**DPPL-** **01**

# DESKRIPSI PERANCANGAN PERANGKAT LUNAK

**Rental Kendaraan**

untuk:

Mahasiswa Telkom University

Dipersiapkan oleh:

Ilham Rizki Hidayat (1301184123)

Meyzo Naufal Romzi (130118429)

Mufidah Alfiah (1301184180)

Rachmat Dwi Putra (1301180201)

Program Studi Informatika Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Prodi S1- Informatika Universitas Telkom** | **Nomor Dokumen** | | **Halaman** |
| ***DPPL-0001*** | | *41* |
| **Revisi** | *A* | *Tgl: 20 April 2020* |

**DAFTAR PERUBAHAN**

|  |  |
| --- | --- |
| **Revisi** | **Deskripsi** |
| **A** | * Perbaikan pada Sequence Diagram * Penambahan Algoritma * Penambahan Antarmuka * Perbaikan pada Class Diagram |
| **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**

[DESKRIPSI PERANCANGAN PERANGKAT LUNAK 1](#_Toc39318049)

[1. Pendahuluan 6](#_Toc39318050)

[1.1 Tujuan Penulisan Dokumen 6](#_Toc39318051)

[1.2 Lingkup Masalah 6](#_Toc39318052)

[1.3 Definisi dan Istilah 6](#_Toc39318053)

[1.4 Referensi 6](#_Toc39318054)

[1.5 Sistematika Pembahasan 7](#_Toc39318055)

[2. Deskripsi Perancangan Global 8](#_Toc39318056)

[2.1 Rancangan Lingkungan Implementasi 8](#_Toc39318057)

[2.2 Deskripsi Arsitektural 8](#_Toc39318058)

[2.3 Deskripsi Komponen 9](#_Toc39318059)

[3. Perancangan Rinci 10](#_Toc39318060)

[3.1 Realisasi Use Case 10](#_Toc39318061)

[3.1.1 Use Case <Registrasi> 10](#_Toc39318062)

[3.1.1.1 Identifikasi Kelas 10](#_Toc39318063)

[3.1.1.2 Sequence Diagram 10](#_Toc39318064)

[3.1.1.3 Diagram Kelas 11](#_Toc39318065)

[3.1.2 Use Case <Login> 11](#_Toc39318066)

[3.1.2.1 Identifikasi Kelas 11](#_Toc39318067)

[3.1.3 Use Case <Pemesanan> 13](#_Toc39318068)

[3.1.3.1 Identifikasi Kelas 13](#_Toc39318069)

[3.1.3.2 Sequence Diagram 13](#_Toc39318070)

[3.1.3.3 Diagram Kelas 14](#_Toc39318071)

[3.1.4 Use Case <Pembayaran> 15](#_Toc39318072)

[3.1.4.1 Identifikasi Kelas 15](#_Toc39318073)

[3.1.4.2 Sequence Diagram 15](#_Toc39318074)

[3.1.4.3 Diagram Kelas 16](#_Toc39318075)

[3.1.5 Use Case <Login Admin> 17](#_Toc39318076)

[3.1.5.1 Identifikasi Kelas 17](#_Toc39318077)

[3.1.5.2 Sequence Diagram 17](#_Toc39318078)

[3.1.5.3 Diagram Kelas 17](#_Toc39318079)

[3.1.6 Use Case <Tambahan Kendaraan> 18](#_Toc39318080)

[3.1.6.1 Identifikasi Kelas 18](#_Toc39318081)

[3.1.6.2 Sequence Diagram 18](#_Toc39318082)

[3.1.6.3 Diagram Kelas 19](#_Toc39318083)

[3.1.7 Use Case <Update Kendaraan> 19](#_Toc39318084)

[3.1.7.1 Identifikasi Kelas 19](#_Toc39318085)

[3.1.7.2 Sequence Diagram 20](#_Toc39318086)

[3.1.7.3 Diagram Kelas 20](#_Toc39318087)

[3.1.8 Use Case <Hapus Kendaraan> 21](#_Toc39318088)

[3.1.8.1 Identifikasi Kelas 21](#_Toc39318089)

[3.1.8.2 Sequence Diagram 21](#_Toc39318090)

[3.1.8.3 Diagram Kelas 21](#_Toc39318091)

[3.2 Perancangan Detil Kelas 22](#_Toc39318092)

[3.2.1 Kelas <Kendaraan> 22](#_Toc39318093)

[3.2.2 Kelas <Konsumen> 22](#_Toc39318094)

[3.2.3 Kelas <Pesanan> 23](#_Toc39318095)

[3.2.4 Kelas <Admin> 23](#_Toc39318096)

[3.2.5 Kelas <Database Akun> 23](#_Toc39318097)

[3.2.6 Kelas <Database Kendaraan> 24](#_Toc39318098)

[3.2.7 Kelas <Database Pesanan> 24](#_Toc39318099)

[3.2.8 Kelas <V\_Admin> 25](#_Toc39318100)

[3.2.9 Kelas <C\_Admin> 25](#_Toc39318101)

[3.2.10 Kelas <V\_Transaksi> 25](#_Toc39318102)

[3.2.11 Kelas <C\_Transaksi> 26](#_Toc39318103)

[3.3 Diagram Kelas Keseluruhan 26](#_Toc39318104)

[3.4 Algoritma/Query 27](#_Toc39318105)

[3.5 Perancangan Antarmuka 30](#_Toc39318106)

[3.6 Perancangan Representasi Persistensi Kelas 40](#_Toc39318107)

[4. Matriks Kerunutan 41](#_Toc39318108)

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini berisi penjelasan mengenai Deskripsi Perancangan Perangakat Lunak (DPPL) untuk Sistem Rental Kendaraan. Tujuan penulisan dokumen ini adalah untuk mendeskripsikan secara rinci mengenai perangkat lunak yang akan dibangun. Dokumen ini akan digunakan untuk acuan pada proses pembangunan perangkat lunak dan evaluasi diakhir pembangunan perangkat lunak.

Pengguna dari dokumen ini adalah tim pengembang dari perangkat lunak dan *stakeholders* yang terlibat dalam sistem ini. Dengan disusunnya dokumen DPPL ini, diharapkan pembangunan perangkat lunak akan

## Lingkup Masalah

Pada era globalisasi, pada umumnya segala aktivitas telah mengimplementasikan sistem yang terdigitalisasi. Kegiatan-kegiatan tersebut bervariasi mulai dari transportasi, makanan, dan pendidikan ikut merasakan dampak nyata dari kemajuan teknologi tersebut. Dokumen ini akan membahas serta menganalisis aplikasi kemajuan teknologi pada sistem pendidikan. Dalam hal ini, kami membuat aplikasi dimana program ini membantu menangani permasalahan pengelolaan rental kendaraan yang didalamnya mencakup beberapa aspek seperti aspek logistic (mengenai kesediaan kendaraan) dan manajemen (mengenai pengaturan penyewaannya). Program ini akan sangat memudahkan admin dalam proses pengelolaan data. Data penyewaan akan menjadi lebih mudah untuk diakses serta memudahkan saat melakukan proses pencarian data.

## Definisi dan Istilah

Berikut ini adalah beberapa definisi, singkatan, dan akronim yang terdapat di dalam dokumen ini:

* DPPL

DPPL adalah singkatan dari Deskripsi Perancangan Perangkat Lunak yaitu dokumen yang mendeskripsikan dan menjabarkan secara terperinci mengenai perancangan perangkat lunak yang akan dibangun.

* SKPL

SKPL adalah singkatan dari Spesifikasi Kebutuhan Perangkat Lunak atau sering juga disebut sebagai *Software Requirements Spesification* (SRS) merupakan spesifikasi dari perangkat lunak yang akan dikembangkan.

