

REKAYASA PERANGKAT LUNAK  
STUDI KASUS BENGKEL ONLINE&OFFLINE  
“Bengkel-in  
(Melayani Online&Offline : Service, Booking Online, dan Consult)”



Disusun oleh :  
**Kelompok 4**  
Saeful Iman (20240050032)  
Azka Putra P (20240050056)  
Muhamad Nazar Z (20240050020)  
Muhamad Daffa Fasya R (20240050041)  
Neng Rista Nur Rahmawati (20240050064)  
Nurlaila (20240050141)  
Dwi zahwa Abdillah (20240050008)

## SOFTWARE DESIGN DOCUMENT

### 1. PENDAHULUAN

#### 1.1 Tujuan Penulisan Dokumen

Dokumen *Software Design Description* (SDD) ini bertujuan untuk memberikan penjelasan yang komprehensif mengenai perancangan dan proses pengembangan sistem Bengkel-in, sebuah platform bengkel yang melayani kebutuhan pengguna secara online dan offline. Dokumen ini disusun untuk memastikan bahwa setiap elemen desain sistem mendukung pencapaian tujuan aplikasi, khususnya dalam penyediaan layanan service kendaraan, booking service secara online, serta konsultasi antara pelanggan dan montir.

#### 1.2 Lingkup Masalah

Website Bengkel-in dirancang sebagai platform publik yang mempermudah pengguna dalam mengakses layanan bengkel secara online dan offline. Sistem ini menyediakan informasi layanan bengkel secara lengkap, termasuk jenis service kendaraan, estimasi biaya, dan ketersediaan produk. Selain itu, website ini memungkinkan pengguna untuk melakukan booking service secara online serta memperoleh konsultasi terkait permasalahan kendaraan. Melalui fitur yang tersedia, pengguna dapat dengan mudah mencari layanan yang dibutuhkan dan menghubungi bengkel atau admin secara langsung melalui kontak yang terintegrasi dengan WhatsApp, sehingga proses komunikasi menjadi lebih cepat dan efisien.

#### 1.3 Definisi dan Istilah

- SPMP (Software Project Management Plant)
- RS (Software Requirements Specification)
- SDD (Software Design Description)

#### 1.4 Referensi

- IEEE, IEEE Draft Standard for Software Design Descriptions. IEEE P1 01 6/D5.0; 1 2 December 2005
- Eka Ismantohadi & Moh. Yani, Software Design Document (SDD). 2018

## 1.5. Ikhtisar Dokumen

BAB	ISI
<b>Bab I</b>	1.1 Tujuan Penulisan Dokumen 1.2 Lingkup Masalah 1.3 Definisi dan Istilah 1.4 Referensi 1.5 Ikhtisar Dokumen
<b>Bab II Deskripsi Perancangan Global</b>	2.1 Rancangan Lingkungan Implementasi 2.2 Deskripsi Data 2.2.1 Definisi Domain (x) 2.2.2 Conceptual Data Model (x) 2.2.3 ERD Logical Data Model 2.2.4 Daftar Tabel Aplikasi (Kamus Data) 2.3 Deskripsi Modul
<b>Bab III Deskripsi Perancangan Rinci</b>	3.1 Diagram Konteks 3.1.1 DFD Level 0 3.1.2 DFD Level 1 Proses M 3.1.3 DFD Level 1 Proses N 3.2 Deskripsi Rinci Tabel 3.2.1 Table A 3.2.2 Table B 3.3 Deskripsi Rinci Modul 3.3.1 Modul 3.3.1.1 Fungsi Modul 3.3.1.2 Spesifikasi Layar Utama 3.3.1.3 Spesifikasi Query (x) 3.3.1.4 Spesifikasi Field Data Layer (x) 3.3.1.5 Spesifikasi Obyek Pada Layer (x) 3.3.1.6 Spesifikasi Proses/Algoritma (x) 3.4 Matriks Keturunan (x)

## 2. Deskripsi Perancangan Global

### 2.1 Rancangan Lingkungan Implementasi

- Rancangan Kebutuhan

NO	Rancangan Kebutuhan	Keterangan
1	Sistem Operasi	<ul style="list-style-type: none"><li>- UML dibuat dengan menggunakan Visual Paradigm CE</li><li>- Design PowerPoint dibuat menggunakan Canva</li><li>- Prototyping Aplikasi dibuat menggunakan Figma</li><li>- Pembuatan Laporan dibuat menggunakan Microsoft Word 2013</li></ul>
2	DBMS	MYSQL
3	Filling System	Dokumen-dokumen dan program disimpan dalam Hard Disk internal pada laptop masing-masing anggota
4	Bahasa Pemrograman	Menggunakan PHP sebagai Bahasa Pemrograman utama untuk pengembangan Website

- Tools yang digunakan

NO	Tools	Jumlah
1	Laptop	7 Unit

## **2.2 Deskripsi Data**

### **1. Data Admin**

Data Admin digunakan untuk menyimpan informasi pengguna yang memiliki hak akses dalam mengelola sistem bengkel. Admin bertanggung jawab dalam proses pencatatan transaksi servis, pengelolaan data master, serta pemantauan pembayaran. Data yang disimpan meliputi identitas admin, informasi akun untuk login ke sistem, serta data pendukung berupa email dan gambar profil sebagai identitas visual admin di dalam sistem.

### **2. Data Pelanggan**

Data Pelanggan digunakan untuk menyimpan informasi pelanggan yang menggunakan layanan bengkel. Data ini berperan penting sebagai referensi utama dalam pencatatan kepemilikan kendaraan dan riwayat servis. Informasi pelanggan yang disimpan meliputi identitas pelanggan, data kontak, alamat, serta akun pengguna yang digunakan pelanggan untuk mengakses sistem berbasis web.

### **3. Data Motor**

Data Motor digunakan untuk menyimpan informasi kendaraan milik pelanggan yang terdaftar di dalam sistem. Setiap data motor terhubung langsung dengan data pelanggan sebagai pemilik kendaraan. Informasi yang dicatat meliputi merek, tipe, dan nomor polisi kendaraan yang digunakan sebagai identitas resmi motor dalam proses servis.

### **4. Data Jasa Servis**

Data Jasa Servis digunakan untuk menyimpan daftar jenis layanan servis yang disediakan oleh bengkel. Data ini mencakup informasi nama jasa dan biaya servis yang menjadi dasar perhitungan biaya pada setiap transaksi servis kendaraan pelanggan.

### **5. Data Mekanik**

Data Mekanik digunakan untuk menyimpan informasi tenaga kerja bengkel yang bertugas menangani proses servis kendaraan. Data ini mencakup identitas mekanik serta bidang keahlian yang dimiliki, sehingga memudahkan penugasan mekanik sesuai dengan jenis perbaikan yang dilakukan.

### **6. Data Servis**

Data Servis merupakan data utama yang mencatat setiap transaksi servis kendaraan yang terjadi di bengkel. Data ini menghubungkan informasi pelanggan, motor, dan mekanik yang terlibat dalam proses servis. Selain itu, data servis juga mencatat tanggal pelaksanaan servis serta status pengerjaan servis, seperti dalam proses, selesai, atau dibatalkan.

## **7. Data Sparepart**

Data Sparepart digunakan untuk menyimpan informasi suku cadang yang tersedia dan digunakan dalam proses servis kendaraan. Data ini mencakup nama sparepart, jumlah stok yang tersedia, serta harga satuan sparepart yang digunakan sebagai dasar perhitungan biaya servis.

## **8. Data Detail Servis**

Data Detail Servis digunakan untuk mencatat rincian penggunaan sparepart dalam setiap transaksi servis. Data ini menghubungkan transaksi servis dengan sparepart yang digunakan, mencakup informasi jumlah sparepart serta subtotal biaya berdasarkan harga satuan dan jumlah yang digunakan.

## **9. Data Pembayaran**

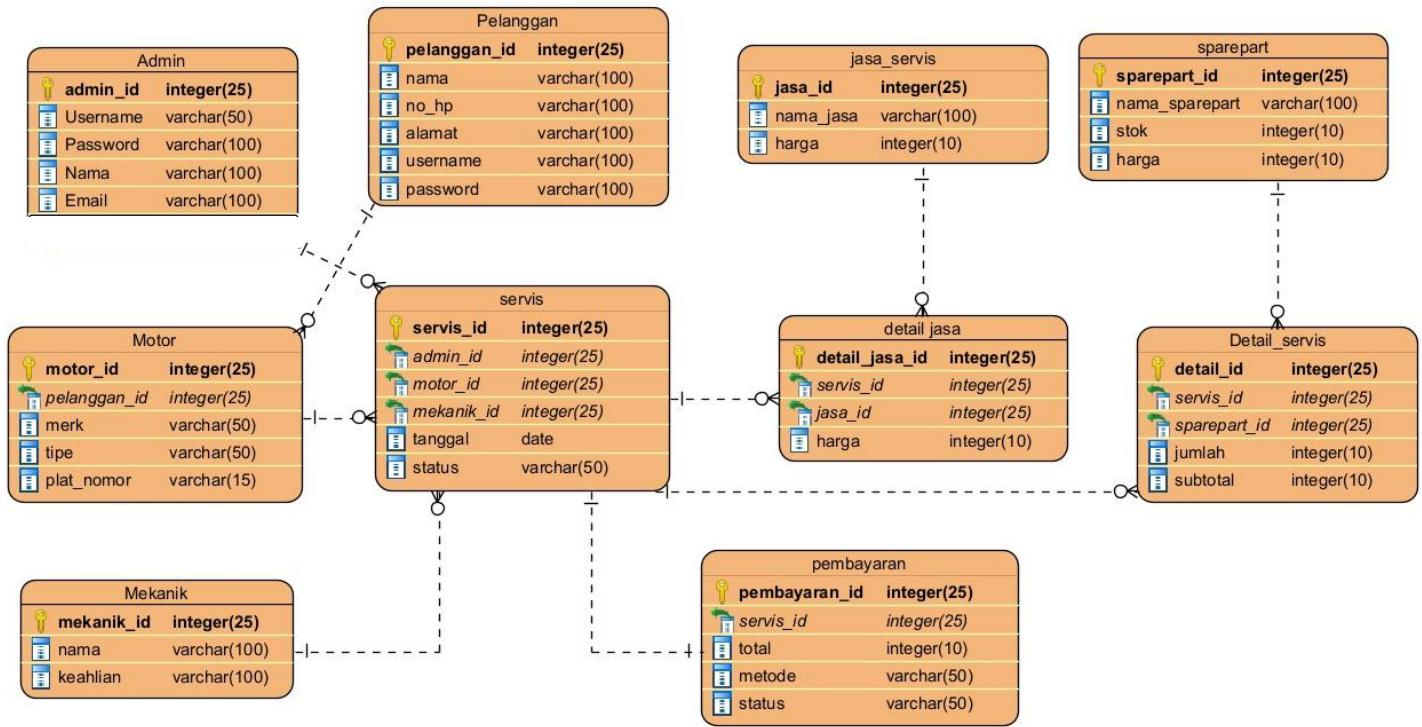
Data Pembayaran digunakan untuk menyimpan informasi pembayaran dari setiap transaksi servis kendaraan. Data ini mencakup total biaya yang harus dibayarkan oleh pelanggan, metode pembayaran yang digunakan, serta status pembayaran sebagai penanda apakah transaksi telah lunas atau belum.

## **10. Data Detail Jasa**

Data Detail Jasa digunakan untuk mencatat rincian jasa servis yang digunakan dalam setiap transaksi servis. Data ini menghubungkan transaksi servis dengan jenis jasa servis yang dipilih, serta menyimpan informasi harga jasa yang dikenakan pada setiap detail jasa sebagai bagian dari total biaya servis.

### 2.2.3 ERD Logical Data Model

Berikut adalah **ERD Logical Data Model** yang disajikan dalam bentuk **entitas dan relasi** (logical level). ERD ini menunjukkan hubungan antar entitas tanpa detail fisik implementasi database.



#### Penjelasan Entitas dan Relasi dari ERD Logical Data Model diatas :

##### 1. Pelanggan – Motor

Satu pelanggan dapat memiliki lebih dari satu motor. Setiap motor hanya dimiliki oleh satu pelanggan. Relasi ini bersifat one-to-many (1:M), di mana satu pelanggan berhubungan dengan banyak motor.

##### 2. Pelanggan – Servis

Setiap pelanggan dapat melakukan transaksi servis lebih dari satu kali dalam periode waktu yang berbeda. Oleh karena itu, relasi yang terbentuk adalah one-to-many (1:M).

##### 3. Motor – Servis

Satu motor dapat memiliki riwayat servis lebih dari satu kali, sedangkan satu transaksi servis hanya terkait dengan satu motor. Relasi ini bersifat one-to-many (1:M).

##### 4. Mekanik – Servis

Satu mekanik dapat menangani banyak transaksi servis, namun satu transaksi servis hanya ditangani oleh satu mekanik. Relasi yang terbentuk adalah one-to-many (1:M).

##### 5. Servis – Detail\_Jasa

Satu transaksi servis dapat mencakup lebih dari satu jenis jasa servis, dan setiap detail\_jasa hanya terkait dengan satu transaksi servis. Relasi ini bersifat one-to-many (1:M).

## **6. Jasa\_Servis – Detail\_Jasa**

Satu jenis jasa servis dapat digunakan dalam banyak detail\_jasa pada transaksi yang berbeda. Oleh karena itu, relasi yang terbentuk adalah one-to-many (1:M).

## **7. Servis – Detail\_Servis**

Satu transaksi servis dapat menggunakan banyak sparepart, yang dicatat dalam detail\_servis. Setiap detail\_servis hanya terkait dengan satu transaksi servis. Relasi ini bersifat one-to-many (1:M).

## **8. Sparepart – Detail\_Servis**

Satu jenis sparepart dapat digunakan dalam banyak detail\_servis pada transaksi yang berbeda. Relasi yang terbentuk adalah one-to-many (1:M).

## **9. Servis – Pembayaran**

Setiap transaksi servis memiliki satu data pembayaran, dan setiap pembayaran hanya untuk satu transaksi servis. Oleh karena itu, relasi ini bersifat one-to-one (1:1).

## **10. Admin – Servis**

Satu admin dapat mencatat dan mengelola banyak transaksi servis, sedangkan satu transaksi servis hanya dicatat oleh satu admin. Oleh karena itu, relasi antara admin dan servis bersifat one-to-many (1:M).

