**DPPL-xx**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

RENTAL MOBIL GAYENG

untuk:

Umum

Dipersiapkan oleh:

M. Firmansyah Arrozi (1301174138)

Subkhan Ibnu Aji (1301172001)

Suryo Adi Gumelar (1301174185)

Viny Gilang Ramadhan (1301174122)

Program Studi Informatika

Fakultas Informatika

Jl. Telekomunikasi 1, Bandung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Prodi S1- Informatika**  **Universitas Telkom** | Nomor Dokumen | | Halaman |
| *DPPL-xx <xx:no grp>* | | *<#>/<jml #* |
| Revisi | *<nomor revisi>* | *Tgl: <isi tanggal>* |

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

[Daftar Tabel 8](#_Toc8823084)

[Daftar Gambar 8](#_Toc8823085)

[1. Pendahuluan 10](#_Toc8823086)

[1.1 Tujuan Penulisan Dokumen 10](#_Toc8823087)

[1.2 Lingkup Masalah 10](#_Toc8823088)

[1.3 Definisi dan Istilah 10](#_Toc8823089)

[1.4 Referensi 10](#_Toc8823090)

[1.5 Sistematika Pembahasan 11](#_Toc8823091)

[2 Deskripsi Perancangan Global 11](#_Toc8823092)

[2.1 Deskripsi Arsitektural 11](#_Toc8823093)

[2.2 Deskripsi Komponen 12](#_Toc8823094)

[3 Perancangan Rinci 14](#_Toc8823095)

[3.1 Realisasi Use Case 14](#_Toc8823096)

[3.1.1 Use Case #1 Memilih Mobil 14](#_Toc8823097)

[3.1.1.1 Perancangan Antarmuka Usecase #1 14](#_Toc8823098)

[3.1.1.1.1 Tabel Identifikasi Antarmuka / Layar / Page 14](#_Toc8823099)

[3.1.1.2 Identifikasi Object Baru 18](#_Toc8823100)

[**TABEL OBJECT PERANCANGAN MEMILIH MOBIL** 18](#_Toc8823101)

[*1.* 18](#_Toc8823102)

[*Halaman Search Mobil* 18](#_Toc8823103)

[*Boundary* 18](#_Toc8823104)

[*2.* 18](#_Toc8823105)

[*Controller halaman pilih mobil* 18](#_Toc8823106)

[*Controller* 18](#_Toc8823107)

[*3.* 18](#_Toc8823108)

[*Database Mobil* 18](#_Toc8823109)

[*Entity* 18](#_Toc8823110)

[*Halaman Form Booking* 18](#_Toc8823111)

[*Boundary* 18](#_Toc8823112)

[*5.* 18](#_Toc8823113)

[*Controller Booking* 18](#_Toc8823114)

[*Controller* 18](#_Toc8823115)

[*6.* 18](#_Toc8823116)

[*Database Transaksi* 18](#_Toc8823117)

[*Entity* 18](#_Toc8823118)

[3.1.1.3 Robustness Diagram 19](#_Toc8823119)

[**1)** **Robustness Memilih Mobil** 19](#_Toc8823120)

[3.1.1.4 Diagram Kelas 20](#_Toc8823121)

[3.1.1.5 Sequence Diagram 20](#_Toc8823122)

[**1)** **sequence diagram memilih mobil** 20](#_Toc8823123)

[3.1.2 Use Case #2 Pemesanan 20](#_Toc8823124)

[3.1.2.1 Perancangan Antarmuka Usecase #2 21](#_Toc8823125)

[3.1.2.1.1 Tabel Identifikasi Antarmuka / Layar / Page 21](#_Toc8823126)

[3.1.2.2 Identifikasi Object Baru 21](#_Toc8823127)

[3.1.2.3 Robustness Diagram 22](#_Toc8823128)

[3.1.2.4 Diagram Kelas 22](#_Toc8823129)

[3.1.2.5 Sequence Diagram 23](#_Toc8823130)

[3.1.3 Use Case #3 Melakukan Pembayaran 23](#_Toc8823131)

[3.1.3.1 Perancangan Antarmuka Use case #3 23](#_Toc8823132)

[3.1.3.1.1 Tabel Identifikasi Antarmuka / Layar / Page 23](#_Toc8823133)

[3.1.3.2 Identifikasi Object Baru 24](#_Toc8823134)

[**TABEL OBJECT PERANCANGAN PEMBAYARAN** 24](#_Toc8823135)

[*1.* 25](#_Toc8823136)

[*Halaman Pembayaran* 25](#_Toc8823137)

[*Boundry* 25](#_Toc8823138)

[*2.* 25](#_Toc8823139)

[*Controller Pembayaran* 25](#_Toc8823140)

[*Controller* 25](#_Toc8823141)

[*3.* 25](#_Toc8823142)

[*Database Transaksi* 25](#_Toc8823143)

[*Entity* 25](#_Toc8823144)

[3.1.3.3 Robustness Diagram 25](#_Toc8823145)

[3.1.3.4 Diagram Kelas 25](#_Toc8823146)

[3.1.3.5 Sequence Diagram 27](#_Toc8823147)

[3.1.4 Use Case #4 <Verifikasi Pembayaran> 27](#_Toc8823148)

[3.1.4.1 Perancangan Antarmuka Use case #4 28](#_Toc8823149)

[3.1.4.1.1 Tabel Identifikasi Antarmuka / Layar / Page 28](#_Toc8823150)

[3.1.4.2 Identifikasi Object Baru 28](#_Toc8823151)

[3.1.4.3 Robustness Diagram 29](#_Toc8823152)

[3.1.4.4 Diagram Kelas 30](#_Toc8823153)

[3.1.4.4.1 Sequence Diagram 30](#_Toc8823154)

[3.1.5 Use Case #Mengelola Data Customer 31](#_Toc8823155)

[3.1.5.1 Perancangan Antarmuka Usecase #5 32](#_Toc8823156)

[3.1.5.1.1 Tabel Identifikasi Antarmuka/Layar/page 32](#_Toc8823157)

[3.1.5.2 Identifikasi Object Baru 34](#_Toc8823158)

[**TABEL OBJECT PERANCANGAN VERIFIKASI PEMBAYARAN** 34](#_Toc8823159)

[1. 34](#_Toc8823160)

[Halaman Data Customer 34](#_Toc8823161)

[Boundry 34](#_Toc8823162)

[2. 35](#_Toc8823163)

[Controller Data Customer 35](#_Toc8823164)

[Controller 35](#_Toc8823165)

[3. 35](#_Toc8823166)

[Database customer 35](#_Toc8823167)

[Entity 35](#_Toc8823168)

[3.1.5.3 Robustness Diagram 35](#_Toc8823169)

[3.1.5.4 Diagram Kelas 36](#_Toc8823170)

[3.1.5.5 Sequence Diagram 37](#_Toc8823171)

[3.1.6 Use Case #Mengelola Data Mobil 37](#_Toc8823172)

[3.1.6.1 perancangan Antarmuka Usecase #6 38](#_Toc8823173)

[3.1.6.1.1 Tabel Identifikasi Antarmuka/Layar/page 38](#_Toc8823174)

[3.1.6.2 Identifikasi Object Baru 39](#_Toc8823175)

[**TABEL OBJECT PERANCANGAN MANAGE DATA MOBIL** 40](#_Toc8823176)

[1. 40](#_Toc8823177)

[Halaman manage moibl 40](#_Toc8823178)

[Boundry 40](#_Toc8823179)

[2. 40](#_Toc8823180)

[Controller Data mobil 40](#_Toc8823181)

[Controller 40](#_Toc8823182)

[3. 40](#_Toc8823183)

[Database mobil 40](#_Toc8823184)

[Entity 40](#_Toc8823185)

[3.1.6.3 Robustness Diagram 41](#_Toc8823186)

[3.1.6.5 Sequence Diagram 43](#_Toc8823187)

[4 Perancangan Detail 43](#_Toc8823188)

[4.1 Perancangan Detail Kelas 43](#_Toc8823189)

[4.2 Perancangan Kelas Persistensi (\*\*\*Basis Data Skema Tabel) 46](#_Toc8823190)

[4.3 Perancangan Algoritma 47](#_Toc8823191)

[4.3.1 Algoritma #1 47](#_Toc8823192)