* ERD

ERD adalah Entity Relationship Diagram yang menunjukkan keterhubungan antar entitas yang ada pada sistem tempat perangkat lunak ini akan diterapkan.

* PHP

PHP adalah singkatan dari PHP Hypertext Preprocessor. PHP merupakan bahasa scripting yang ditanamkan dalam sebuah file html.

## Referensi

Dokumen ini merujuk pada dokumen SKPL sistem informasi rental kendaraan yang berkaitan dengan spesifikasi kebutuhan user dalam proses pembuatan aplikasi.

## Sistematika Pembahasan

Dokumen DPPL ini dibagi menjadi tiga bagian utama. Pertama berisi penjelasan tentang dokumen DPPL yang mencakup tujuan pembuatan dokumen ini, lingkup masalah yang diselesaikan oleh pengembangan website, definisi , referensi, dan deskripsi umum. Kedua berisi penjelasan secara umum mengenai aplikasi yang akan dikembangkan meliputi fungsi dari perangkat lunak, karakteristik pengguna, batasan, dan asumsi yang diambil dalam pengembangan aplikasi. Bagian ketiga berisi uraian kebutuhan aplikasi secara lebih rinci

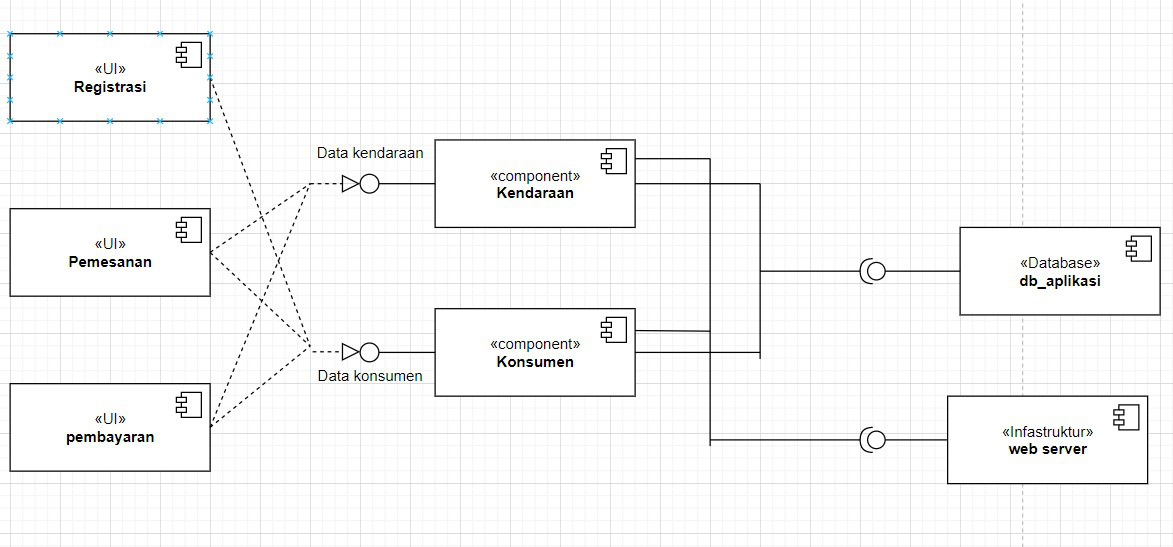
# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

Perangkat lunak pada sisi *server* yang dibutuhkan oleh sistem penyewaan kendaraan adalah:

* Sistem Operasi : Microsoft server 2003 atau linux server
* *Harddisk :* Min. 1 TB
* *Memory* : Min. 4 *GB RAM*
* *Database* : *MySQL Database*
* Perangkat Lunak sisi *client* yang dibutuhkan :
* Sistem Operasi : *Microsoft Windows 7/8/10*
* RAM : Min. 2 GB

## Deskripsi Arsitektural

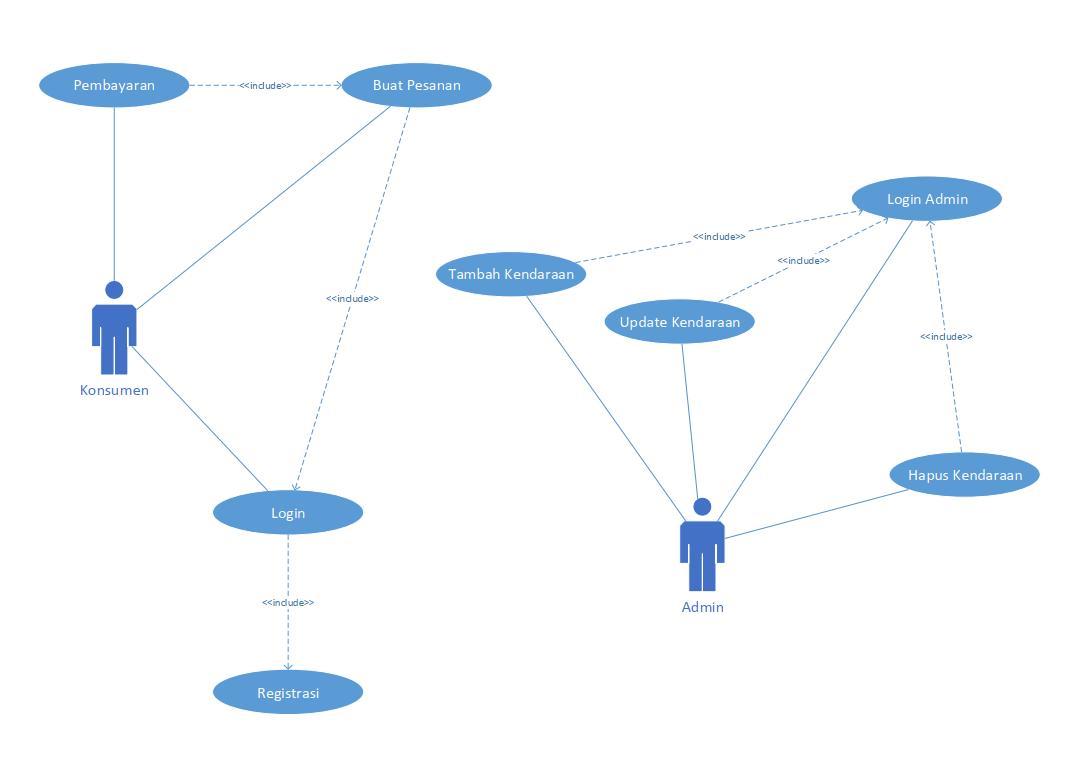


## Deskripsi Komponen

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Komponen** | **Keterangan** |
| 1 | Pembayaran | Komponen yang berisi info pembayaran yang dilakukan oleh konsumen |
| 2 | Pemesanan | Komponen yang berisi info dari detail pemesanan yang dilakukan oleh konsumen |
| 3 | Kendaraan | Komponen yang mewakili informasi mengenai kendaraan yang tersedia untuk disewakan |
| 4 | Registrasi | Komponen yang mewakili informasi yang diperlukan untuk konsumen melakukan proses pendaftaran akun |
| 5 | Konsumen | Komponen yang mewakili informasi mengenai konsumen |
| 6 | DB\_Aplikasi | Komponen yang berisi info mengenai database yang digunakan |
| 7 | Web Server | Komponen yang berisi info mengenai server yang digunakan untuk website |

