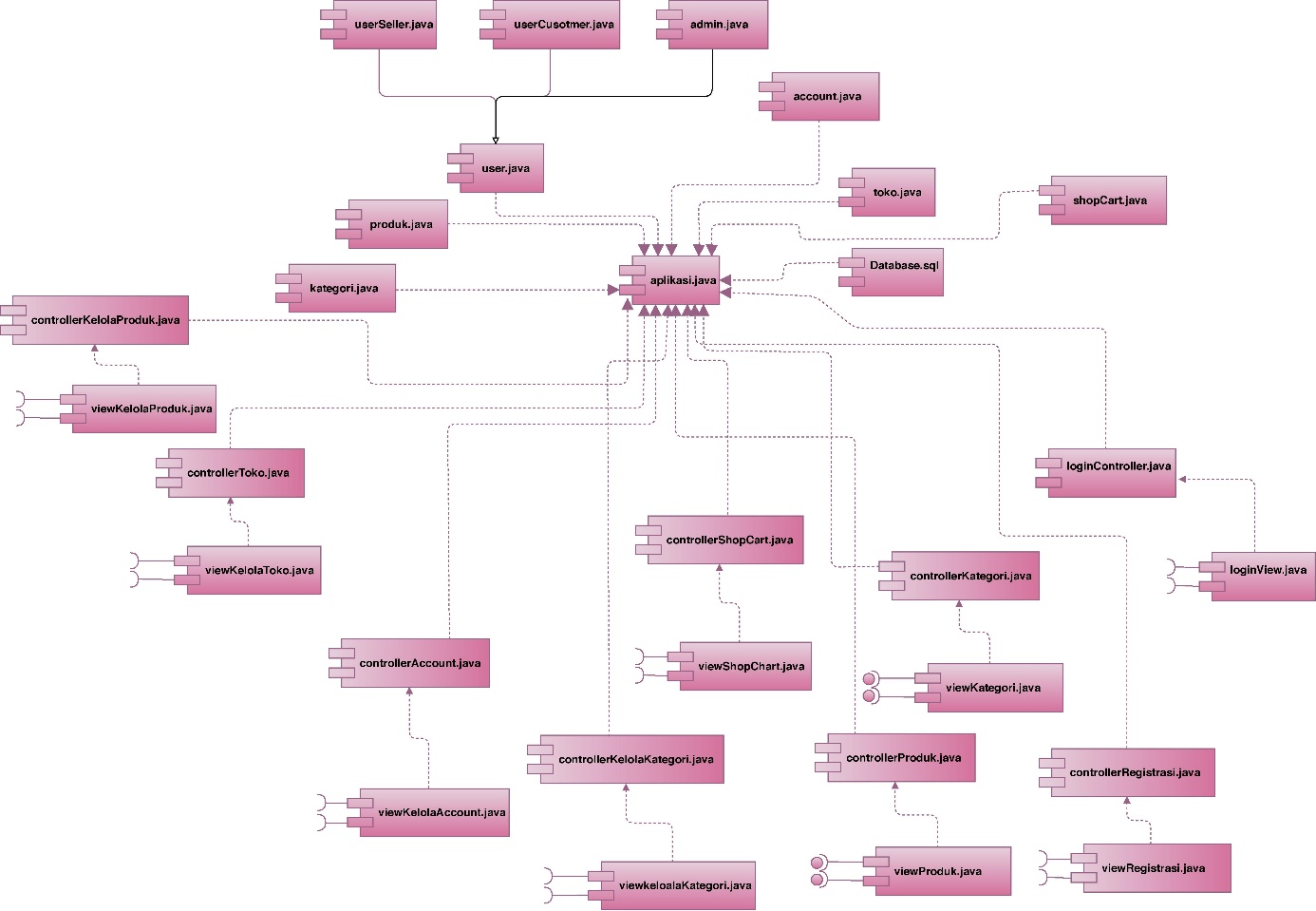
1. Hadware :

* Processor QuadCore, 1.2 GHz
* RAM 512 MB
* Internal storage free space 100 MB
* Smartphone dengan layar sentuh

1. Software :

* Sistem operasi minimum Jelly Bean
* DBMS Oracle 11g
* Bahasa pemrograman Java
* Sistem operasi windows server 2003

## Deskripsi Arsitektural



## Deskripsi Komponen

| **No** | **Nama Komponen** | **Keterangan** |
| --- | --- | --- |
| 1 | Database.sql | File *database* |
| 2 | kategori.java | File kelas model kategori |
| 3 | produk.java | File kelas model produk |
| 4 | user.java | File kelas model *user* |
| 5 | account.java | File kelas model *account* |
| 6 | toko.java | File kelas model toko |
| 7 | menu.java | File kelas model menu utama |
| 8 | aplikasi.java | File kelas *driver* untuk menjalankan seluruh aplikasi |
| 9 | viewKelolaProduk.java | File kelas view kelola barang |
| 10 | viewKelolaToko.java | File kelas view kelola toko |
| 11 | viewKelolaAccount.java | File kelas view kelola account |
| 12 | viewKategori.java | File kelas view kategori |
| 13 | viewKelolaKategori.java | File kelas view kelola kategori |
| 14 | viewShopCart.java | File kelas view keranjang belanja |
| 15 | viewProduk.java | File kelas view produk |
| 16 | viewLogin.java | File kelas view login |
| 17 | viewRegistrasi.java | File kelas view registrasi |
| 18 | controllerKelolaProduk.java | File kelas controller kelola barang |
| 19 | controllerKelolaToko.java | File kelas controller kelola toko |
| 20 | controllerKelolaAccount.java | File kelas controller kelola account |
| 21 | controllerKategori.java | File kelas controller kategori |
| 22 | controllerCart.java | File kelas controller detail dari keranjang |
| 23 | controllerKelolaKategori.java | File kelas controller kelola kategori |
| 24 | controllerProduk.java | File kelas controller produk |
| 25 | controllerLogin.java | file kelas controller login |
| 26 | controllerRegistrasi.java | file kelas controller registrasi |

# Perancangan Rinci

Dalam bab perancangan rinci akan dijelaskan mengenai realisasi use case, perancangan detil kelas, diagram kelas keseluruhan, algoritma/query, diagram statechart, perancangan antarmuka, dan perancangan representasi persistensi kelas.

## Realisasi Use Case

Subbab ini akan menjelaskan realisasi dari semua *use case* yang telah dirancang pada dokumen spesifikasi kebutuhan perangkat lunak (SKPL).

### Use Case Kelola Kategori

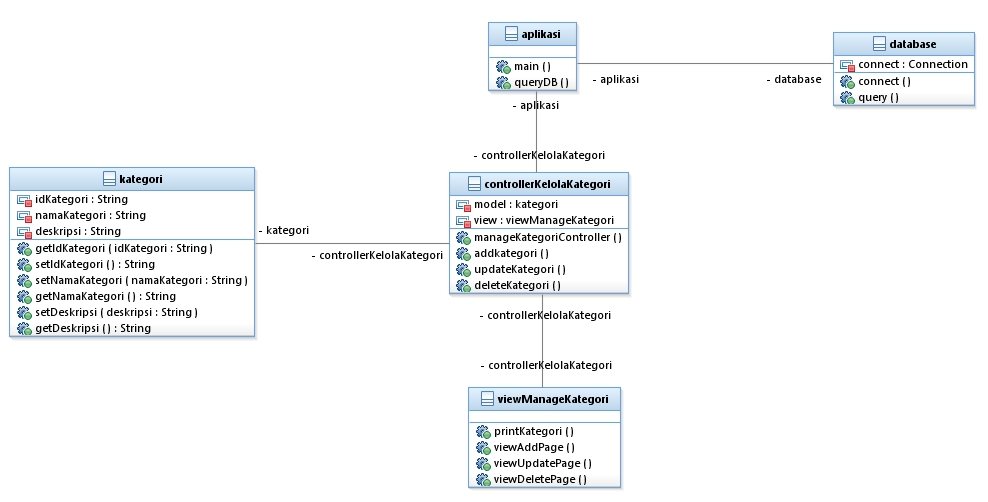
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewKelolaKategori | kategori |
| 2 | controllerKelolaKategori |  |
| 3 | kategori |  |
| 4 | aplikasi |  |
| 5 | database |  |