[4.4 Perancangan Query 47](#_Toc8823193)

[5 Matriks Kerunutan (Requirement Traceability Matrix) 48](#_Toc8823194)

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# Daftar Tabel

[Table 1 komponen diagram 12](#_Toc8822795)

[Table 2 realisasi use case 14](#_Toc8822796)

[Table 3 UI use case #1 16](#_Toc8822797)

[Table 4 page menu user 16](#_Toc8822798)

[Table 5 page search car 17](#_Toc8822799)

[Table 6 choose car 17](#_Toc8822800)

[Table 7 memilih mobil 19](#_Toc8822801)

[Table 8 UI use case #2 21](#_Toc8822802)

[Table 9 page pemesanan 21](#_Toc8822803)

[Table 10 object perancangan #2 22](#_Toc8822804)

[Table 11 UI use case #3 24](#_Toc8822805)

[Table 12 page payment 24](#_Toc8822806)

[Table 13 pembayaran\ 25](#_Toc8822807)

[Table 14 UI Use case #4 28](#_Toc8822808)

[Table 15 page verifikasi pembayaran 28](#_Toc8822809)

[Table 16 object perancangan verifikasi pembayaran 28](#_Toc8822810)

[Table 17 UI usecase #5 33](#_Toc8822811)

[Table 18 page manage customer data 33](#_Toc8822812)

[Table 19 page edt data 34](#_Toc8822813)

[Table 20 object manage data 35](#_Toc8822814)

[Table 21 UI use case #6 38](#_Toc8822815)

[Table 22 page manage mobil data 39](#_Toc8822816)

[Table 23 object perancangan manage data mobil 40](#_Toc8822817)

[Table 24 Perancagan class diagram 45](#_Toc8822818)

[Table 25 query 48](#_Toc8822819)

[Table 26 matriks kerunutan 48](#_Toc8822820)

# Daftar Gambar

[Gambar 1 Deployment diagram 11](#_Toc8822821)

[Gambar 2diagram komponen customer 13](#_Toc8822822)

[Gambar 3komponen admin 13](#_Toc8822823)

[Gambar 4 menu user 15](#_Toc8822824)

[Gambar 5earch car 15](#_Toc8822825)

[Gambar 6 choose car 16](#_Toc8822826)

[Gambar 7 robustness memilh mobil 19](#_Toc8822827)

[Gambar 8 class diagram memilih mobil 20](#_Toc8822828)

[Gambar 9 sequence diagram memilih mobil 20](#_Toc8822829)

[Gambar 10 robustness pemesanan 22](#_Toc8822830)

[Gambar 11 class diagram informasi pemesanan 23](#_Toc8822831)

[Gambar 12 sequence diagram memilih mobil 23](#_Toc8822832)

[Gambar 13 payment 24](#_Toc8822833)

[Gambar 14 robustness pembayaran 25](#_Toc8822834)

[Gambar 15 class diagram pembayaran 26](#_Toc8822835)

[Gambar 16 sequence pembayaran 27](#_Toc8822836)

[Gambar 17 verifikasi pembayaran 28](#_Toc8822837)

[Gambar 18 robustness diagram verifikasi pembayaran 29](#_Toc8822838)

[Gambar 19 class diagram verifikasi pembayaran 30](#_Toc8822839)

[Gambar 20 sequence verifikasi pembayaran 31](#_Toc8822840)

[Gambar 21 manage customer data 32](#_Toc8822841)

[Gambar 22 edit data 32](#_Toc8822842)

[Gambar 23 robustness manage data customer 35](#_Toc8822843)

[Gambar 24 class diagram manage data customer 36](#_Toc8822844)

[Gambar 25 sequence diagram manage data customer 37](#_Toc8822845)

[Gambar 26 manage mobil 38](#_Toc8822846)

[Gambar 27 robustness diagram manage data 41](#_Toc8822847)

[Gambar 28 class diagram manage data mobil 42](#_Toc8822848)

[Gambar 29 sequence manage data mobil 43](#_Toc8822849)

[Gambar 30 skema relasi 46](#_Toc8822850)

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen SKPL ini berisi Spesifikasi Kebutuhan Perangkat Lunak untuk Rental Mobil Gayeng. Adapun tujuan dari penulisan dokumen ini yaitu :

1. Untuk memberikan penjelasan dan gambaran mengenai kebutuhan perangkat lunak yang akan dibangun baik berupa gambaran umum maupun penjelasan yang detail.
2. Pengguna dari dokumen ini adalah pengembang perangkat lunak Sistem Informasi Rental Mobil. Dokumen ini akan digunakan sebagai acuan rancangan perangkat lunak untuk jadi bahan evaluasi pada saat pengembangan perangkat lunak.
3. Dengan adanya dokumen ini, diharapkan pengembangan perangkat lunak Rental Mobil Gayeng dapat lebih terarah.

## Lingkup Masalah

Perangkat lunak yang akan dikembangkan berisi suatu sistem informasi rental mobil berbasis web untuk melakukan customeran mobil secara online yang mempermudah customer yang ingin menyewa mobil di suatu wilayah, akan tetapi tidak mengetahui tempat dimana untuk menyewa mobil di suatu daerah tersebut. Dan kami sudah menyediakan beberapa armada mobil dari berbagai macam merk dan jenis mobil untuk disewakan. Tujuannya untuk mempersingkat waktu customer dalam pencarian mobil pada daerah yang diinginkan.

## Definisi dan Istilah

1. SKPL

Spesifikasi Kebutuhan Perangkat Lunak adalah kondisi atau kemampuan yang harus dimiliki untuk memenuhi apa yang diisyaratkan oleh perangkat lunak atau diinginkan oleh pemakai.

1. ERD

Entity Relationship Diagram adalah salah satu model yang digunakan untuk mendesain database dengan tujuan menggambarkan data yang berelasi pada sebuah database.

1. DPPL

Dokumen Deskripsi Perancangan Perangkat Lunak merupakan dokumen deskripsi dari perancangan perangkat lunak yang akan dikembangkan.

## Referensi

- SKPL Rental Mobil Gayeng

- Template Dokumen Deskripsi Perancangan Perangkat Lunak jurusan Informatika Telkom University.

-https://www.academia.edu/9440127/DOKUMEN\_PERSYARATAN\_PERANGKAT\_LUNAK\_DPPL\_Program\_Studi\_Teknik\_Informatika\_Fakultas\_Teknologi\_dan\_Informasi

## Sistematika Pembahasan

DPPL ini menjelaskan tentang perancangan perangkat lunak yang merupakan transformasi dari SKPL. Dokumen ini terdiri dari bab dengan rincian sebagai berikut.

a. Bab 1 Pendahuluan, merupakan pengantar dokumen DPPL yang berisi tujuan penulisan dokumen dari sistem informasi Rental Mobil Gayeng, lingkup masalah, definisi dan istilah yang digunakan dalam dokumen ini, aturan penomoran dan penamaan, referensi dibuatnya dokumen ini, dan sistematika dokumen yang merupakan rincian dari penjelasan bab per-bab.

b. Bab 2 Deskripsi Perancangan Global yang berisi rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen dari Rental Mobil Gayeng.

c. Bab 3 Perancangan rinci yang berisi realisasi Use Case, Perancangan Antarmuka, Identifikasi Object, Sequence Diagram dan Perancangan Detail Kelas dari Rental Mobil Gayeng.

d. Bab 4 Matriks Keruntutan.