# Perancangan Rinci

## Realisasi Use Case

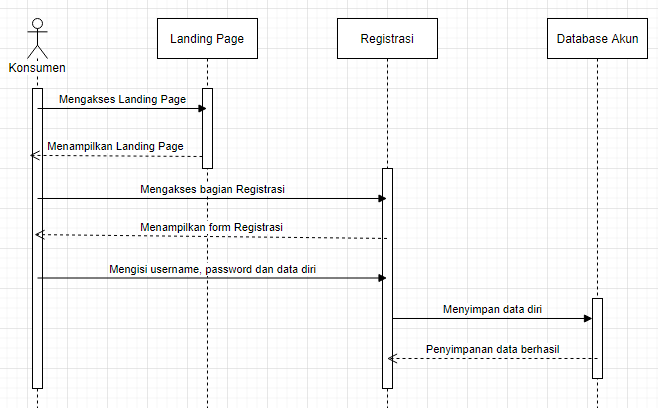


### Use Case <Registrasi>

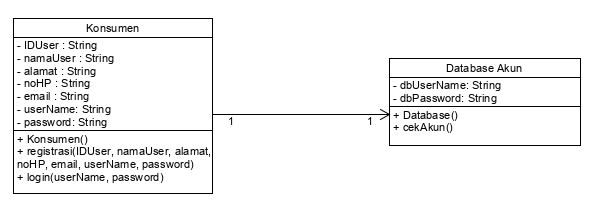
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Akun | Database |
| 2 | Halaman Registrasi | Boundary |
| 3 | Konsumen | Actor |

#### Sequence Diagram



#### Diagram Kelas

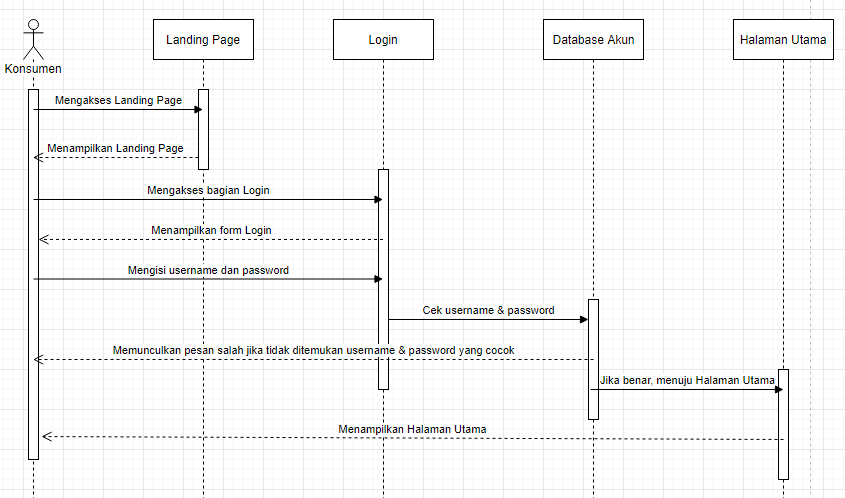


### Use Case <Login>

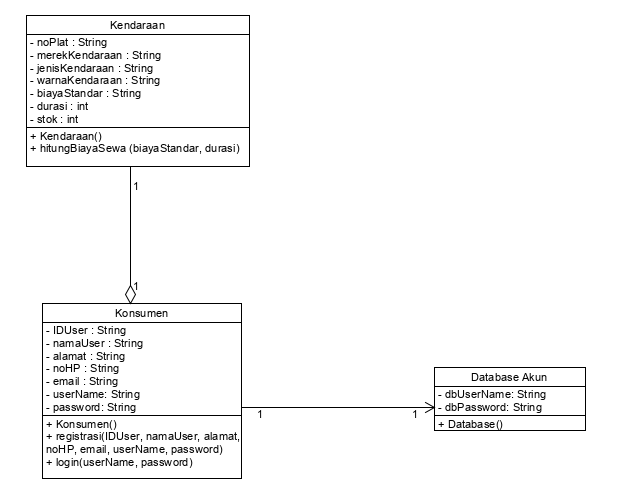
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Akun | Database |
| 2 | Halaman Login | Boundary |
| 3 | Konsumen | Actor |

1. **Sequence Diagram**



1. **Diagram Kelas**

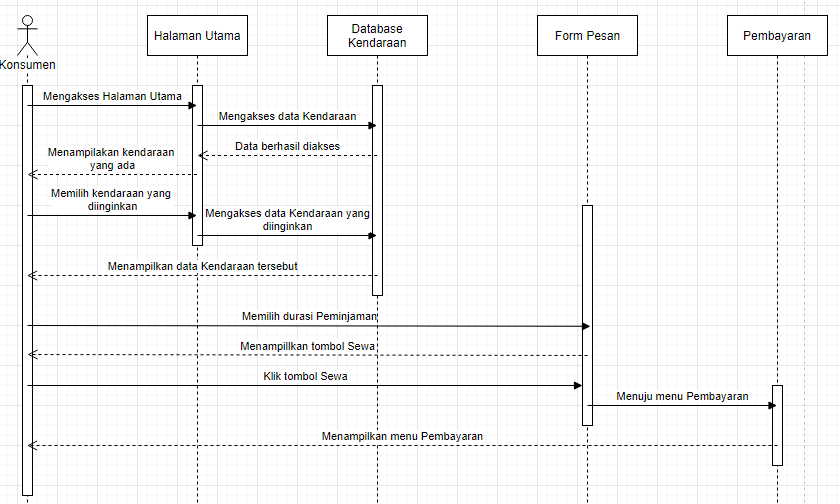


### Use Case <Pemesanan>

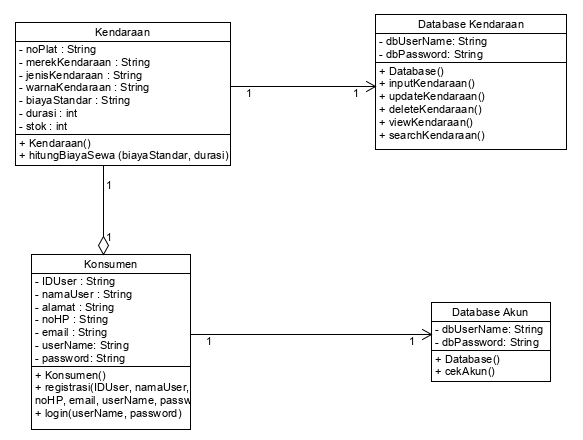
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Kendaraan | Database |
| 2 | Homepage Rental | Boundary |
| 3 | Konsumen | Actor |

#### Sequence Diagram



#### Diagram Kelas

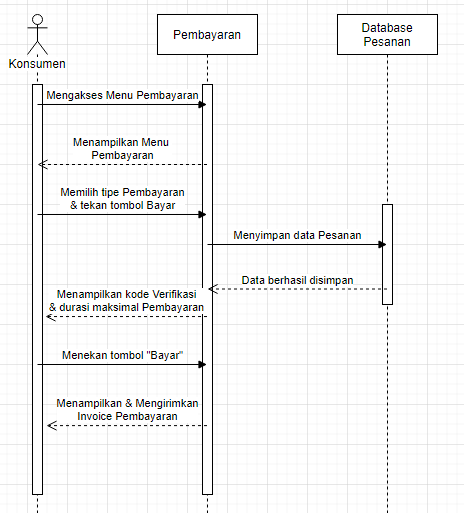


### Use Case <Pembayaran>

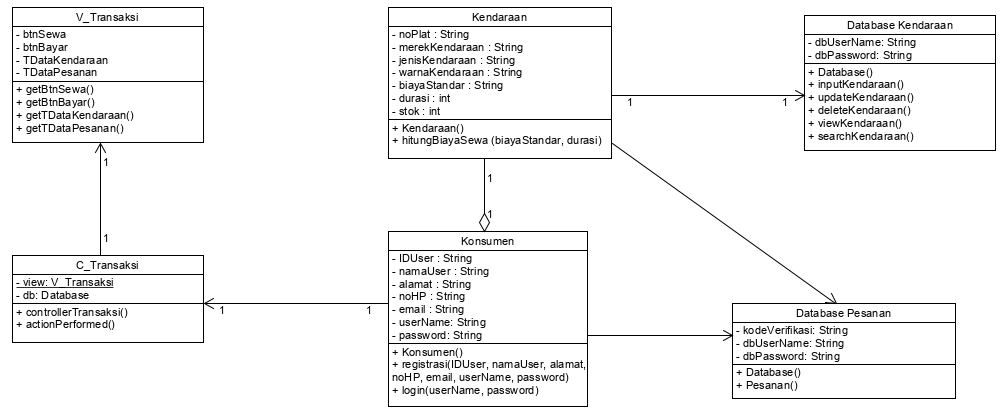
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Kendaraan | Database |
| 2 | Homepage Rental | Boundary |
| 3 | Konsumen | Actor |
| 4 | Database Pesanan | Database |