### **2.2.4 Daftar Tabel Aplikasi (Kamus Data)**

- **Tabel Admin**

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
Admin_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan identitas unik yang digunakan untuk membedakan setiap data admin yang tersimpan di dalam sistem.
Username	Varchar	50		Tidak	NOT NULL, UNIQUE	Digunakan sebagai nama akun admin yang berfungsi untuk proses autentikasi saat admin melakukan login ke sistem.
Password	Varchar	100		Tidak	NOT NULL	Berisi kata sandi admin yang digunakan untuk menjaga keamanan akses terhadap sistem.
Nama	Varchar	100		Tidak	NOT NULL	Menyimpan nama lengkap admin yang terdaftar dan digunakan sebagai

						informasi identitas admin.
Email	Varchar	100		Tidak	NOT NULL, UNIQUE	Digunakan untuk menyimpan alamat email admin sebagai sarana komunikasi dan pemulihan akun.

- Tabel Pelanggan

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
Pelanggan_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan sebagai identitas pelanggan di dalam sistem.
Nama	Varchar	100		Ya	NOT NULL, REFERENCES User(User_id)	Digunakan sebagai penghubung antara data pelanggan dengan data user yang telah terdaftar.
No_Hp	Varchar	100		Tidak	NOT NULL	Menyimpan nomor telepon pelanggan yang digunakan untuk keperluan komunikasi dan konfirmasi servis.
Alamat	Varchar	100		Tidak	NOT NULL	Menyimpan alamat lengkap pelanggan sebagai informasi pendukung dalam proses pelayanan.
Username	Varchar	100		Tidak	NOT NULL	Digunakan sebagai nama pengguna pelanggan untuk melakukan login ke dalam sistem berbasis web.
Password	Varchar	100		Tidak	NOT NULL	Digunakan untuk menyimpan kata sandi pelanggan sebagai bagian dari proses autentikasi sistem.

- Tabel Motor

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
-----------	------	--------	------	-------------	---------------------	-----------

Motor_id	Integer	25	Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan untuk mengidentifikasi setiap data motor yang terdaftar di dalam sistem.
Pelanggan_id	Integer	25	Ya	NOT NULL, REFERENCES Pelanggan(Pelanggan_id)	Digunakan untuk menunjukkan kepemilikan motor oleh pelanggan tertentu.
Merk	Varchar	50	Tidak	NOT NULL	Menyimpan informasi mengenai merek sepeda motor yang dimiliki pelanggan.
Tipe	Varchar	50	Tidak	NOT NULL	Digunakan untuk menyimpan jenis atau tipe sepeda motor yang terdaftar.
Plat_Nomor	Varchar	15	Tidak	NOT NULL, UNIQUE	Menyimpan nomor polisi kendaraan yang digunakan sebagai identitas resmi motor.

- Tabel Jasa\_Servis

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
Jasa_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan untuk mengidentifikasi setiap jenis jasa servis.
Nama_jasa	Varchar	100		Tidak	NOT NULL	Menyimpan nama layanan servis yang disediakan oleh bengkel.
Harga	Integer	10		Tidak	NOT NULL	Menyimpan informasi biaya jasa servis yang harus dibayarkan oleh pelanggan.

- Tabel Mekanik

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi

Mekanik_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan sebagai identitas setiap mekanik yang bekerja di bengkel.
Nama	Varchar	100		Tidak	NOT NULL	Menyimpan nama lengkap mekanik yang bertugas melakukan servis kendaraan.
Keahlian	Varchar	100		Tidak	NOT NULL	Menyimpan informasi mengenai bidang keahlian mekanik dalam menangani jenis perbaikan tertentu.

- Tabel Servis

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
Sevis_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan untuk mengidentifikasi setiap transaksi servis yang terjadi.
Pelanggan_id	Integer	25		Ya	NOT NULL, REFERENCES Pelanggan(Pelanggan_id)	Digunakan untuk menunjukkan pelanggan yang melakukan servis kendaraan.
Motor_id	Integer	25		Ya	NOT NULL, REFERENCES Motor(Motor_id)	Menyimpan informasi motor yang sedang atau telah dilakukan proses servis.
Mekanik_id	Integer	25		Ya	NOT NULL, REFERENCES Mekanik(Mekanik_id)	Digunakan untuk menunjukkan mekanik yang bertanggung jawab dalam proses servis.
Tanggal	Date	-		Tidak	NOT NULL	Digunakan untuk mencatat tanggal dilakukannya proses servis kendaraan.
Status	Teks	50		Tidak	NOT NULL	Menyimpan informasi status pengerjaan servis, seperti proses, selesai, atau dibatalkan.

- Tabel Sparepart

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi

Sparepart_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan untuk mengidentifikasi setiap data sparepart.
Nama_Sparepart	Varchar	100		Tidak	NOT NULL	Menyimpan nama sparepart yang tersedia dan digunakan dalam proses servis.
Stok	Integer	10		Tidak	NOT NULL	Menyimpan jumlah ketersediaan sparepart yang masih ada di dalam gudang.
Harga	Integer	10		Tidak	NOT NULL	Menyimpan harga satuan sparepart yang digunakan dalam perhitungan biaya servis.

- Tabel Detail\_Servis

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
Detail_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan untuk mengidentifikasi detail penggunaan sparepart dalam servis.
Servis_id	Integer	25		Ya	NOT NULL, REFERENCES Servis(Servis_id)	Digunakan sebagai penghubung antara data detail servis dengan data servis utama.
Sparepart_id	Integer	25		Ya	NOT NULL, REFERENCES Sparepart(Sparepart_id)	Menyimpan informasi sparepart yang digunakan dalam suatu proses servis.
Jumlah	Varchar	10		Tidak	NOT NULL	Menyimpan jumlah sparepart yang digunakan dalam proses servis kendaraan.
Subtotal	Integer	10		Tidak	NOT NULL	Menyimpan total biaya sparepart berdasarkan jumlah dan harga satuan.

- Tabel Pembayaran

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
Pembayaran_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan untuk mengidentifikasi setiap transaksi pembayaran.
Servis_id	Integer	25		Ya	NOT NULL, REFERENCES Servis(Servis_id)	Digunakan untuk menghubungkan data pembayaran dengan transaksi servis terkait.
Total	Integer	10		Tidak	NOT NULL	Menyimpan total biaya yang harus dibayarkan oleh pelanggan.
Metode	Varchar	50		Tidak	NOT NULL	Menyimpan informasi metode pembayaran yang digunakan oleh pelanggan.
Status	Teks	50		Tidak	NOT NULL	Menyimpan informasi status pembayaran, seperti lunas atau belum lunas.

- Tabel Detai\_Jasa

Data Item	Type	Volume	Laju	Primary Key	Constrain Integrity	Deskripsi
Detail_jasa_id	Integer	25		Ya	NOT NULL, AUTO INCREMENT	Merupakan kode unik yang digunakan sebagai identitas pelanggan di dalam sistem.
Servis-id	Integer	25		Tidak	NOT NULL, REFERENCES servis(servis_id)	Digunakan sebagai penghubung antara detail jasa dengan transaksi servis yang sedang atau telah dilakukan.
Jasa_id	Integer	25		Tidak	NOT NULL, REFERENCES jasa_servis(jasa_id)	Digunakan sebagai penghubung antara detail jasa dengan jenis jasa servis yang digunakan pada transaksi.

Harga	Integer	10		Tidak	NOT NULL	Menyimpan harga jasa servis yang dikenakan pada detail jasa sesuai dengan jenis jasa yang dipilih.
-------	---------	----	--	-------	----------	--

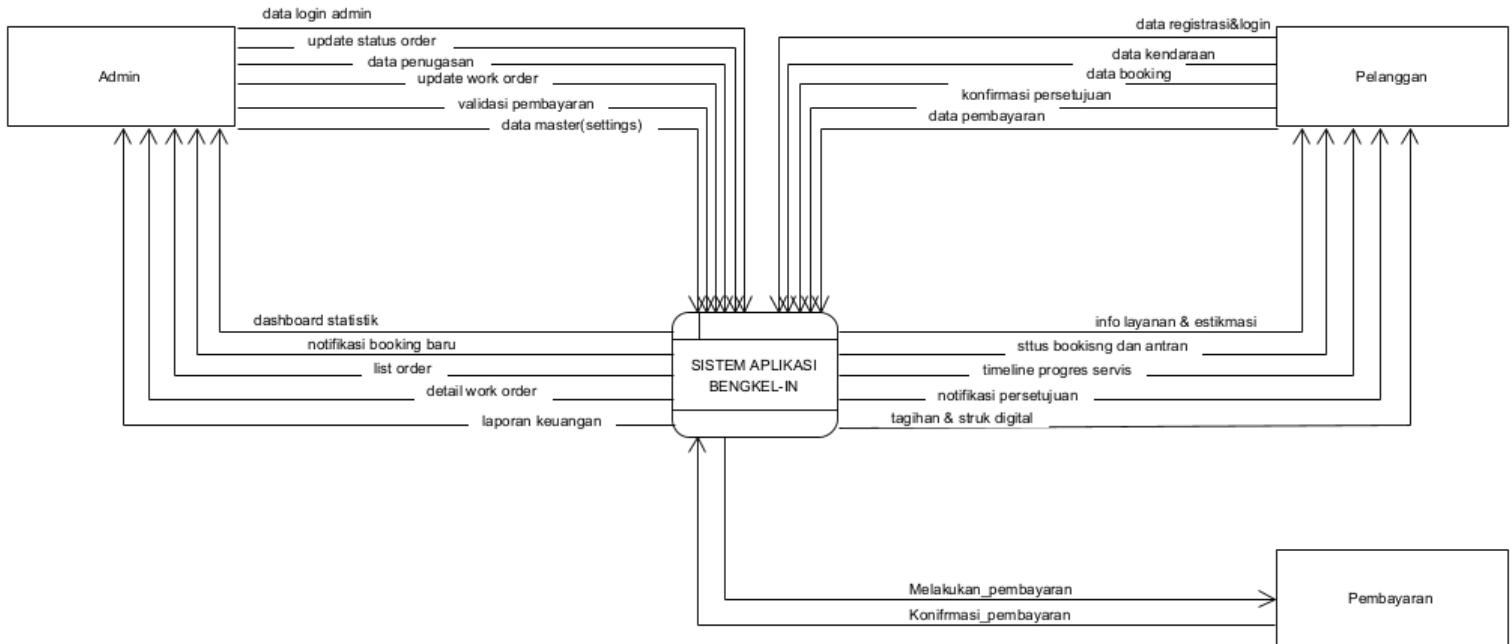
### 2.3 Deskripsi Modul

- 1. Modul Admin :** Modul ini digunakan untuk mengelola data admin sistem, termasuk proses login, pengaturan hak akses, serta pencatatan transaksi servis. Admin berperan sebagai pengelola utama sistem.
- 2. Modul Pelanggan :** Modul pelanggan berfungsi untuk mengelola data pelanggan bengkel, seperti pendaftaran pelanggan baru, perubahan data pelanggan, serta melihat riwayat servis pelanggan.
- 3. Modul Motor :** Modul motor digunakan untuk mencatat dan mengelola data kendaraan milik pelanggan, termasuk nomor polisi, merek, tipe, dan tahun kendaraan.
- 4. Modul Mekanik :** Modul ini berfungsi untuk mengelola data mekanik bengkel, mencakup informasi identitas, keahlian, dan penugasan mekanik pada transaksi servis.
- 5. Modul Servis :** Modul servis merupakan modul utama yang menangani transaksi servis kendaraan, mulai dari pencatatan data motor, mekanik, jasa servis, sparepart, hingga perhitungan total biaya.
- 6. Modul Jasa Servis :** Modul jasa servis digunakan untuk mengelola jenis-jenis jasa servis yang tersedia di bengkel beserta tarifnya.
- 7. Modul Sparepart :** Modul sparepart berfungsi untuk mengelola data suku cadang, termasuk harga dan stok sparepart yang tersedia di bengkel.
- 8. Modul Pembayaran :** Modul pembayaran digunakan untuk mencatat proses pembayaran servis, memilih metode pembayaran, serta menyimpan riwayat transaksi pembayaran

### 3. Deskripsi Perancangan Rinci

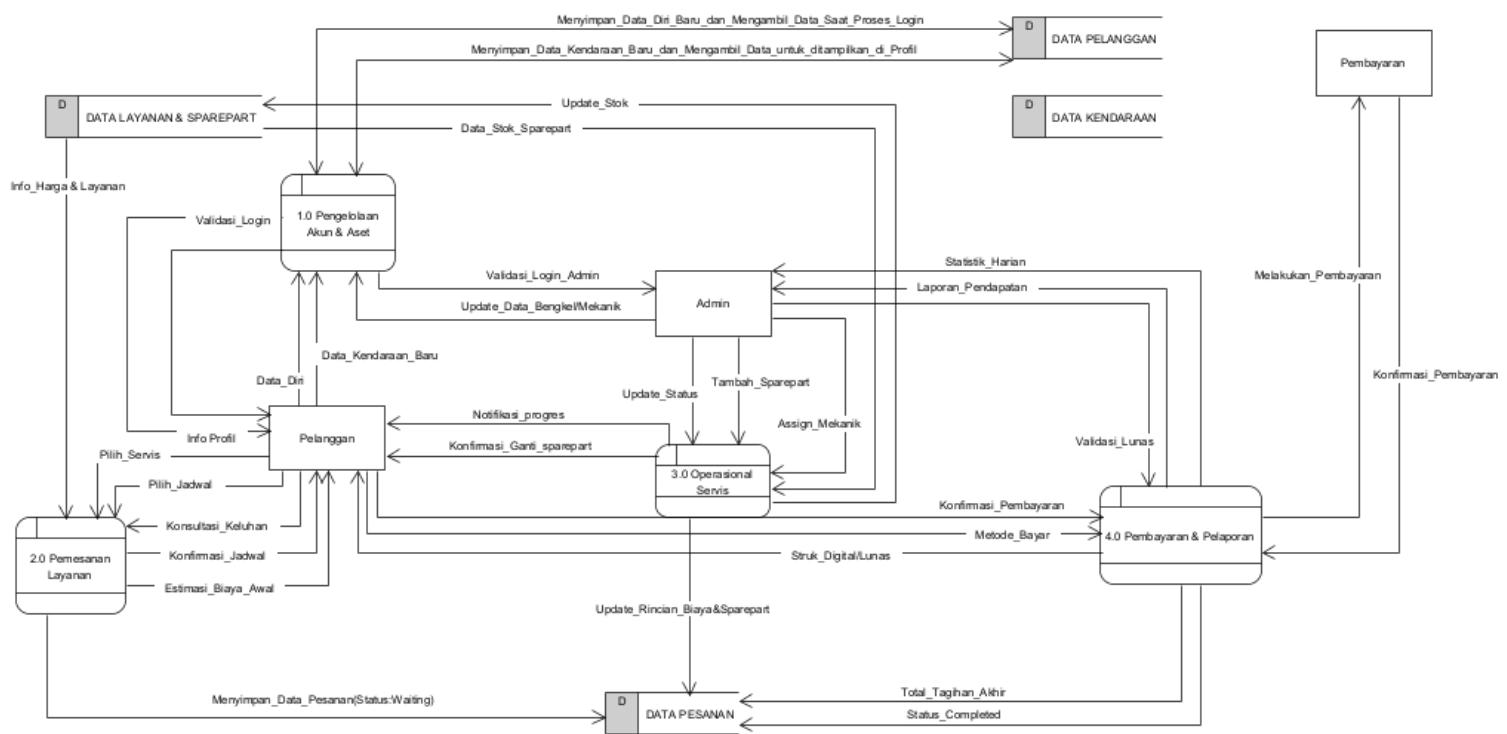
#### 3.1 Diagram Konteks

Diagram konteks merupakan tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses, menunjukkan sistem secara keseluruhan. Berikut merupakan diagram konteks dari website ini.