# Deskripsi Perancangan Global

## Deskripsi Arsitektural

*- Spesifikasi yang dibutuhkan untuk konstruksi Perangkat Lunak Rental Mobil Gayeng adalah:*

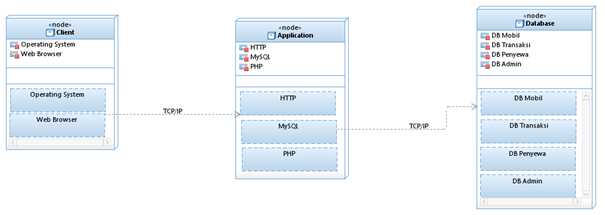
*o Sistem operasi : Microsoft Windows XP/Vista/7/8/10*

*o Web server : Personal Web Server (PWS)*

*o Scripting language : Hypertext Preprocessor (PHP)*

*o DBMS : MySQL*

*- Deployment Diagram*

**

Gambar 1 Deployment diagram

## Deskripsi Komponen

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

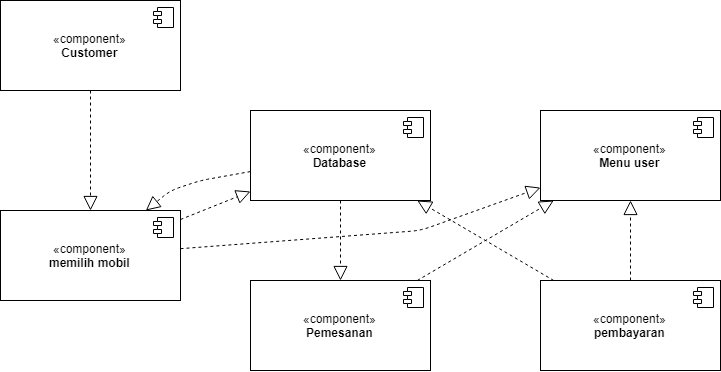
***Isinya komponen diagram***

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Komponen** | **Keterangan** |
| 1 | Login Admin | Admin dapat login untuk memasuki kedalam sistem |
| 2 | Login customer | Customer dapat login untuk memasuki kedalam sistem |
| 3 | Database | Tempat penyimpanan semua data pada sistem ini |
| 4 | Registrasi | customer dapat melakukan registrasi dahulu apabila tidak mempunyai akun |
| 5 | Memilih mobil | Sistem mengeluarkan list mobil yang disajikan untuk customer pada halaman memilih mobil |
| 6 | Informasi pemesanan | sistem menampilkan informasi pemesanan |
| 7 | Mobil | Database menyimpan semua data mobil yang tersedia di rental mobil gayeng |
| 8 | customer | Database menyimpan semua data customer yang dapat mengakses aplikasi rental mobil gayeng |
| 9 | Transaksi | Database menyimpan semua data bookingan di aplikasi rental mobil gayeng |
| 10 | Mengelola data customer | untuk mengelola data costumer yang sudah memiliki akun |

Table komponen diagram

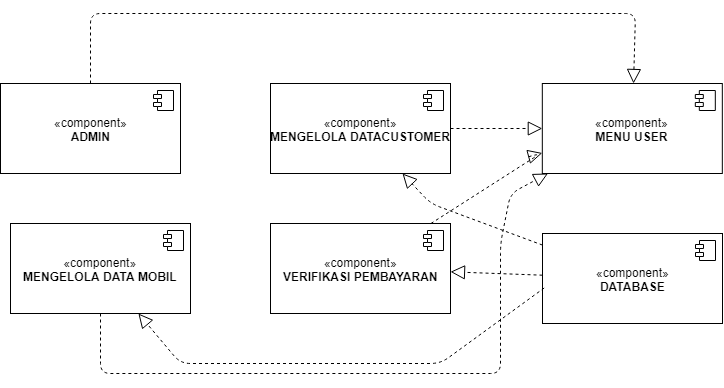
**-** Diagram Komponen

* Diagram Komponen Customer

****

Gambar diagram komponen customer

* Diagram Komponen Admin



Gambar komponen admin

# Perancangan Rinci

## Realisasi Use Case

Berisi TABEL USE CASE sebagai berikut :

|  |  |  |
| --- | --- | --- |
| No | Nama Use Case | Deskripsi Use Case |
| #1 | Memilih Mobil | User menginputkan data - data customeran seperti tanggal peminjaman, jenis mobil, dll yang sesuai dengan kriteria user dan tersedia |
| #2 | Pemesanan | User menerima rincian pesanan sesuai dengan apa yang diinputkan oleh user di menu memilih mobil |
| #3 | Melakukan Pembayaran | Sistem menerima pilihan mobil yang dipilih oleh customer dan system menampilkan biaya yang harus dibayar |
| #4 | Verifikasi Pembayaran | Sistem menerima masukan data konfirmasi pembayaran yang dilakukan oleh customer dan akan diverifikasi oleh admin |
| #5 | Mengelola Data Customer | Admin dapat mengelola data customer berupa edit data dan delete data customer |
| #6 | Mengelola Data Mobil | Admin dapat mengelola data mobil berupa input, edit data dan delete data mobil |
|  |  |  |
|  |  |  |

Table realisasi use case

### Use Case #1 Memilih Mobil

*Skenario Use Case #1*

*Pra-Kondisi : Customer telah berada pada halaman search mobil untuk melakukan pencarian mobil*

*Primary Flow :*

1. *Aktor memilih menu search mobil yang disewakan*
2. *Sistem menampilkan form pencarian*
3. *Aktor mengisi data pencarian mobil yang diinginkan*
4. *Sistem menampilkan mobil sesuai dengan inputan*
5. *Aktor memilih mobil yang diinginkan dan mengisi form booking*
6. *Sistem melakukan penyimpanan pada database*

*Pasca-Kondisi : Aktor telah memilih mobil yang diinginkan dan dapat melihat informasi pemesanan*

*Alternate Flow :*

1. *sistem akan mengirimkan notifikasi mobil tidak ditemukan apabila mobil yang dicari tidak ada pada database*

#### Perancangan Antarmuka Usecase #1

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case*

*Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya.*

##### Tabel Identifikasi Antarmuka / Layar / Page

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *MenuUser* | Gambar menu user | *Layar menampilkan halaman menu utama untuk user(customer). Di menu ini terdapat 3 tombol yang terdiri dari tombol Search Car, Booking Information, dan Log Out.* |
| *SearchCar* | Gambar earch car | *Tampilan layar ini menampilkan menu pencarian mobil. Pencarian mobil ini adalah fungsi utama dari aplikasi rental mobil. Disini, user yang ingin menyewa dapat mencari mobil sesuai dengan kriteria yang telah disediakan.* |
| *Choose Car* | Gambar choose car | *Tampilan ini adalah tampilan antarmuka setelah user menginputkan kriteria - kriteria pencarian mobil yang telah disediakan. Akan muncul tampilan dari hasil pencarian mobil yang dikehendaki oleh user* |

Table UI use case #1

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

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

*Page MENU USER*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *SearchCar* | *Button* | *SEARCH CAR* | *Jika diklik, akan menampilkan menu Search Car* |
| *Booking* | *Button* | *BOOKING* | *Jika diklik akan menampilkan menu Booking Information dan dapat melakukan pembayaran didalamnya* |
| *LogOut* | *Button* | *LOG OUT* | *Jika diklik, akan keluar dari sistem* |

Table page menu user

Page Search Car

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *Search\_* | *Button* | *SEARCH CAR* | *Jika diklik, akan menampilkan list mobil* |
| *Box\_City* | *Combo Box* | *CITY* | *Jika diklik, akan menampilkan list kota yang tersedia* |
| *Box\_DateStart* | *Combo Box* | *DATE START* | *Jika diklik, akan menampilkan list tanggal* |
| *Box\_DateEnd* | *Combo Box* | *DATE END* | *Jika diklik, akan menampilkan list tanggal* |
| *Box\_Pickuptime* | *Combo Box* | *PICKUP TIME* | *Jika diklik, akan menampilkan list jam* |
| *Box\_Passanger* | *Combo Box* | *AMOUNT OF PASSANGER* | *Jika diklik, akan menampilkan list jumlah penumpang* |

Table page search car

*Page Choose Car*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Id\_Objek*** | ***JENIS*** | ***LABEL\**** | ***Keterangan\*\**** |
| *Select\_* | *Button* | *SELECT* | *Jika diklik, akan menampilkan form customeran* |
|  | *Textfield* |  | *Keterangan data mobil* |

