**SKPL-001**

**SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK**

**RentWheel.com**

untuk:

Pemilik Bisnis Rental Kendaraan

Dipersiapkan oleh:

Tubagus Ibrohim – 1301174655  
Mochammad Alfi Rizky R – 1301174637  
Muhammad Arief Rahman – 1301173704  
Syafrullah Anwar – 1301170562

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Program Studi S1 Informatika**  **-**  **Fakultas Informatika** | **Nomor Dokumen** | | **Halaman** |
| **SKPL-001** | | < isi halaman > |
| **Revisi** | < nomor revisi > | Tgl : < isi tanggal> |

# Daftar Perubahan

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

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

# Daftar Halaman Perubahan

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

# Daftar Isi

[Daftar Perubahan 2](#_Toc32396772)

[Daftar Halaman Perubahan 3](#_Toc32396773)

[Daftar Isi 4](#_Toc32396774)

[1. Pendahuluan 5](#_Toc32396775)

[1.1. Tujuan Penulisan Dokumen 5](#_Toc32396776)

[1.2. Ruang Lingkup / Cakupan Dokumen 5](#_Toc32396777)

[1.3. Definisi, Singkatan, dan Akronim 5](#_Toc32396778)

[1.4. Referensi 5](#_Toc32396779)

[2. Deskripsi Global Perangkat Lunak 6](#_Toc32396780)

[2.1. Statement of Objective Perangkat Lunak 6](#_Toc32396783)

[2.2. Perspektif dan Fungsi Perangkat Lunak 6](#_Toc32396784)

[2.3. Profil dan Karakteristik Pengguna 6](#_Toc32396785)

[2.4. Arsitektur Lingkungan Operasi Perangkat Lunak (digambarkan) 6](#_Toc32396786)

[2.5. Kebutuhan Perangkat Keras 6](#_Toc32396787)

[2.6. Asumsi dan Batasan Perangkat Lunak 7](#_Toc32396788)

[3. Deskripsi Rinci Perangkat Lunak 8](#_Toc32396789)

[3.1. Deskripsi Kebutuhan 8](#_Toc32396791)

[3.1.1. Kebutuhan Fungsional 8](#_Toc32396792)

[3.1.2. Kebutuhan Non-Fungsional 8](#_Toc32396798)

[3.2 Pemodelan Analisis 8](#_Toc32396799)

[3.2.1. Usecase Diagram 8](#_Toc32396801)

[3.3. Class Diagram 9](#_Toc32396802)

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen in berisi spesifikasi kebutuhan perangkat lunak (SKPL) atau *software requirement* *specification* (SRS) yang akan dikembangkan. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak pada tahap selanjutnya.

## Ruang Lingkup / Cakupan Dokumen

RentWheel.com merupakan aplikasi berbasis web untuk menyediakan informasi-informasi rental kendaraan dengan menyediakan informasi kendaraan bagi yang membutuhkan kendaran dan menyediakan kendaraan untuk disewakan. RentWheel.com . Program ini dirancang dalam bahasa pemprograman berbasis web.

## Definisi, Singkatan, dan Akronim

Tabel 1. Daftar definisi dan Akronim

|  |  |
| --- | --- |
| **Kata Kunci** | **Definisi atau Akronim** |
| SKPL | Spesifikasi Kebutuhan Perangkat Lunak atau dalam bahasa Inggris-nya sering juga disebut sebagai *Software Requirements Specification* (SRS) dan merupakan spesifikasi perangkat lunak yang akan dikembangkan |
| SKPL-SPSPU-xxx | kode yang digunakan untuk merepresentasikan kebutuhan (*requirement*) pada SPSPU dengan SPSPU merupakan kode perangkat lunak SPSPU adalah kode fase, dan xxx adalah digit/nomor kebutuhan (requirement) |
| SRS | Software Requirement Specification |
| IEEE | *Institute of Electrical and Electronics Engineers*  Standar internasional untuk pengembangan dan rancangan produk |
| ANSI | *American Standart Institute*  Lembaga standardisasi Amerika |
| RPL | Rekayasa Perangkat Lunak, kegiatan pengembangan perangkat lunak |