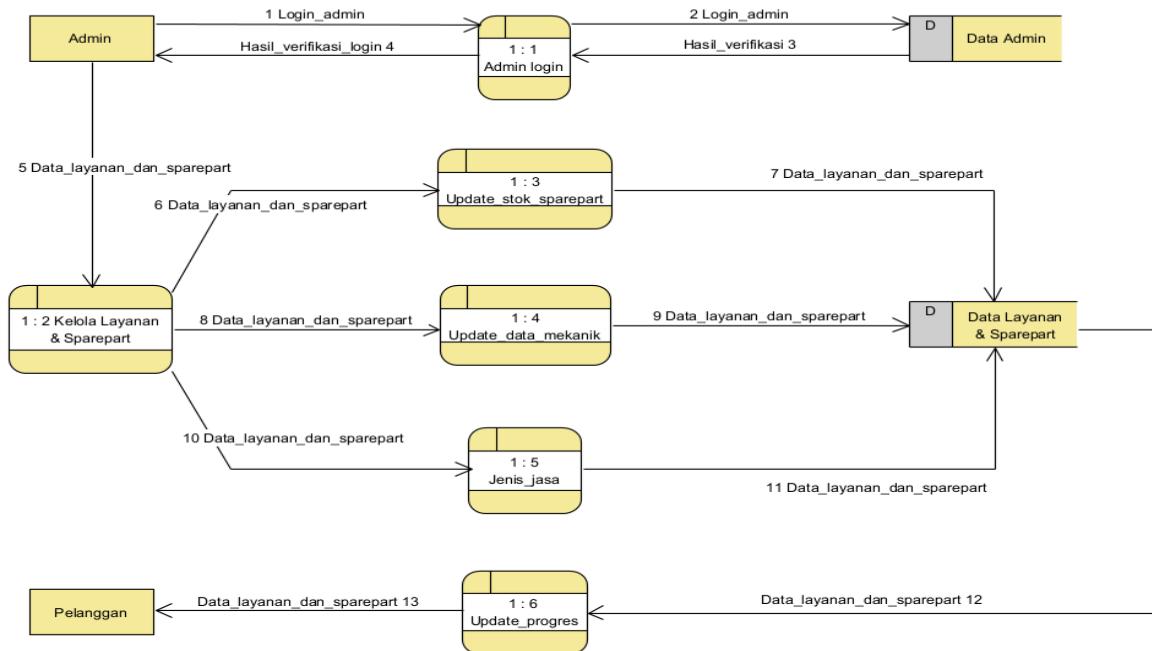
### 3.1.1 DFD Level 0

Diagram 0 adalah diagram yang menggambarkan proses dari data flow diagram. Diagram 0 memberikan pandangan secara menyeluruh mengenai sistem yang ditangani, menunjukkan tentang fungsi-fungsi utama atau proses yang ada, aliran data, dan eksternal entity. Berikut merupakan diagram 0 dari aplikasi ini.



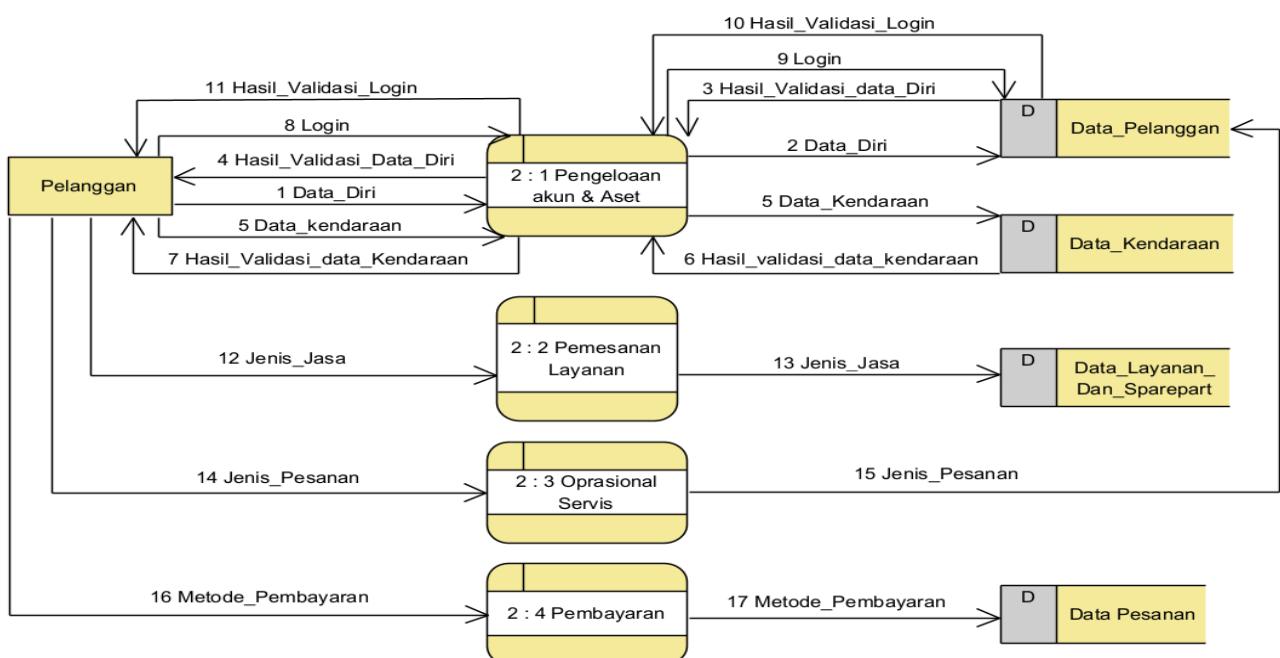
### 3.1.2 DFD Level 1 Proses M

**DFD (Data Flow Diagram) Level 1 Proses M** adalah diagram yang memberikan pandangan mendetail mengenai sub-proses yang terjadi di dalam satu proses besar (yang diberi label **M** atau angka tertentu) pada sistem.



### 3.1.3 DFD Level 1 Proses N

**DFD (Data Flow Diagram) Level 1** adalah diagram yang memberikan pandangan lebih rinci dibandingkan Level 0 (Diagram Konteks). Jika Anda menyebutkan "**Proses N**", ini merujuk pada dekomposisi atau pemecahan satu proses besar menjadi beberapa sub-proses yang lebih spesifik.



### 3.2 Deskripsi Rinci Tabel

- Tabel Admin

Data Item	Type	Deskripsi
admin_id	Integer	Identitas unik admin
username	Varchar	Nama akun admin untuk login
Password	Varchar	Kata sandi admin
Nama	Varchar	Nama lengkap admin
Email	Varchar	Email admin

- Tabel Pelanggan

Data Item	Type	Deskripsi
Pelanggan_id	Integer	Identitas unik pelanggan
Nama	Varchar	Nama pelanggan
No_hp	Varchar	Nomor telepon pelanggan
Alamat	Varchar	Alamat Pelanggan
Username	Varchar	Username Pelanggan
Password	Varchar	Kata sandi pelanggan

- Tabel Motor

Data Item	Type	Deskripsi
Motor_id	Integer	Identitas unik motor
Pelanggan_id	Integer	Pemilik motor
Merk	Varchar	Merek motor
Tipe	Varchar	Tipe motor
Plat_nomor	Varchar	Nomor polisi motor

- Tabel Jasa\_Servis

Data Item	Type	Deskripsi
Jasa_id	Integer	Identitas unik jasa servis
Nama_jasa	Varchar	Nama layanan servis
Harga	Integer	Biaya jasa servis

- Tabel Mekanik

Data Item	Type	Deskripsi
Mekanik_id	Integer	Identitas unik mekanik
Nama	Varchar	Nama mekanik
Keahlian	Varchar	Keahlian mekanik

• **Tabel Servis**

Data Item	Type	Deskripsi
Servis_id	Integer	Identitas transaksi servis
Pelanggan_id	Integer	Pelanggan yang servis
Motor_id	Integer	Motor yang diservis
Mekanik_id	Integer	Mekanik yang menangani
Tanggal	Date	Tanggal servis
Status	Text	Status servis

• **Tabel Sparepart**

Data Item	Type	Deskripsi
Sparepart_id	Integer	Identitas unik sparepart
Nama_Sparepart	Varchar	Nama sparepart
Stok	Integer	Jumlah stok sparepart
Harga	Integer	Harga sparepart

• **Tabel Detail\_Servis**

Data Item	Type	Deskripsi
Detail_id	Integer	Identitas detail servis
Servis_id	Integer	Id transaksi servis
Sparepart_id	Integer	Sparepart yang digunakan
Jumlah	Varchar	Jumlah sparepart
Subtotal	Integer	Total biaya sparepart

• **Tabel Pembayaran**

Data Item	Type	Deskripsi
Pembayaran_id	Integer	Identitas pembayaran
Servis_id	Integer	Transaksi servis terkait
Total	Integer	Total pembayaran
Metode	Varchar	Metode pembayaran
Status	Text	Status pembayaran

• **Tabel Detail\_Jasa**

Data Item	Type	Deskripsi
Detail_Jasa_id	Integer	Identitas detail jasa
Servis_id	Integer	Transaksi servis
Jasa_id	Integer	Jasa yang digunakan
Harga	Integer	Harga jasa servis

**3.2.1 Table A**

Data Item	Type	Volume	Primary Key	Constraint Integrity	Deskripsi
Admin_id	Integer	25	Ya	NOTNULL,AUTO INCREMENT	Identitas unit admin.
Username	Varchar	50	Tidak	NOT NULL, UNIQUE	Nama akun untuk autentikasi.
Password	Varchar	100	Tidak	NOT NULL	Kata sandi untuk keamanan akses.
Nama	Varchar	100	Tidak	NOT NULL	Nama Lengkap admin.
Email	Varchar	100	Tidak	NOT NULL,UNIQUE	Alamat email untuk komunikasi.

**3.2.2 Table B**

Data Item	Type	Volume	Primary Key	Constraint integrity	Deskripsi
Pelanggan_id	Integer	25	Ya	NOT NULL,AUTO INCREMENT	Kode unik identitas pelanggan.
Nama	Varchar	100	Ya	NOT NUL,REF USER(User_id)	Penghubung dengan data.
No_Hp	Varchar	100	Tidak	NOT NULL	Nomor telepon untuk konfirmasi.
Alamat	Varchar	100	Tidak	NOT NULL	Alamat lengkap pelanggan.
Username	Varchar	100	Tidak	NOT NULL	Nama pengguna untuk login web
Password	Varchar	100	Tidak	NOT NULL	Kata sandi autentikasi

### 3.3 Deskripsi Rinci Modul

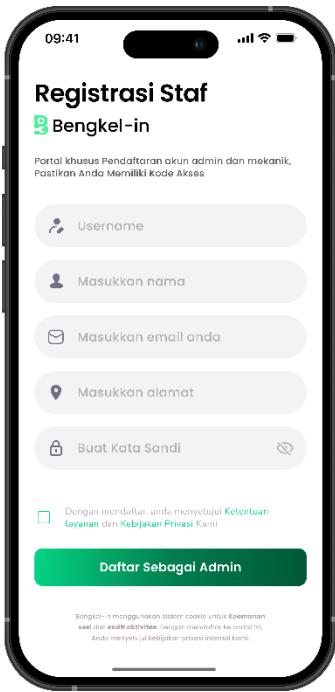
#### 3.3.1 Modul

##### 3.3.1.1 Modul Register Admin

##### 3.3.1.2 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Input Username	Form Input	Admin / Staf	Mobile App
2	Input Nama Lengkap	Form Input	Admin / Staf	Mobile App
3	Input Email Staf	Form Input	Admin / Staf	Mobile App
4	Input Alamat Domisili	Form Input	Admin / Staf	Mobile App
5	Input Kata Sandi	Form Input	Admin / Staf	Mobile App
6	Checkbox Ketentuan Layanan	Checkbox Input	-	Mobile App
7	Tombol Daftar Sebagai Admin	Button	Admin / Staf	Mobile App
8	Validasi Kelengkapan Data	Validasi	-	Mobile App
9	Simpan Data Staf Baru	Process / Query	Admin / Staf	Mobile App

### 3.3.1.2 Spesifikasi Layar Utama



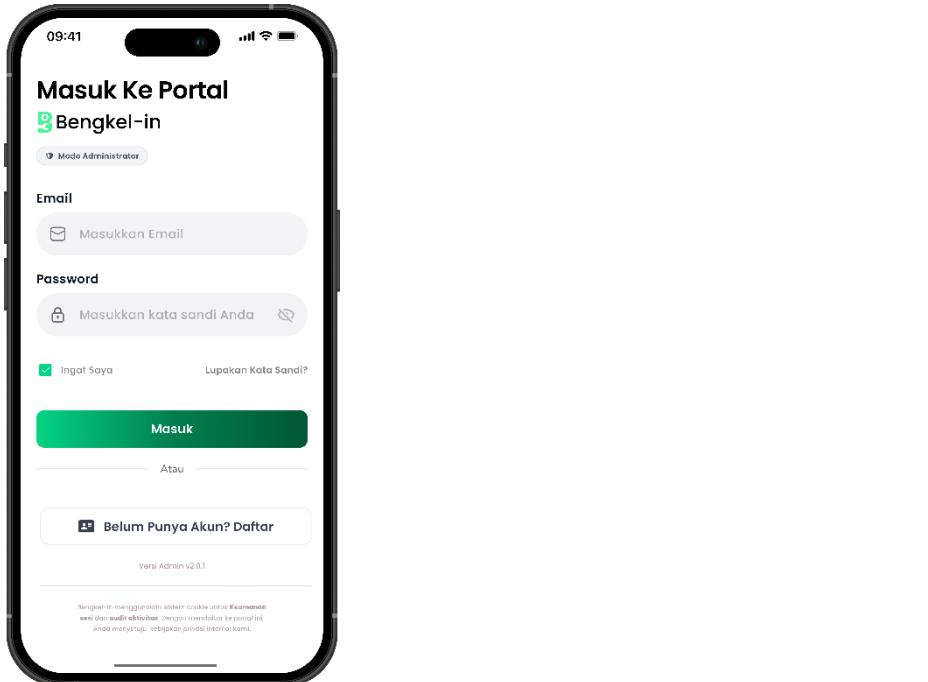
### 3.3.2 Modul Login Admin

#### 3.3.2.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Input Email Admin	Form Input	Admin / Staf	Mobile App
2	Input Kata Sandi	Form Input	Admin / Staf	Mobile App
3	Validasi Kredensial (Hak Akses)	Validasi	Admin / Staf	Mobile App
4	Checkbox "Ingat Saya"	Checkbox Input	-	Mobile App
5	Tombol Masuk Portal	Button	Admin / Staf	Mobile App
6	Link Lupa Kata Sandi	Link	-	Mobile App