Table choose car

#### Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

###### **TABEL OBJECT PERANCANGAN MEMILIH MOBIL**

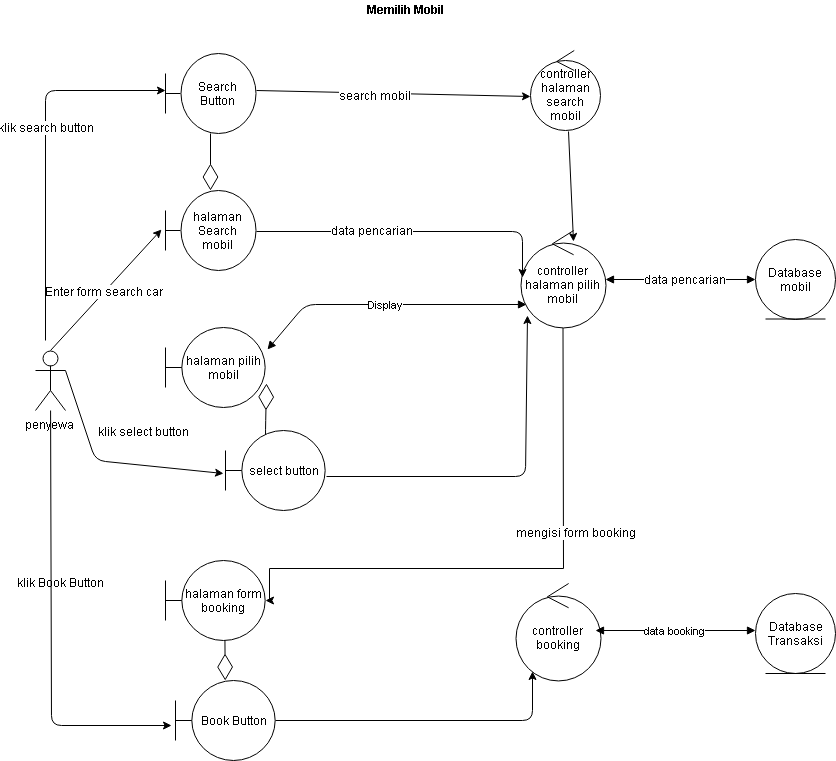
### 

|  |  |  |
| --- | --- | --- |
| ***No*** | ***Nama Object Baru*** | ***Jenis / Tipe Kelas*** |
| *1.* | *Halaman Search Mobil* | *Boundary* |
| *2.* | *Controller halaman pilih mobil* | *Controller* |
| *3.* | *Database Mobil* | *Entity* |
|  | *Halaman Form Booking* | *Boundary* |
| *5.* | *Controller Booking* | *Controller* |
| *6.* | *Database Transaksi* | *Entity* |

Table memilih mobil

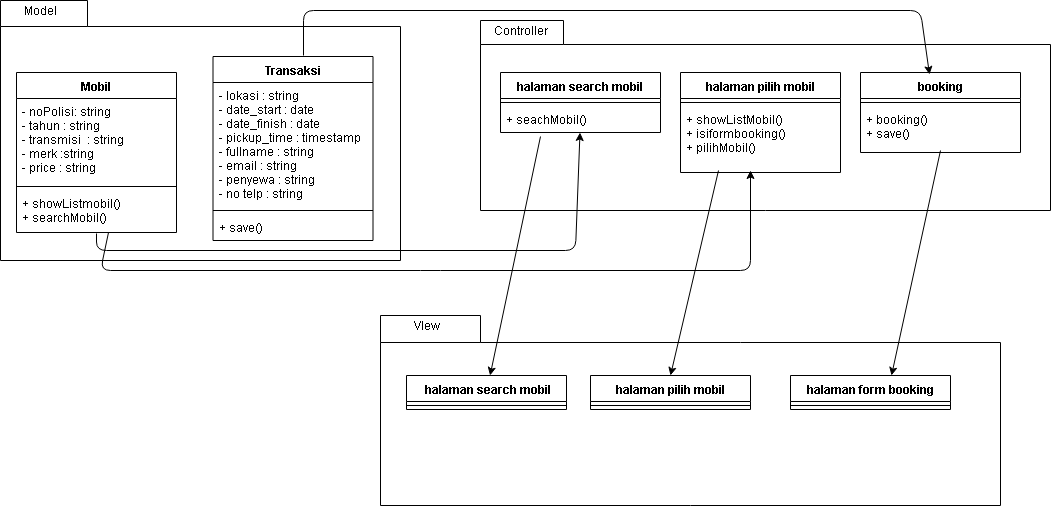
#### Robustness Diagram

###### **1)** **Robustness Memilih Mobil**



Gambar robustness memilh mobil

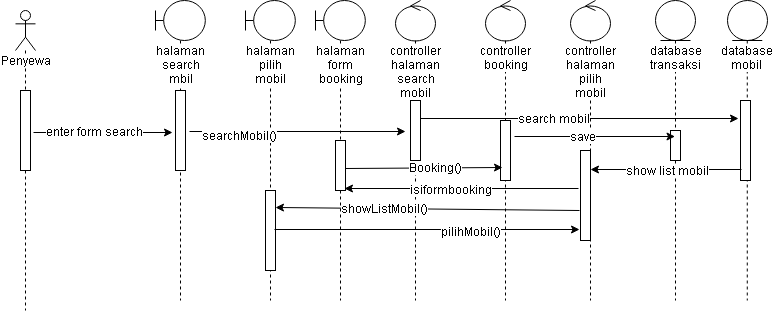
#### Diagram Kelas

Class Diagram Memilih Mobil

Gambar class diagram memilih mobil

#### Sequence Diagram

###### **1)** **sequence diagram memilih mobil**



Gambar sequence diagram memilih mobil

### Use Case #2 Pemesanan

*Skenario Use Case #2*

*Pra-kondisi: customer sudah melakukan pemilihan mobil dan berada di halaman informasi pemesanan*

*Primary Flow:*

1. *Sistem melakukan loading data dari database Transaksi*
2. *customer mengecek detail pemesanan*
3. *customer mengklik tombol "Payment" setelah mengecek detail pemesanan*
4. *Sistem mengarahkan customer untuk masuk ke halaman pembayaran*

*Pasca-kondisi: customer berada di halaman pembayaran*

*Alternate Flow:*

1. *Apabila ada detail pemesanan yang tidak sesuai dengan keinginan customer, maka customer kembali ke halaman memilih mobil untuk menginputkan data - data yang diinginkan oleh customer*

#### Perancangan Antarmuka Usecase #2

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case*

*Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya.*

##### Tabel Identifikasi Antarmuka / Layar / Page

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

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

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *InfPemesanan* |  | *Layar menampilkan halaman informasi pemesanan. Detail informasi pemesanan yang sudah diinputkan oleh customer pada saat memilih mobil ditampilkan di halaman ini.* |

Table UI use case #2

*Page PEMESANAN*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *Payment* | *Button* | *PAYMENT* | *Jika diklik, akan memanggil halaman pembayaran* |

Table page pemesanan

#### Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

TABEL OBJECT PERANCANGAN

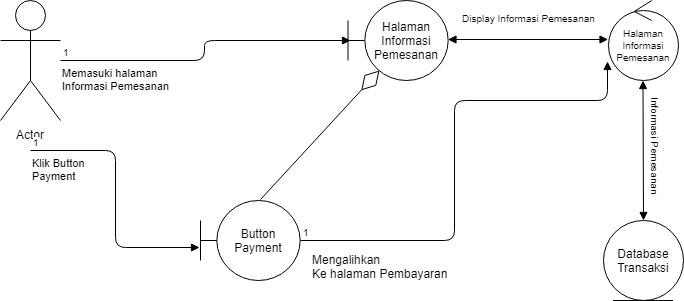
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1.* | *Halaman Informasi Pemesanan* | *Boundary* |
| *2.* | *Halaman Informasi Pemesanan* | *Controller* |
| *3.* | *Database Transaksi* | *Entity (Database)* |