#### Sequence Diagram



#### Diagram Kelas

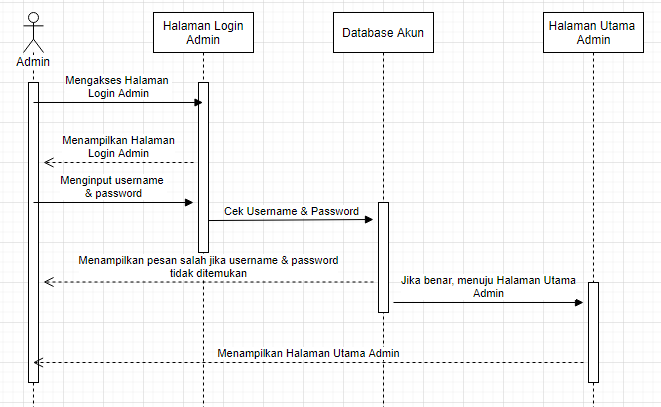


### Use Case <Login Admin>

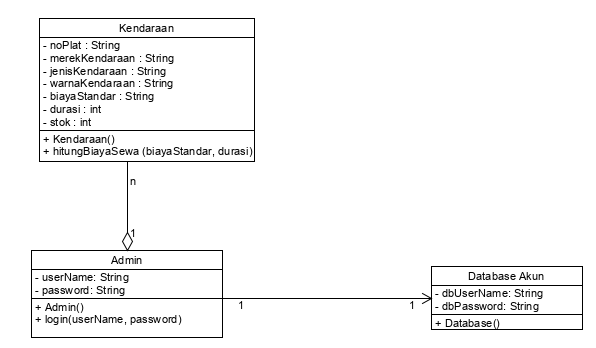
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Akun | Database |
| 2 | Halaman Login Admin | Boundary |
| 3 | Admin | Actor |

#### Sequence Diagram



#### Diagram Kelas

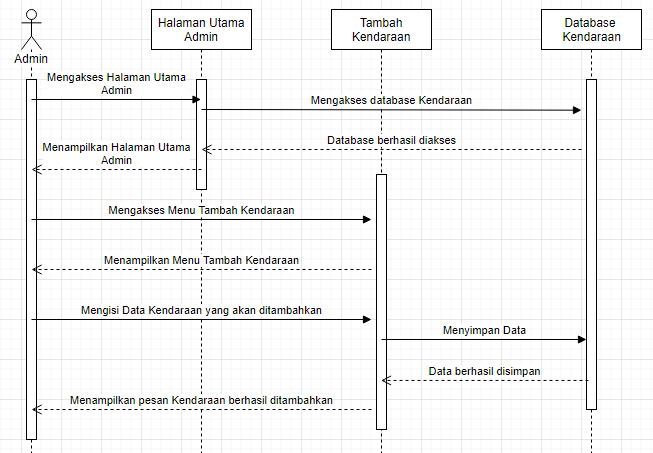


### Use Case <Tambahan Kendaraan>

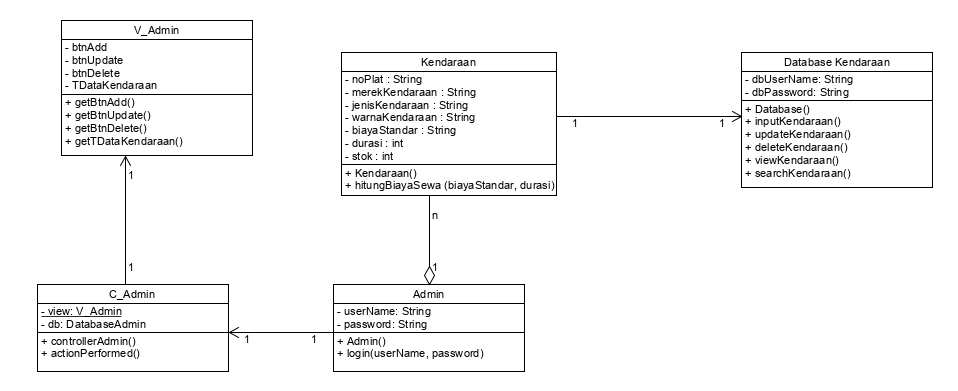
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Kendaraan | Database |
| 2 | Homepage Admin | Boundary |
| 3 | Admin | Actor |

#### Sequence Diagram



#### Diagram Kelas

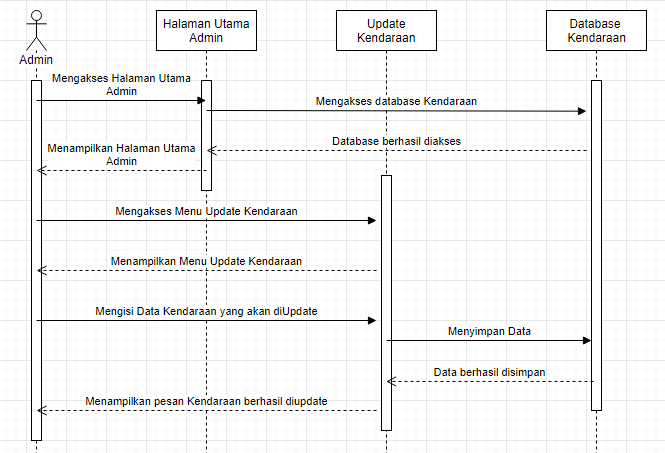


### Use Case <Update Kendaraan>

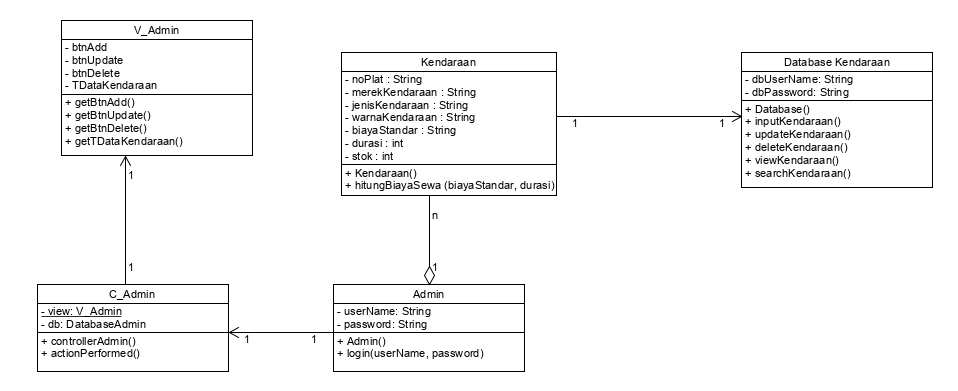
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Kendaraan | Database |
| 2 | Homepage Admin | Boundary |
| 3 | Admin | Actor |