7	Tombol Daftar Akun Baru	Button Link	-	Mobile App
8	Simpan Sesi Admin	Session	Admin / Staf	Mobile App
9	Pesan Error (Akses Ditolak)	Notifikasi	-	Mobile App

### 3.3.2.2 Spesifikasi Layar Utama



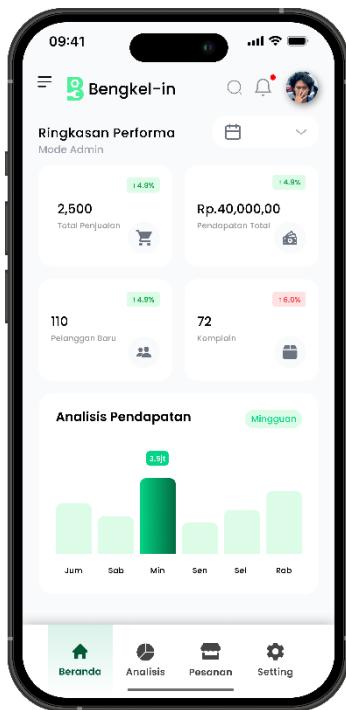
### 3.3.3 Modul Home Dashboard

#### 3.3.3.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Tampil Ringkasan Penjualan	Query / View	Transaksi	Mobile App (Admin)
2	Tampil Total Pendapatan	Query / View	Pembayaran	Mobile App (Admin)

3	Tampil Jumlah Pelanggan Baru	Query / View	Pelanggan / Users	Mobile App (Admin)
4	Tampil Data Komplain	Query / View	Komplain / Feedback	Mobile App (Admin)
5	Grafik Analisis Pendapatan	Chart / Visualization	Pembayaran	Mobile App (Admin)
6	Filter Periode (Tanggal)	Dropdown Input	-	Mobile App (Admin)
7	Tombol Notifikasi	Button Link	Notifikasi	Mobile App (Admin)
8	Tombol Profil Admin	Button Link	Admin / Staf	Mobile App (Admin)
9	Menu Sidebar (Hamburger)	Button	-	Mobile App (Admin)
10	Navigasi Menu Bawah	Navigation Bar	-	Mobile App (Admin)

### 3.3.3.2 Spesifikasi Layar Utama



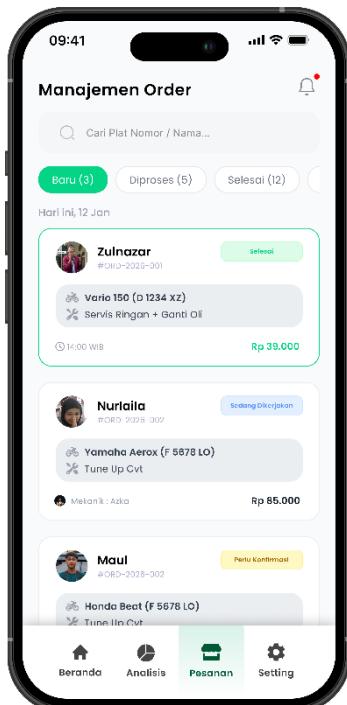
### 3.3.4 Modul Manajemen Pesanan

#### 3.3.4.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	<b>Pencarian Pesanan (Search Bar)</b>	Form Input	Transaksi / Pelanggan	Mobile App (Admin)
2	<b>Filter Status (Baru, Diproses, Selesai)</b>	Button Filter	-	Mobile App (Admin)
3	<b>Menampilkan Daftar Pesanan</b>	Query / View	Transaksi	Mobile App (Admin)
4	<b>Menampilkan Data Pelanggan</b>	Query / View	Pelanggan / Users	Mobile App (Admin)

5	Menampilkan Detail Kendaraan	Query / View	Kendaraan	Mobile App (Admin)
6	Menampilkan Status Pengerjaan	Output Status	Transaksi	Mobile App (Admin)
7	Menampilkan Info Mekanik	Query / View	Mekanik / Staf	Mobile App (Admin)
8	Tombol Notifikasi	Button Link	Notifikasi	Mobile App (Admin)
9	Navigasi Menu Bawah	Navigation Bar	-	Mobile App (Admin)

### 3.3.4.2 Spesifikasi Layar Utama



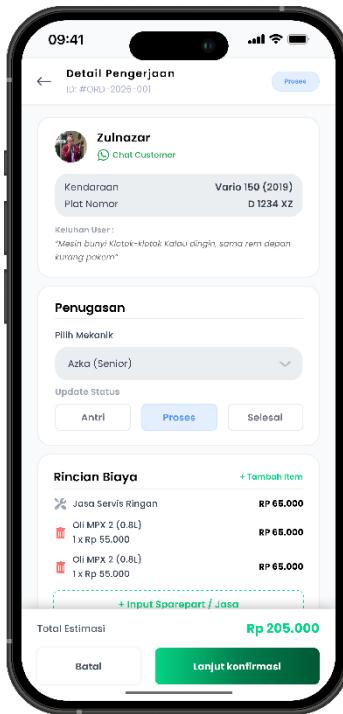
### 3.3.5 Modul Manajemen Pesanan Detail Penggerjaan

#### 3.3.5.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Info Pelanggan	Query / View	Pelanggan / Users	Mobile App (Admin)
2	Menampilkan Data Kendaraan	Query / View	Kendaraan	Mobile App (Admin)
3	Update Status Penggerjaan	Dropdown / Button	Transaksi	Mobile App (Admin)
4	Tambah Item Jasa/Servis	Button Input	Detail_Transaksi	Mobile App (Admin)
5	Tambah Item Sparepart	Button Input	Detail_Transaksi	Mobile App (Admin)
6	Pilih Mekanik Penanggungjawab	Dropdown Input	Staf / Mekanik	Mobile App (Admin)
7	Hapus Item (Servis/Part)	Button Delete	Detail_Transaksi	Mobile App (Admin)
8	Kalkulasi Total Tagihan	Process (Auto)	Transaksi	Mobile App (Admin)
9	Konfirmasi Pembayaran (Lunas)	Switch / Button	Pembayaran	Mobile App (Admin)

10	Cetak Struk / Invoice	Button Output	-	Mobile App (Admin)
----	-----------------------	---------------	---	--------------------

### 3.3.5.2 Spesifikasi Layar Utama



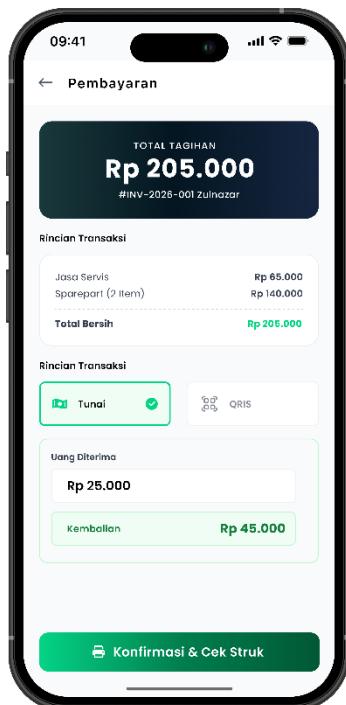
### 3.3.6 Modul Admin Konfirmasi Pembayaran

#### 3.3.6.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	<b>Tampil Total Tagihan &amp; ID Invoice</b>	Query / View	Transaksi / Pembayaran	Mobile App (Admin)
2	<b>Tampil Rincian Transaksi (Jasa &amp; Part)</b>	Query / View	Detail_Transaksi	Mobile App (Admin)
3	<b>Pilih Metode Pembayaran (Tunai/QRIS)</b>	Button Selection	Pembayaran	Mobile App (Admin)

4	Input Nominal Uang Diterima	Form Input	Pembayaran	Mobile App (Admin)
5	Kalkulasi Kembalian (Otomatis)	Process / Calculation	-	Mobile App (Admin)
6	Tombol Konfirmasi & Cek Struk	Button / Update	Transaksi / Pembayaran	Mobile App (Admin)
7	Tombol Kembali (Back)	Navigation	-	Mobile App (Admin)

### 3.3.6.2 Spesifikasi Layar Utama

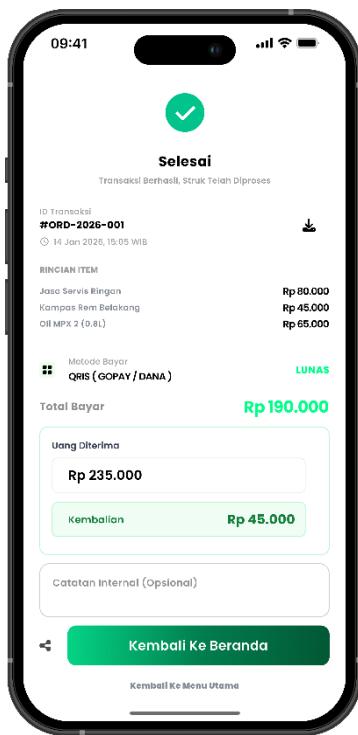


### 3.3.7 Modul Transaksi Sukses Pelanggan oleh Admin

#### 3.3.7.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Status Sukses	Output / View	-	Mobile App (Admin)
2	Menampilkan ID & Tanggal Transaksi	Query / View	Transaksi	Mobile App (Admin)
3	Tombol Unduh Struk (Download)	Button / Process	Transaksi / Laporan	Mobile App (Admin)
4	Menampilkan Rincian Item (Jasa/Part)	Query / View	Detail_Transaksi	Mobile App (Admin)
5	Menampilkan Status Pembayaran (Lunas)	Query / View	Pembayaran	Mobile App (Admin)
6	Menampilkan Kalkulasi Uang & Kembalian	Query / View	Pembayaran	Mobile App (Admin)
7	Input Catatan Internal (Opsional)	Form Input / Update	Transaksi	Mobile App (Admin)
8	Tombol Bagikan (Share)	Button Link	-	Mobile App (Admin)
9	Tombol Kembali Ke Beranda	Navigation	-	Mobile App (Admin)

### 3.3.7.2 Spesifikasi Layar Utama



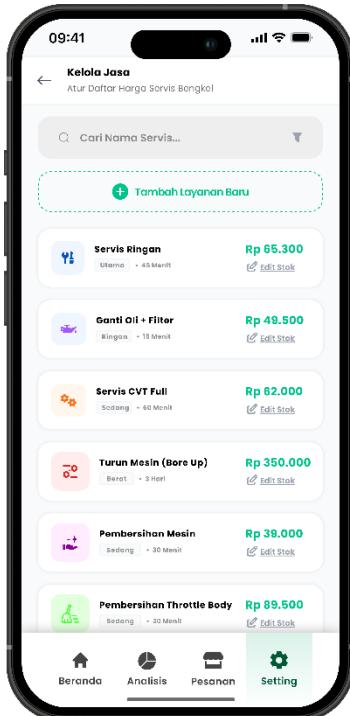
### 3.3.8 Modul Kelola Data Jasa

#### 3.3.8.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Pencarian Nama Servis	Form Input	Jasa_Servis	Mobile App (Admin)
2	Filter Kategori Servis	Button Filter	Jasa_Servis	Mobile App (Admin)
3	Tombol Tambah Layanan Baru	Button Link	Jasa_Servis	Mobile App (Admin)
4	Menampilkan Daftar Nama Jasa	Query / View	Jasa_Servis	Mobile App (Admin)

5	Menampilkan Harga Layanan	Query / View	Jasa_Servis	Mobile App (Admin)
6	Menampilkan Estimasi Waktu	Query / View	Jasa_Servis	Mobile App (Admin)
7	Tombol Edit Layanan/Stok	Button Update	Jasa_Servis	Mobile App (Admin)
8	Navigasi Menu Bawah (Setting)	Navigation Bar	-	Mobile App (Admin)
9	Tombol Kembali (Back)	Navigation	-	Mobile App (Admin)

### 3.3.8.2 Spesifikasi Layar Utama

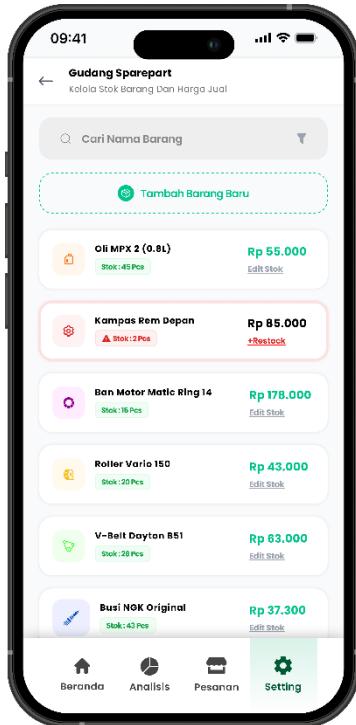


### 3.3.9 Modul Kelola Data Sparepart

#### 3.3.9.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Pencarian Nama Barang	Form Input	Sparepart / Inventaris	Mobile App (Admin)
2	Filter Kategori/Stok	Button Filter	Sparepart / Inventaris	Mobile App (Admin)
3	Tombol Tambah Barang Baru	Button Link	Sparepart / Inventaris	Mobile App (Admin)
4	Menampilkan Daftar Barang	Query / View	Sparepart / Inventaris	Mobile App (Admin)
5	Menampilkan Harga Jual	Query / View	Sparepart / Inventaris	Mobile App (Admin)
6	Menampilkan Jumlah Stok	Query / View	Sparepart / Inventaris	Mobile App (Admin)
7	Validasi Stok Minimum (Alert)	Logic / Validation	Sparepart	Mobile App (Admin)
8	Tombol Edit Stok / Restock	Button Update	Sparepart / Inventaris	Mobile App (Admin)
9	Navigasi Menu Bawah (Setting)	Navigation Bar	-	Mobile App (Admin)