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

Table object perancangan #2

#### Robustness Diagram

*Buatlah diagram robustness untuk masing – masing use case*

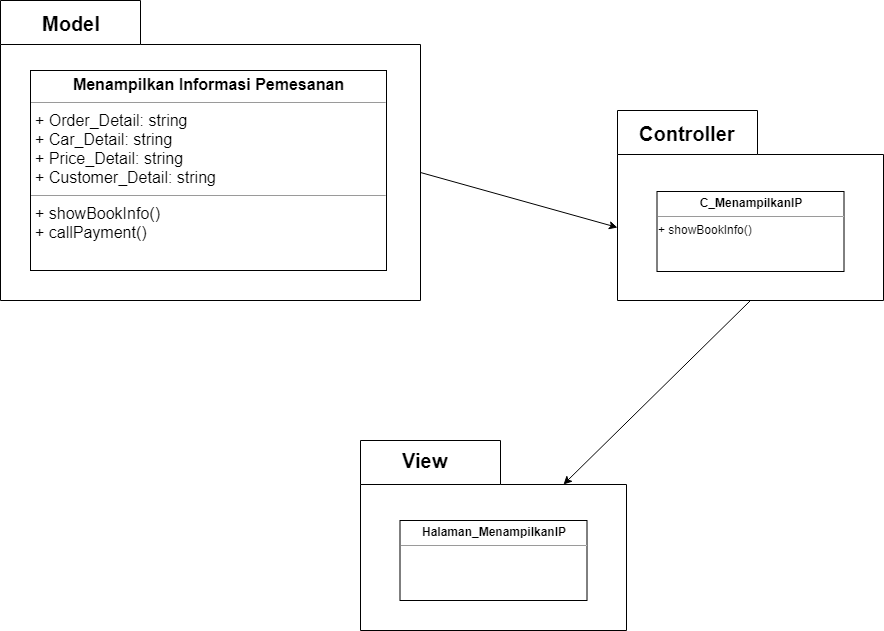


Gambar robustness pemesanan

#### Diagram Kelas

*Buatlah diagram kelas untuk masing – masing use case*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis*

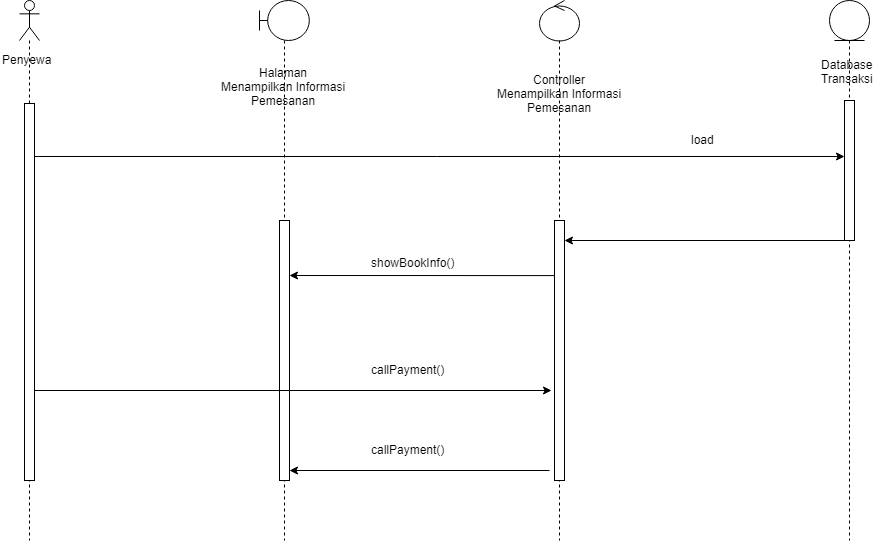


Gambar class diagram informasi pemesanan

#### Sequence Diagram

*Buatlah diagram sequence untuk masing - masing use case sesuai skenario.*

*Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.*



Gambar sequence diagram memilih mobil

### Use Case #3 Melakukan Pembayaran

*Pra-kondisi: customer telah berada pada halaman Pembayaran, dan telah melakukan transfer*

*Primary Flow*

1. *customer mengirim bukti pembayaran dengan mengklik tombol upload invoice*
2. *Sistem menerima bukti pembayaran dan disimpan pada database Transaksi*
3. *Sistem mengirimkan tampilan upload successfull*

*Pasca-kondisi: bukti transfer yang di upload tersimpan pada Database*

*Alternate Flow*

*I. Apabila customer tidak melakukan proses pembayaran dengan tenggang waktu yang disediakan, maka proses pembayaran akan dibatalkan.*

#### Perancangan Antarmuka Use case #3

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case*

*Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya.*

##### Tabel Identifikasi Antarmuka / Layar / Page

|  |  |  |
| --- | --- | --- |
| ID. LAYAR | *NAMA LAYAR* | DESKRIPSI |
| 03-Payment | Gambar payment | Tampilan ini adalah tampilan pembayaran dimana user harus membayar sesuai dengan tenggat waktu yang tertera. Di dalam tampilan ini juga tertera besaran jumlah yang harus dibayarkan dan nomor rekening tujuan. |

Table UI use case #3

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

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

*Page Payment*

|  |  |  |  |
| --- | --- | --- | --- |
| ***Id\_Objek*** | ***JENIS*** | ***LABEL\**** | ***Keterangan\*\**** |
| *UploadInvoice\_* | *Button* | *UPLOAD INVOICE* | *Jika diklik, akan menampilkan tampilan upload invoice* |
|  | *Text Area* | *FINISH YOUR TRANSACTION* | *Penjelasan tentang pembayaran sewa mobil* |

Table page payment

#### Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

TABEL OBJECT PERANCANGAN

###### **TABEL OBJECT PERANCANGAN PEMBAYARAN**

### 

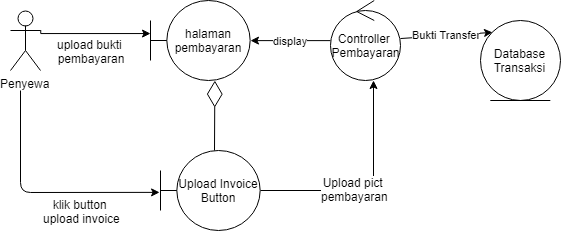
|  |  |  |
| --- | --- | --- |
| ***No*** | ***Nama Object Baru*** | ***Jenis / Tipe Kelas*** |
| *1.* | *Halaman Pembayaran* | *Boundry* |
| *2.* | *Controller Pembayaran* | *Controller* |
| *3.* | *Database Transaksi* | *Entity* |

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

Table pembayaran\

#### Robustness Diagram

*Buatlah diagram robustness untuk masing – masing use case*

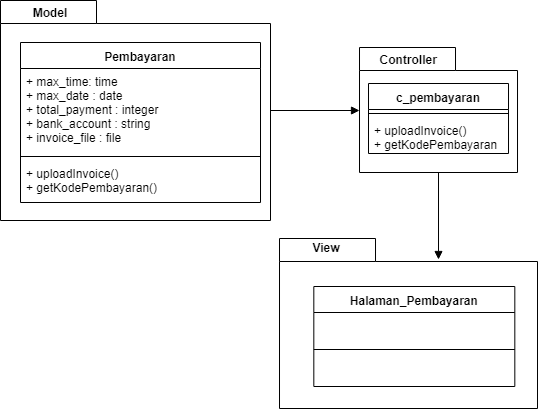
**

Gambar robustness pembayaran

#### Diagram Kelas

*Buatlah diagram kelas untuk masing – masing use case*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis*

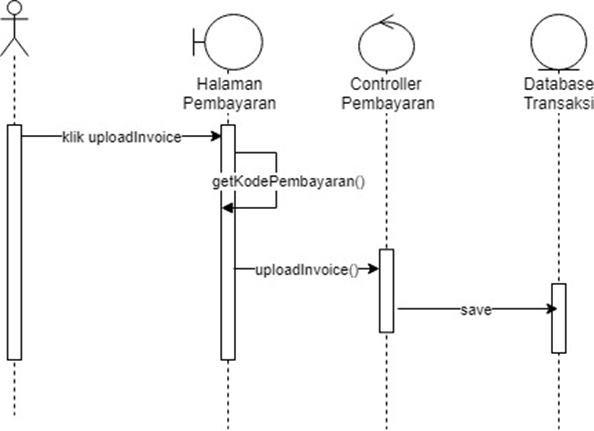
**

Gambar class diagram pembayaran

#### Sequence Diagram

*Buatlah diagram sequence untuk masing - masing use case sesuai skenario.*

*Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.*

**

Gambar sequence pembayaran

### Use Case #4 <Verifikasi Pembayaran>

*Skenario Use Case #4*

*Pra-Kondisi : Admin berada di halaman utama admin.*

*Primary Flow*

1. *Admin memilih menu verifikasi pembayaran.*
2. *Sistem menerima inputan admin dan mencari semua pesanan di database transaksi.*
3. *Sistem menampilkan menu verifikasi pembayaran yang berisi semua pesanan di database transaksi.*
4. *Admin meng-klik tombol confirm untuk transaksi yang sudah upload bukti pembayaran ke sistem.*
5. *Sistem menerima inputan admin dan memunculkan pesanan yang dipilih di confirmed order.*
6. *Admin meng-klik tombol delete untuk transaksi yang melewati batas akhir pembayaran.*
7. *Sistem menerima inputan admin dan menghapus pesanan yang dipilih dari database.*