#### Sequence Diagram



#### Diagram Kelas

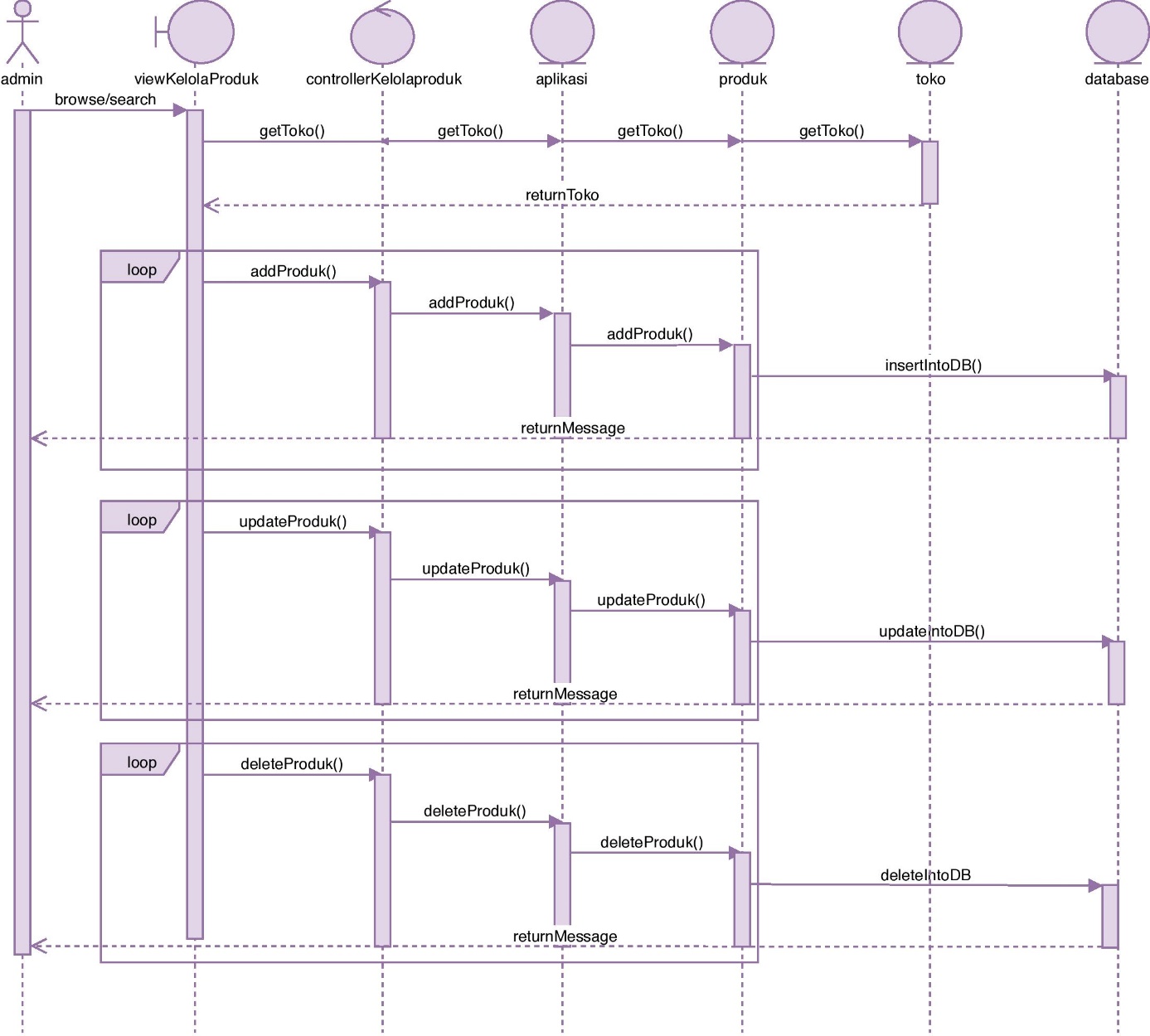


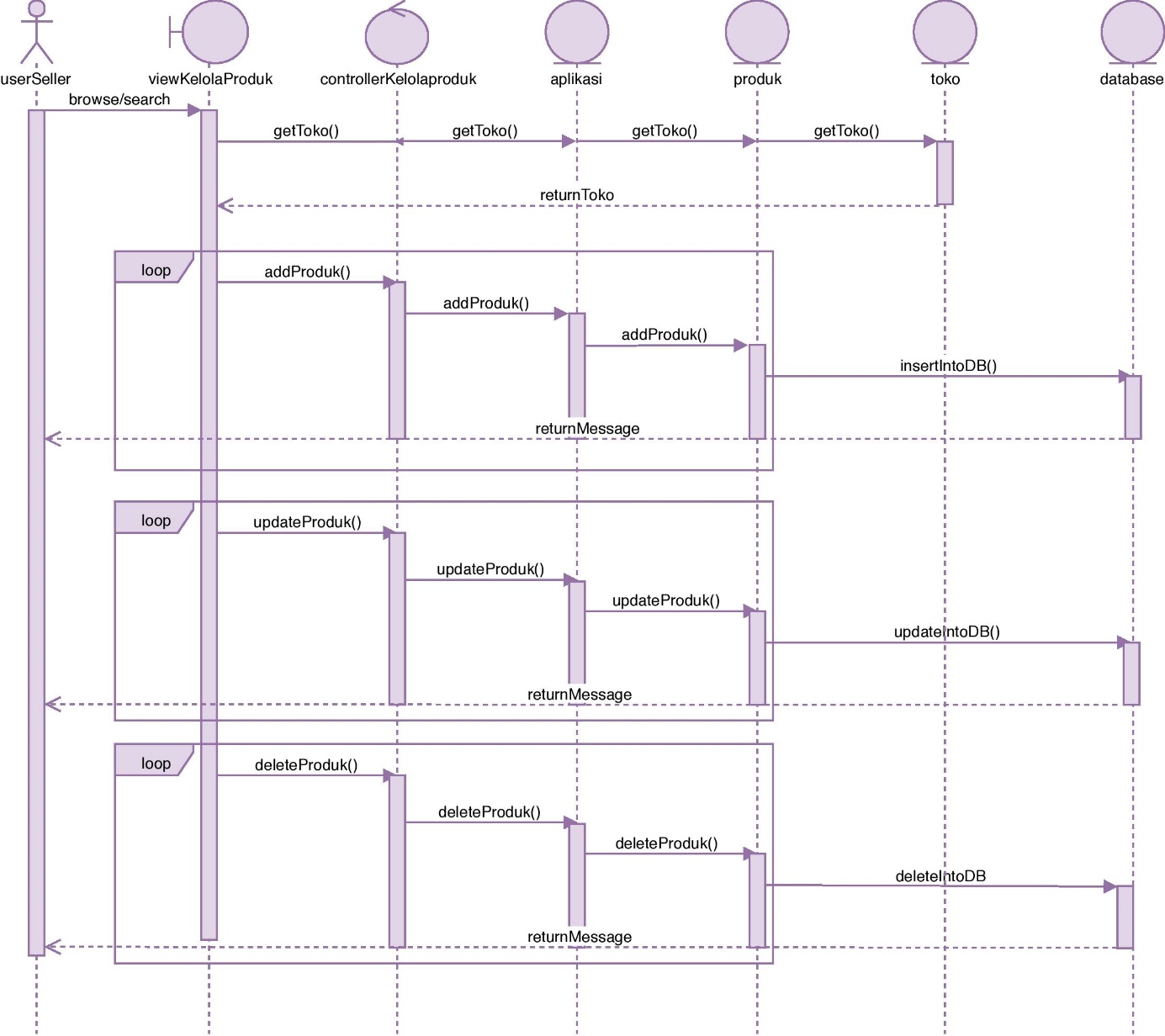
### Use Case Kelola Produk

#### Identifikasi Kelas

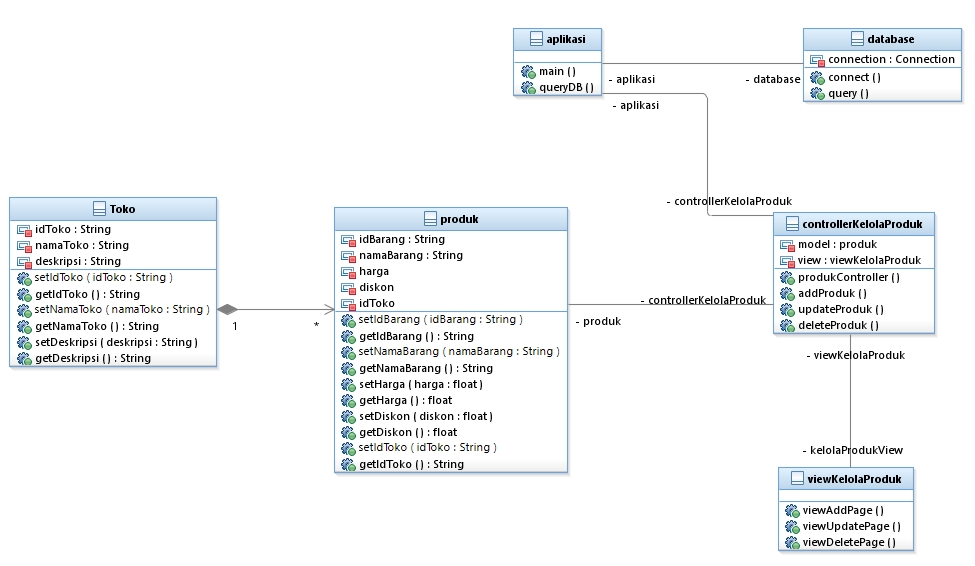
|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewKelolaProduk | produk |
| 2 | controllerKelolaProduk | toko |
| 3 | controllerToko |  |
| 4 | produk |  |
| 5 | aplikasi |  |
| 6 | database |  |

#### Sequence Diagram





#### Diagram Kelas

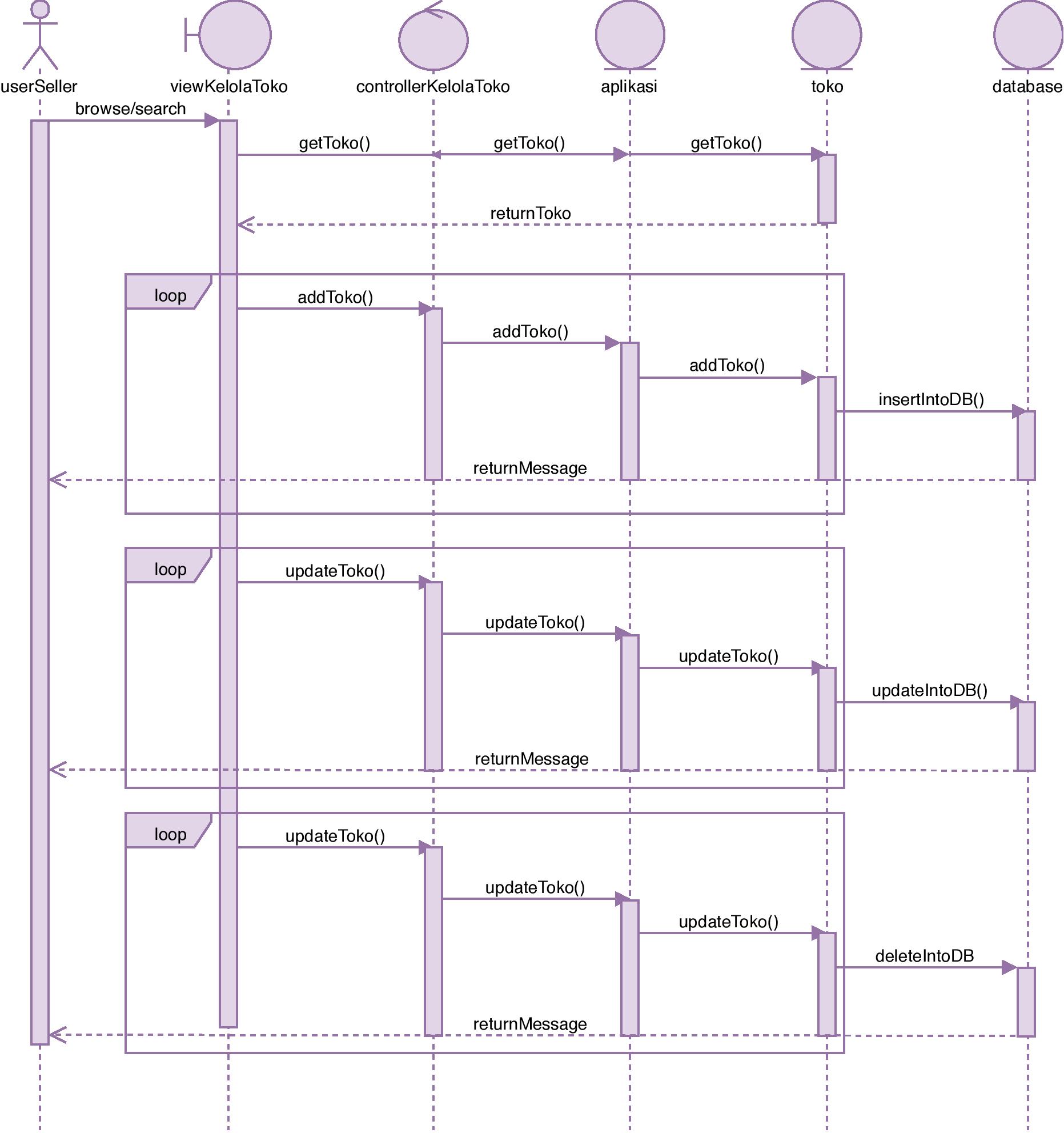


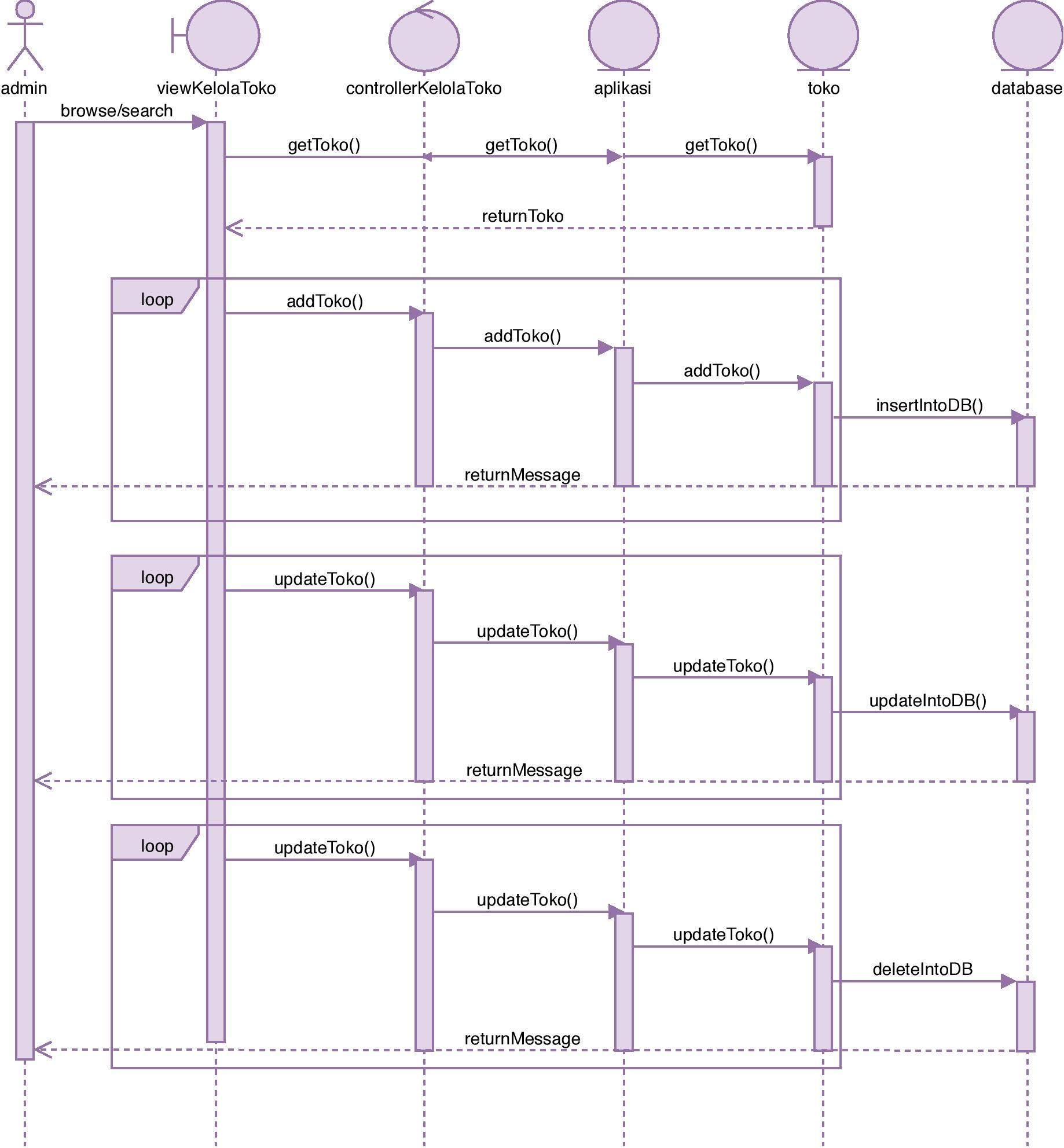
### Use Case Kelola Toko

#### Identifikasi Kelas

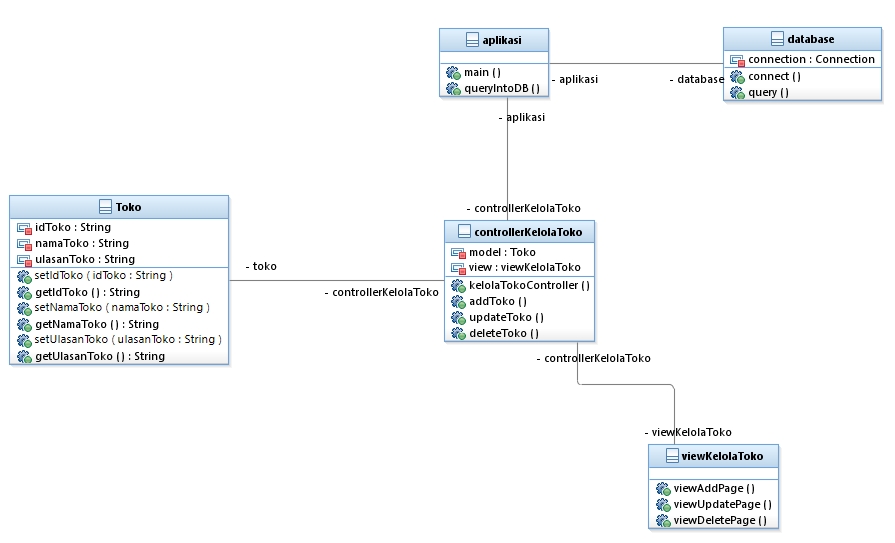
|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewKelolaToko | produk |
| 2 | controllerKelolaToko | toko |
| 3 | toko |  |
| 4 | aplikasi |  |
| 5 | database |  |