Tabel 2. Daftar Istilah

|  |  |
| --- | --- |
| **Istilah** | **Definisi** |
| *Vehicle Provider* | *Stakeholder* yang merepresentasikan *user* yang menyediakan kendaraan. Merupakan salah satu jenis *user* yang di kategorikan dalam pengoperasian sistem informasi RentWheel.com |
| *Customer* | Istilah yang mengacu pada *stackholder* utama yang berperan dalam menjalankan bisnis utama. |
| *Admin* | *Stakeholder* yang merepresentasikan *user* yang berada pada sistem. *Admin* berfungsi dalam mengawasi dan dapat memanipulasi sistem yang ada. |

## Referensi

Beberapa referensi yang kami gunakan dalam pembuatan dokumen ini antara lain :

* *IEEE Std 830-1993, IEEE Recommended Practice for Software Requirement Specifications.*
* *IEEE Std 610.12-1990 IEEE Standard Glossary of Software Engineering Terminology (ANSI).*
* *Jurusan Teknik Informatika – Institut Teknologi Bandung Panduan GL01, Panduan Penggunaan dan Pengisian Spesifikasi Kebutuhan Perankat Lunak.*

# Deskripsi Global Perangkat Lunak



## Statement of Objective Perangkat Lunak

RentWheel.com adalah suatu perangkat lunak berbasis *website* yang menampung informasi mengenai jasa rental kendaraan disediakan oleh *Vehicle* yang menyediakan kendaraan. RentWheel.com merupakan aplikasi yang dibuat dan dikembangkan berdasarkan sistem yang sudah ada, karena aplikasi ini dibuat atas dasar permintaan perusahaan. RentWheel.com akan menyimpan data inputan user (*Vehicle Provider , Customer*) berupa data akun yang akan masuk ke database dan siap diproses oleh Admin sebagai operator. Dan menyimpan data inputan *Vehicle Provider* berupa data kendaraan yang akan masuk ke database dan siap diproses oleh Admin.

## Perspektif dan Fungsi Perangkat Lunak (FULLIN)

RentWheel.com merupakan aplikasi yang berbasis web yang memudahkan pemilik rental sewa kendaraan maupun pencari kendaraan untuk disewa dalam menerima informasi mengenai info penyewaan tersedia. Pada sistem ini memiliki berbagai jenis user sesuai kebutuhan yaitu user *admin*, dan user *customer*.

Fungsi yang didapat jika menggunakan perangkat lunak ini adalah:

1. Fungsi untuk *Customer*
2. Mendaftarkan diri sebagai user dari website RentWheel.com
3. Mendapatkan infomasi tentang kendaraan yang akan tersedia untuk disea
4. Meminjam dan memilih paket kendaraan
5. Melakukan pembayaran sebelum menggunakan kendaraan
6. *Admin*
7. Admin mendapatkan fee sebagai media partner yang sifatnya saling menguntungkan berbagai pihak
8. Mengelola database kendaraan
9. Melakukan peningkatan kualitas sistem sesuai kebutuhan user.

## Mempermudah dalam proses transaksi untuk ticketing event, serta dalam pembagian sertifikat akan mudah karena tinggal distore dalam bentuk digital dan peserta tinggal mendownload file.Profil dan Karakteristik Pengguna

Pengguna paling penting dari website ini adalah orang-orang yang mempunyai Kartu Identitas Penduduk dan Surat Izin Mengemudi yang sedang mencari kendaraan yang sesuai dengan keinginannya (*Customer*), dan orang-orang menyewakan kendaraanya (*Vehicle Provider*).

Terdapat juga pengguna yang berperan sebagai operator, dapat mengawasi dan memanipulasi sistem yaitu *admin*. Dalam proses bisnis utama *admin* dapat menyetujui kendaraan yang akan di publikasi oleh *Vehicle Provider* dan menambahkannya dalam sistem jika disetujui.

## Arsitektur Lingkungan Operasi Perangkat Lunak (digambarkan)

Aplikasi ini dapat berfungsi pada spesifikasi berikut :

* + - Windows/Linux/macOS
    - Android/Ios
    - Web Server

## Kebutuhan Perangkat Keras

Agar aplikasi ini dapat berjalan dengan baik, maka dibutuhkan

seperangkat komputer dengan spesifikasi:

Prosesor : Processor Intel(R) HD Graphics 4000 / 2.50 GHz

RAM : 4gb

HDD : 500gb

Monitor : 14inch

Karena aplikasi ini bersifat WEB, maka perangkat dengan spesifikasi dibawah

perangkat keras yang digunakan pada pembuatan aplikasi ini , dapat

menggunakan dan mengakses aplikasi ini dengan baik

## Asumsi dan Batasan Perangkat Lunak

Asumsi :

* Admin memiliki otoritas penuh dalam pengolahan data penyewaan kendaraan
* Website ini dibuat untuk orang-orang yang ingin mencari kendaraan
* Aplikasi hanya dapat berfungsi/ digunakan jika sudah ada yang menyewakan kendaraan..
* Aplikasi hanya bisa digunakan pada platform web.
* *Admin* hanya mengelola akun dan kendaraan yang akan disewakan..
* Pekerjaan yang tersedia hanya berlokasi di Indonesia.