### 3.3.9.2 Spesifikasi Layar Utama



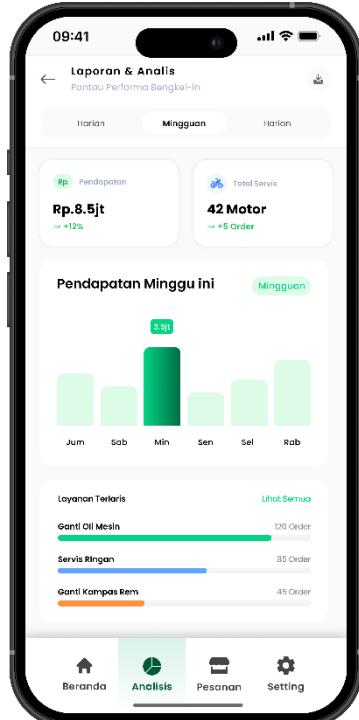
### 3.3.10 Modul Analisis Penjualan (Admin)

#### 3.3.10.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Filter Periode (Harian/Mingguan)	Button Filter	-	Mobile App (Admin)
2	Tombol Unduh Laporan (PDF/Excel)	Button Output	Laporan / Transaksi	Mobile App (Admin)
3	Tampil Total Pendapatan	Query Aggregate	Pembayaran	Mobile App (Admin)
4	Tampil Total Volume Servis	Query Aggregate	Transaksi	Mobile App (Admin)

5	Indikator Pertumbuhan (Percentase)	Logic / Calculation	Pembayaran	Mobile App (Admin)
6	Grafik Batang Pendapatan	Chart Visualization	Pembayaran	Mobile App (Admin)
7	Tampil Layanan Terlaris	Query Aggregate	Detail_Transaksi	Mobile App (Admin)
8	Link "Lihat Semua" (Detail)	Link	-	Mobile App (Admin)
9	Navigasi Menu Bawah (Analisis)	Navigation Bar	-	Mobile App (Admin)

### 3.3.10.2 Spesifikasi Layar Utama

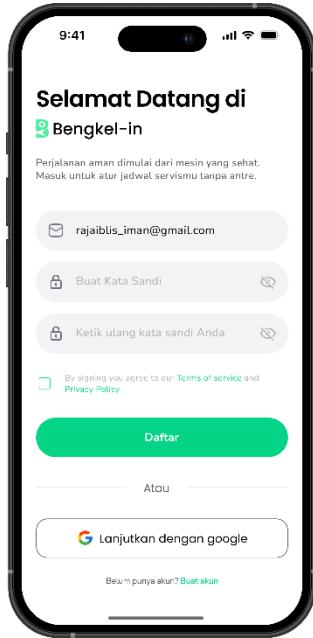


### 3.3.11 Modul Registrasi Pelanggan

#### 3.3.11.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Input Email	Form Input	Pelanggan / Users	Mobile App (Pelanggan)
2	Input Kata Sandi Baru	Form Input	Pelanggan / Users	Mobile App (Pelanggan)
3	Input Konfirmasi Kata Sandi	Form Input	-	Mobile App (Pelanggan)
4	Lihat/Sembunyikan Sandi (Eye Icon)	Button Toggle	-	Mobile App (Pelanggan)
5	Checkbox Syarat & Ketentuan	Checkbox Input	-	Mobile App (Pelanggan)
6	Validasi Kesamaan Sandi	Logic / Validation	-	Mobile App (Pelanggan)
7	Tombol Daftar (Register)	Button Process	Pelanggan / Users	Mobile App (Pelanggan)
8	Daftar via Google	API Button	Users	Mobile App (Pelanggan)
9	Link Pindah ke Halaman Login	Navigation Link	-	Mobile App (Pelanggan)

### 3.3.11.2 Spesifikasi Layar Utama



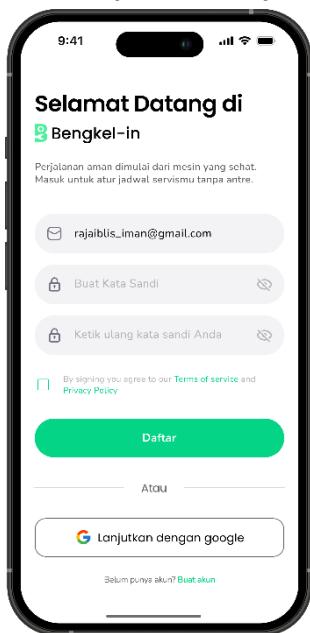
### 3.3.12 Modul Login Pelanggan

#### 3.3.12.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Input Email Pelanggan	Form Input	Pelanggan / Users	Mobile App (Pelanggan)
2	Input Kata Sandi	Form Input	Pelanggan / Users	Mobile App (Pelanggan)
3	Lihat/Sembunyikan Sandi (Eye Icon)	Button Toggle	-	Mobile App (Pelanggan)
4	Checkbox Syarat & Ketentuan	Checkbox Input	-	Mobile App (Pelanggan)
5	Validasi Kredensial (Cek Akun)	Logic / Validation	Pelanggan / Users	Mobile App (Pelanggan)

6	Tombol Masuk (Login)	Button Process	Pelanggan / Users	Mobile App (Pelanggan)
7	Login via Google	API Button	Users	Mobile App (Pelanggan)
8	Login via Apple	API Button	Users	Mobile App (Pelanggan)
9	Simpan Sesi Login	Session	Users	Mobile App (Pelanggan)

### 3.3.12.2 Spesifikasi Layar Utama



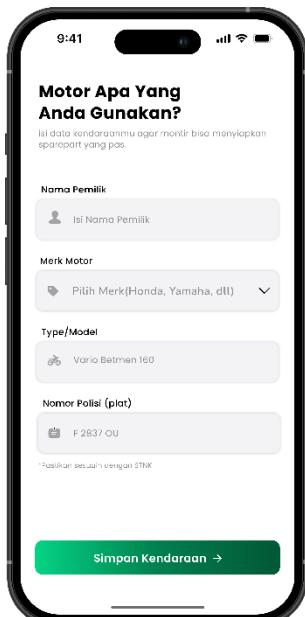
### 3.3.13 Modul Input Data Motor

#### 3.3.13.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Input Nama Pemilik	Form Input	Kendaraan	Mobile App (Pelanggan)

2	Pilih Merk Motor	Dropdown Input	Kendaraan / Ref_Merk	Mobile App (Pelanggan)
3	Input Tipe/Model Motor	Form Input	Kendaraan	Mobile App (Pelanggan)
4	Input Nomor Polisi (Plat)	Form Input	Kendaraan	Mobile App (Pelanggan)
5	Tombol Simpan Kendaraan	Button Process	Kendaraan	Mobile App (Pelanggan)
6	Validasi Input (Wajib Isi)	Logic / Validation	-	Mobile App (Pelanggan)
7	Simpan Data ke Database	Process / Query	Kendaraan	Mobile App (Pelanggan)
8	Tombol Kembali (Back)	Navigation	-	Mobile App (Pelanggan)

### 3.3.13.2 Spesifikasi Layar Utama



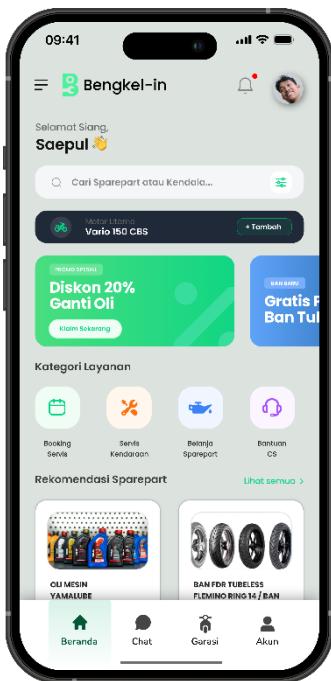
### 3.3.14 Modul Homepage Pelanggan

#### 3.3.14.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Nama Pengguna	Query / View	Pelanggan / Users	Mobile App (Pelanggan)
2	Pencarian (Sparepart/Jasa)	Form Input	Sparepart / Jasa	Mobile App (Pelanggan)
3	Tombol Filter Pencarian	Button Filter	-	Mobile App (Pelanggan)
4	Menampilkan Motor Utama	Query / View	Kendaraan	Mobile App (Pelanggan)
5	Tombol Tambah Motor (+)	Button Link	Kendaraan	Mobile App (Pelanggan)
6	Menampilkan Banner Promo	Query / View	Promo / Banner	Mobile App (Pelanggan)
7	Menu Navigasi Layanan (Booking, dll)	Button Link	-	Mobile App (Pelanggan)
8	Menampilkan Rekomendasi Sparepart	Query / View	Sparepart	Mobile App (Pelanggan)
9	Link "Lihat Semua" (Sparepart)	Link	-	Mobile App (Pelanggan)

10	Tombol Notifikasi (Lonceng)	Button Link	Notifikasi	Mobile App (Pelanggan)
11	Navigasi Menu Bawah	Navigation Bar	-	Mobile App (Pelanggan)

### 3.3.14.2 Spesifikasi Layar Utama



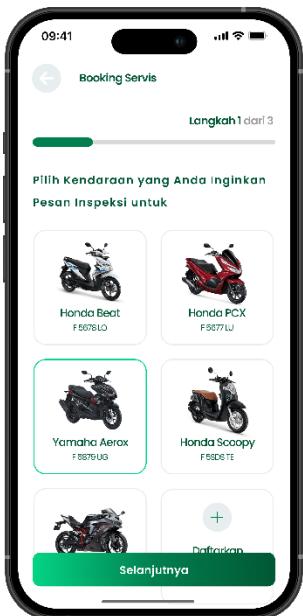
### 3.3.15 Modul Pilih Kendaraan (Booking Servis)

#### 3.3.15.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Progress Bar (Langkah 1)	UI Indicator	-	Mobile App (Pelanggan)
2	Menampilkan Daftar Kendaraan Saya	Query / View	Kendaraan	Mobile App (Pelanggan)
3	Pilih Kartu Kendaraan (Selection)	Button Input	Kendaraan / Booking	Mobile App (Pelanggan)

4	Tombol Tambah Kendaraan Baru (+)	Button Link	-	Mobile App (Pelanggan)
5	Tombol Selanjutnya (Next)	Button Nav	Booking	Mobile App (Pelanggan)
6	Validasi Pilihan (Wajib Pilih)	Logic / Validation	-	Mobile App (Pelanggan)
7	Tombol Kembali (Back)	Navigation	-	Mobile App (Pelanggan)

### 3.3.15.2 Spesifikasi Layar Utama



### 3.3.16 Modul Pilih Jenis Layanan (Booking Servis)

#### 3.3.16.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Progress Bar (Langkah 2)	UI Indicator	-	Mobile App (Pelanggan)

2	Pilih Jenis Servis	Dropdown Input	Jasa_Servis	Mobile App (Pelanggan)
3	Input Keluhan Tambahan	Form/Dropdown Input	Booking / Transaksi	Mobile App (Pelanggan)
4	Pilih Sparepart (Opsional)	Dropdown Input	Sparepart	Mobile App (Pelanggan)
5	Pilih Mekanik	Dropdown Input	Mekanik / Staf	Mobile App (Pelanggan)
6	Tombol Selanjutnya (Next)	Button Nav	Booking	Mobile App (Pelanggan)
7	Validasi Kelengkapan Data	Logic / Validation	-	Mobile App (Pelanggan)
8	Tombol Kembali (Back)	Navigation	-	Mobile App (Pelanggan)

### 3.3.16.2 Spesifikasi Layar Utama



### 3.3.17 Modul Pilih Jadwal (Booking Servis)

#### 3.3.17.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Progress Bar (Langkah 3)	UI Indicator	-	Mobile App (Pelanggan)
2	Menampilkan Kalender (Hari & Tanggal)	Query / View	Jadwal_Operasional	Mobile App (Pelanggan)
3	Pilih Tanggal Kedatangan	Button Input	Booking / Transaksi	Mobile App (Pelanggan)
4	Menampilkan Slot Waktu (Available)	Query / Logic	Booking	Mobile App (Pelanggan)

5	Pilih Jam Servis	Button Input	Booking / Transaksi	Mobile App (Pelanggan)
6	Validasi Ketersediaan Slot	Logic / Validation	Booking	Mobile App (Pelanggan)
7	Tombol Selanjutnya (Konfirmasi)	Button Nav	Booking	Mobile App (Pelanggan)
8	Tombol Kembali (Back)	Navigation	-	Mobile App (Pelanggan)

### 3.3.17.2 Spesifikasi Layar Utama

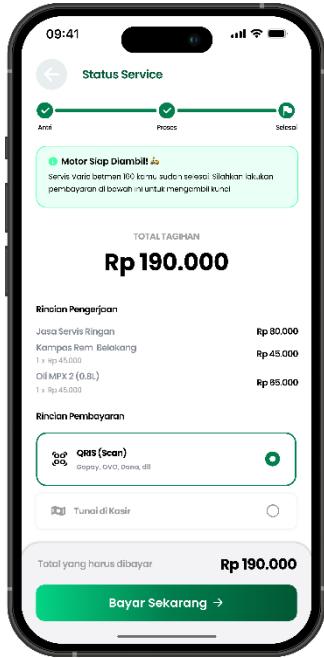


### 3.3.18 Modul Pembayaran (Booking Servis)

#### 3.3.18.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Progress Status (Selesai)	UI Indicator	Transaksi	Mobile App (Pelanggan)
2	Menampilkan Notifikasi "Motor Siap"	Logic / View	Transaksi	Mobile App (Pelanggan)
3	Menampilkan Total Tagihan	Query / View	Transaksi	Mobile App (Pelanggan)
4	Menampilkan Rincian Item (Jasa/Part)	Query / View	Detail_Transaksi	Mobile App (Pelanggan)
5	Pilih Metode Pembayaran (QRIS/Tunai)	Radio Button Input	Pembayaran	Mobile App (Pelanggan)
6	Kalkulasi Subtotal Item	Calculation	-	Mobile App (Pelanggan)
7	Tombol "Bayar Sekarang"	Button Process	Pembayaran	Mobile App (Pelanggan)
8	Tombol Kembali (Back)	Navigation	-	Mobile App (Pelanggan)

### 3.3.18.2 Spesifikasi Layar Utama



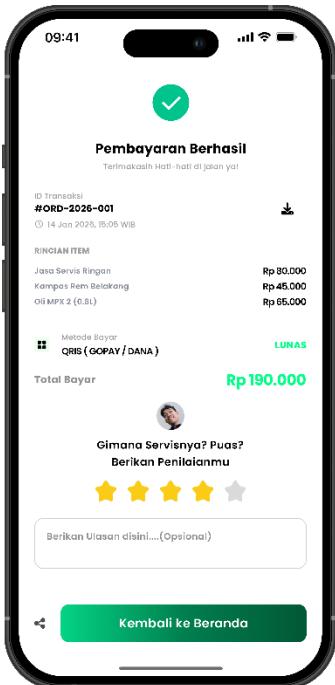
### 3.3.19 Modul Transaksi Sukses (Booking Servis)

#### 3.3.19.1 Fungsi Modul

No	Fungsi	Jenis	Tabel Terkait	Kategori
1	Menampilkan Status Berhasil	UI / View	-	Mobile App (Pelanggan)
2	Menampilkan ID & Tanggal Transaksi	Query / View	Transaksi	Mobile App (Pelanggan)
3	Tombol Unduh Struk (Download)	Button Process	Transaksi / Laporan	Mobile App (Pelanggan)
4	Menampilkan Rincian Item (Jasa/Part)	Query / View	Detail_Transaksi	Mobile App (Pelanggan)

5	Menampilkan Status Pembayaran (Lunas)	Query / View	Pembayaran	Mobile App (Pelanggan)
6	Input Rating Bintang (1-5)	Input Rating	Ulasan / Feedback	Mobile App (Pelanggan)
7	Input Teks Ulasan (Opcional)	Form Input	Ulasan / Feedback	Mobile App (Pelanggan)
8	Tombol Share (Bagikan)	Button Link	-	Mobile App (Pelanggan)
9	Tombol Kembali Ke Beranda	Navigation	-	Mobile App (Pelanggan)

### 3.3.19.2 Spesifikasi Layar Utama



# SOFTWARE PROJECT MANAGEMENT PLAN

## 1. PENDAHULUAN

### 1.1 Gambaran Proyek

Bengkel-In adalah aplikasi berbasis mobile (Android/iOS) yang dirancang untuk memudahkan pengguna kendaraan bermotor dalam menemukan, memesan, dan mengelola layanan bengkel secara digital. Aplikasi ini memungkinkan pelanggan untuk mencari bengkel terdekat, melihat layanan dan harga, melakukan booking servis, serta memantau status penggerjaan kendaraan secara real-time.