#### Sequence Diagram



#### Diagram Kelas

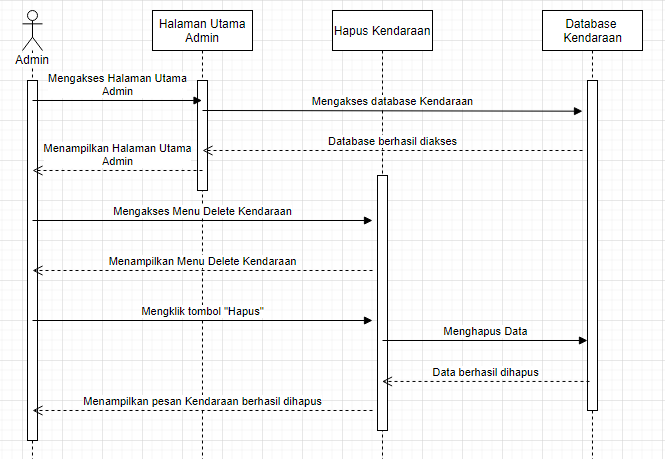


### Use Case <Hapus Kendaraan>

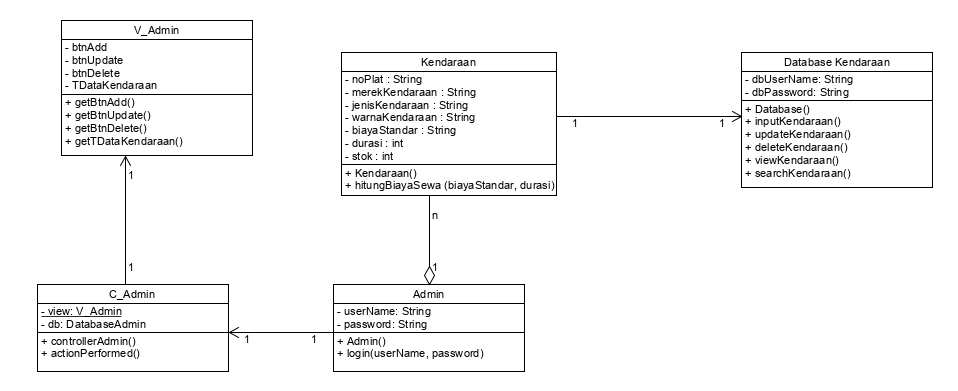
#### Identifikasi Kelas

|  |  |  |
| --- | --- | --- |
| No | Nama Kelas Perancangan | Tipe Kelas |
| 1 | Database Kendaraan | Database |
| 2 | Homepage Admin | Boundary |
| 3 | Admin | Actor |

#### Sequence Diagram



#### Diagram Kelas



## Perancangan Detil Kelas

### Kelas <Kendaraan>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Kendaraan(noPlat: String, merekKendaraan: String, jenisKendaraan: String, warnaKendaraan: String, biayaStandar: String, durasi: int, stok: int) | Public | constructor kendaraan untuk membuat objek kendaraan |
| hitungBiayaSewa(biayaStandar: String, durasi: int) | Public | Method untuk menghitung biaya sewa |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| noPlat | private | String |
| merekKendaraan | private | String |
| jenisKendaraan | private | String |
| warnaKendaraan | private | String |
| biayaStandar | private | String |
| durasi | private | Integer |
| stok | private | Integer |

### Kelas <Konsumen>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Konsumen(IDUser: String, namaUser: String, alamat: String, noHP: String, email: String, userName: String, password: String) | Public | constructor konsumen untuk membuat objek konsumen |
| registrasi(IDUser: String, namaUser: String, alamat: String, noHP: String, email: String, username: String, password: String) | Public | Method untuk melakukan registrasi ke dalam sistem |
| login(userName: String, password: String) | Public | Method untuk melakukan login konsumen ke dalam sistem |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| IDUser | private | String |
| namaUser | private | String |
| alamat | private | String |
| noHP | private | String |
| email | private | String |
| userName | private | String |
| password | private | String |

### Kelas <Pesanan>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Pesanan(kodeVerifikasi: String) | Public | constructor pesanan untuk membuat objek pesanan |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| kodeVerifikasi | private | String |

### Kelas <Admin>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Admin(userName: String, password: String) | Public | constructor admin untuk membuat objek admin |
| login(userName: String, password: String) | Public | Method untuk melakukan login admin ke dalam sistem |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| userName | private | String |
| password | private | String |

### Kelas <Database Akun>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Database(dbUserName: String, dbPassword: String) | Public | constructor database untuk membuat objek database |
| cekAkun() | Public | Method yang berisi query untuk mengecek apakah username ada atau tidak |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| dbUserName | private | String |
| dbPassword | private | String |

### Kelas <Database Kendaraan>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Database(dbUserName: String, dbPassword: String) | Public | constructor database untuk membuat objek database |
| inputKendaraan() | Public | Method yang berisi query untuk input kendaraan ke database |
| updateKendaraan() | Public | Method yang berisi query untuk update kendaraan ke database |
| deleteKendaraan() | Public | Method yang berisi query untuk delete kendaraan dari database |
| viewKendaraan() | Public | Method yang berisi query untuk view data kendaraan di database |
| searchKendaraan() | Public | Method yang berisi query untuk mencari suatu data kendaraan tertentu |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| dbUserName | private | String |
| dbPassword | private | String |

### Kelas <Database Pesanan>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| Database(dbUserName: String, dbPassword: String) | Public | constructor database untuk membuat objek database |
| Pesanan() | Public | Method yang berisi query untuk input pesanan ke database |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| kodeVerifikasi | Private | String |
| dbUserName | private | String |
| dbPassword | private | String |

### Kelas <V\_Admin>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| V\_Admin(btnAdd: button, btnUpdate: button, btnDelete: button, TDataKendaraan: table) | Public | constructor V\_Admin untuk membuat objek V\_Admin |
| getBtnAdd() | Public | Button untuk menambahkan kendaraan |
| getBtnUpdate() | Public | Button untuk mengupdate kendaraan |
| getBtnDelete() | Public | Button untuk menghapus kendaraan |
| getTDataKendaraan() | Public | Table untuk melihat data kendaraan |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| btnAdd | private | Button |
| btnUpdate | private | Button |
| btnDelete | Private | Button |
| TDataKendaraan | private | Table |

### Kelas <C\_Admin>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| controllerAdmin() | Public | Controller untuk admin |
| actionPerformed() | Public | Controller untuk button yang ada di admin |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| view | private | V\_Admin |
| db | private | DatabaseAdmin |