Batasan Perangkat Lunaknya berupa :

* Sistem operasi : Windows 7 je atras
* Database : Xampp 5,6
* Aplikasi Pembuatan : Visual Studio Code
* Browser : Google Chrome

# Deskripsi Rinci Perangkat Lunak (ISIIN)

Pada perangkat lunak RentWheel.com yang di rancang memiliki 3 aktor dan 15 use case.

Pada use case yang merepresentasikan suatu goal dari system dan mendeskripsikan urutan aktivitas antar user dalam mencapai tujuannya. Dan untuk relasinya untuk menghubungkan antara actor dengan use case ada beberapa yaitu semua use case kecuali registrasi dan validasi akun harus login terlebih dahulu sebelum menggunakan fitur tersebut dan di dalam use case login akan melanjutkan ke proses validasi akun.

## Deskripsi Kebutuhan

### Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Kode Kebutuhan** | **Fungsi** | **Deskripsi** |
| 1. | FR-01 | Registrasi | Fungsi ini digunakan oleh *Customer* untuk daftar sebagai anggota ke sistem, registrasi diharuskan oleh user untuk menggunakan semua fitur yang ada di aplikasi ini dengan menginputkan username : String, password : String, dan email : String.. |
| 2. | FR-02 | Login | Fungsi ini dapat digunakan oleh user untuk melakukan login ke sistem. Fungsi ini adalah langkah pertama agar bisa mempunyai kewenangan untuk menggunakan fitur yang terdapat dalam system. Data yang diinput antara lain username : String, password : String. |
| 3. | FR-03 | Peminjaman Kendaraan | User |
| 4. | FR-04 | Pembayaran | Fungsi ini dapat digunakan oleh user untuk melakukan pembayaran melalui metode yang telah disediakan oleh sistem. Fungsi ini dapat digunakan oleh akun yang sudah teregistrasi. Fungsi ini menghitung berapa jumlah yang harus dibayar oleh sistem, sistem menyediakan jenisPembayaran yang disediakan antara lain transfer, bayar langsung. |
| 5. | FR-05 | Manage Peminjaman Kendaraan | Fungsi ini digunakan oleh admin untuk mengedit data peminjaman dari sistem, user dapat mengedit seluruh info yang ada di akun tersebut, yaitu idPeminjaman : String, JenisKendaraan,NamaKendaraan : String, dan StatusPembayaran : String. |
| 6. | FR-06 | Manage Pembayaran | Fungsi ini digunakan oleh admin untuk mengedit data pembayaran dari sistem, user dapat mengedit seluruh info yang ada di akun tersebut, yaitu idPembayaran : String, JenisPembayaran : String, dan StatusPembayaran : String. |
| 7. | FR-07 | Input Kendaraan | Fungsi ini digunakan oleh *Vehicle Provider* untuk menginputkan kendaraan ke system dan terupdate di database, dengan menginputkan nama kendaraan dan jenis kendaraan : string, lalu jumlah kendaraan : int. |
| 8. | FR-08 | Delete Kendaraan | Fungsi ini digunakan oleh admin untuk menghapus data kendaraan yang telah tidak tersedia. Fungsi ini menghapus kendaraan yang dipilih oleh admin.Jika terhapus maka kendaraan yang terdapat di sistem tidak ada. |
| 9. | FR-09 | View Kendaraan | Fungsi ini digunakan oleh *Vehicle Provider* untuk melihat semua data kendaraan dari system. Fungsi ini dapat digunakan oleh akun yang sudah teregistrasi. Fungsi ini dilakukan oleh admin dan menyediakan semua info dari kendaraan yang telah teregistrasi dan tersimpan dalam system. |