Di sisi lain, pemilik bengkel dapat mengelola data layanan, jadwal servis, antrean pelanggan, serta laporan transaksi melalui sistem terintegrasi. Bengkel-In bertujuan meningkatkan efisiensi operasional bengkel sekaligus memberikan kenyamanan dan transparansi bagi pelanggan.

### 1.2 Dokumen-dokumen Dalam Proyek

Dokumen yang digunakan dalam pengembangan Aplikasi Bengkel Berbasis Mobile meliputi:

- SPMP (Software Project Management Plan)
- SRS (Software Requirements Specification)
- SDD (Software Design Document)
- STD (Software Testing Document)

### 1.3 Evolusi SPMP

SPMP ini disusun sebagai pedoman dalam mengelola proyek pengembangan Aplikasi Bengkel Berbasis Mobile. Setiap anggota tim bertanggung jawab terhadap tugas dan dokumen yang telah ditentukan. SPMP dapat mengalami perubahan apabila terdapat penyesuaian kebutuhan pengguna, ruang lingkup proyek, atau metode pengembangan.

### 1.4 Material Acuan

- IEEE Standards
- <https://www1.in.tum.de/stars.globalse.org/stars1/docs/SPMP/Examples/Examples.htm>
- <https://www.slideshare.net/ttchenok/spmp-18728488>
- <https://github.com/firstiaulyaa/RPL-D-5/blob/master/SPMP.md>
- <https://github.com/oksar3110-0110/RPL-D-7/blob/master/SPMP.md>

### 1.5 Definisi, Akronim, dan Singkatan

Android	Sistem operasi mobile yang dikembangkan oleh Google untuk perangkat smartphone dan tablet.
iOS	Sistem operasi mobile yang dikembangkan oleh Apple untuk perangkat iPhone dan iPad.
SPMP (Software Project Management Plan)	Dokumen perencanaan yang menjelaskan pengelolaan, pengawasan, dan pengendalian proyek perangkat lunak.

Aplikasi Bengkel Berbasis Mobile	Aplikasi layanan bengkel kendaraan yang berjalan pada perangkat mobile untuk mengelola servis, pelanggan, mekanik, dan transaksi.
IEEE (Institute of Electrical and Electronics Engineers)	Organisasi profesi internasional yang mengembangkan standar di bidang teknologi dan rekayasa.

## 2. ORGANISASI PROYEK

### 2.1 Struktur Organisasi

Model proses yang digunakan dalam pengembangan aplikasi Bengkel-In adalah Agile dengan framework Scrum. Metode ini dipilih karena mampu mengakomodasi perubahan kebutuhan pengguna serta cocok untuk pengembangan aplikasi mobile yang membutuhkan iterasi cepat dan umpan balik berkelanjutan.

#### 2.1.1 Definisi Scrum

Scrum adalah kerangka kerja Agile yang membagi pengembangan ke dalam iterasi singkat yang disebut *sprint* (1–4 minggu). Setiap sprint menghasilkan fitur aplikasi Bengkel-In yang siap diuji dan dievaluasi.

#### 2.1.2 Kelebihan Metode Scrum

- Fleksibilitas Tinggi**  
Mudah menyesuaikan fitur aplikasi mobile sesuai kebutuhan bengkel dan pelanggan.
- Iterasi Singkat**  
Fitur dapat diuji dan diperbaiki lebih cepat.
- Kolaborasi Tim yang Kuat**  
Daily Scrum meningkatkan koordinasi antar anggota tim.
- Prioritas yang Jelas**  
Fitur inti aplikasi dikerjakan lebih dahulu.
- Peningkatan Berkelanjutan**  
Evaluasi rutin meningkatkan kualitas aplikasi dan proses pengembangan.

### 2.2 Struktur Organisasi

Saeful Iman  
Project manager

Neng Rista & M.  
Nazar Zulfikar  
Database Engineer

Nurlaila & Dwi  
Zahwa Abdillah  
System Analyst

M. Daffa Fasya Ramadhan  
& Azka Putra Priatna  
UI/UX Designer

## 2.3 Lingkup dan Tanggung Jawab

No.	Tugas	Penjelasan
1.	Project Manager	Mengintegrasikan berbagai kegiatan yang berbeda untuk mencapai tujuan tertentu, bertanggung jawab dalam pengambilan keputusan, dan menjadi pusat koordinasi antar tim. Sebagai komunikator utama, Project Manager memastikan setiap anggota tim mendapatkan dukungan yang diperlukan untuk menyelesaikan tugas mereka. Project Manager juga aktif membantu anggota tim dengan memberikan arahan yang jelas, menyelesaikan hambatan teknis maupun non-teknis, serta memastikan setiap tugas dapat dikerjakan secara efisien. Dengan pendekatan ini, Project Manager berperan sebagai pemimpin sekaligus mitra kerja bagi seluruh anggota tim.
2.	Database Engineer	Bertugas sebagai pengelola pusat kendali data dalam proyek, mencakup pengelolaan sistem basis data, perancangan struktur data, hingga memastikan data yang dibutuhkan dapat diakses dengan optimal oleh tim lain, seperti System Analyst dan UI/UX Designer, sesuai kebutuhan proyek.
3.	System Analyst	Bertanggung jawab menganalisis kebutuhan sistem yang akan diimplementasikan, mengevaluasi kelebihan dan kekurangan sistem, serta memberikan hasil analisis yang menjadi panduan bagi Database Engineer dan UI/UX Designer untuk pengembangan lebih lanjut.
4.	UI/UX Designer	Merancang desain aplikasi secara menyeluruh, mulai dari antarmuka hingga pengalaman pengguna. Bekerja sama erat dengan System Analyst untuk memastikan desain sesuai kebutuhan fungsional, serta berkoordinasi dengan Database Engineer untuk mendukung data yang diperlukan dalam desain.

## 3. Proses Manajerial

### 3.1 Tujuan dan Prioritas Manajemen

Proyek Aplikasi Bengkel-In bertujuan untuk mengembangkan sebuah aplikasi mobile yang dapat membantu pengelolaan operasional bengkel kendaraan secara efektif, terstruktur, dan real-time. Aplikasi ini dirancang untuk meningkatkan kualitas layanan bengkel melalui kemudahan pencatatan servis, manajemen data pelanggan dan kendaraan, pemantauan status pekerjaan, serta pengelolaan transaksi pembayaran.

#### 3.2 Asumsi, Keterkaitan, dan Batasan

##### Asumsi Proyek

Asumsi yang digunakan dalam pengembangan Aplikasi Bengkel Berbasis Mobile adalah sebagai berikut:

- Tim pengembang terdiri dari 7 orang, dengan pembagian tugas:

1. Project Manager

Bertanggung jawab mengatur timeline proyek, memastikan setiap tahapan SDLC berjalan sesuai rencana, serta mengoordinasikan komunikasi antar anggota tim.

2. UI/UX Designer (2 orang)

Mendesain antarmuka aplikasi mobile yang responsif, menarik, dan mudah digunakan, serta membuat mockup atau prototype menggunakan tools seperti Figma atau Canva.

3. Database Engineer (2 orang)

Merancang dan mengembangkan database untuk menyimpan data pelanggan, kendaraan, servis, suku cadang, mekanik, dan transaksi.

4. System Analyst (2 orang)

Menganalisis kebutuhan sistem dan membuat model sistem menggunakan UML seperti Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram.

- Ketersediaan Peralatan dan Perangkat Lunak

Seluruh anggota tim memiliki akses ke perangkat keras (laptop/PC) dengan spesifikasi yang memadai serta perangkat lunak pendukung pengembangan seperti Android Studio, Visual Paradigm, Draw.io, Figma, dan tools pendukung lainnya.

Substansi dari proyek ini adalah:

- Aplikasi berbasis mobile yang dirancang untuk mendukung operasional bengkel kendaraan.
- Aplikasi memungkinkan pengguna untuk:
  - Admin Bengkel: Mengelola data pelanggan, kendaraan, layanan servis, suku cadang, dan laporan transaksi.
  - Mekanik: Melihat daftar pekerjaan servis dan memperbarui status pengerjaan kendaraan.
  - Pelanggan: Melakukan pendaftaran servis, melihat status servis kendaraan, serta riwayat perawatan melalui aplikasi mobile.

### Kendala Proyek

Kendala yang dihadapi dalam proyek Aplikasi Bengkel-In antara lain:

- Keterbatasan anggaran, terutama untuk penggunaan software atau tools desain dan pengembangan versi premium seperti Figma Plugin, Visual Paradigm, atau layanan cloud tertentu.
- Keterbatasan waktu dan koordinasi tim, terutama ketika pengerjaan dilakukan secara bersamaan di luar jam perkuliahan.
- Kendala teknis, seperti perbedaan spesifikasi perangkat mobile yang digunakan untuk pengerjaan projek aplikasi.

### 3.3 Manajemen Risiko

Resiko	Manajemen Resiko
Estimasi biaya dan waktu tidak sesuai	<ul style="list-style-type: none"> <li>Melakukan analisis kebutuhan proyek secara menyeluruh sebelum pengembangan dimulai.</li> <li>Menyusun spesifikasi aplikasi Bengkel-In secara formal dan terperinci.</li> <li>Melakukan diskusi dengan calon pengguna (pemilik bengkel dan pelanggan).</li> <li>Membuat prototype aplikasi mobile untuk memvalidasi rencana proyek.</li> </ul>
Keterlambatan pengembangan fitur aplikasi	<ul style="list-style-type: none"> <li>Memperketat pengendalian jadwal sprint.</li> <li>Membatasi perubahan fitur yang terlalu sering di luar backlog prioritas.</li> <li>Meningkatkan komunikasi antar anggota tim melalui daily meeting.</li> </ul>
Kegagalan pada komponen sistem	<ul style="list-style-type: none"> <li>Melakukan analisis awal terhadap arsitektur aplikasi mobile.</li> <li>Mengidentifikasi keterbatasan perangkat, framework, dan sumber daya sejak awal.</li> </ul>
Kegagalan kinerja aplikasi (performance)	<ul style="list-style-type: none"> <li>Menerapkan prosedur jaminan kualitas (quality assurance).</li> <li>Menggunakan desain sistem yang sesuai dengan kebutuhan aplikasi mobile.</li> <li>Melakukan pengujian performa pada berbagai perangkat.</li> </ul>
Kesulitan teknis dalam pengembangan mobile	<ul style="list-style-type: none"> <li>Melakukan analisis teknis untuk mengidentifikasi masalah kompleks sejak awal.</li> <li>Meninjau kembali analisis biaya dan manfaat untuk memastikan kelayakan teknis.</li> <li>Melakukan simulasi dan uji coba fitur secara bertahap.</li> </ul>
Kegagalan integrasi fitur eksternal (maps, notifikasi, dll.)	<ul style="list-style-type: none"> <li>Menggunakan metode benchmarking dengan aplikasi sejenis.</li> <li>Menguji integrasi API eksternal secara bertahap.</li> <li>Mengoptimalkan kinerja sistem melalui tuning dan penyempurnaan.</li> </ul>

### 3.4 Mekanisme Pemantauan dan Pengendalian

- Kerja Kelompok 4x dalam Seminggu.
- Penyimpanan dokumen secara Bersama.

### 4. Proses Teknis

#### 4.1 Metode, Alat, dan Teknik

Proyek **Bengkel-In** menggunakan kerangka kerja **Agile** dengan pendekatan **Scrum** untuk memastikan proses pengembangan perangkat lunak berjalan secara fleksibel dan iteratif.

Metode ini diterapkan melalui beberapa sprint yang dilakukan secara berulang, di mana setiap sprint menghasilkan peningkatan fungsi aplikasi yang dapat langsung diuji dan dievaluasi. Pendekatan ini memungkinkan tim pengembang menyesuaikan sistem Bengkel-In secara bertahap sesuai dengan kebutuhan pengguna serta meminimalkan risiko kesalahan pada tahap akhir pengembangan.

#### 4.2 Dokumentasi Perangkat Lunak

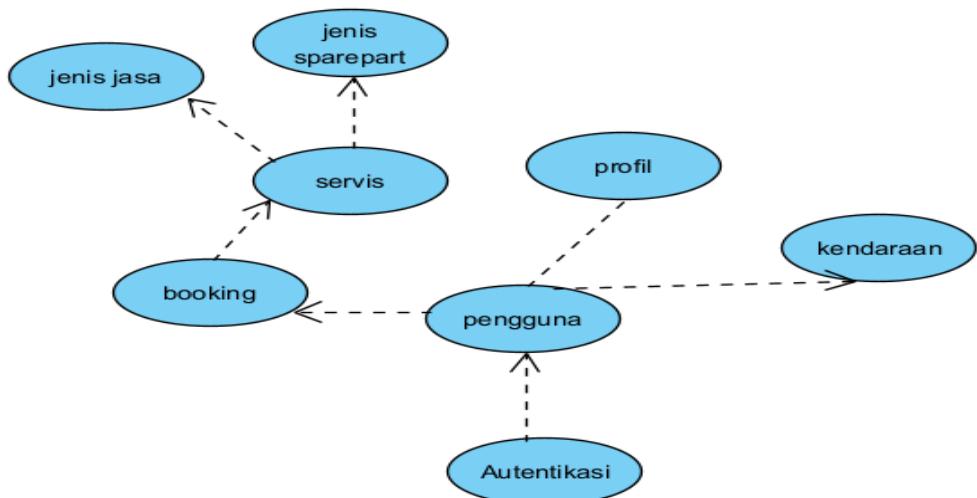
Dokumentasi perangkat lunak dalam proyek Bengkel-In disusun secara sistematis dan berkesinambungan selama proses pengembangan berlangsung. Dokumentasi yang dihasilkan meliputi Piagam Proyek, Dokumentasi Kebutuhan Bisnis, Spesifikasi Fungsional, Analisis Manfaat dan Biaya, Spesifikasi Teknis, Detail Design Document, Rencana Pelaksanaan, serta Dokumen Realisasi Manfaat. Seluruh dokumen ini berfungsi sebagai acuan utama dalam pengembangan, pengujian, dan evaluasi aplikasi Bengkel-In agar berjalan sesuai dengan tujuan proyek.