#### Sequence Diagram





#### Diagram Kelas

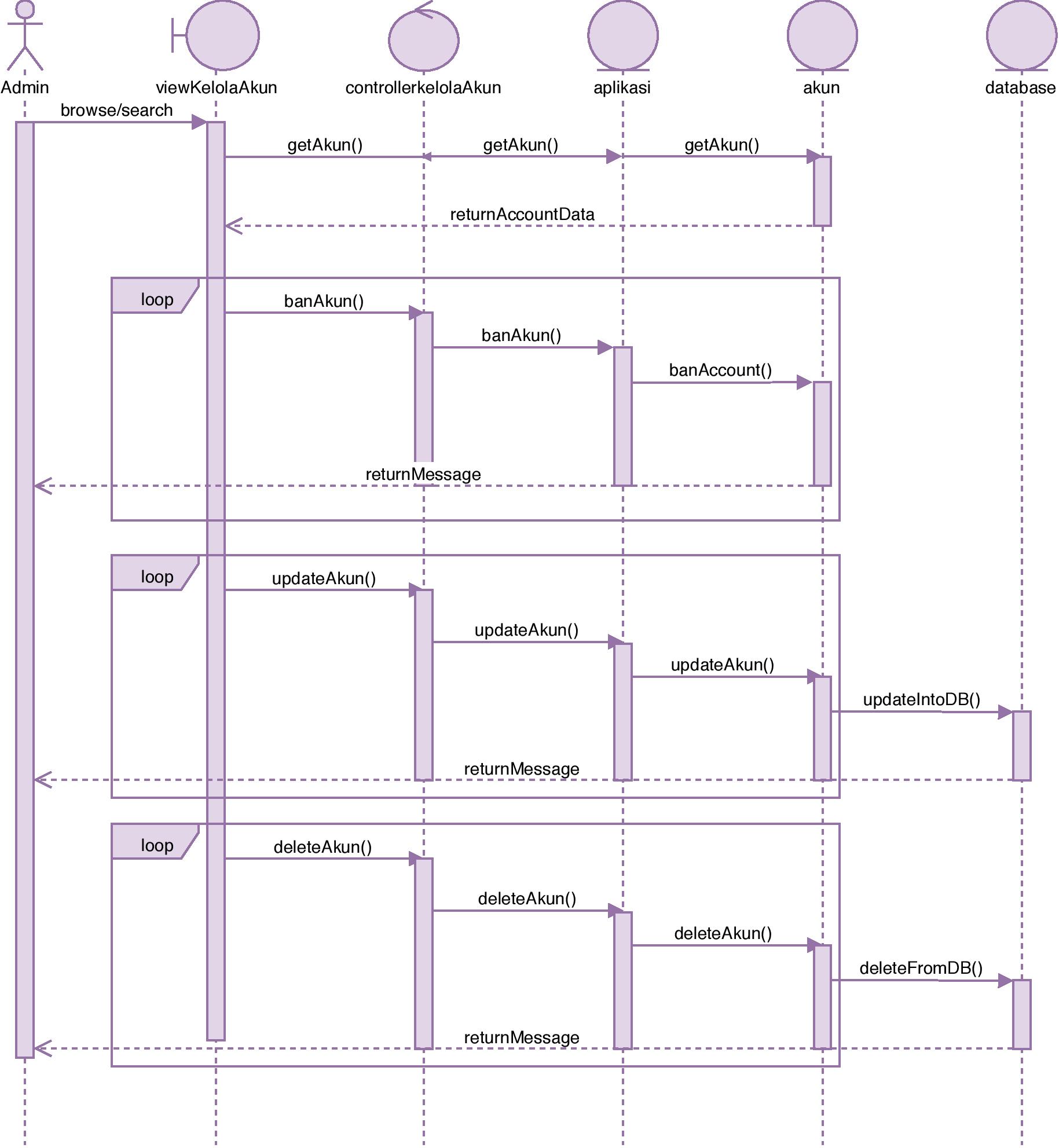


### Use Case Kelola Akun

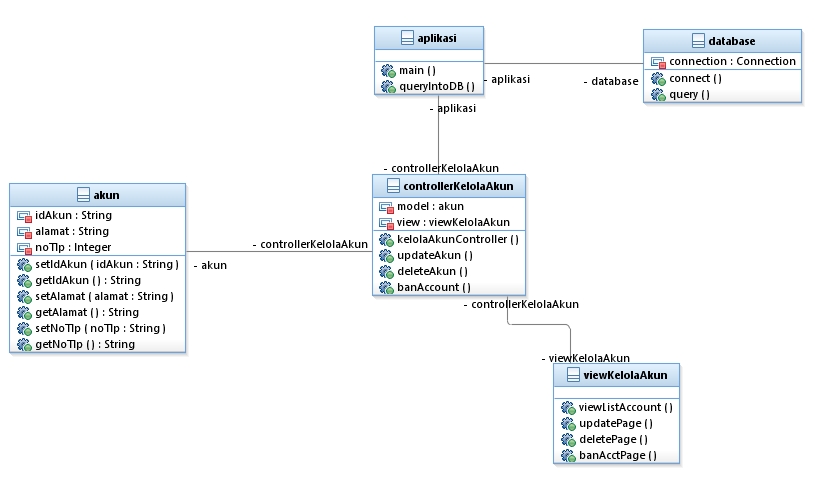
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewKelolaAkun | produk |
| 2 | controllerKelolaAkun | toko |
| 3 | akun |  |
| 4 | aplikasi |  |
| 5 | database |  |

#### Sequence Diagram



#### Diagram Kelas

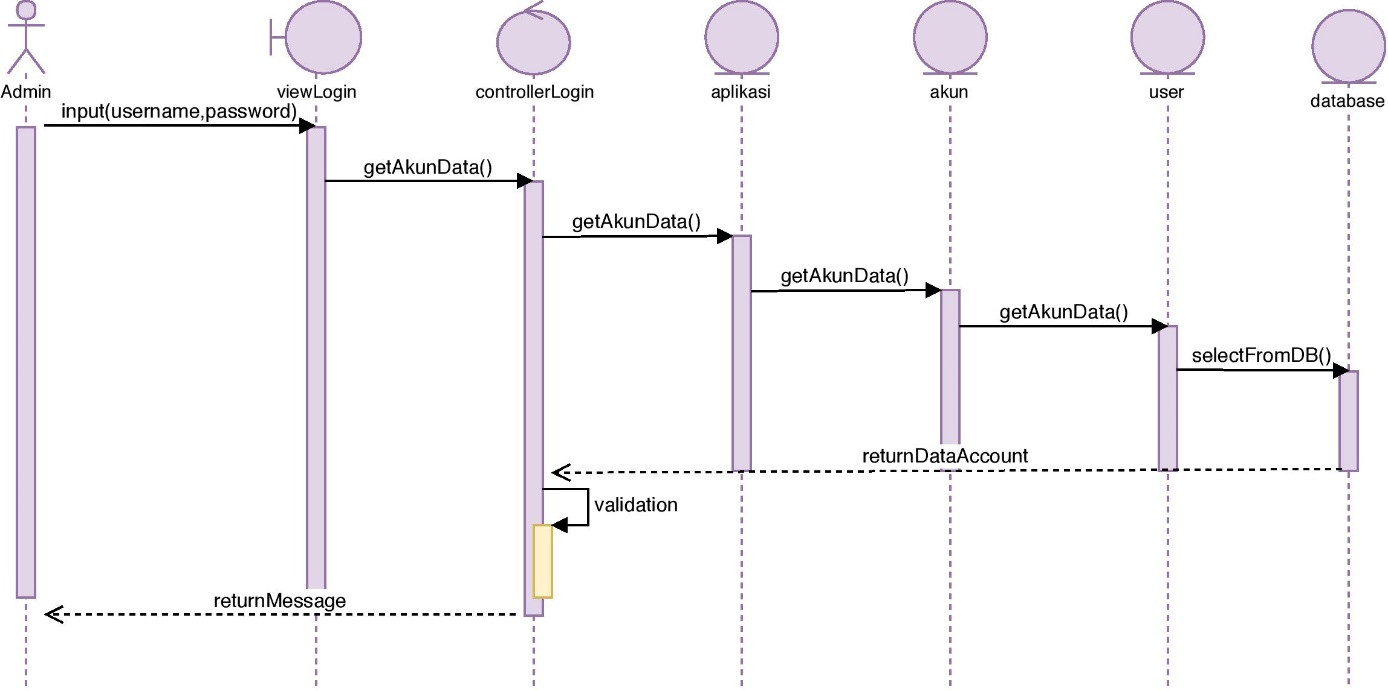


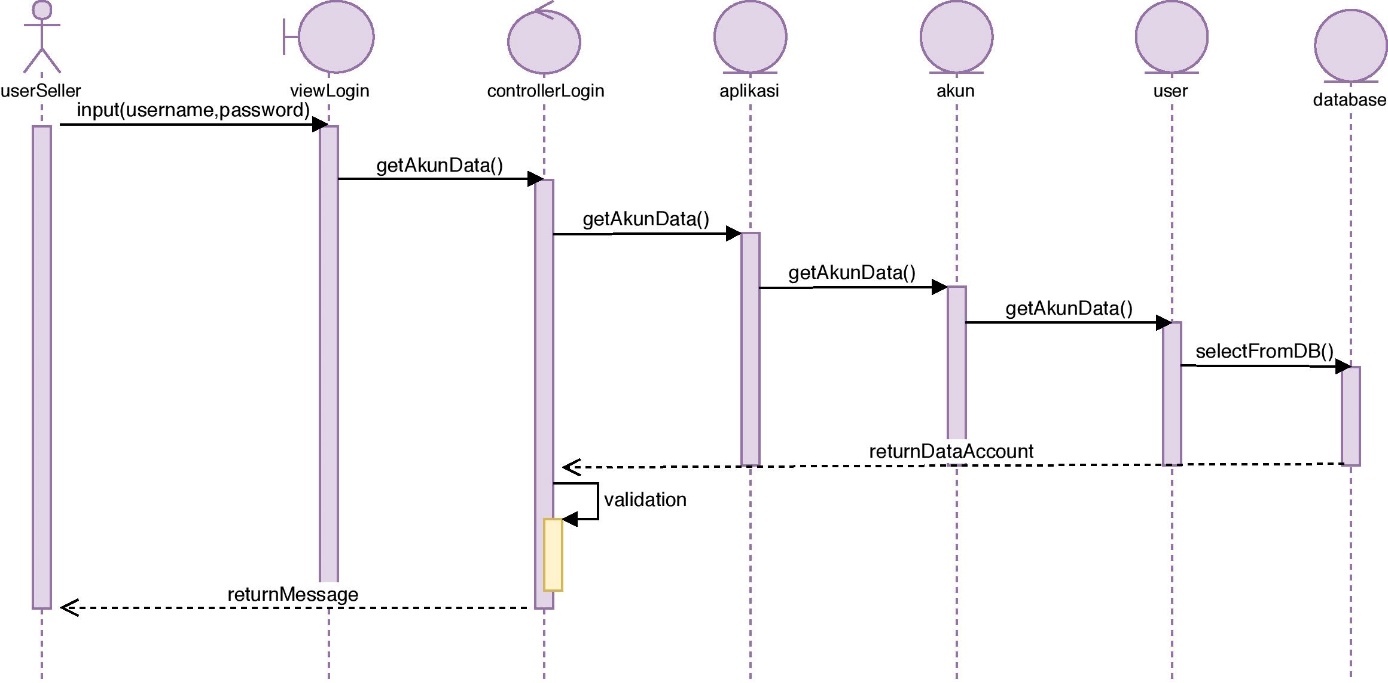
### Use Case Login

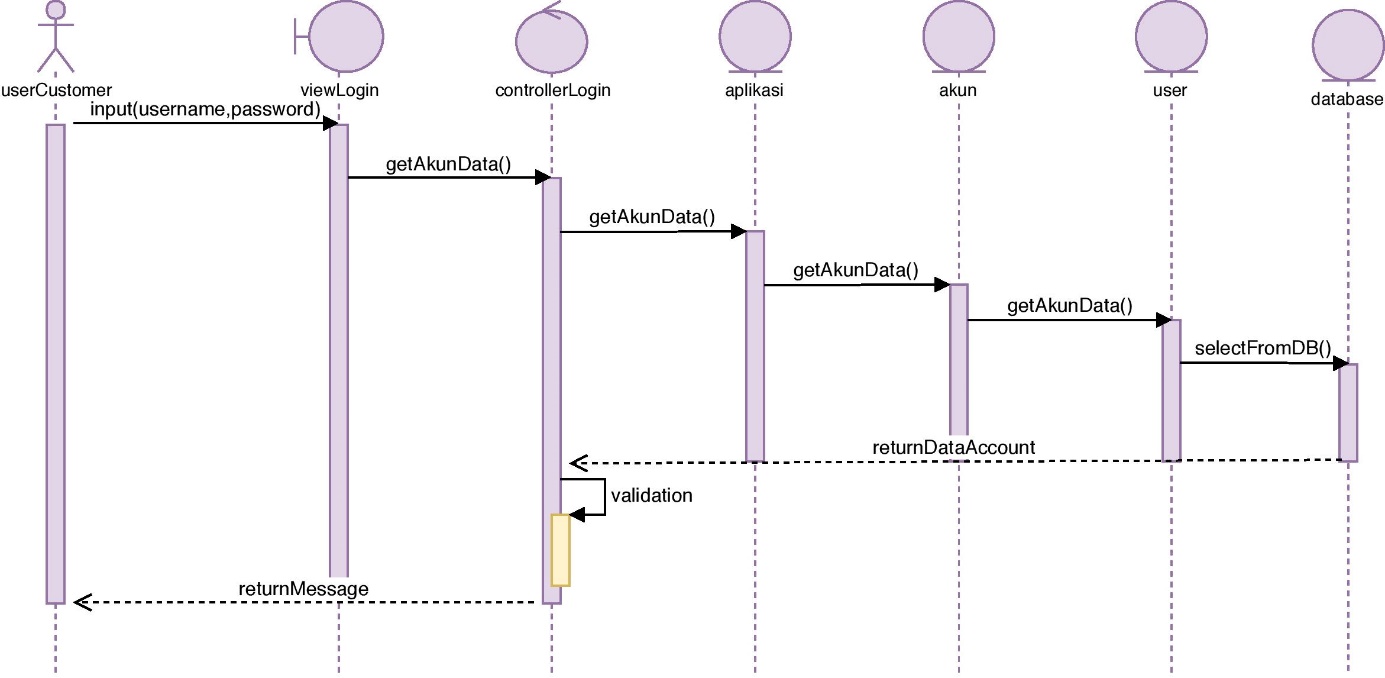
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewLogin | user |
| 2 | controllerLogin | akun |
| 3 | akun |  |
| 4 | aplikasi |  |
| 5 | database |  |
| 6 | user |  |