*Pasca-Kondisi: Transaksi yang sudah lunas berhasil dikonfirmasi dan transaksi belum lunas yang melewati batas maksimal pembayaran sudah dihapus dari transaksi.*

*Alternate Flow*

1. *Admin hanya melakukan verifikasi untuk transaksi yang sudah melunasi pembayaran dan menghapus transaksi belum lunas yang melewati batas waktu maksimal pembayaran.*

#### Perancangan Antarmuka Use case #4

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case*

*Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.*

##### Tabel Identifikasi Antarmuka / Layar / Page

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *04-VerifikasiPayment* | Gambar verifikasi pembayaran | *Layar menampilkan tampilan order sewa mobil dari customer yang digunakan untuk memverifikasi transaksi yang sudah melunasi pembayaran, dan menghapus transaksi belum lunas yang melewati batas waktu maksimal pembayaran. Order tersebut muncul apabila ada order dari customer.* |

Table UI Use case #4

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

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

*Page Verifikasi Pembayaran*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *Confirm* | *TextBox* | *CONFIRM* | *Jika di-klik akan menyimpan data transaksi ke tabel order yang sudah dikonfirmasi di database.* |
| *Delete* | *TextBox* | *DELETE* | *Jika di-klik akan menghapus data transaksi dari tabel transaksi di database.* |
| *Description* | *Textfield* |  | *Rincian order dari customer.* |

Table page verifikasi pembayaran

#### Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

TABEL OBJECT PERANCANGAN VERIFIKASI PEMBAYARAN

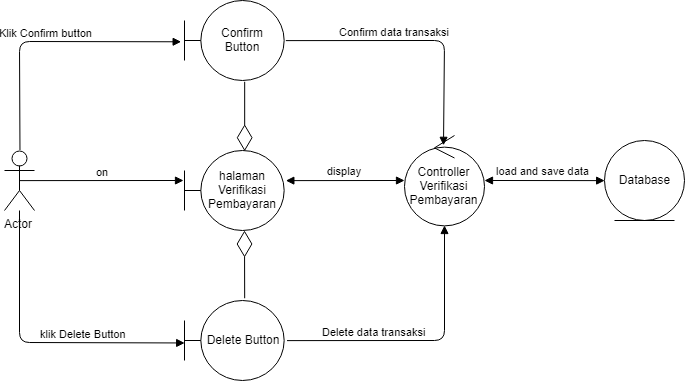
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| 1. | *Halaman Verifikasi Pembayaran* | *Boundry* |
| 2. | *Controller Verifikasi Pembayaran* | *Controller* |
| 3. | *Database Transaksi* | *Entity* |

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

Table object perancangan verifikasi pembayaran

#### Robustness Diagram

*Buatlah diagram robustness untuk masing – masing use case*

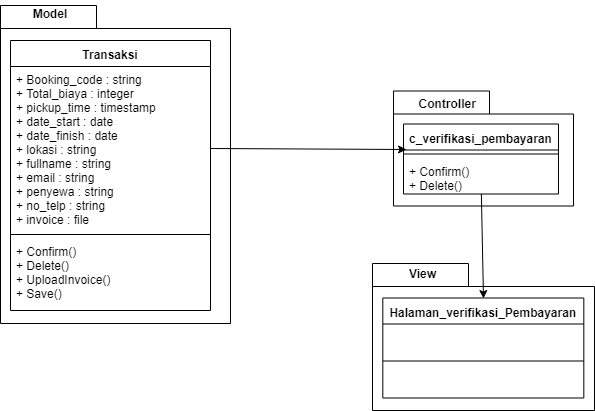


Gambar robustness diagram verifikasi pembayaran

#### Diagram Kelas

*Buatlah diagram kelas untuk masing – masing use case*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis*

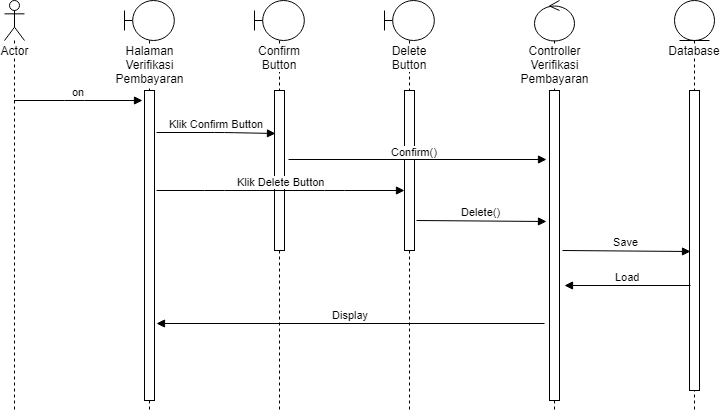


Gambar class diagram verifikasi pembayaran

##### Sequence Diagram

*Buatlah diagram sequence untuk masing - masing use case sesuai skenario.*

*Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.*



Gambar sequence verifikasi pembayaran

### 3.1.5 Use Case #Mengelola Data Customer

*Skenario Use Case #5*

Pra-kondisi: Admin berada pada halaman manage data customers, data customer telah tersimpan pada database

Primary Flow

1. SIstem mengirim list data customer pada halaman manageData
2. Admin mengklik button edit pada list nama customer yang dibutuhkan
3. Sistem mengirim form data customer yang dipilih pada halaman edit data
4. Admin mengedit data customer seperlunya, setelah itu meng-klik button save.
5. Sistem menyimpan data yang telah di edit pada database

Pasca-kondisi: Data customer yang telah di edit,telah tersimpan pada database

Alternate Flow

I. Admin mengklik button delete pada list nama customer

II. System menghapus data customer tersebut pada database

#### 

#### 3.1.5.1 Perancangan Antarmuka Usecase #5

##### 3.1.5.1.1 Tabel Identifikasi Antarmuka/Layar/page

|  |  |  |
| --- | --- | --- |
| ID. LAYAR | *NAMA LAYAR* | DESKRIPSI |
| Manage Customer Data | Gambar manage customer data | Tampilan ini adalah tampilan Manage customers Data, yang dimana dalam halaman ini menampilkan list customer yang tersimpan pada database.admin dapat melakukan edit atau delete customer dengan menklik button yang tersedia |
| Edit Data | Gambar edit data | Layar halaman edit data. Halaman edit data ini digunakan untuk mengedit data customer yang dilakukan oleh admin |

Table UI usecase #5

*Page Manage Customer Data*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *edit* | *Button* | *Edit data* | *Jika diklik, akan menampilkan halaman edit data* |
| *delete* | *Button* | *delete* | *Jika diklik akan menghapus semua data customer yang dipilih* |

Table page manage customer data

*Page edit data*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| save | button | save data | Jika diklik akan menyimpan semua perubahan data customer yang telah di edit oleh admin |
| box\_fullname | combo box | full name | jika diklik akan menjadi tempat pengisian fullname |
| box\_email | combo box | email | jika diklik akan menjadi tempat pengisian email |
| box\_username | combo box | username | jika diklik akan menjadi tempat pengisian username |
| box\_password | combo box | password | jika diklik akan menjadi tempat pengisian password |
| box\_mobilenumber | combo box | mobile number | jika diklik akan menjadi tempat pengisian nomor telfon |
| box\_idcard | combo box | id card | jika diklik akan menjadi tempat pengisian id card |
| box\_gender | combo box | gender | jika diklik akan menampilkan list gender |