#### 4.3 Fungsi Pendukung Proyek

Fungsi pendukung proyek Bengkel-In mencakup seluruh aktivitas yang mendukung kelancaran pengembangan perangkat lunak, termasuk pengelolaan dokumen, koordinasi antar anggota tim, serta pemantauan progres proyek. Seluruh dokumen pendukung akan diselesaikan pada fase yang sesuai dengan tahapan pengembangan, sehingga setiap proses dapat berjalan terstruktur, terkontrol, dan selaras dengan rencana proyek yang telah ditetapkan.

### 5. Jadwal, dan Budget

#### 5.1 Paket Pekerjaan



No	NIM	Nama	Job
1.	20240050032	Saeful Iman	<ul style="list-style-type: none"><li>SPMP</li><li>STD</li><li>SDD: Dokumen dan Modul</li></ul>

2.	20240050141 20240050008	Nurlaila Dwi Zahwa Abdillah	<ul style="list-style-type: none"><li>• SRS: UML</li><li>• SDD: DFD, ERD</li></ul>
3.	20240050056 20240050041	Azka Putra Priyatna M. Daffa fasya R	<ul style="list-style-type: none"><li>• SDD</li><li>• SRS: Mockup, Tabel Non Fungsional</li></ul>
4.	20240050064 20240050020	Neng Rista M. Nazar Zulfikar	<ul style="list-style-type: none"><li>• SDD: DFD, ERD</li><li>• SDD: Tabel Database</li></ul>

## 5.2 Jadwal

Kegiatan	Sep W3	Sep W4	Okt W1	Okt W2	Okt W3	Okt W4	Nov W1	Nov W2	Nov W3	Nov W4	Des W1	Des W2	Des W3	Des W4	Jan W1	Jan W2	Jan W3
Persiapan Presentasi																	
Pelaksanaan Presentasi																	
Revisi Laporan Akhir																	

### 5.3 Ketergantungan/Keterikatan

Dari	Kepada	Kaitan
Project Manager	Anggota	Manager menjadi pengawas anggotanya jika anggota lalai dengan tugasnya maka manager berhak menegur anggotanya, tetapi manager juga tidak berhak semena-mena dengan jabatannya
System Analyst	UI/UX Designer	Bertugas untuk memberi gambaran projek dan alur design kepada UI / UX
System Analyst	Database Engineer	System Analyst menjelaskan kebutuhan teknis dan alur data kepada Database Engineer agar struktur basis data mendukung sistem yang dirancang.
UI/UX Designer	Project Manager	UI/UX Designer menyajikan desain antarmuka kepada manager untuk mendapat evaluasi dan persetujuan sebelum dilanjutkan ke tahap berikutnya.
Database Engineer	System Analyst	Database Engineer berkoordinasi dengan System Analyst untuk memastikan struktur basis data sesuai kebutuhan sistem dan mendukung kelancaran proses kerja.

#### 5.4 Kebutuhan sumber daya

Sumber Daya	Jenis Kebutuhan	Alasan
Sumber Daya Manusia	7 Orang anggota pekerja projek	Dengan jumlah tersebut untuk keseimbangan antara penggerjaan projek, dokumen-dokumen dan pencarian masukan dari pasar, dengan demikian kita dapat mempercepat proses pekerjaan projek
Sumber Daya Perangkat Lunak	Visual Paradigm	Digunakan untuk membuat UML
	Draw.io	
	MySQL	Software ini digunakan untuk pembuatan database
	Figma	Digunakan untuk pembuatan Desain dan Prototyping Aplikasi
Sumber Daya Perangkat Keras	Perangkat Laptop	Perangkat induk yang digunakan dalam penggerjaan projek
		Perangkat yang digunakan untuk uji coba prototyping

#### 5.5 budget

No	Kebutuhan	Jumlah Barang	Satuan	Harga	Jumlah Harga
1	Laptop	4	Unit	Rp. 8.000.000	Rp. 32.000.000
2	Smartphone	5	Unit	Rp. 5.000.000	Rp. 20.000.000
3	Internet	1		Rp. 600.000	Rp. 600.000
4	Konsumsi	360	Pcs	Rp. 20.000	Rp. 7.200.000
5	Transportasi	90	Liter	Rp. 20.000	Rp. 3.600.000
6	Gaji Anggota	3	Bulan	Rp. 25.000	Rp. 9.000.000
Jumlah Biaya					Rp. 72.400.000

# SOFTWARE REQUIREMENTS SPECIFICATION

## 1. PENDAHULUAN

### 1.1.Tujuan

Tujuan penyusunan dokumen software Requirement Specification (SRS) ini adalah untuk mendeskripsikan secara rinci spesifikasi kebutuhan system pada aplikasi Bengkel-in, yaitu sebuah platform bengkel yang menyediakan layanan servis kendaraan secara online dan offline.

Dokumen ini disusun sebagai acuan dalam proses analisis dan pengembangan system agar aplikasi yang dibangun sesuai dengan kebutuhan pengguna, baik pelanggan maupun pihak bengkel, serta dukungan proses bisnis yang berjalan.

Dokumen SRS ini mencakup gambaran umum sistem, analisis kebutuhan pengguna, kebutuhan fungsional dan non-fungsional, serta deskripsi awal rancangan system yang akan menjadi dasar dalam tahap perancangan dan implementasi aplikasi Bengkel-in.

### 1.2 Lingkup Masalah

Aplikasi Bengkel-in merupakan platform publik berbasis website yang dirancang untuk mempermudah masyarakat dalam mengakses layanan bengkel secara lebih efektif dan efisien. Sistem ini memungkinkan pengguna untuk memperoleh informasi terkait layanan servis kendaraan, melakukan booking servis secara online, serta melakukan konsultasi mengenai permasalahan kendaraan dengan pihak bengkel atau mekanik.

Lingkup sistem Bengkel-in meliputi pengelolaan data pengguna, data pelanggan, data kendaraan, layanan servis, mekanik, sparepart, serta proses transaksi servis dan pembayaran. Sistem ini juga mendukung komunikasi antara pengguna dan bengkel melalui media yang terintegrasi, sehingga dapat meningkatkan kualitas layanan dan kepuasan pelanggan.

Dokumen ini membatasi pembahasan pada analisis kebutuhan sistem dan rancangan konseptual aplikasi Bengkel-in, tanpa membahas secara mendalam aspek teknis implementasi seperti detail pemrograman atau struktur basis data.

### 1.3 Overview

Dokumen *Software Requirements Specification* (SRS) ini disusun sebagai acuan dalam pengembangan aplikasi Bengkel-in, yaitu sistem layanan bengkel berbasis website yang menyediakan layanan servis kendaraan secara online dan offline. Dokumen ini bertujuan

untuk memberikan gambaran yang jelas mengenai kebutuhan sistem agar seluruh pihak yang terlibat memiliki pemahaman yang sama.

Dokumen SRS ini terdiri dari beberapa bagian, yaitu pendahuluan, gambaran umum produk, spesifikasi kebutuhan fungsional dan non-fungsional, karakteristik pengguna, serta batasan sistem. Selain itu, dokumen ini juga dilengkapi dengan model perancangan menggunakan UML, ERD, serta mockup aplikasi sebagai pendukung perancangan sistem.

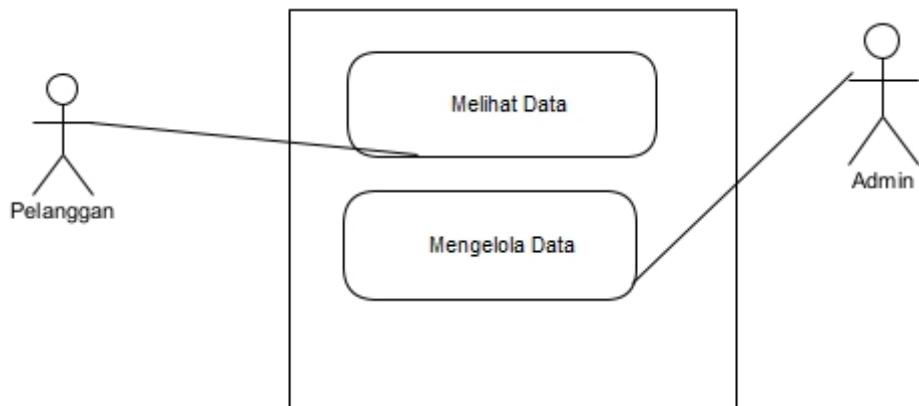
## 2. Gambaran Umum

### 2.1. Perspektif Produk

Aplikasi Bengkel-in merupakan sistem layanan bengkel berbasis website yang ditujukan untuk memudahkan pengguna dalam mengakses informasi dan layanan servis kendaraan secara online maupun offline. Sistem ini berfungsi sebagai media penghubung antara pelanggan dan pihak bengkel.

Bengkel-in menyediakan fitur informasi layanan servis, booking servis, konsultasi kendaraan, serta pengelolaan data servis dan pembayaran yang terintegrasi dengan basis data. Sistem dapat diakses melalui browser pada perangkat yang terhubung ke internet tanpa memerlukan instalasi aplikasi tambahan.

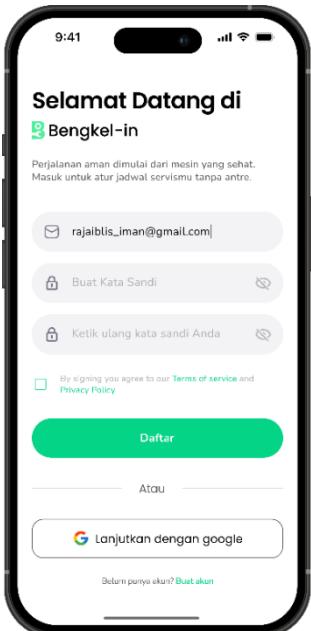
#### 2.1.1 Antarmuka Sistem



## 2.1.2. Antarmuka Pengguna

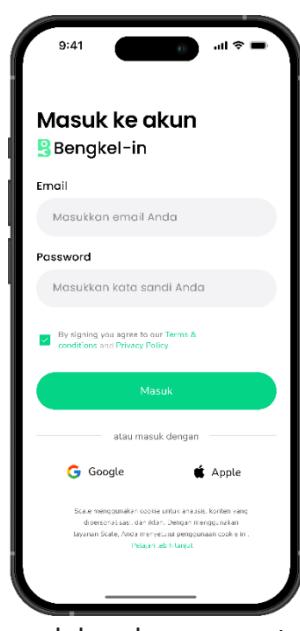
### Pelanggan

- Halaman Selamat Datang



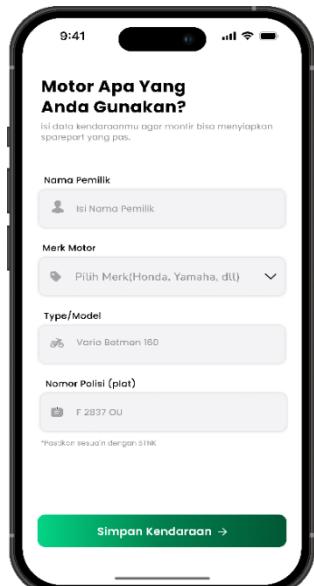
Menampilkan halaman awal aplikasi yang digunakan pelanggan untuk **melakukan pendaftaran akun** dengan mengisi data diri atau menggunakan akun Google.

- Halaman Login



Digunakan oleh pelanggan untuk masuk ke sistem dengan memasukkan email dan password yang telah terdaftar.

- Halaman Data Kendaraan



Halaman untuk menginput data kendaraan pelanggan yang akan

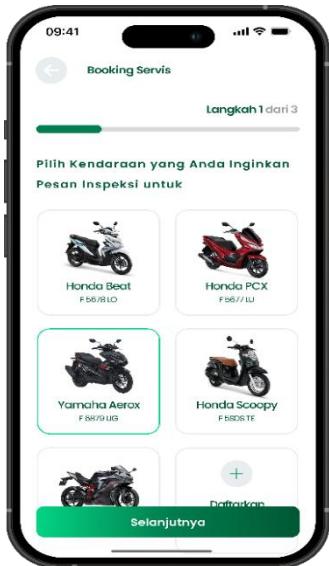
- Halaman Beranda



Merupakan **halaman utama** setelah login yang menampilkan informasi, promo, serta menu navigasi aplikasi.

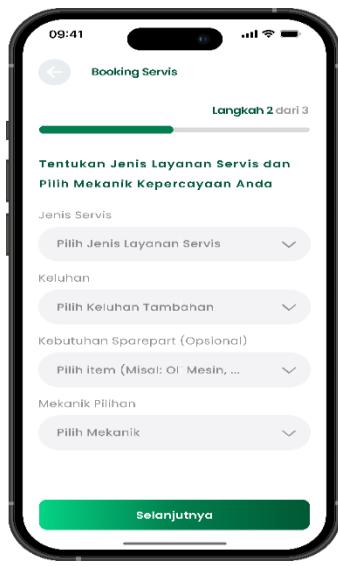
digunakan sebagai acuan dalam menampilkan produk atau layanan yang sesuai

- **Pilih Kendaraan**



Menampilkan daftar kendaraan pelanggan yang akan dipilih untuk melakukan booking servis.