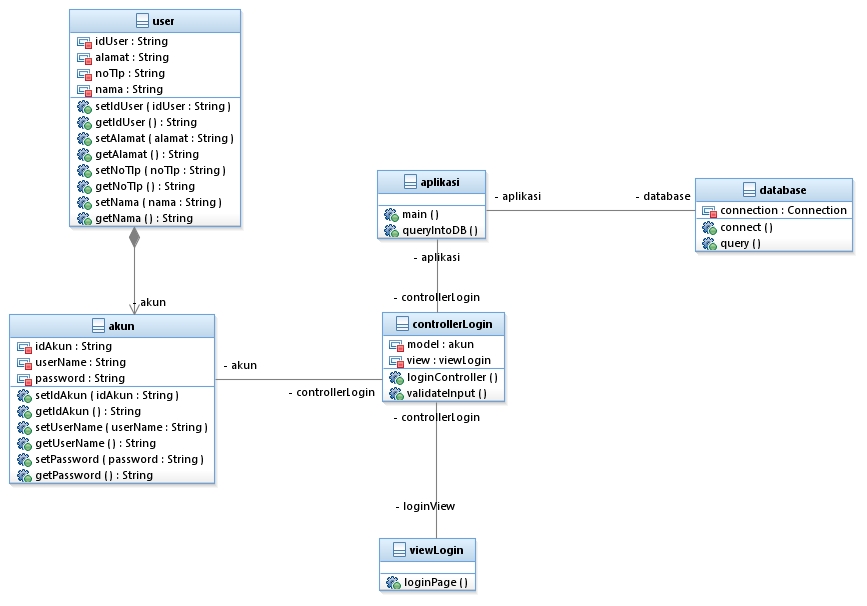
#### Sequence Diagram







#### Diagram Kelas

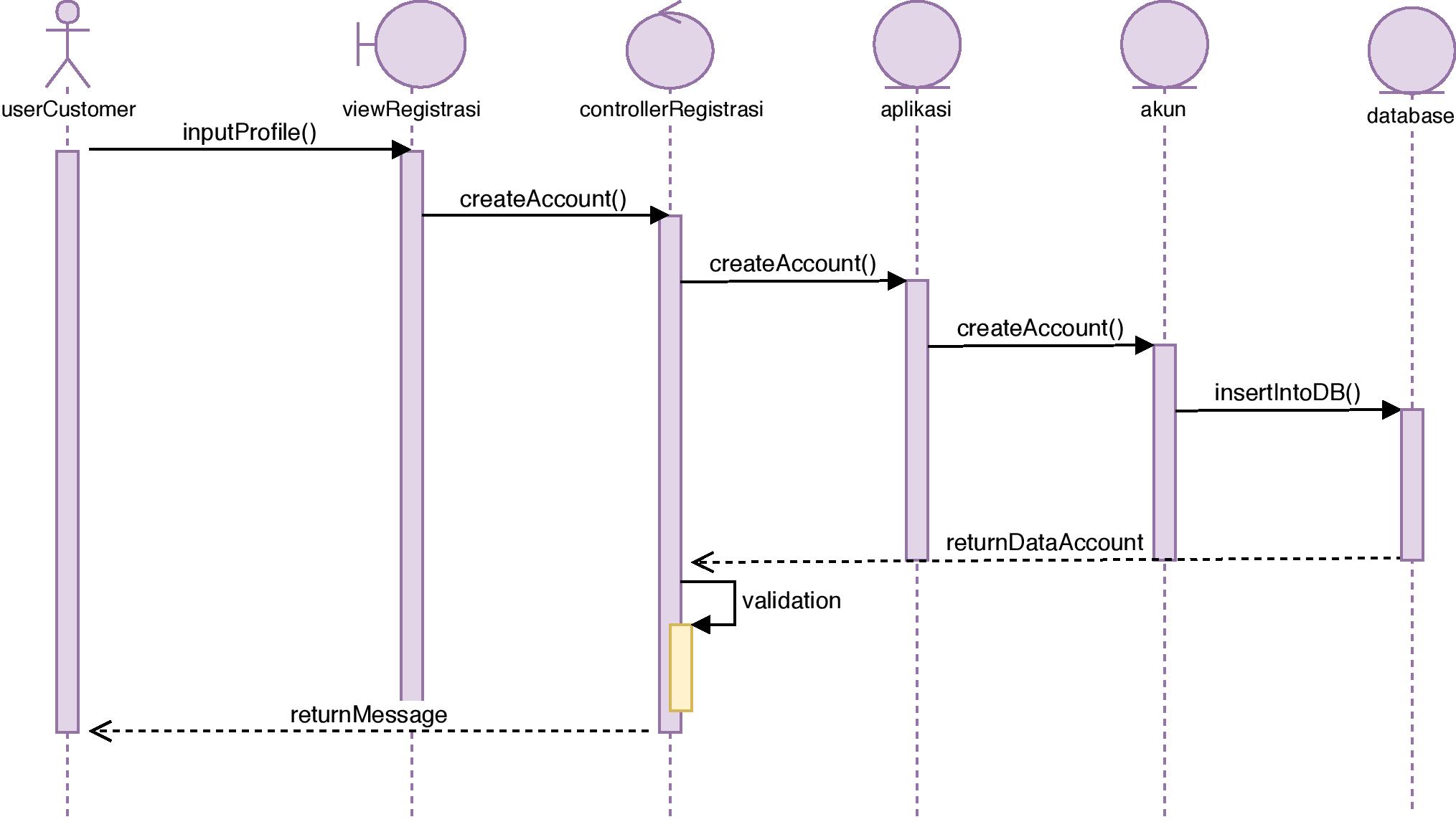


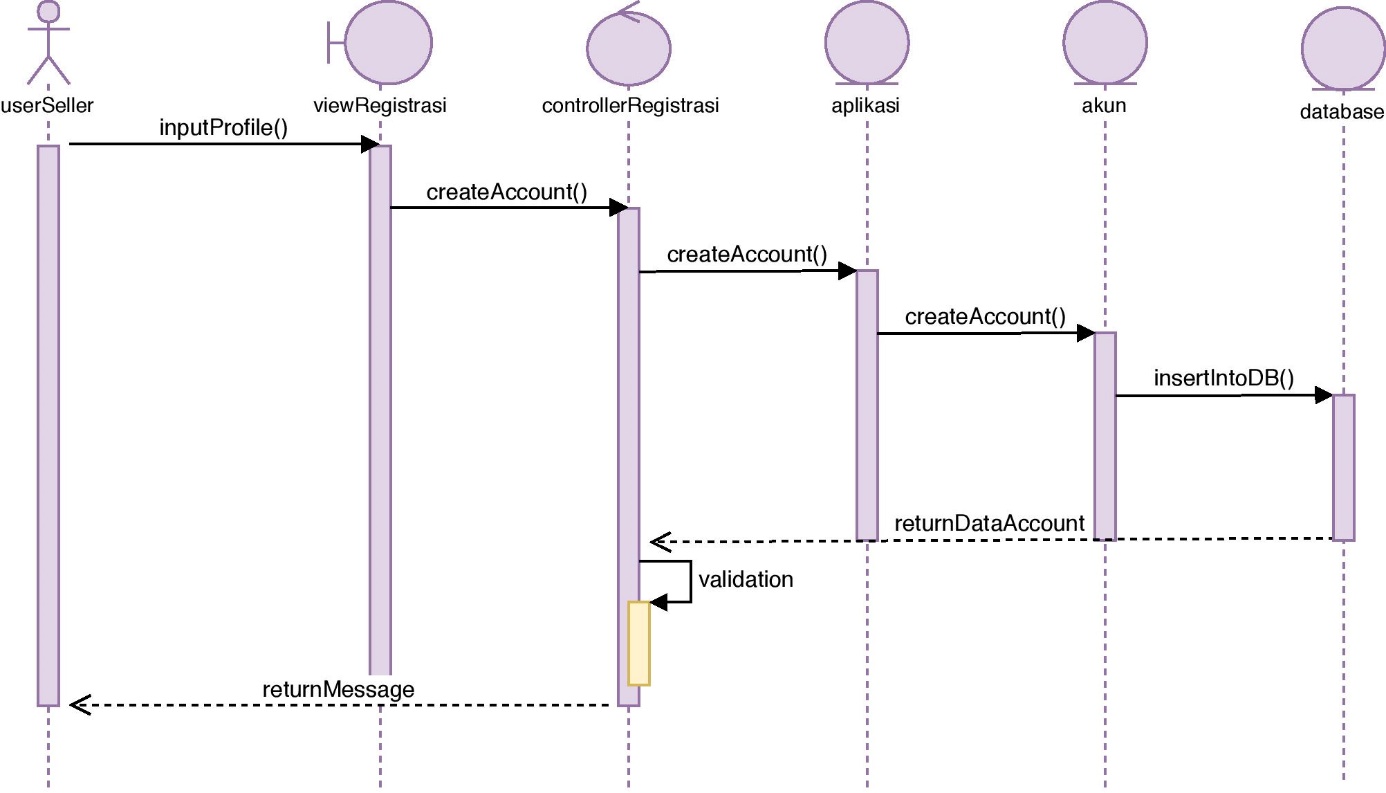
### Use Case Registrasi

#### Identifikasi Kelas

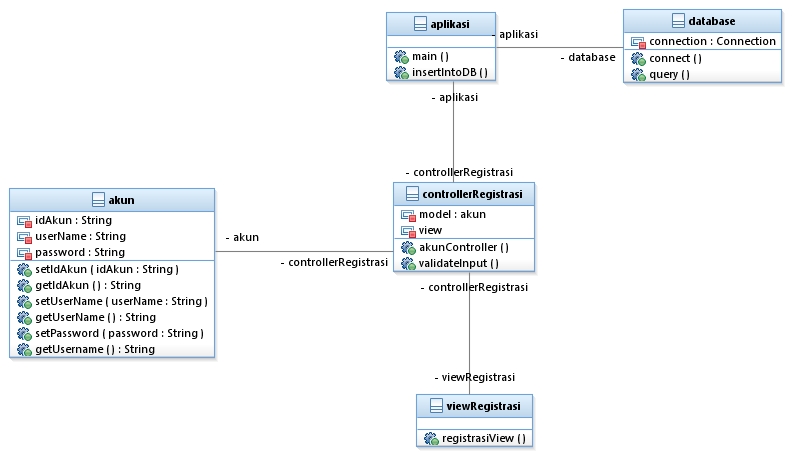
|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewRegistrasi | akun |
| 2 | controlleRegistrasi |  |
| 3 | akun |  |
| 4 | aplikasi |  |
| 5 | database |  |