Table page edt data

#### 3.1.5.2 Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

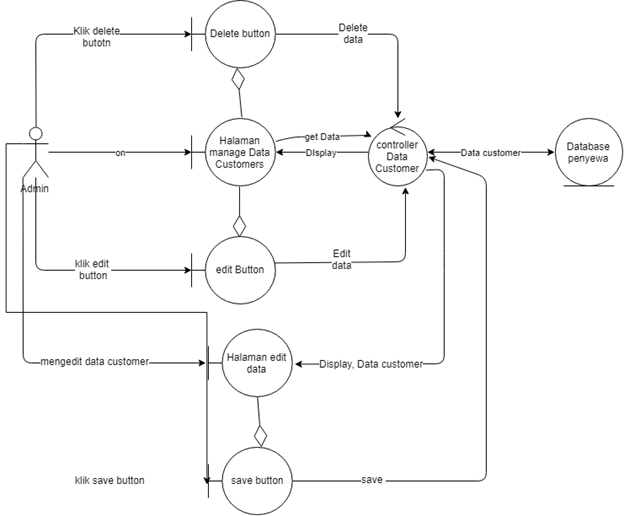
###### **TABEL OBJECT PERANCANGAN VERIFIKASI PEMBAYARAN**

### 

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| 1. | Halaman Data Customer | Boundry |
| 2. | Controller Data Customer | Controller |
| 3. | Database customer | Entity |

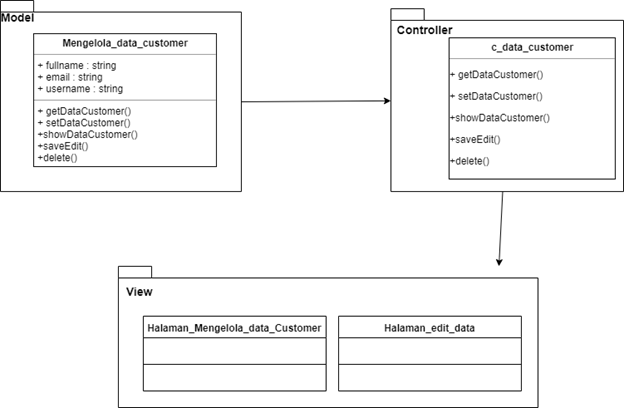
Table object manage data

#### 3.1.5.3 Robustness Diagram

****

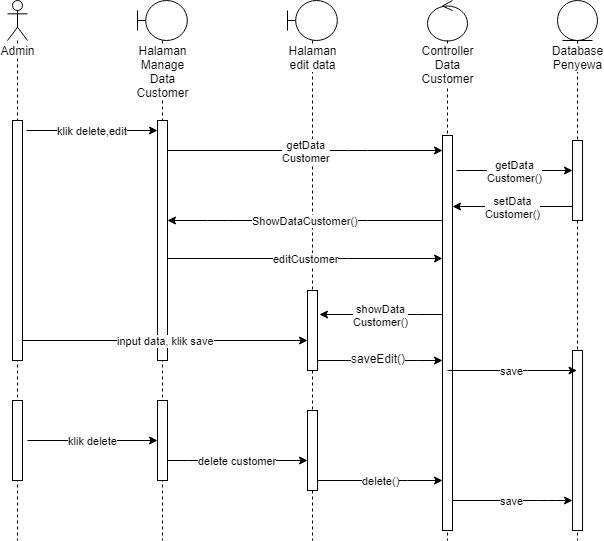
Gambar robustness manage data customer

#### 3.1.5.4 Diagram Kelas

****

Gambar class diagram manage data customer

#### 3.1.5.5 Sequence Diagram

****

Gambar sequence diagram manage data customer

### 3.1.6 Use Case #Mengelola Data Mobil

*Skenario Use Case #6*

Pra-kondisi: Admin sudah berada pada halaman manage mobil, data mobil telah tersimpan pada database

Primary Flow

I. System menampilkan form pengisian data mobil pada halaman manage mobil

II. Admin mengisi form data mobil apabila ingin memasukan data mobil baru, dan mengklik tombol save

III. System menerima data mobil dan menyimpan pada database

IV. Admin akan mengisi fom data mobil berdasarkan plat nomor mobil yang akan di edit, dan mengklik tombol update

V. System menerima data mobil dan menyimpan pada database

VI. Admin mengisi plat nomor mobil,sesuai dengan data mobil yang akan dihapus,dan mengklik tombol delete

VII. System menerima data dan menyimpan pada database

Pasca-kondisi: Data mobil yang telah di manage akan tersimpan didatabase,dan akan ditampilkan pada search mobil.

Alternate Flow

I. Apabila form pengisian data mobl tidak diisi semua oleh admin maka,system akan menampilkan alert

II. Apabila mobil yang akan di edit atau delete tidak tersedia, maka system akan mengirimkan alert mobil tidak ditemukan.

#### 3.1.6.1 perancangan Antarmuka Usecase #6

##### 3.1.6.1.1 Tabel Identifikasi Antarmuka/Layar/page

|  |  |  |
| --- | --- | --- |
| ID. LAYAR | *NAMA LAYAR* | DESKRIPSI |
| Manage mobil | Gambar manage mobil | Tampilan ini adalah tampilan Manage mobil, yang dimana dalam halaman ini terdapat form pengisian data mobil ,yang akan diisi oleh actor .actor dapat melakukan delete,input,dan update. |

Table UI use case #6

*Page Manage mobil Data*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *update* | *Button* | *Edit data* | *Jika diklik, akan menyimpan data yang telah diupdate* |
| *delete* | *Button* | *delete* | *Jika diklik akan menghapus semua data moboil yang dipilih* |
| *save* | *Button* | *save* | *Jika diklik akan menyimpan data mobil yang baru diinputkan* |
| *box\_brand* | *combo box* | *brand* | jika diklik akan menjadi tempat pengisian merk mobil |
| *box\_licenseplate* | *combo box* | *license plate* | jika diklik akan menjadi tempat pengisian plat nomor |
| *box\_year* | *combo box* | *year* | jika diklik akan menjadi tempat pengisian year |
| *box\_dateavailable* | *combo box* | *date* | jika diklik akan menjadi tempat pengisian tanggal mobil |
| *box\_city* | *combo box* | *city* | jika diklik akan menjadi tempat pengisian kota |
| *box\_rentPrice* | *combo box* | *rent price* | jika diklik akan menjadi tempat pengisian rent price |

Table page manage mobil data

#### 3.1.6.2 Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

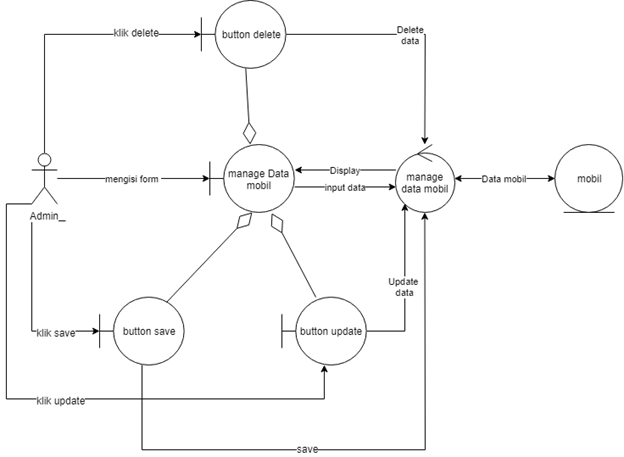
###### **TABEL OBJECT PERANCANGAN MANAGE DATA MOBIL**

### 

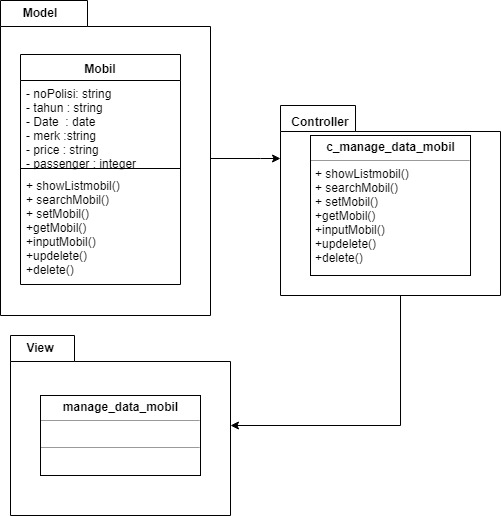
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| 1. | Halaman manage moibl | Boundry |
| 2. | Controller Data mobil | Controller |
| 3. | Database mobil | Entity |