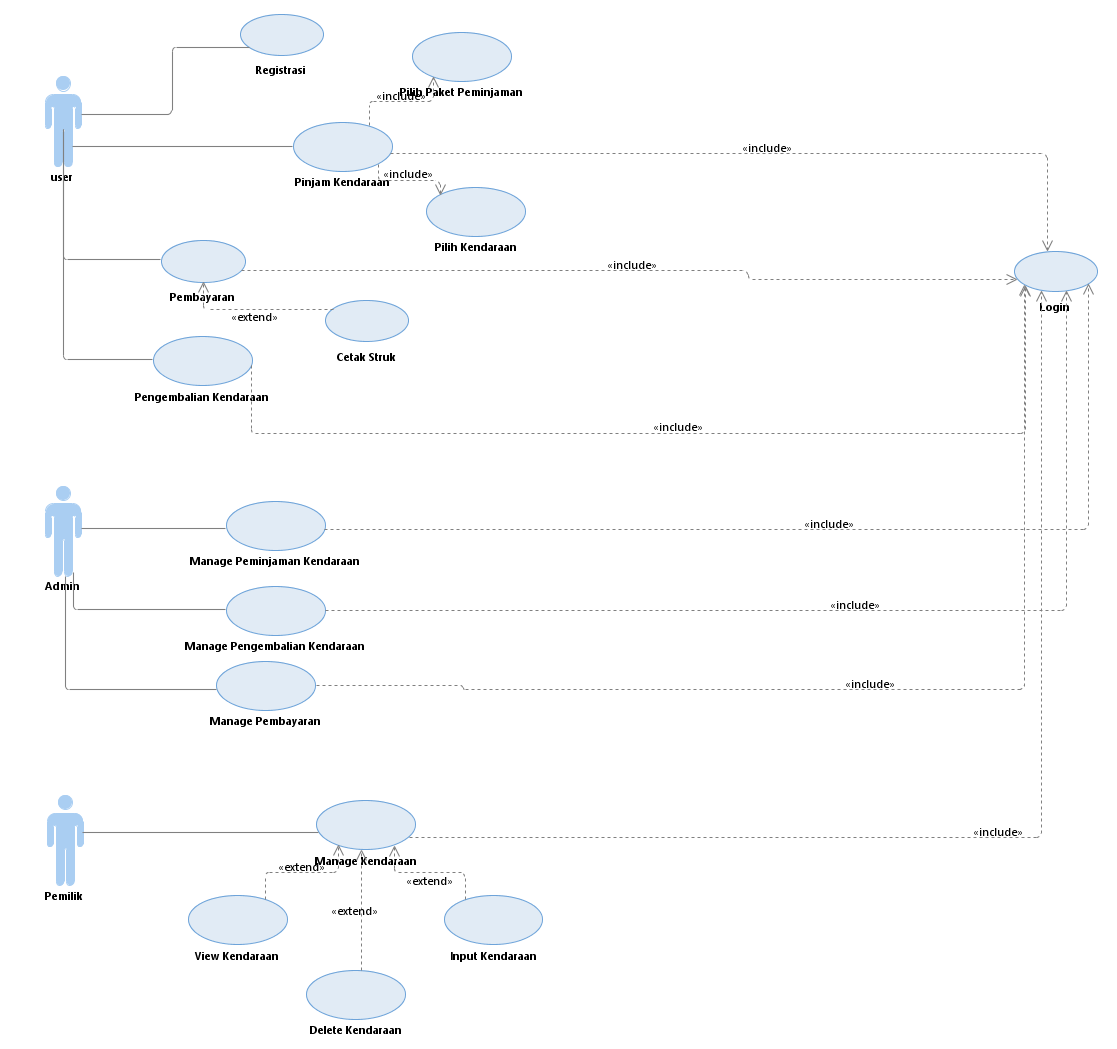
### Kebutuhan Non-Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Quality** | **Kode Kebutuhan** | **Deskripsi** |
| 1. | Security  Safety | NFR-01 | Akun dari setiap user harus dilindungi dengan enkripsi untuk menghindari tindakan hacking dan bocornya data informasi. |
| 2. | *Platform Compability* | NFR-02 | Akun dapat diakses hanya di web browser dan dengan OS *Windows/Linux/macOS/Android* atau/dan *Ios.* |
| 3. | *Availability* | NFR-03 | Sistem dapat diakses dimana dan kapan saja melalui web browser android atau *Personal Computer* selama 24 jam dalam sehari |
| 4. | *Respons Time* | NFR-04 | Kecepatan memuat setiap halaman selama 5 detik dengan jaringan internet 4g dan kecepatan download 250kb/s |
| 5. | Ergonomi | NFR-05 | 1. Sistem mudah digunakan oleh semua kalangan terutama mahasiswa 2. Dapat di akses semua tempat dengan internet 3. Dapat di akses semua waktu dengan internet 4. Dapat di akses semua aplikasi browser |