#### Sequence Diagram





#### Diagram Kelas

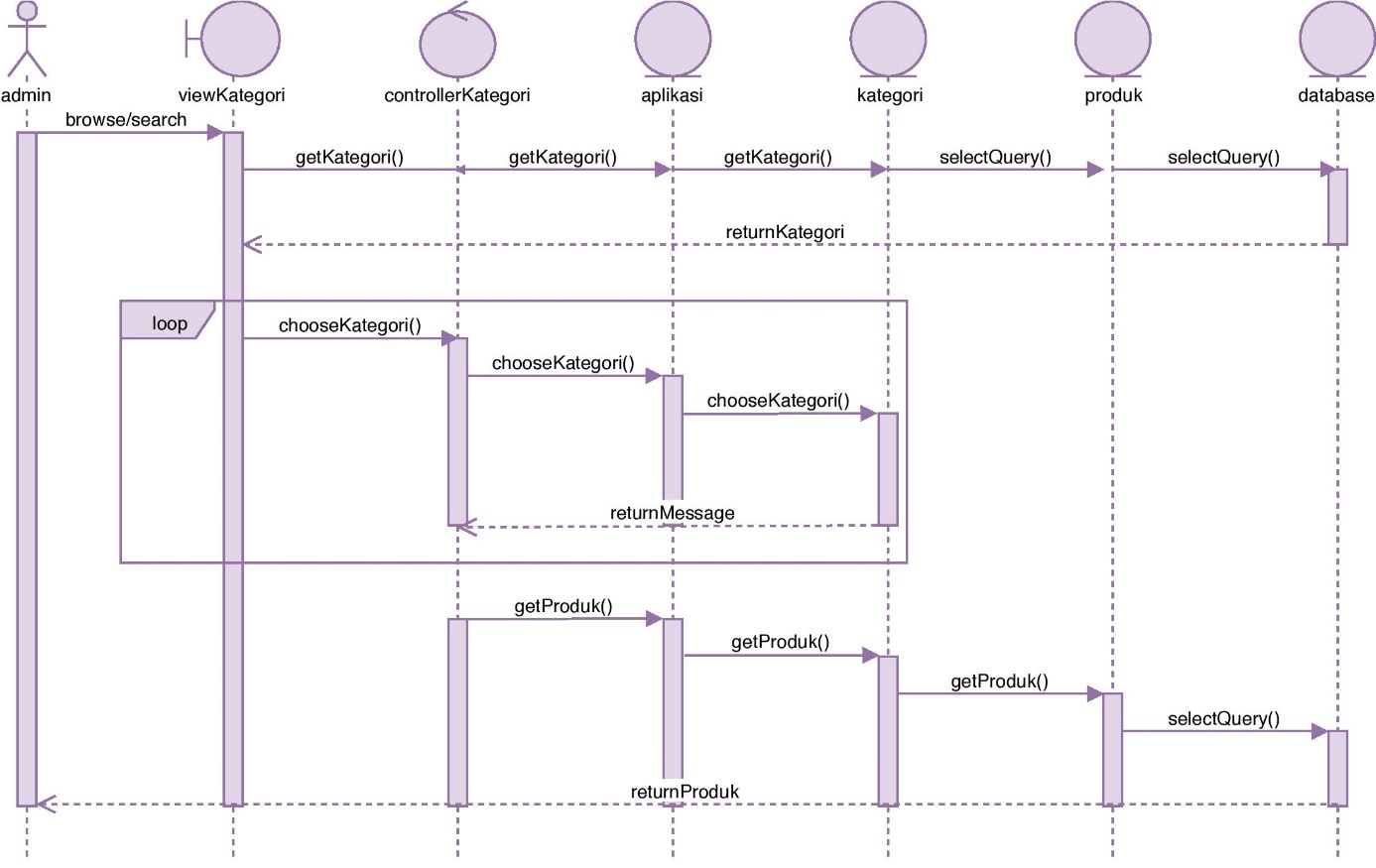


### Use Case Pilih Kategori

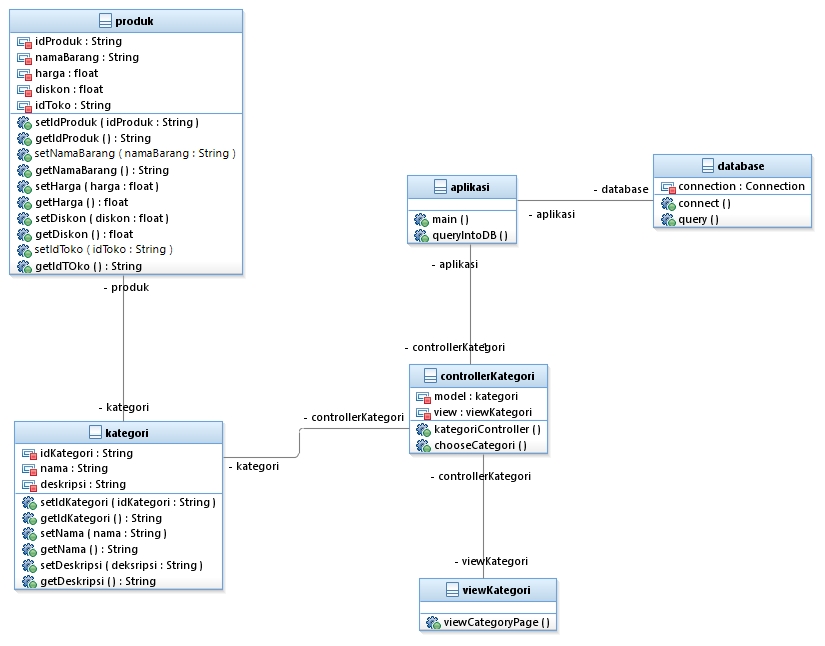
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewKatgori | kategori |
| 2 | controllerKategori | produk |
| 3 | controllerProduk |  |
| 4 | kategori |  |
| 5 | aplikasi |  |
| 6 | database |  |

#### Sequence Diagram



#### Diagram Kelas

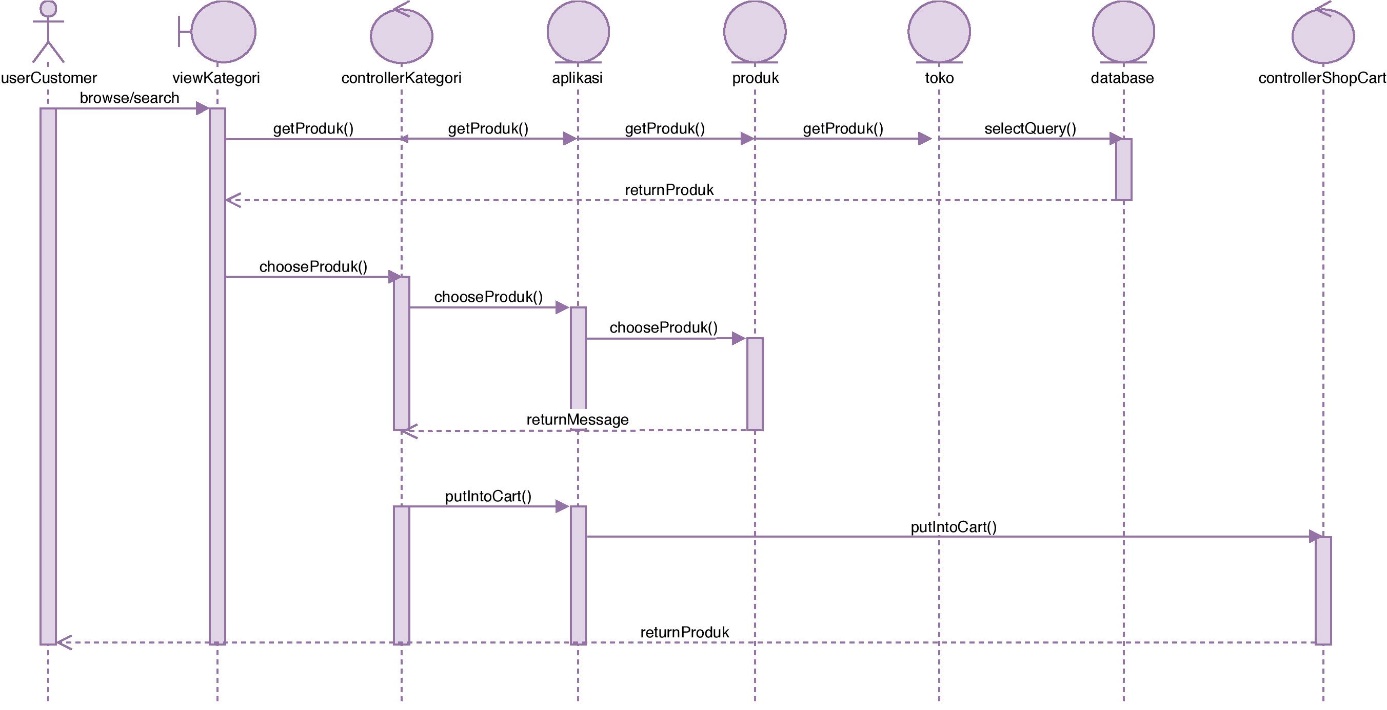


### Use Case Pilih Barang

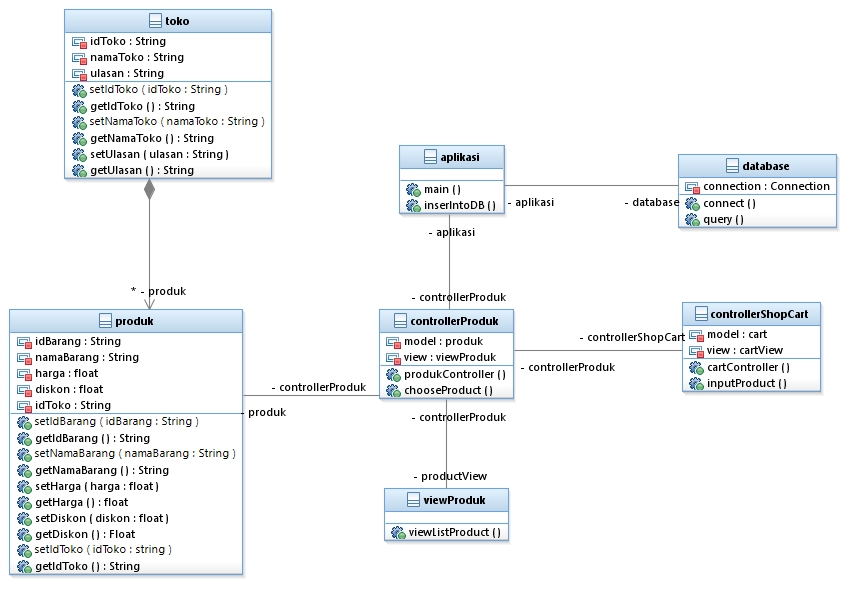
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | viewProduk | cart |
| 2 | controllerProduk | produk |
| 3 | controllerCart |  |
| 4 | produk |  |
| 5 | toko |  |
| 6 | aplikasi |  |
| 7 | database |  |

#### Sequence Diagram



#### Diagram Kelas



## Perancangan Detil Kelas

Perancangan detil kelas berisi tentang daftar seluruh kelas yang akan dibuat saat pembangunan perangkat lunak nanti. Daftar kelas-kelas beserta rinciannya terdapat pada tabel dan subbab di bawah ini.

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Nama Kelas Analisis Terkait |
| 1 | Database.sql | user.java |
| 2 | kategori.java | userSeller.java |
| 3 | produk.java | userCustomer.java |
| 4 | user.java | account.java |
| 5 | account.java | admin.java |
| 6 | toko.java | shopCart.java |
| 7 | menu.java | detailCart.java |
| 8 | aplikasi.java | produk.java |
| 9 | viewKelolaProduk.java | toko.java |
| 10 | viewKelolaToko.java | kategori.java |
| 11 | viewKelolaAccount.java |  |
| 12 | viewKategori.java |  |
| 13 | viewKelolaKategori.java |  |
| 14 | viewShopCart.java |  |
| 15 | viewProduk.java |  |
| 16 | viewLogin.java |  |
| 17 | viewRegistrasi.java |  |
| 18 | controllerKelolaProduk.java |  |
| 19 | controllerKelolaToko.java |  |
| 20 | controllerKelolaAccount.java |  |
| 21 | controllerKategori.java |  |
| 22 | controllerCart.java |  |
| 23 | controllerKelolaKategori.java |  |
| 24 | controllerProduk.java |  |
| 25 | controllerLogin.java |  |
| 26 | controllerRegistrasi.java |  |

### Kelas Akun

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

*Nama Kelas : Akun*

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| setIdAkun() | Public | Mengakses data dari variable IdAkun |
| getIdAkun() | Public | Mengisi data ke dalam variable IdAkun |
| setAlamat() | Public | Mengakses data dari variable alamat |
| getAlamat() | Public | Mengisi data ke dalam variable Alamat |
| setNoTelp() | Public | Mengakses data dari variable NoTelp |
| getNoTelp() | Public | Mengisi data ke dalam variable NoTelp |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| idAkun | Private | String |
| Alamat | Private | String |
| noTlp | Private | String |

### Kelas Toko

Nama Kelas : Toko

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| setIdToko() | Public | Mengakses data dari variable Id Toko |
| getIdToko() | Public | Mengisi data ke dalam variable Id Toko |
| setNamaToko() | Public | Mengakses data dari variable Nama Toko |
| getNamaToko() | Public | Mengisi data ke dalam variable Nama Toko |
| setUlasanToko() | Public | Mengakses data dari variable Ulasan Toko |
| getUlasanToko() | Public | Mengisi data ke dalam variable Ulasan Toko |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| idToko | Private | String |
| namaToko | Private | String |
| ulasanToko | Private | String |

### Kelas Kategori

Nama Kelas : Kategori

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| setIdKategori() | Public | Mengakses data dari variable Id Kategori |
| getIdKategori () | Public | Mengisi data ke dalam variable Id Kategori |
| setNamaKategori () | Public | Mengakses data dari variable Nama Kategori |
| getNamaKategori() | Public | Mengisi data ke dalam variable Nama Kategori |
| setDeskripsi() | Public | Mengakses data dari variable Deskripsi |
| getDeskripsi() | Public | Mengisi data ke dalam variable Deskripsi |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| idKategori | Private | String |
| namaKategori | Private | String |
| Deskripsi | Private | String |

### Kelas Aplikasi

Nama Kelas : Aplikasi

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| main() | Public | Memanggil semua kelas yang telah dibuat |
| queryDB () | Public | Buat mengakses database dan melakukan query ke database |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |
|  |  |  |

### Kelas Shop Cart

Nama Kelas : Cart

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| setIdCart() | Public | Mengakses data dari variable Id Cart |
| getIdCart() | Public | Mengisi data ke dalam variable Id Cart |
| setTanggalSimpan() | Public | Mengakses data dari variable Tanggal Simpan |
| getTanggalSimpan() | Public | Mengisi data ke dalam variable Tanggal Simpan |
| setIdBarang() | Public | Mengakses data dari variable Id Barang |
| getIdBarang() | Public | Mengisi data ke dalam variable Id Barang |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| idCart | Private | String |
| tanggalSimpan | Private | String |
| idBarang | Private | String |

### Kelas View Kelola Produk

Nama Kelas : ViewKelolaProduk

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| viewAddPage() | Public | Menampilkan user interface add produk |
| viewUpdatePage () | Public | Menampilkan user interface update produk |
| viewDeletePage() | Public | Menampilkan user interface delete produk |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |
|  |  |  |

### Kelas View Kelola Kategori

Nama Kelas : ViewKelolaKategori

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| printKategori() | public | Menampilkan list kategori |
| viewAddPage() | Public | Menampilkan user interface add produk |
| viewUpdatePage () | Public | Menampilkan user interface update produk |
| viewDeletePage() | Public | Menampilkan user interface delete produk |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |
|  |  |  |

### Kelas Controller Kelola Kategori

Nama Kelas : ControllerKelolaKategori

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| manageKategoriController() | Public | Menjembatani antara view class manage kategori controller dengan model class kategori dan mengatur aliran data |
| addKategori() | Public | Menangani proses penambahan kategori yang akan dimasukkan kedalam database |
| updateKategori() | Public | Menangani proses pembaharian kategori kedalam database |
| deleteKategori() | Public | Menangani proses penghapusan kategori dari database |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | Private | kategori |
| view | Private | ManageKategoriView |

### Kelas Controller Kelola Akun

Nama Kelas : ControllerKelolaAkun

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| kelolaAkunController() | Public | Menjembatani antara view class kelola akun controller dengan model class akun dan mengatur aliran data |
| updateAkun() | Public | Menangani proses pembaharuan akun yang akan disimpan pada database |
| deleteAkun() | Public | Menangani proses penghapusan akun dari database |
| banAkun() | Public | Menangani proses penangguhan account |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | Private | kategori |
| view | Private | Manage kategori view |

### Kelas View Kelola Account

Nama Kelas : ViewKelolaAccount

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| viewListAccount () | Public | Menampilkan halaman interface akun |
| updatePage () | Public | Menyediakan interface untuk halaman update page |
| deletePage () | Public | Menyediakan interface untuk halaman penghapusan |
| banAcctPage () | Public | Prosedure yang akan menangani proses menangguhkan akun |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |

### Kelas View Produk

Nama Kelas : ViewProduk

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| viewListProduct () | Public | Menampilkan halaman interface produk |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |

### Kelas View Kategori

Nama Kelas : View Kategori

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| viewKategoriPage () | Public | Menampilkan halaman interface kategori |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |

### Kelas Controller Login

Nama Kelas : ControllerLogin

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| loginController () | Public | Menjembatani antara class view login dengan model class account dan mengatur aliran datanya |
| validateInput () | Public | Prosedure yang menangani proses validasi hasil input dari user |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | Private | akun |
| view | Private | loginView |

### Kelas View Login

Nama Kelas : ViewLogin

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| loginPage () | Public | Menampilkan interface halaman login |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |

### Kelas User

Nama Kelas : User

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| setIdUser() | public | Mengakses data dari variabel idUser |
| getIdUser() | public | Mengisi data ke dalam variabel idUser |
| setAlamat() | public | Mengakses data dari variabel alamat |
| getAlamat() | public | Mengisi data ke dalam variabel alamat |
| setNoTlp() | public | Mengakses data dari variabel noTlp |
| getNoTlp() | public | Mengisi data ke dalam variabel noTlp |
| setNama() | public | Mengakses data dari variabel nama |
| getNama() | public | Mengisi data ke dalam variabel nama |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| idUser | private | String |
| Alamat | private | String |
| noTlp | private | String |
| nama | private | String |

### Kelas Produk

Nama kelas : Produk

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| setIdBarang() | public | Mengakses data dari variabel idBarang |
| getIdBarang() | public | Mengisi data ke dalam variabel idBarang |
| setNamaBarang() | public | Mengakses data dari variabel namaBarang |
| getNamaBarang() | public | Mengisi data ke dalam variabel namaNarang |
| setHarga() | public | Mengakses data dari variabel harga |
| getHarga() | public | Mengisi data ke dalam variabel harga |
| setDiskon() | public | Mengakses data dari variabel diskon |
| getDiskon() | public | Mengisi data ke dalam variabel diskon |
| setIdToko() | public | Mengakses data dari variabel idToko |
| getIdToko() | public | Mengisi data ke dalam variabel idToko |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| idBarang | private | String |
| namaBarang | private | String |
| harga | private | String |
| diskon | private | String |
| idToko | private | String |

### Kelas Database

Nama Kelas : Database

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| connect() | public | Membuat koneksi ke database |
| query() | public | Melakukan query kelam database |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| connection | private | Connection |

### Kelas Controller Kelola Produk

Nama Kelas : ControllerKelolaProduk

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| produkController() | public | Menjembatani antara view class dengan model class dan mengatur aliran data |
| addProduk() | public | Prosedure yang akan menangani proses penambahan produk |
| updateProduk() | public | Prosedure yang akan menangani proses pembaharuan produk |
| deleteProduk() | public | Prosedure yang akan menangani proses penghapusan produk |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | private | produk |
| view | private | kelolaProdukView |

### Kelas Controller Kelola Toko

Nama Kelas : ControllerKelolaToko

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| kelolaProdukController() | public | Menjembatani antara view class dengan model class dan mengatur aliran data |
| addToko() | public | Prosedure yang menangani proses penambahan toko |
| updateToko() | public | Prosedure yang menangani proses pembaharuan toko |
| deleteToko() | public | Prosedure yang menangani proses penghapusan toko |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | private | Toko |
| view | private | kelolaTokoView |

### Kelas View Kelola Toko

Nama Kelas : ViewKelolaToko

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| viewAddPage() | public | Merupakan procedure untuk menampilkan interface halaman penambahan |
| viewUpdatePage() | public | Prosedure untuk menampilkan interface halaman pembaharuan |
| viewDeletePage() | public | Prosedure untuk menampilkan interface halaman penghapusan |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |

### Kelas View Registrasi

Nama Kelas : ViewRegistrasi

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| registrasiView() | public | Menampilkan interface halaman registrasi. |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |

### Kelas Controller Registrasi

Nama Kelas : ControllerRegistrasi

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| akunController() | public | Menjembatani antara view class dengan model class dan sebagai tempat menyalurkan data |
| validateInput() | public | Prosedure yang akan menangani proses validasi input dari user |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | private | akun |
| view | private | registrasiView |

### Kelas Controller Kategori

Nama Kelas : ControllerKategori

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| kategoriController() | public | Menjembatani antara view class dengan model class dan sebagai tempat mengalirkan data |
| chooseCategory() | public | procedure yang menangani pemilihan kategori |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | private | kategori |
| view | private | kategoriView |

### Kelas Controller Cart

Nama Kelas : ControllerCart

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| cartController() | public | Menjembatani antara view class dengan model class dan sebagai tempat mengalirkan data |
| inputProduk() | public | procedure yang menangani pemilihan produk kedalam keranjang |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | private | cart |
| view | private | kategoriView |

### Kelas Controller produk

Nama Kelas : ControllerProduk

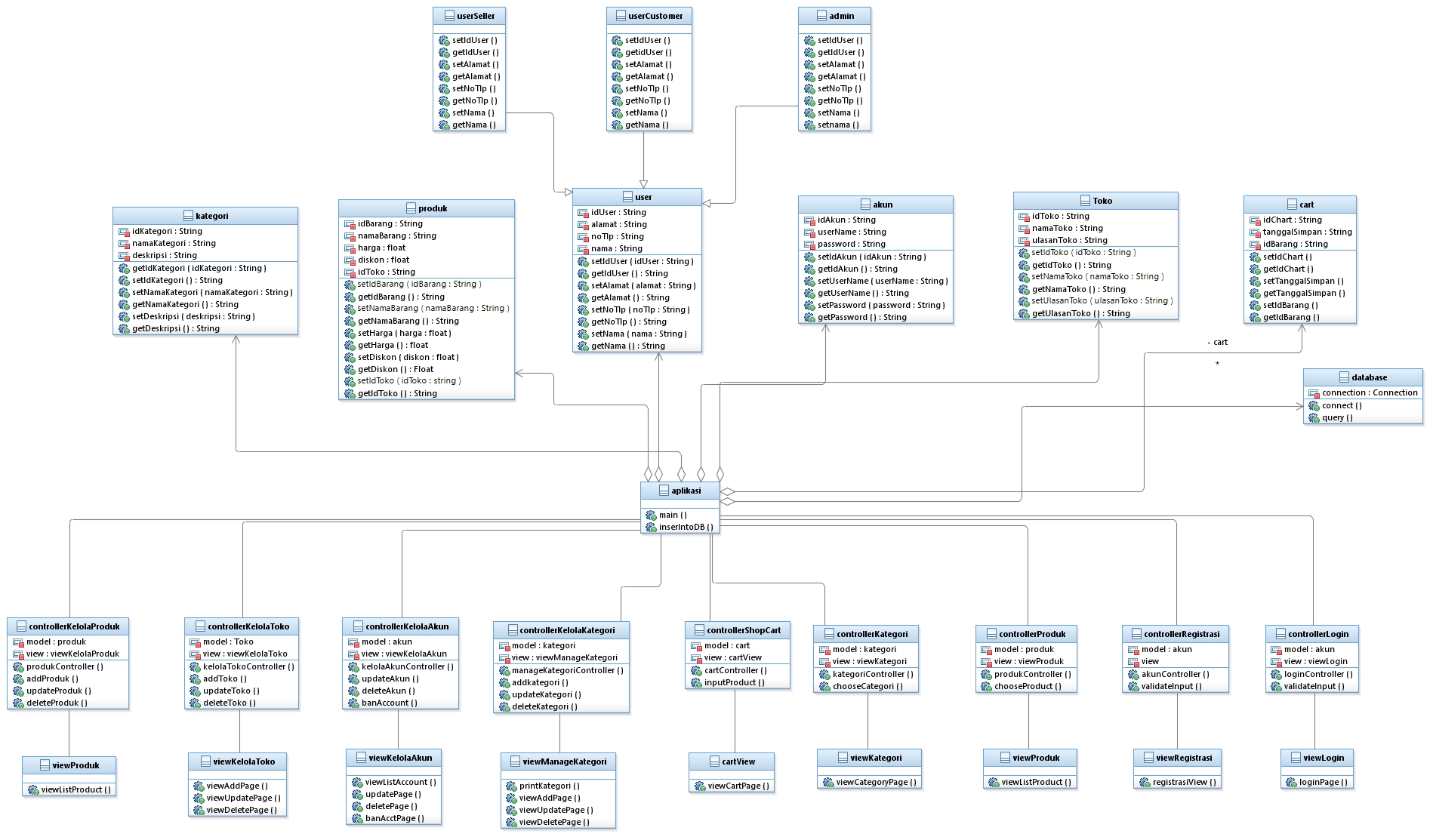
|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| kelolaKategoriController() | public | Menjembatani antara view class dengan model class dan sebagai tempat mengalirkan data |
| chooseProduk | public | Menangani psoses pemilihan barang atau produk |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| model | private | produk |
| view | private | produkView |

### Kelas View Cart

Nama Kelas : ViewCart

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| viewCartPage() | public | Procedure yang memberikan interface pada halaman cart |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
|  |  |  |
|  |  |  |

## Diagram Kelas Keseluruhan



## Algoritma/Query

Berikut ini merupakan beberapa algoritma dan query yang dianggap paling penting pada perangkat lunak yang akan kami kembangkan.

Nama Kelas : database

Nama Operasi : connect

Algoritma : (Algo-001)

private String server = "jdbc:mysql://localhost/yangpentingdiskonDB";

private String dbuser = "root";

private String dbpassword = "";

protected Statement statement;

protected Connection connection;

public void connectDB() {

try {

connection = DriverManager.getConnection(server, dbuser, dbpassword);

statement = connection.createStatement();

} catch (SQLException ex) {

ex.printStackTrace();

}

}

Nama Kelas : controllerKelolaProduk

Nama Operasi : addProduk

Algoritma : (Algo-002)

public void addProduk(Produk P){

try{

database db = new database();

String s = “INSERT INTO table\_produk (‘”+P.getIdBarang()+” ’, ‘”+P.getNamaBarang()+”’,”+ Float.toString(P.getHarga())+”’,”+ Float.toString(P.getDiskon())+”’,”+ P .getIdToko()+”’)”;

db.query(s);

OptionPane.showMessageDialog(null, " Add Data Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerKelolaProduk

Nama Operasi : updateProduk

Algoritma : (Algo-003)

public void updateProduk(Produk P){

try{

database db = new database();

String s = “Update table\_produk SET

idBarang = ‘”+P.getIdBarang()+

“’,namaBarang = ‘”+P.getNamaBarang()+

“’,harga = ‘”+ Float.toString(P.getHarga())+

“’,diskon = ‘”+Float.toString(P.getDiskon())+

“’,idToko = ‘”+ P .getIdToko()+”’)”;

db.query(s);

JOptionPane.showMessageDialog(null, "Update Data Successfull");

}catch(SQLException ex){

JOptionPane.showMessageDialog(null,"Update Data Failed");

}

}

Nama Kelas : controllerKelolaProduk

Nama Operasi : deleteProduk

Algoritma : (Algo-004)

public void deleteProduk(Produk P){

try{

database db = new database();

String s = “DELETE FROM table\_produk WHERE idBarang = ‘”+P.getIdBarang();

db.query(s);

OptionPane.showMessageDialog(null, "Delete Data Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerKelolaToko

Nama Operasi : addToko

Algoritma : (Algo-005)

public void addToko(Toko T){

try{

database db = new database();

String s = “INSERT INTO table\_toko (‘”+T.getIdToko()+” ’, ‘”+T.getNamaToko()+”’,”+ T.getUlasanToko()+”’)”;

db.query(s);

OptionPane.showMessageDialog(null, "Add Data Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerKelolaToko

Nama Operasi : updateToko

Algoritma : (Algo-006)

public void updateToko(Toko T){

try{

database db = new database();

String s = “Update table\_produk SET

idToko = ‘”+T.getIdToko()+

“’,namaToko = ‘”+ T.getNamaToko()+

“’,ulasanToko = ‘”+T.getUlasanToko())”;

db.query(s);

JOptionPane.showMessageDialog(null, "Update Data Successfull");

}catch(SQLException ex){

JOptionPane.showMessageDialog(null,"Update Data Failed");

}

}

Nama Kelas : controllerKelolaToko

Nama Operasi : hapusToko

Algoritma : (Algo-007)

public void deleteToko(Toko T){

try{

database db = new database();

String s = “DELETE FROM table\_toko WHERE idToko = ‘”+T.getIdToko();

db.query(s);

OptionPane.showMessageDialog(null, "Delete Data Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerKelolaKategori

Nama Operasi : addKategori

Algoritma : (Algo-008)

public void addKategori(Kategori K){

try{

database db = new database();

String s = “INSERT INTO table\_kategori (‘”+K.getIdKategori()+” ’, ‘”+K.getNamaKategori()+”’,”+ K.getDeskripsi()+”’)”;

db.query(s);

OptionPane.showMessageDialog(null, "Add Data Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerKelolaKategori

Nama Operasi : updateKategori

Algoritma : (Algo-009)

public void updateKategori (Kategori K){

try{

database db = new database();

String s = “Update table\_kategori SET

idkategori = ‘”+K.getIdkategori()+

“’,namaKategori= ‘”+ K.getNamaKategori()+

“’,deskripsi = ‘”+K.getDeskripsi())”;

db.query(s);

JOptionPane.showMessageDialog(null, "Update Data Successfull");

}catch(SQLException ex){

JOptionPane.showMessageDialog(null,"Update Data Failed");

}

}

Nama Kelas : controllerKelolaKategori

Nama Operasi : deleteKategori

Algoritma : (Algo-010)

public void deleteKategori (kategori K){

try{

database db = new database();

String s = “DELETE FROM table\_Kategori WHERE idKategori = ‘”+K.getIdKategori();

db.query(s);

OptionPane.showMessageDialog(null, "Delete Data Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerLogin

Nama Operasi : login

Algoritma : (Algo-011)

public void login (Akun A){

try{

database db = new database();

String s = “SELECT username, password from table\_akun WHERE username like ‘”+A.getUserName()+”’and password like ‘”,’”+A.getPassword()’”;

db.query(s);

OptionPane.showMessageDialog(null, " The username and password has successfully verfied!!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerRegistrasi

Nama Operasi : registrasi

Algoritma : (Algo-0012)

public void addKategori(Registrasi R){

try{

database db = new database();