Table object perancangan manage data mobil

#### 3.1.6.3 Robustness Diagram

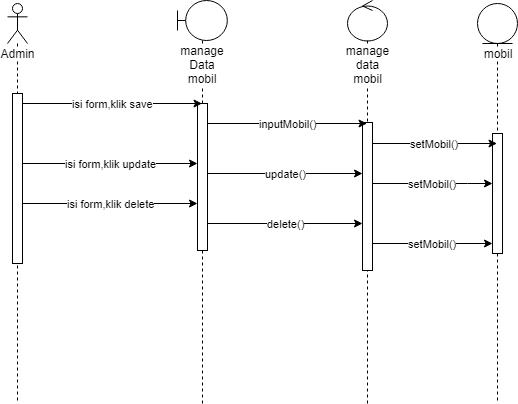
****

Gambar robustness diagram manage data

**3.1.6.4 Diagram Kelas **

Gambar class diagram manage data mobil

#### 3.1.6.5 Sequence Diagram



Gambar sequence manage data mobil

# Perancangan Detail

## Perancangan Detail Kelas

*Bagian ini diisi dengan daftar keseluruhan kelas yang akan digunakan dalam PL. menggunakan model MVC*

**TABEL KELAS :**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID Kelas** | **Nama Kelas Perancangan** | **Atribute (visibility)** | **Method / Operation** |
| *001* | *mobil* | *public* | * *showListMobil()* * *searchMobil()* * *setMobil()* * *getMobil()* * *inputMobil()* * *updete()* * *delete()* |
| *002* | *Transaksi* | *public* | * *cofirm()* * *save()* * *delete()* * *uploadInvoice()* |
| *003* | *halaman search mobil* | *public* | * *searchMobil()* |
| *004* | *halaman pilih mobil* | *public* | * *showListMobil* * *isiformbooking()* * *pilihmobil()* |
| *005* | *booking* | *public* | * *booking()* * *save()* |
| *006* | *menampilkan informasi pemesanan* | *public* | * *showBookinfo()* * *callPayment()* |
| *007* | *C\_MenampilkanIP* | *public* | * *showBookinfo()* |
| *008* | *transaksi* | *public* | * *confirm()* * *delete()* * *uploadInvoice()* * *save()* |
| *009* | *c\_verfikasi\_pembayaran* | *public* | * *confirm()* * *delete* |
| *010* | *mengelola\_data\_customer* | *public* | * *getDataCustomer()* * *setDataCustomer()* * *showDataCustomer()* * *saveEdit()* * *delete()* |
| *011* | *c\_data\_customer* | *public* | * *getDataCustomer()* * *setDataCustomer()* * *showDataCustomer()* * *saveEdit()* * *delete()* |
| *012* | *halaman\_verifikasi\_pembayaran* | *public* |  |
| *013* | *halaman\_Mengelola\_Data\_Customer* | *public* |  |
| *014* | *halaman\_edit\_data* | *public* |  |
| *015* | *c\_manage\_data\_mobil* | *public* | * *showListMobil()* * *searchMobil()* * *setMobil()* * *getMobil()* * *inputMobil()* * *updete()* * *delete()* |
| *016* | *manage\_data\_mobil* | *public* |  |
| *017* | *halaman search mobil* | *public* |  |
| *018* | *halaman pilih mobil* | *public* |  |
| *019* | *halaman form booking* | *public* |  |
| *020* | *halaman\_menampilkanIP* | *public* |  |
| *021* | *pembayaran* | *public* | *- uploadInvoice()*  *- getKodePembayaran()* |
| *022* | *c\_pembayaran* | *public* | *- uploadInvoice()*  *- getKodePembayaran()* |
| *023* | *Halaman\_Pembayaran* | *Public* |  |
|  |  |  |  |

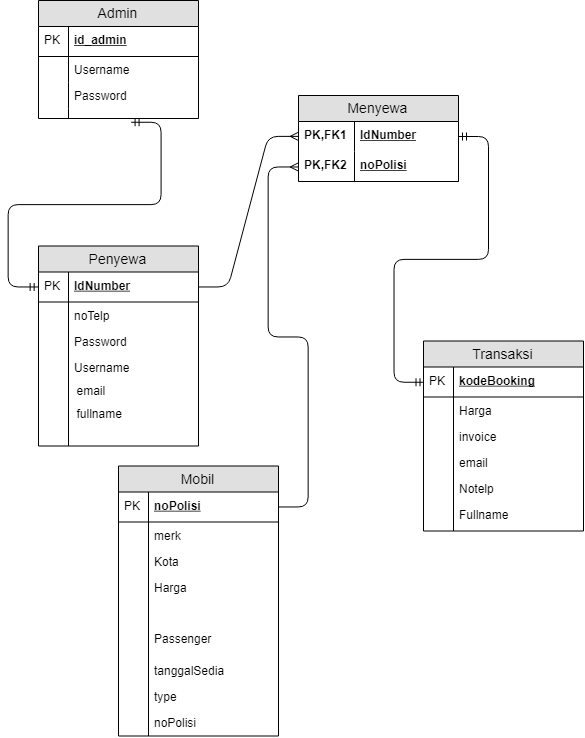
Table Perancagan class diagram

*Untuk setiap kelas:*

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

## Perancangan Kelas Persistensi (\*\*\*Basis Data Skema Tabel)

Skema Relasi database yang digunakan



Gambar skema relasi

## Perancangan Algoritma

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

### Algoritma #1

Contoh:

*Nama Kelas : halaman pilih mobil*

*Nama Operasi : memilihMobil()*

*Algoritma :*

|  |
| --- |
| Procedure memilihMobil(price p, date\_finish df, date\_start ds)  var  biaya : integer  begin  biaya = p \* (df - ds)  biaya = biaya + idNumber/1000  -> biaya  end |

## Perancangan Query

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

*Query :*

|  |  |  |  |
| --- | --- | --- | --- |
| *No Query* | *Nama Use Case* | *Query* | *Keterangan* |
| *Q-001* | *Memilih mobil* | *SELECT \* FROM MOBIL WHERE merk=merk* | *Query untuk menampilkan sebagian data mobil di dalam database* |
| *Q-002* | *Memilih Mobil* | *INSERT INTO TRANSAKSI(lokasi,date\_start,date\_finish,pickup\_time,fullname,email,customer,notelp)* | *Query untuk menambahkan data Booking ke database transaksi* |
| *Q-003* | *Pemesanan* | *Select \* FROM TRANSAKSI WHERE idOrder=idOrder;* | *Query untuk menampilkan data informasi pemesanani dari database transaksi* |
| *Q-004* | *Verifikasi Pembayaran* | *SELECT \* FROM TRANSAKSI WHERE BookingCode=BookingCode* | *Query untuk menampilkan data pembayaran pada admin dari database transaksi* |
| *Q-005* | *Verifikasi Pembayaran* | *UPDATE TRANSAKSI SET status =valueStatus, where booking\_code == valueidBookingcode* | *Query untuk update Status pembayaran customer dari datahase transaksi* |
| *Q-006* | *Mengelola Data Customer* | *UPDATE CUSTOMER SET \*=values\*, where idCustomer=idCustomer* | *Query untuk mengupdate data customer dari database customer* |
| *Q-007* | *Mengelola Data Customer* | *DELETE \* FROM CUSTOMER WHERE idCustomer=valueidCustomer* | *Query untuk menghapus data customer dari database customer* |
| *Q-008* | *Mengelola Data Mobil* | *INSERT INTO MOBIL values (noPolisi,tahun,DAte,merk,price,passanger)* | *Query untuk menambahkan data mobil ke dalam database mobil* |
| *Q-009* | *Mengelola Data Mobil* | *UPDATE MOBIL SET \*=values\* ,where noPolisi=valuesnoPolisi* | *Query untuk mengupdate data mobil di database mobil* |
| *Q-010* | *Mengelola Data Mobil* | *DELETE \* FROM MOBIL WHERE noPolisi=valuesnoPolisi* | *Query untuk menghapus data mobil dari database mobil* |

Table query

# Matriks Kerunutan (Requirement Traceability Matrix)

*Mapping requirement dengan Use Case yang direalisasikan*

|  |  |  |
| --- | --- | --- |
| **Kode FR** | **Nama Functional Requirement** | **Nama Use Case** |
| FR-001 | memilih mobil | memilih mobil |
| FR-002 | menampilkan informasi pemesanan | menampilkan informasi pemesanan |
| FR-003 | proses pembayaran | proses pembayaran |
| FR-004 | verifikasi pemabayaran | verifikasi pemabayaran |
| FR-005 | megelola data customer | megelola data customer |
| FR-006 | mengelola data admin | mengelola data admin |

Table matriks kerunutan

# 