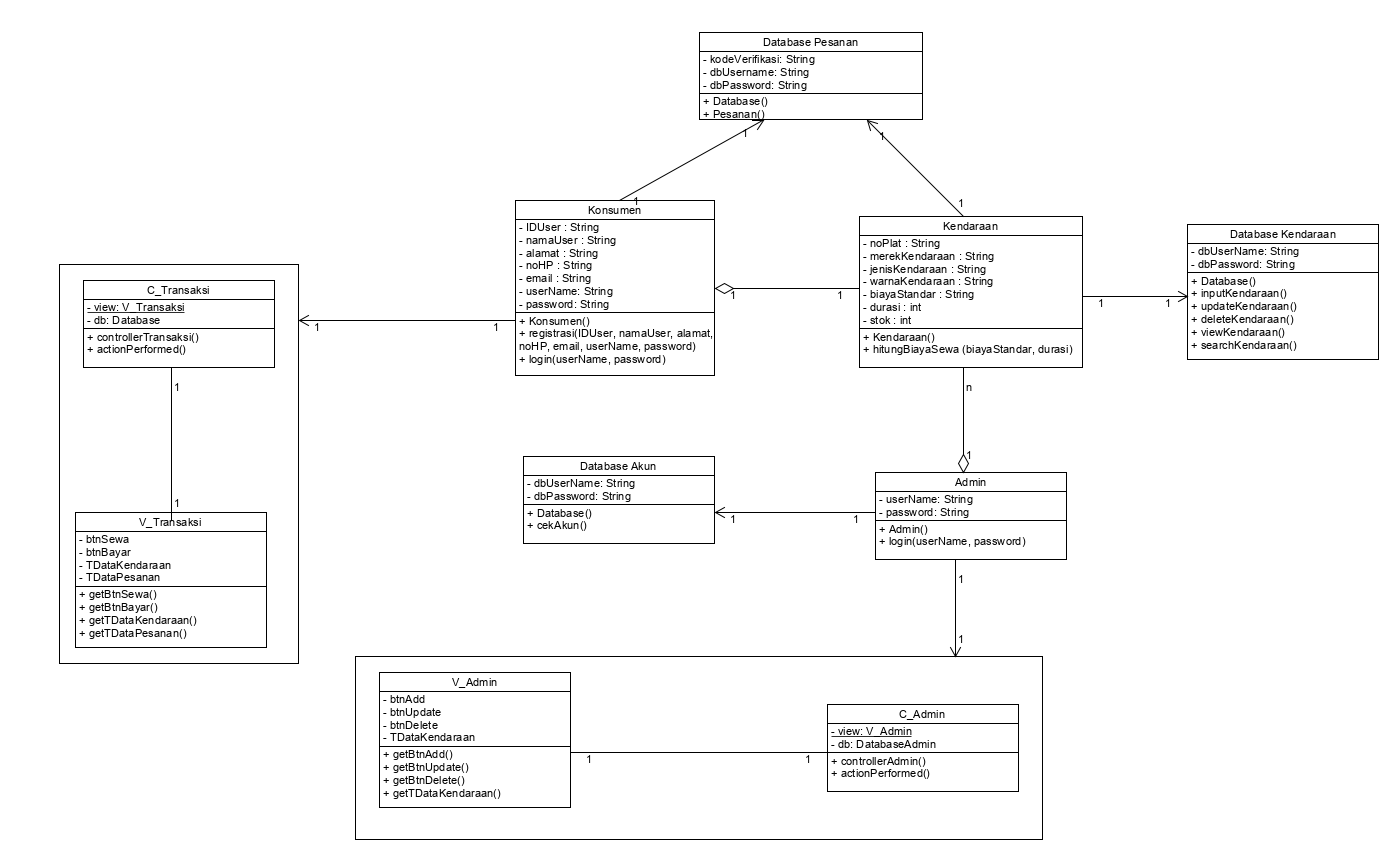
### Kelas <V\_Transaksi>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| V\_Transaksi(btnSewa: button, btnBayar: button, TDataKendaraan: table, TDataPesanan: table) | Public | constructor V\_Transaksi untuk membuat objek V\_Transaksi |
| getBtnSewa() | Public | Button untuk melakukan pesanan |
| getBtnBayar() | Public | Button untuk melakukan pembayaran |
| getTDataKendaraan() | Public | Table untuk melihat data kendaraan yang akan dipesan |
| getTDataPesanan() | Public | Table untuk melihat data pesanan |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| btnSewa | private | Button |
| btnBayar | private | Button |
| TDataKendaraan | Private | Table |
| TDataPesanan | private | Table |

### Kelas <C\_Transaksi>

|  |  |  |
| --- | --- | --- |
| **Nama Operasi** | **Visibility**  **(private, public)** | **Keterangan** |
| controllerTransaksi() | Public | Controller untuk transaksi |
| actionPerformed() | Public | Controller untuk button yang ada di admin |
| **Nama Atribut** | **Visibility**  **(private, public)** | **Tipe** |
| view | private | V\_Transaksi |
| db | private | Database |

## Diagram Kelas Keseluruhan

*.*

## Algoritma/Query

*Nama Kelas : Konsumen*

*Nama Operasi : registrasi*

*Algoritma :*

*data = array(IDUser, namaUser, alamat, noHP, email, username, password);*

*if (data != NULL) {*

*Username = data[username];*

*Password = data[password];*

*if (Q-001 == true) {*

*return false;*

*} else {*

*Q-002;*

*Return true;*

*}*

*}*

*Nama Kelas : Konsumen, Admin*

*Nama Operasi : login*

*Algoritma :*

*U = userName;*

*P = password;*

*if (U != NULL && P != NULL) {*

*if(Q-001 == true){*

*return false;*

*} else {*

*return true;*

*}*

*Nama Kelas : Kendaraan*

*Nama Operasi : hitungBiayaSewa*

*Algoritma :*

*A1 = biayaStandar; A2 = durasi;*

*return A1 \* A2;*

*Nama Kelas : Database Kendaraan*

*Nama Operasi : inputKendaraan*

*Algoritma :*

*dataKendaraan = array(noPlat, merekKendaraan, jenisKendaraan, warnaKendaraan, biayaStandar, durasi, stok);*

*if(dataKendaraan != NULL) {*

*if(Q-004 == true) {*

*return true;*

*} else {*

*return false;*

*}*

*}*

*Nama Kelas : Database Kendaraan*

*Nama Operasi : updateKendaraan*

*Algoritma :*

*kondisi = noPlat;*

*//dataUpdate adalah array dari data Kendaraan yang akan diubah*

*dataKendaraan = dataUpdate;*

*if(dataKendaraan != NULL) {*

*if(Q-005 == true) {*

*return true;*

*} else {*

*return false;*

*}*

*}*

*Nama Kelas : Database Kendaraan*

*Nama Operasi : deleteKendaraan*

*Algoritma :*

*kondisi = noPlat;*

*if(kondisi != NULL) {*

*if(Q-006 == true) {*

*return true;*

*} else {*

*return false;*

*}*

*}*

*Nama Kelas : Database Kendaraan*

*Nama Operasi : searchKendaraan*

*Algoritma :*

*kondisi = noPlat;*

*if(kondisi != NULL) {*

*if(Q-009 == true) {*

*return true;*

*} else {*

*return false;*

*}*

*}*

*Nama Kelas : Database Pesanan*

*Nama Operasi : Pesanan*

*Algoritma :*

*Pesanan = array(noPlat, IDUser, kodeVerifikasi);*

*if(Pesanan != NULL) {*

*if(Q-010 == true) {*

*return true;*

*} else {*

*return false;*

*}*

*}*

|  |  |  |
| --- | --- | --- |
| **No Query** | **Query** | **Keterangan** |
| Q-001 | *SELECT \* FROM akun WHERE userName=Username & password=Password;* | Mengecek Username dan Password yang diinputkan oleh user sudah ada di database atau tidak |
| Q-002 | *INSERT INTO akun VALUES(IDUser, namaUser, alamat, noHP, email, username, password);* | Melakukan input konsumen baru saat registrasi |
| Q-003 | *SELECT \* FROM kendaraan;* | Menampilkan data yang ada di tabel kendaraan |
| Q-004 | *INSERT INTO kendaraan VALUES(noPlat, merekKendaraan, jenisKendaraan, warnaKendaraan, biayaStandar, durasi, stok);* | Melakukan input kendaraan |
| Q-005 | *UPDATE kendaraan SET kolom=”data” WHERE kondisi* | Melakukan update data kendaraan |
| Q-006 | *DELETE FROM kendaraan WHERE kondisi* | Melakukan hapus kendaraan |
| Q-007 | *INSERT INTO pesanan VALUES(kodeVerifikasi)* | Melakukan pembuatan pesanan baru |
| Q-008 | *SELECT \* FROM pesanan JOIN kendaraan* | Menampilkan suatu detail pesanan |
| Q-009 | *SELECT \* FROM kendaraan WHERE kondisi* | Menampilkan kendaraan dengan kondisi tertentu |
| Q-010 | *INSERT INTO pesanan VALUES(noPlat, IDUser, kodeVerifikasi);* | Melakukan input suatu pesanan |