String s = “INSERT INTO table\_akun (‘”+R.getIdUser()+” ’, ‘”+R.getAlamat()+”’,”+ R.getNoTlp()+”’ ,”+ R.getNama()+”’)”;

db.query(s);

OptionPane.showMessageDialog(null, "Registration Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

Nama Kelas : controllerCart

Nama Operasi : inputProduct

Algoritma : (Algo-0013)

public void inputProduct (Produk P){

try{

database db = new database();

String s = “INSERT INTO table\_cart (‘”+P.getIdCart()+” ’, ‘”+P.getTanggalSimpan()+”’,”+ P.getIdBarang()+”’)”;

db.query(s);

OptionPane.showMessageDialog(null, "Insert Into Cart Success!");

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

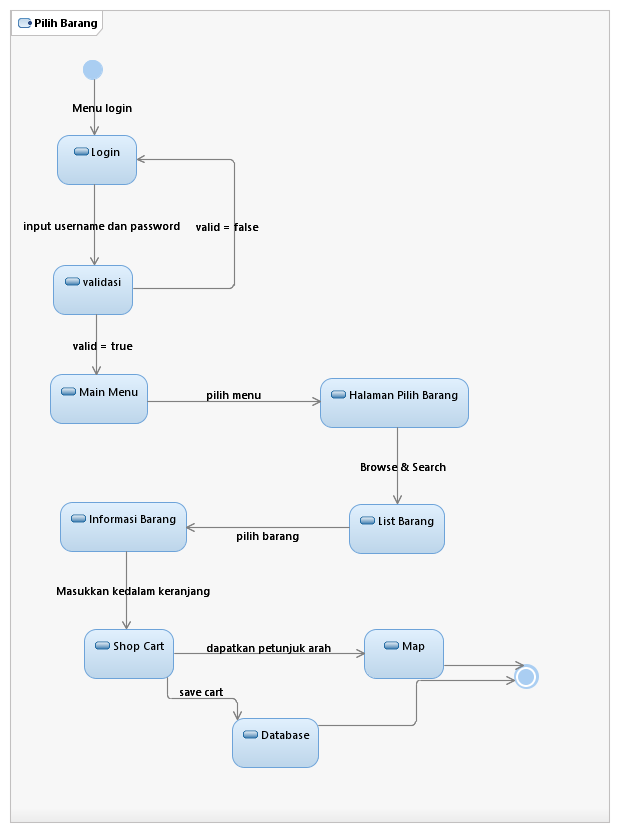
}

}

Query :

|  |  |  |
| --- | --- | --- |
| No Query | Query | Keterangan |
| Q-001 | INSERT INTO table\_produk (P.getIdBarang(), P.getNamaBarang(), Float.toString(P.getHarga()), Float.toStreng(P.getDiskon()), P.getIdToko()); | Melakukan penambahan kepada table produk |
| Q-002 | UPDATE table\_produk SET  idBarang = P.getIdBarang,  namaBarang = P.getNamaBarang(),  harga = Float.toString(P.getHarga()),  diskon = Float.toString(P.getDiskon()),  idToko = P .getIdToko()); | Melakukan pembaharuan dari table produk |
| Q-003 | DELETE FROM table\_produk WHERE idBarang = P.getIdBarang(); | Melakukan penghapusan pada isi dari table produk |
| Q-004 | INSERT INTO table\_toko (T.getIdToko(), T.getNamaToko(), T.getUlasanToko()); | Melakukan penambahan toko pada table toko di database |
| Q-005 | Update table\_produk SET  idToko = T.getIdToko(),  namaToko = T.getNamaToko(),  ulasanToko = T.getUlasanToko()); | Melakukan pembaharuan pada table produk di database |
| Q-006 | DELETE FROM table\_toko WHERE idToko = T.getIdToko(); | Melakukan penghapusan data yang ada pada table toko di database |
| Q-007 | INSERT INTO table\_kategori (K.getIdKategori(), K.getNamaKategori(), K.getDeskripsi()); | Melakukan penambahan kategori pada table toko di database |
| Q-008 | Update table\_kategori SET  idkategori = K.getIdkategori,  namaKategori= K.getNamaKategori(),  deskripsi = K.getDeskripsi()); | Melakukan pembaharuan pada isi dari table kategori di database |
| Q-009 | DELETE FROM table\_Kategori WHERE idKategori = K.getIdKategori(); | Melakukan penghapusan data dari table kategori di database |
| Q-010 | SELECT username, password from table\_akun WHERE username like A.getUserName() and password like A.getPassword(); | Melakukan pengambilan data username dan password dari database untuk digunakan dalam validasi ketika melakukan login |
| Q-011 | INSERT INTO table\_akun (R.getIdUser(), R.getAlamat(), R.getNoTlp(), R.getNama()); | Melakukan penambahan data akun pada table akun di database |
| Q-012 | INSERT INTO table\_cart (P.getIdCart(), P.getTanggalSimpan(), P.getIdBarang()); | Melakukan penambahan pada table cart di database. |

## Diagram Statechart



## Perancangan Antarmuka

Antarmuka : Login

*Gambar 3.1 Halaman Login*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFUname | TF Box | Username | Isi text box sesuai dengan data username yang disimpan dalam database. |
|  | TFPass | TF Box | Password | Isi text box sesuai dengan data password yang disimpan dalam database. |
|  | BtnGo | Button | GO | Jika di klik, maka akan dilakukan pencocokkan dari username dan password yang dimasukkan. Jika cocok, maka akan masuk ke halaman utama namun jika tidak cocok maka akan muncul notifikasi “Username atau Password yang Anda masukkan salah” |
|  | BtnNtMbr | Button | Not Member Yet | Jika di klik, maka akan masuk ke halaman registrasi untuk mendaftar sebagai akun. |

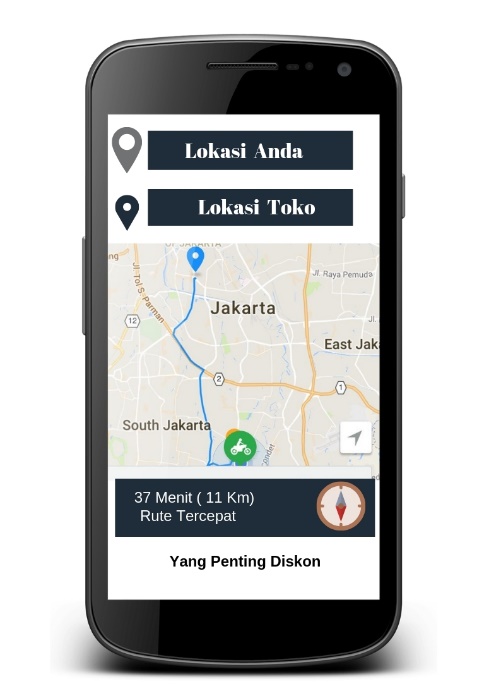
Antarmuka : Registrasi



*Gambar 3.2 Halaman Registrasi*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFUname | TF Box | Username | Isi text box sesuai dengan username yang Anda inginkan dan akan disimpan dalam database. |
|  | TFFname | TF Box | Fullname | Isi text box sesuai dengan full name Anda dan akan disimpan dalam database. |
|  | TFTelp | TF Box | Telephone Number | Isi text box sesuai dengan nomor telepon Anda dan akan disimpan dalam database. |
|  | TFAddr | TF Box | Address | Isi text box sesuai dengan alamat Anda dan akan disimpan dalam database. |
|  | TFEmail | TF Box | Email Address | Isi text box sesuai dengan alamat email Anda dan akan disimpan dalam database. |
|  | TFPass | TF Box | Password | Isi text box sesuai dengan password Anda inginkan dan akan disimpan dalam database. |
|  | BtnAgree | Button | I Agree All Terms of Service | Jika di klik, maka akan menandakan bahwa Anda menyetujui semua aturan dan ketentuan dari aplikasi ini. |
|  | BtnSbmt | Button | Submit | Jika di klik, maka akan dilakukan validasi dari data yang sudah dimasukkan. Jika valid, maka data akan tersimpan dalam database dan bisa melakukan proses login. |

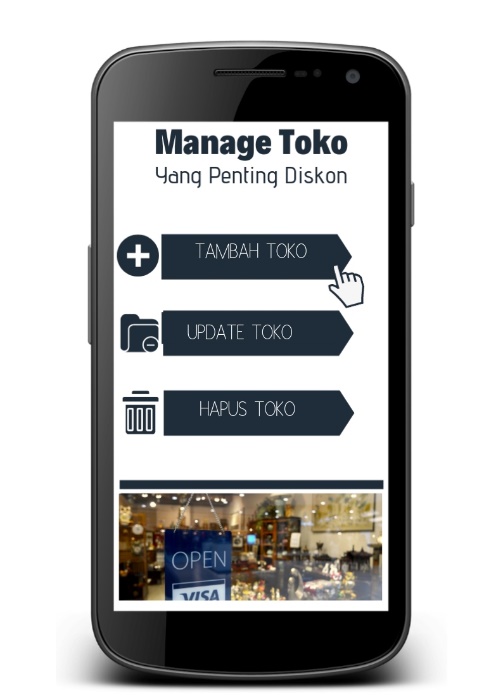
Antarmuka : gMap



*Gambar 3.3 Halaman API Google Maps*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFLok | TF Box | Lokasi Anda | Isi text box sesuai dengan lokasi asal. |
|  | TFLokTk | TF Box | Lokasi Toko | Isi text box sesuai dengan lokasi tujuan yang ingin dikunjungi. |

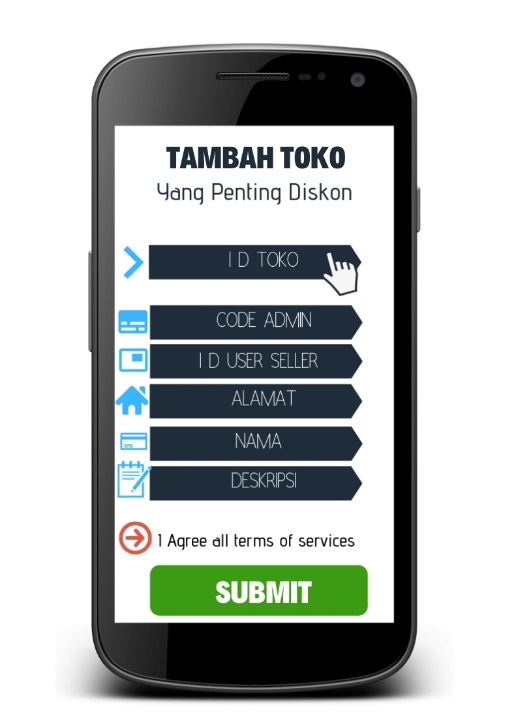
Antarmuka : Kelola Toko



*Gambar 3.4 Halaman Manage Toko*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | BtnTmbhTk | Button | Tambah Toko | Jika di klik, maka akan masuk ke halaman tambah toko. |
|  | BtnUpdtTk | Button | Update Toko | Jika di klik, maka akan masuk ke halaman update toko. |
|  | BtnHpsTk | Button | Hapus Toko | Jika di klik, maka akan masuk ke halaman hapus toko. |

Antarmuka : tambah toko



*Gambar 3.5 Halaman Tambah Toko*

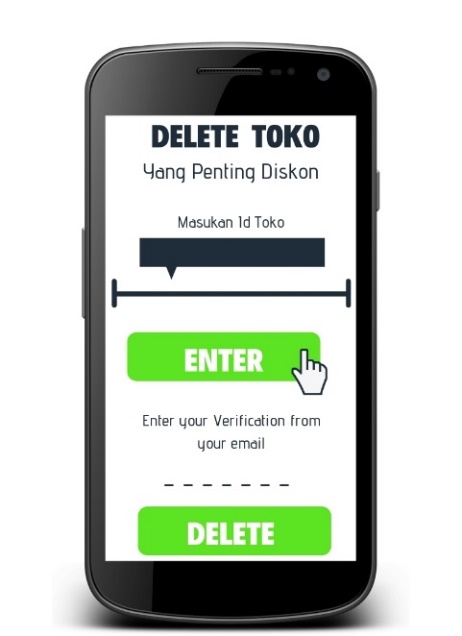
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFIdToko | TF Box | ID Toko | Isi text box sesuai dengan ID Toko yang disimpan dalam database Toko. |
|  | TFFKdAdmn | TF Box | Kode Admin | Isi text box sesuai dengan kode admin yang disimpan dalam database admin. |
|  | TFIdUsrSlr | TF Box | ID User Seller | Isi text box sesuai dengan ID User Seller Anda yang disimpan dalam database user seller. |
|  | TFAlmt | TF Box | Alamat | Isi text box sesuai dengan alamat yang disimpan dalam database. |
|  | TFNama | TF Box | Nama | Isi text box sesuai dengan nama barang yang diinginkan dan akan disimpan dalam database. |
|  | TFDesk | TF Box | Deskripsi | Isi text box sesuai dengan deskripsi barang yang diinginkan dan akan disimpan dalam database. |
|  | BtnAgree | Button | I Agree All Terms of Service | Jika di klik, maka akan menandakan bahwa Anda menyetujui semua aturan dan ketentuan dari aplikasi ini. |
|  | BtnSbmt | Button | Submit | Jika di klik, maka akan dilakukan validasi dari data yang sudah dimasukkan. Jika valid, maka data akan tersimpan dalam database. |

Antarmuka : cart shop



*Gambar 3.6 Halaman Cart*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | BtnEntr | Button | Enter | Jika di klik, maka barang yang dipilih akan ditambahkan ke dalam tabel cart. |

Antarmuka : delete toko

*Gambar 3.7 Halaman Hapus Toko*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFIdTk | TF Box | Id Toko | Isi text box sesuai dengan ID Toko yang disimpan dalam database Toko. |
|  | BtnEntr | Button | Enter | Jika di klik, maka akan dilakukan validasi Id Toko. Jika sesuai, maka akan dikirimkan kode verifikasi ke email. |
|  | BtnDlt | Button | Delete | Jika di klik, maka Toko akan terhapus dari database. |