## Pemodelan Analisis



### Usecase Diagram



#### **Usecase Skenario #1**

|  |  |
| --- | --- |
| **Use Case** | Registrasi |
| **Input** | Data User(Nama\_User, No\_KTP, TTL\_Customer, Email, Username, Password) |
| **Output** | Data User(Nama\_User, No\_KTP, TTL\_Customer, Email, Username, Password) |
| **Actor** | User |
| **Precondition** | User ingin mendaftar |
| **Postcondition** | User telah terdaftar |
| **Description** | User melakukan pendaftaran untuk menyimpan data diri pada Database\_User |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Membuka halaman registrasi | 2. Menampilkan halaman registrasi | | 3. User memasukkan data User |  | | 4. Menekan tombol Daftar | 5. Jika Username telah tersedia, maka akan mengeluarkan notifikasi “Username Telah Tersedia”. 6. Jika terdapat input yang kosong, maka akan mengeluarkan notifikasi “Registration Failed”. 7. Jika langkah 5 &amp; 6 tidak terpenuhi, maka system akan merekam Data Customer yang diinputkan oleh Customer. 8. Memproses dan menyimpan Data Customer kedalam Database Akun. 9. Mengeluarkan Notifikasi “Registration Success”. | |

#### **Usecase Skenario #2**

|  |  |
| --- | --- |
| **Use Case** | Login |
| **Input** | Username, Password |
| **Output** | Halaman Utama |
| **Actor** | User, Admin |
| **Precondition** | User/Admin ingin melakukan login |
| **Postcondition** | User/Admin telah melakukan login |
| **Description** | User/Admin ingin menggunakan program |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Membuka halaman login |  | |  | 2. menampilkan halaman login | | 3. Memasukkan Username dan Password |  | |  | 4. Jika Username/Password salah maka program akan menampilkan notifikasi "username/password  salah" 5. Jika Username/Password benar, maka program akan menampilkan halaman utama | |

#### **Usecase Skenario #3**

|  |  |
| --- | --- |
| **Use Case** | Pinjam Kendaraan |
| **Input** | Paket, Kendaraan |
| **Output** | Struk Pembayaran |
| **Actor** | User |
| **Precondition** | User ingin memilih kendaraan dan paket |
| **Postcondition** | User telah memilih paket dan kendaraan |
| **Description** | User melakukan peminjaman kendaraan |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Menekan tombol peminjaman |  | |  | 2. menampilkan halaman peminjaman | | 3. Memilih kendaraan dan paket  peminjaman |  | | 4. Menekan tombol submit |  | |  | 5. Melakukan pengecekan kendaraan pada database\_kendaraan 6. Jika kendaraan tidak tersedia maka program akan menampilkan notifikasi "kendaraan tidak tersedia" 7. Jika kendaraan tersedia, maka program akan menampilkan info pembayaran | |

#### **Usecase Skenario #4**

|  |  |
| --- | --- |
| **Use Case** | Pembayaran |
| **Input** | Opsi pembayaran |
| **Output** | Struk Pembayaran |
| **Actor** | User |
| **Precondition** | User ingin membayar |
| **Postcondition** | User telah membayar |
| **Description** | User melakukan pembayaran atas kendaraan yang dipinjam |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Memilih Opsi Pembayaran |  | |  | 2. menampilkan halaman pembayaran | | 3. Melakukan pembayaran |  | |  | 4. Menampilkan informasi bahwa  pembayaran berhasil | |

#### **Usecase Skenario #5**

|  |  |
| --- | --- |
| **Use Case** | Input Kendaraan |
| **Input** | Data Kendaraan |
| **Output** | Penambahan jumalah kendaraan |
| **Actor** | Pemilik |
| **Precondition** | Pemilik ingin menambahkan kendaraan |
| **Postcondition** | Pemilik telah menambahkan kendaraan |
| **Description** | Pemilik memasukkan Data\_Kendaraan ke Database\_Kendaraan |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Menekan tombol "tambah kendaraan" |  | |  | 2. menampilkan halaman tambah kendaraan | | 3. Melakukan input kendaraan |  | |  | 4. Menyimpan data\_kendaraan ke dalam database\_kendaraan 5. menampilkan data\_kendaraan di menu kendaraan | |