## Perancangan Antarmuka

Antarmuka: Landing Page

A picture containing car, driving, truck, red

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *btnCustomer* | *Button* | *Customer* | *Jika diklik, akan masuk ke halaman Login* |
| *btnAdmin* | *Button* | *Admin* | *Jika diklik, akan masuk ke halaman Login Admin* |

Antarmuka: Login

A close up of a car

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *tfUsername* | *Text Field* | *Username* | *Text field untuk mengisi username pada akun untuk masuk ke halaman utama* |
| *tfPassword* | *Text Field* | *Password* | *Text field untuk mengisi password pada akun untuk masuk ke halaman utama* |
| *btnLogin* | *Button* | *Login* | *Jika diklik, maka akan masuk ke halaman utama dengan username dan password yang telah diisi* |

Antarmuka: Registrasi

A screenshot of a cell phone

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *tfNamaLengkap* | *Text Field* | *Nama Lengkap* | *Text field untuk mengisi nama lengkap untuk membuat akun* |
| *tfNoHP* | *Text Field* | *No HP* | *Text field untuk mengisi no hp untuk membuat akun* |
| *tfEmail* | *Text Field* | *Email* | *Text field untuk mengisi email untuk membuat akun* |
| *tfAlamat* | *Text Field* | *Alamat* | *Text field untuk mengisi alamat untuk membuat akun* |
| *tfUsername* | *Text Field* | *Username* | *Text field untuk mengisi username untuk membuat akun* |
| *tfPassword* | *Text Field* | *Password* | *Text field untuk mengisi password untuk membuat akun* |
| *btnRegistrasi* | *Button* | *Registrasi* | *Jika diklik, data yang sudah diisi akan dimasukkan ke dalam database (jika username yang dimasukkan belum ada di database)* |

Antarmuka: Halaman Utama

A close up of a car

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *btnPemesanan* | *Button* | *Pemesanan* | *Jika diklik, akan masuk halaman Pemesanan kendaraan* |
| *btnPembayaran* | *Button* | *Pembayaran* | *Jika diklik, akan masuk halaman Pembayaran* |

Antarmuka: Pemesanan

A screenshot of a cell phone

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *ddKendaraan* | *Dropdown* | *Kendaraan* | *Akan muncul pilihan kendaraan yang dapat dipesan* |
| *ddDurasi* | *Dropdown* | *Durasi* | *Akan muncul pilihan durasi untuk sewa* |
| *btnSewa* | *Button* | *Sewa* | *Jika diklik, akan masuk ke halaman Pembayaran dengan data yang sudah dipilih* |

Antarmuka: Pembayaran

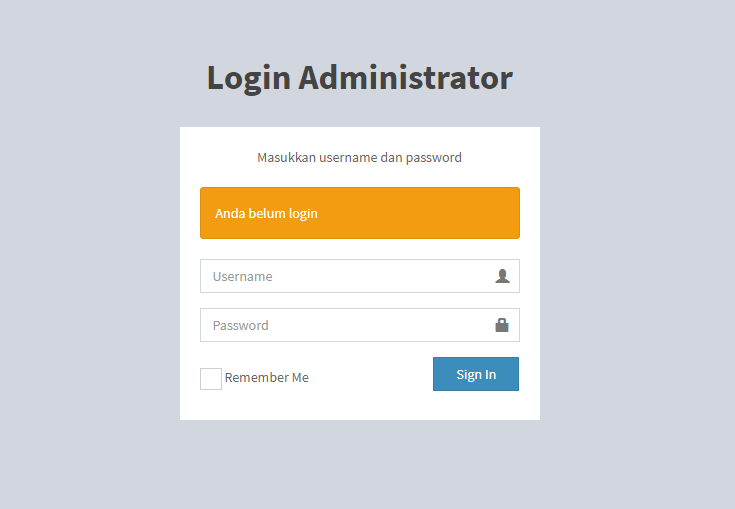
A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

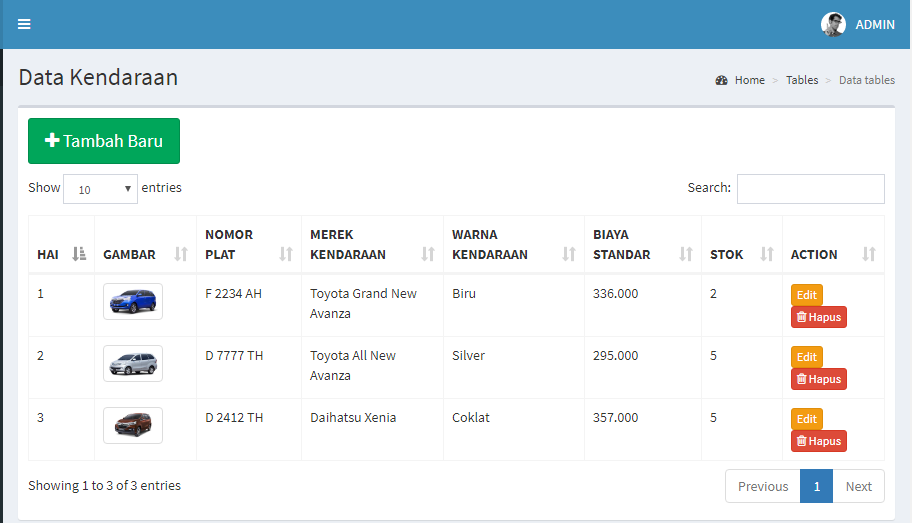
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *ddMetode* | *Dropdown* | *Metode Pembayaran* | *Akan muncul pilihan metode pembayaran yang dapat dilakukan* |
| *btnBayar* | *Dropdown* | *Bayar* | *Jika diklik, maka pembayaran telah dilakukan* |

Antarmuka: Login Admin



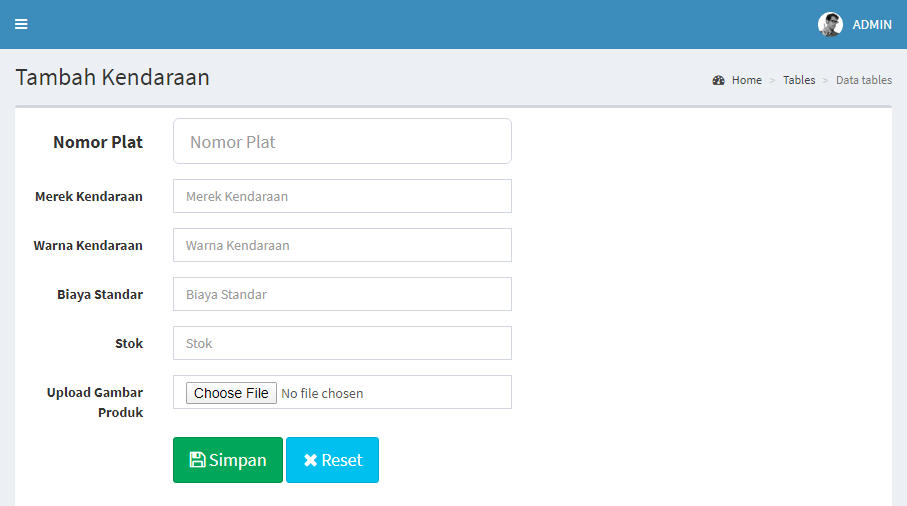
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *tfUsername* | *Text Field* | *Username* | *Text field untuk mengisi username pada akun untuk masuk ke halaman khusus admin* |
| *tfPassword* | *Text Field* | *Password* | *Text field untuk mengisi password pada akun untuk masuk ke halaman khusus admin* |
| *bSignIn* | *Button* | *Sign In* | *Jika diklik, maka akan masuk ke halaman admin dengan username dan password yang telah diisi* |