Antarmuka : kategori barang



*Gambar 3.8 Halaman Pilih Kategori*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFSrch | TF Box | Search | Isi text box sesuai nama kategori yang ingin dicari. |
|  | BtnElktr | Button | Elektronik | Jika di klik, maka akan masuk ke daftar barang berdasarkan kategori elektronik. |
|  | BtnKcntkn | Button | Kecantikan | Jika di klik, maka akan masuk ke daftar barang berdasarkan kategori kecantikan. |
|  | BtnMkn | Button | Minuman Makanan | Jika di klik, maka akan masuk ke daftar barang berdasarkan kategori minuman makanan. |
|  | BtnPkn | Button | Pakaian | Jika di klik, maka akan masuk ke daftar barang berdasarkan kategori pakaian. |
|  | BtnAks | Button | Aksesoris | Jika di klik, maka akan masuk ke daftar barang berdasarkan kategori aksesoris. |
|  | BtnSpt | Button | Sepatu | Jika di klik, maka akan masuk ke daftar barang berdasarkan kategori sepatu. |

Antarmuka : pilih barang

*Gambar 3.9 Halaman Pilih Produk*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | BtnBrg | Button | Barang | Semua barang yang ada pada list diatas dapat dipilih, disini kami memilih salah satu yaitu misalkan tas yang nantinya setelah di klik maka akan masuk ke halaman cart. |

Antarmuka : Update Toko

*Gambar 3.10 Halaman Update Toko*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFIdTk | TF Box | Id Toko | Isi text box sesuai dengan ID Toko yang tersimpan dalam database Toko. |
|  | TFNmTk | TF Box | Nama Toko | Isi text box sesuai dengan nama toko yang ingin di ubah, dan akan disimpan dalam database. |
|  | TFAlmtTk | TF Box | Alamat Toko | Isi text box sesuai dengan alamat toko yang ingin di ubah, dan akan disimpan dalam database. |
|  | TFDesk | TF Box | Deskripsi | Isi text box sesuai dengan deskripsi toko yang ingin di ubah, dan akan disimpan dalam database. |
|  | BtnUpdt | Button | Update | Jika di klik, maka akan dilakukan validasi dari data yang diupdate. Jika sudah sesuai, maka data baru yang di update akan disimpan dalam database. |

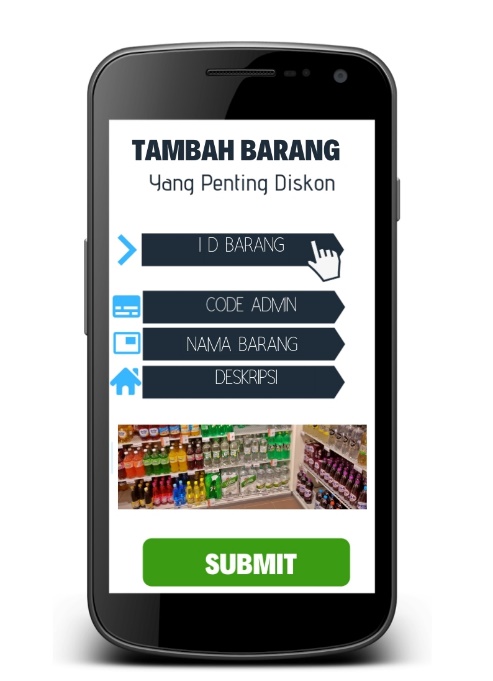
Antarmuka : Kelola Barang



*Gambar 3.11 Halaman Kelola Barang*

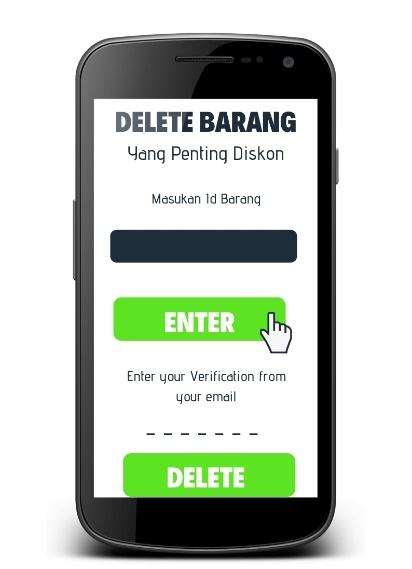
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | BtnTmbhBrg | Button | Tambah Barang | Jika di klik, maka akan masuk ke halaman tambah barang. |
|  | BtnUpdtBrg | Button | Update Barang | Jika di klik, maka akan masuk ke halaman update barang. |
|  | BtnHpsBrg | Button | Hapus Barang | Jika di klik, maka akan masuk ke halaman hapus barang. |

Antarmuka : Tambah Barang



*Gambar 3.12 Halaman Tambah Barang*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | BtnSubmit | Button | Enter | Jika di klik, maka barang yang dipilih akan ditambahkan ke dalam tabel cart. |

Antarmuka : Delete Barang

*Gambar 3.13 Halaman Delete Barang*

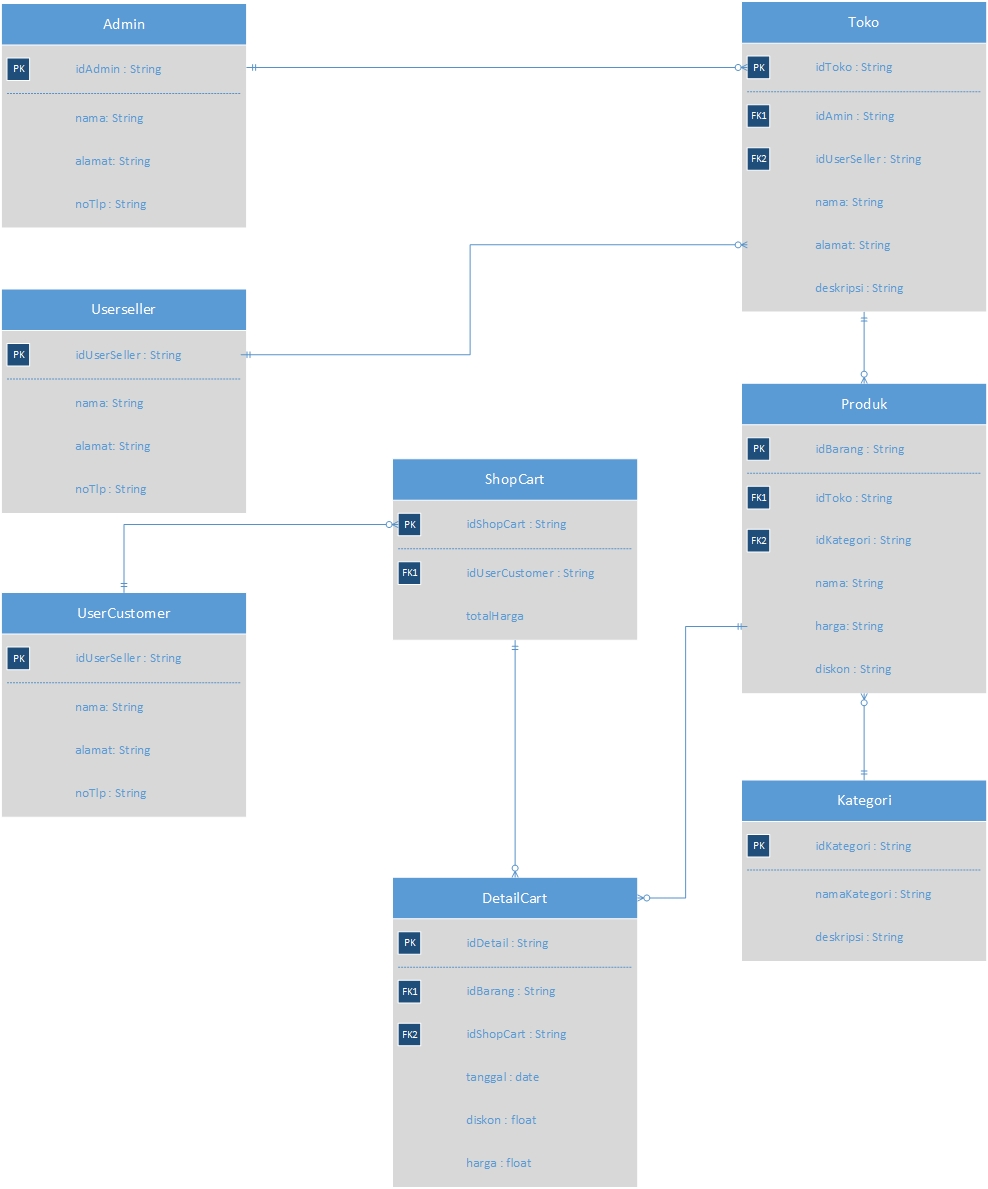
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFIdBrg | TF Box | Id Barang | Isi text box sesuai dengan ID Barang yang disimpan dalam database Barang. |
|  | BtnEntr | Button | Enter | Jika di klik, maka akan dilakukan validasi Id Barang. Jika sesuai, maka akan dikirimkan kode verifikasi ke email. |
|  | BtnDlt | Button | Delete | Jika di klik, maka barang akan terhapus dari database. |

Antarmuka : Update Barang

*Gambar 3.14 Halaman Update Barang*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Antarmuka** | **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
|  | TFIdBrg | TF Box | Id Barang | Isi text box sesuai dengan ID Barang yang tersimpan dalam database Barang. |
|  | TFNmBrg | TF Box | Nama Barang | Isi text box sesuai dengan nama barang yang ingin di ubah, dan akan disimpan dalam database. |
|  | TFHrgBrg | TF Box | Harga Barang | Isi text box sesuai dengan harga barang yang ingin di ubah, dan akan disimpan dalam database. |
|  | TFDesk | TF Box | Deskripsi | Isi text box sesuai dengan deskripsi barang yang ingin di ubah, dan akan disimpan dalam database. |
|  | BtnUpdt | Button | Update | Jika di klik, maka akan dilakukan validasi dari data yang diupdate. Jika sudah sesuai, maka data baru yang di update akan disimpan dalam database. |

## Perancangan Representasi Persistensi Kelas

**

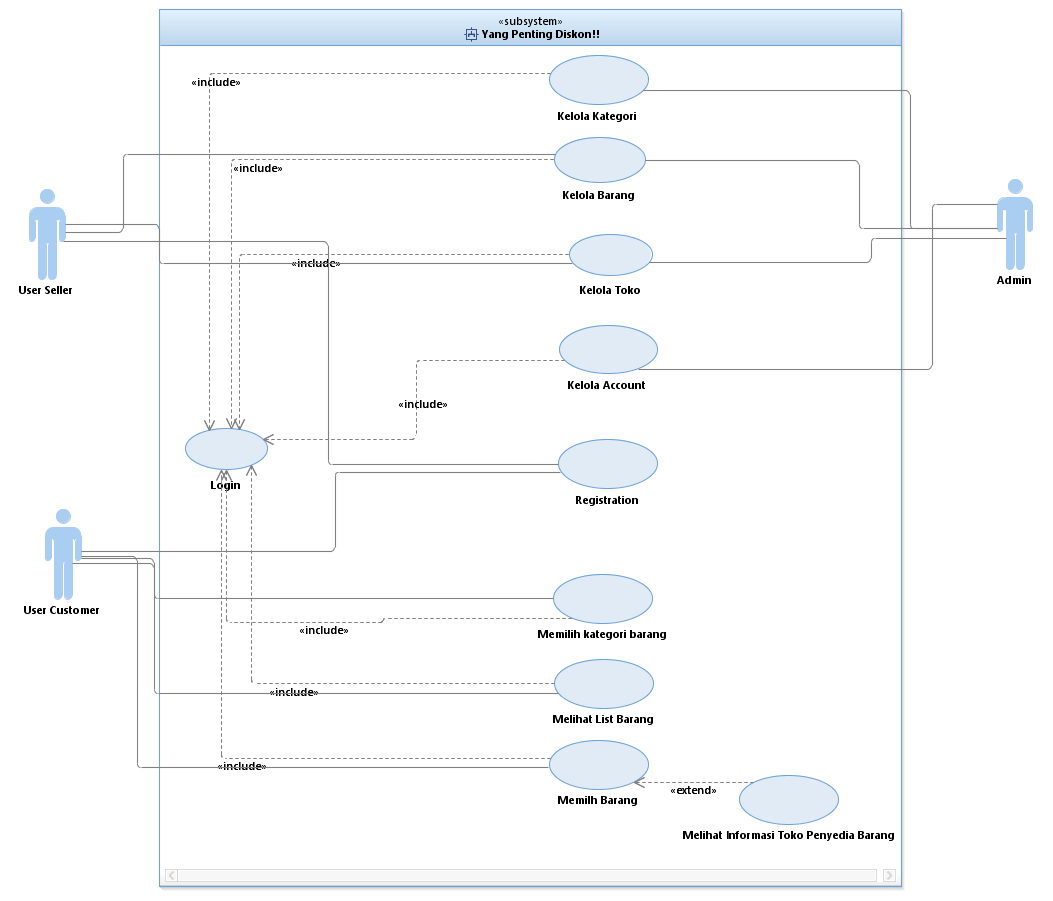
# Matriks Kerunutan

Mapping use case dengan kelas-kelas terkait :

|  |  |
| --- | --- |
| **Use Case Terkait** | **Kelas** |
| Login | viewLogin  controllerLogin  akun  aplikasi  database  user |
| Registrasi | viewRegistrasi controlleRegistrasi  akun  aplikasi  database |
| Kelola Kategori | viewKelolaKategori controllerKelolaKategori  kategori  aplikasi  database |
| Kelola Produk | viewKelolaProduk  controllerKelolaProduk  controllerToko  produk  aplikasi  database |
| Kelola Toko | viewKelolaToko  controllerKelolaToko  toko  aplikasi  database |
| Kelola Akun | viewKelolaAkun  controllerKelolaAkun  akun  aplikasi  database |
| Memilih Kategori Produk | viewKatgori  controllerKategori controllerProduk  kategori  aplikasi  database |
| Memilih Barang | viewProduk  controllerProduk  controllerCart  produk  toko  aplikasi  database |

# Lampiran Lain

## Use Case



## Deployment Diagram