#### **Usecase Skenario #6**

|  |  |
| --- | --- |
| **Use Case** | Pengembalian Kendaraan |
| **Input** | Data Kendaraan |
| **Output** | Pesan “terima kasih” lalu ke Menu Utama |
| **Actor** | User |
| **Precondition** | User ingin mengembalikan kendaraan |
| **Postcondition** | User telah mengembalikan kendaraan |
| **Description** | User melakukan pengembalian kendaraan |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Menekan tombol "Pengembalian Kendaraan" |  | |  | 2. menampilkan halaman pengembalian kendaraan | | 3. Melakukan input nomor kendaraan yanng dipinjam |  | |  | 4. Menyimpan data\_kendaraan ke dalam database\_kendaraan 5. menampilkan pesan “terima kasih telah melakukan pengembalian” | |

#### **Usecase Skenario #7**

|  |  |
| --- | --- |
| **Use Case** | Manage Peminjaman Kendaraan |
| **Input** | Data Kendaraan |
| **Output** | Penambahan jumlah peminjaman kendaraan, pengurangan jumlah ketersediaan kendaraan |
| **Actor** | Admin |
| **Precondition** | Admin ingin menambahkan kendaraan yang dipinjam |
| **Postcondition** | Admin telah menambahkan kendaraan yang dipinjam |
| **Description** | Admin memasukkan data\_kendaraan ke data\_peminjaman |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Menekan tombol "Manage Peminjaman Kendaraan" |  | |  | 2. menampilkan halaman manage peminjaman kendaraan | | 3. Melakukan input kendaraan |  | |  | 4. Menyimpan data\_kendaraan ke dalam database\_peminjaman 5. menampilkan jika da telah ditambahkan | |

#### **Usecase Skenario #8**

|  |  |
| --- | --- |
| **Use Case** | Manage Pengembalian Kendaraan |
| **Input** | Data Kendaraan |
| **Output** | Penambahan jumlah ketersediaan kendaraan, pengurangan jumlah kendaraan yang dipinjam |
| **Actor** | Admin |
| **Precondition** | Admin ingin menambahkan kendaraan yang tersedia |
| **Postcondition** | Admin telah menambahkan kendaraan yang tersedia |
| **Description** | Admin memasukkan data\_peminjaman ke data\_kendaraan |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Menekan tombol "Manage Pengembalian Kendaraan" |  | |  | 2. menampilkan halaman manage pengembalian kendaraan | | 3. Melakukan input kendaraan |  | |  | 4. Menyimpan data\_kendaraan ke database\_kendaraan 5. menampilkan pesan jika data telah ditambahkan | |

#### **3.2.1.9 Usecase Skenario #9**

|  |  |
| --- | --- |
| **Use Case** | Manage Pembayaran |
| **Input** | Nomor Struk |
| **Output** | Penambahan info pembayaran |
| **Actor** | Admin |
| **Precondition** | Admin ingin menambahkan info pembayaran |
| **Postcondition** | Admin telah menambahkan info pembayaran |
| **Description** | Admin memasukkan info pembyaran ke database\_pembayaran |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Menekan tombol "Manage Pembayaran" |  | |  | 2. menampilkan halaman manage pembayaran | | 3. Melakukan input info pembayaran |  | |  | 4. Menyimpan ke database\_pembayaran 5. menampilkan pesan jika data telah ditambahkan | |

#### **Usecase Skenario #10**

|  |  |
| --- | --- |
| **Use Case** | Manage Kendaraan |
| **Input** | Data Kendaraan |
| **Output** | Penambahan Kendaraan |
| **Actor** | Pemilik |
| **Precondition** | Pemilik ingin menambahkan kendaraan |
| **Postcondition** | Pemilik telah menambahkan kendaraan |
| **Description** | Pemilik melakukan inputan baru ke data\_kendaraan |
| **Typical Course Of Event** | |  |  | | --- | --- | | **Actor** | **System** | | 1. Menekan tombol "Manage Kendaraan" |  | |  | 2. menampilkan halaman manage kendaraan | | 3. Melakukan input kendaraan |  | |  | 4. Menyimpan data\_kendaraan ke database\_kendaraan 5. menampilkan pesan jika data telah ditambahkan | |

## Class Diagram