Antarmuka : View Kendaraan



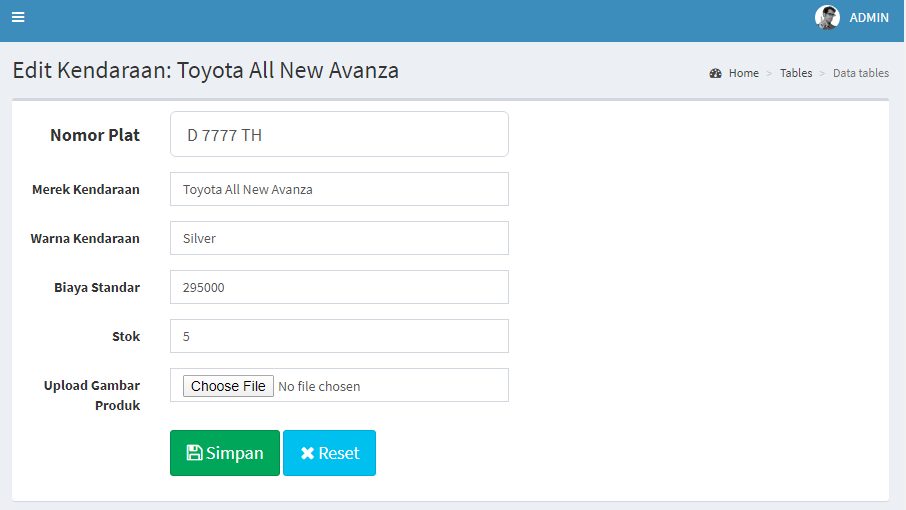
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *btnTambahBaru* | *Button* | *Tambah Baru* | *Jika diklik, maka akan masuk ke halaman untuk input kendaraan* |
| *tiSearch* | *Text Input* | *Search* | *Jika text field diisi dan ditekan enter, maka sistem akan menampilkan merek kendaraan yang dicari* |
| *btnEdit* | *Button* | *Edit* | *Jika diklik, maka akan masuk ke halaman untuk edit kendaraan tersebut* |
| *btnHapus* | *Button* | *Hapus* | *Jika diklik, maka akan masuk ke halaman peringatan untuk menghapus kendaraan tersebut* |

Antarmuka : Create Kendaraan



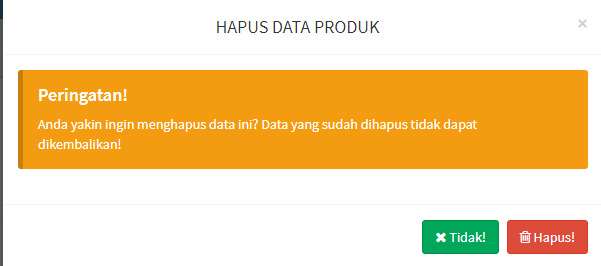
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *tfNomorPlat* | *Text Field* | *Nomor Plat* | *Text Field untuk mengisi nomor plat kendaraan yang akan diinputkan* |
| *tfMerekKendaraan* | *Text Field* | *Merek Kendaraan* | *Text Field untuk mengisi merek kendaraan yang akan diinputkan* |
| *tfWarnaKendaraan* | *Text Field* | *Warna Kendaraan* | *Text Field untuk mengisi warna kendaraan yang akan diinputkan* |
| *tfBiayaStandar* | *Text Field* | *Biaya Standar* | *Text Field untuk mengisi biaya standar dari kendaraan yang akan diinputkan* |
| *tfStok* | *Text Field* | *Stok* | *Text Field untuk mengisi stok kendaraan yang akan diinputkan* |
| *fGambar* | *File* | *Gambar* | *Untuk mengupload gambar dari kendaraan yang akan diinputkan* |
| *btnSimpan* | *Button* | *Simpan* | *Jika diklik, maka data kendaraan akan disimpan ke dalam database kendaraan* |
| *btnReset* | *Button* | *Reset* | *Jika diklik, maka data yang telah diisi akan Kembali hilang ke kondisi awal (kosong)* |

Antarmuka: Edit Kendaraan



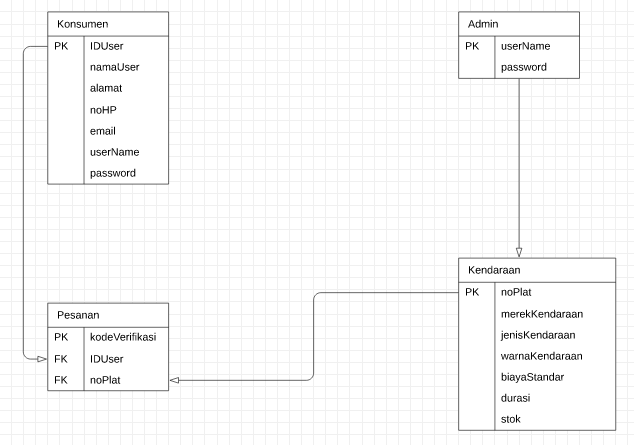
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *tfNomorPlat* | *Text Field* | *Nomor Plat* | *Text Field untuk mengisi / menampilkan nomor plat kendaraan yang akan diedit* |
| *tfMerekKendaraan* | *Text Field* | *Merek Kendaraan* | *Text Field untuk mengisi / menampilkan merek kendaraan yang akan diedit* |
| *tfWarnaKendaraan* | *Text Field* | *Warna Kendaraan* | *Text Field untuk mengisi / menampilkan warna kendaraan yang akan diedit* |
| *tfBiayaStandar* | *Text Field* | *Biaya Standar* | *Text Field untuk mengisi / menampilkan biaya standar dari kendaraan yang akan diedit* |
| *tfStok* | *Text Field* | *Stok* | *Text Field untuk mengisi / menampilkan stok kendaraan yang akan diedit* |
| *fGambar* | *File* | *Gambar* | *Untuk mengupload gambar dari kendaraan yang akan diedit* |
| *btnSimpan* | *Button* | *Simpan* | *Jika diklik, maka data kendaraan akan disimpan ke dalam database kendaraan* |
| *btnReset* | *Button* | *Reset* | *Jika diklik, maka data yang telah diisi akan Kembali hilang ke kondisi awal (kosong)* |

Antarmuka: Delete Kendaraan



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *btnTidak* | *Button* | *Tidak* | *Jika diklik, maka akan Kembali ke halaman daftar kendaraan dan kendaraan tersebut tidak dihapus dari database* |
| *btnHapus* | *Button* | *Hapus* | *Jika diklik, maka data kendaraan tersebut akan dihapus dari database* |

## Perancangan Representasi Persistensi Kelas



# Matriks Kerunutan

Mapping use case dengan kelas-kelas terkait

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Usecase Terkait** | **Kelas** |
| FR-01 | Register | Konsumen, database |
| FR-02 | Login | Konsumen, database |
| FR-03 | Login Admin | Admin, database |
| FR-04 | Pemesanan | Konsumen, Kendaraan, Pesanan, database |
| FR-05 | Pembayaran | Konsumen, Kendaraan, Pesanan, database |
| FR-06 | Input Kendaraan | Admin, Kendaraan, database |
| FR-07 | Update Kendaraan | Admin, Kendaraan, database |
| FR-08 | Delete Kendaraan | Admin, Kendaraan, database |