- **Pilih Jenis Servis**

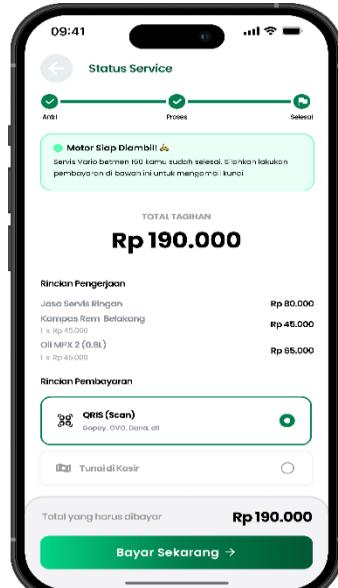


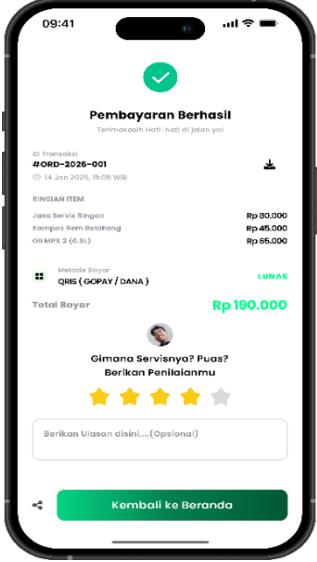
Digunakan untuk menentukan jenis layanan servis, keluhan, kebutuhan sparepart, dan mekanik.

- **Pilih Tanggal & Waktu**

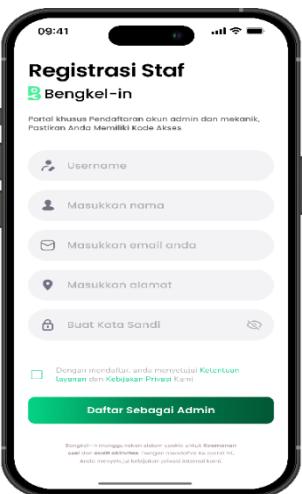
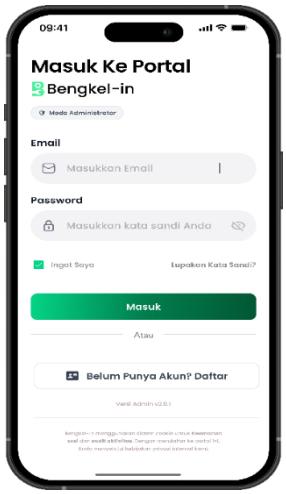


- **Status Servis**

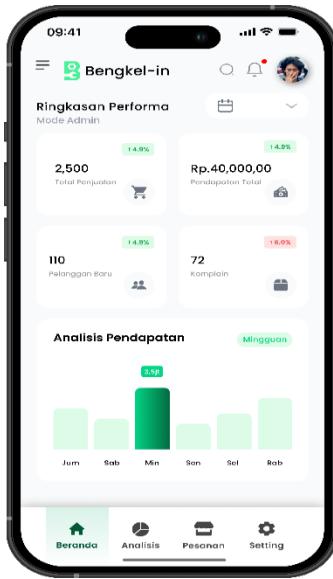


<p>Halaman untuk menentukan jadwal kedatangan servis sesuai dengan waktu yang tersedia.</p>	<p>Menampilkan status pengerajan servis, detail biaya, dan metode pembayaran pelanggan.</p>
<ul style="list-style-type: none"> <li><b>Pembayaran Berhasil</b></li> </ul>  <p>Menampilkan konfirmasi pembayaran yang telah berhasil dilakukan beserta rincian transaksi dan total biaya servis</p>	

## Admin

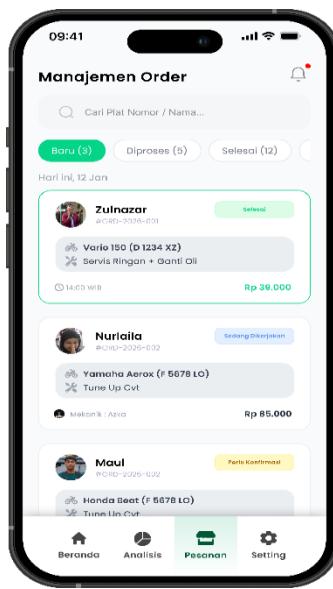
<ul style="list-style-type: none"> <li><b>Registrasi Staf</b></li> </ul>  <p>Menampilkan form pendaftaran staf baru untuk mengelola akses dan peran pengguna dalam sistem.</p>	<ul style="list-style-type: none"> <li><b>Masuk ke Portal</b></li> </ul>  <p>Menampilkan halaman login bagi admin atau staf untuk mengakses sistem bengkel.</p>
---	---

### • Pebayaran Berhasil



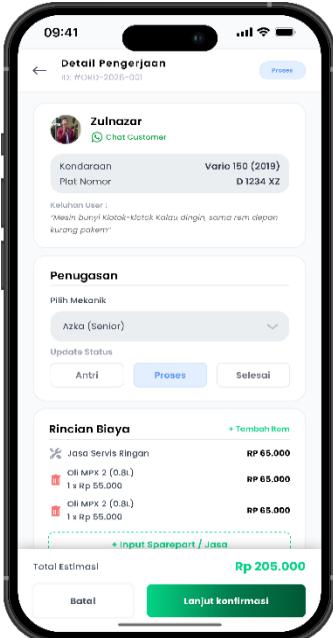
Menampilkan ringkasan performa bengkel, jumlah transaksi, pendapatan, serta analisis grafik secara real-time.

### • Manajemen Order



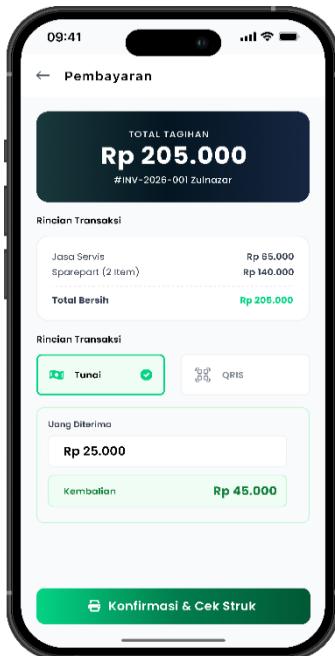
Menampilkan daftar pesanan servis pelanggan, status pengerjaan, serta informasi teknis dan biaya.

### • Detail Penggerjaan Servis



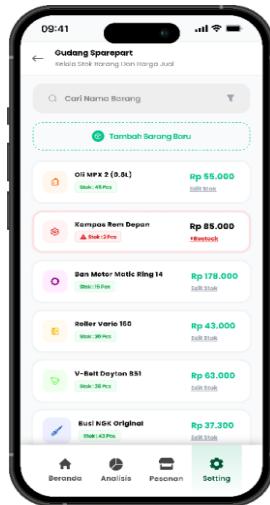
Menampilkan informasi detail pelanggan, teknisi yang bertugas, rincian layanan, dan estimasi biaya servis.

### • Pembayaran



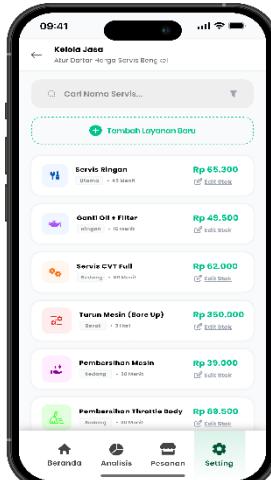
Menampilkan total tagihan servis, metode pembayaran, dan ringkasan transaksi pelanggan.

### • Gudang Bengkel



Menampilkan daftar stok barang bengkel, harga, dan ketersediaan untuk mendukung kebutuhan servis.

### • Tampilan Data Jasa



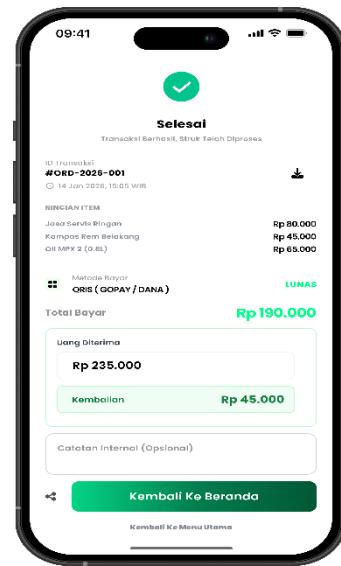
Menampilkan daftar layanan servis bengkel beserta harga, serta fitur tambah, ubah, dan hapus layanan untuk memudahkan pengelolaan jasa.

### • Laporan & Analisis



Menampilkan ringkasan performa bengkel, total pendapatan, jumlah servis, serta grafik analisis pendapatan berdasarkan periode waktu.

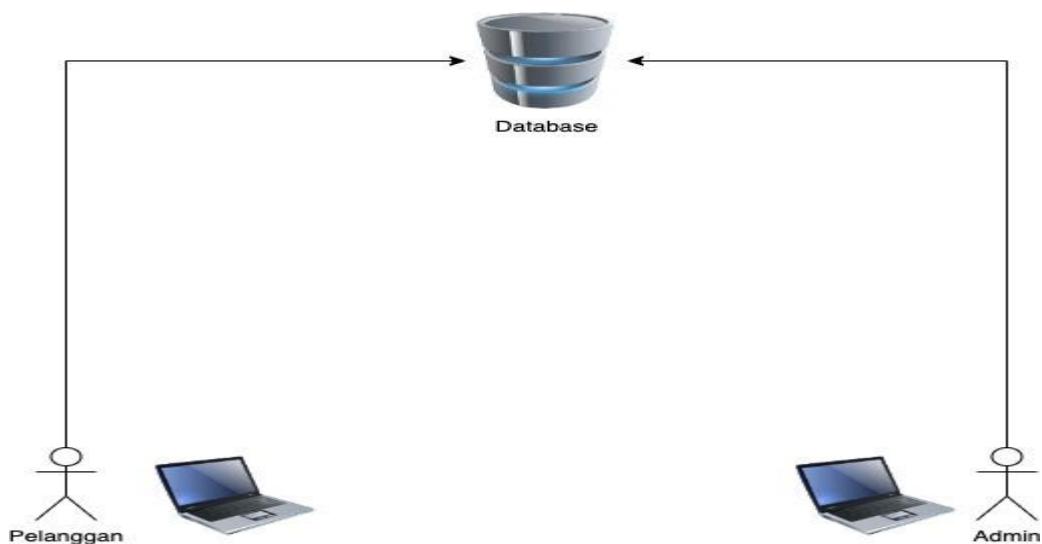
### • Transaksi Selesai



Menampilkan detail transaksi servis yang telah diselesaikan, mencakup rincian layanan, metode pembayaran, total bayar, dan kembalian.

### 2.1.3. Antarmuka Perangkat Keras

Aplikasi Bengkel-In merupakan sistem berbasis web mobile yang diakses menggunakan smartphone sebagai perangkat utama oleh User (Customer) melalui browser dengan koneksi internet. Admin Bengkel mengakses sistem menggunakan **laptop atau komputer** untuk mengelola data dan layanan bengkel. Seluruh data disimpan pada **database server** dan dikelola secara terpusat.



### 2.1.4. Antarmuka Perangkat Lunak

Aplikasi Bengkel-In diakses melalui web browser berbasis mobile tanpa perlu instalasi aplikasi tambahan. Sistem dirancang mobile-friendly sehingga dapat digunakan dengan baik pada perangkat smartphone yang terhubung ke internet.

## 2.2. Spesifikasi Kebutuhan Fungsional

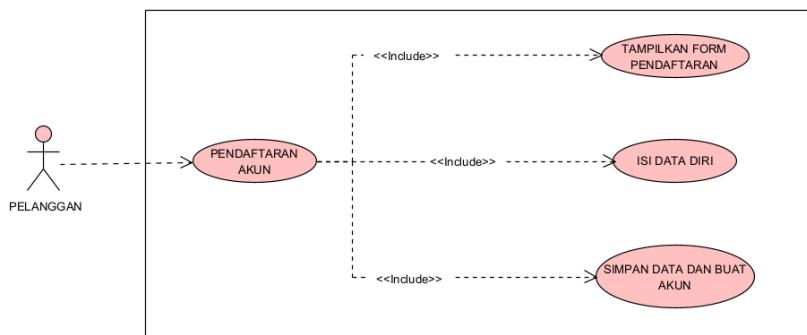
NO	DESKRIPSI	PRIORITAS
1	Sistem dapat menampilkan halaman pendaftaran pelanggan	Must
2	Sistem dapat menampilkan menu login dan logout untuk pelanggan dan admin	Must
3	Sistem dapat menampilkan halaman utama aplikasi bengkel	Must
4	Sistem dapat menampilkan daftar jasa servis	Must
5	Sistem dapat menampilkan jadwal servis yang tersedia	Must
6	Sistem dapat menerima dan menyimpan data booking servis dari pelanggan	Must
7	Sistem dapat menyimpan data motor milik pelanggan	Must
8	Sistem dapat memvalidasi data booking servis	Must

9	Sistem dapat menampilkan ringkasan booking dan estimasi biaya servis	Must
10	Sistem dapat memproses pembayaran servis melalui payment gateway	Must
11	Sistem dapat menyimpan data pembayaran servis	Must
12	Sistem dapat mengirimkan status pembayaran kepada pelanggan	Must
13	Sistem dapat menghasilkan dan mengirim resi booking servis	Must
14	Sistem dapat menyimpan data transaksi servis	Must
15	Sistem dapat menampilkan riwayat servis pelanggan	Must
16	Sistem dapat menampilkan daftar booking servis kepada admin	Must
17	Sistem dapat memungkinkan admin menentukan mekanik untuk servis	Must
18	Sistem dapat memungkinkan admin menginput detail jasa dan sparepart servis	Must
19	Sistem dapat menghitung total biaya jasa dan sparepart	Must
20	Sistem dapat memperbarui status servis (menunggu, proses, selesai)	Must
21	Sistem dapat menyimpan perubahan data servis	Must
22	Sistem dapat mengelola data jasa servis	Must
23	Sistem dapat mengelola data sparepart dan stok	Must
24	Sistem dapat menampilkan laporan transaksi servis	Must
25	Sistem dapat menghubungkan pelanggan dengan admin melalui WhatsApp	Optional

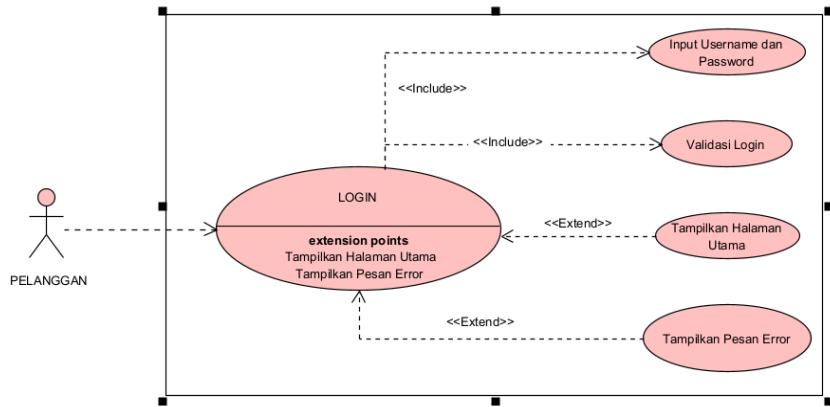
## 2.3. UseCase Diagram

### 2.3.1 Pelanggan

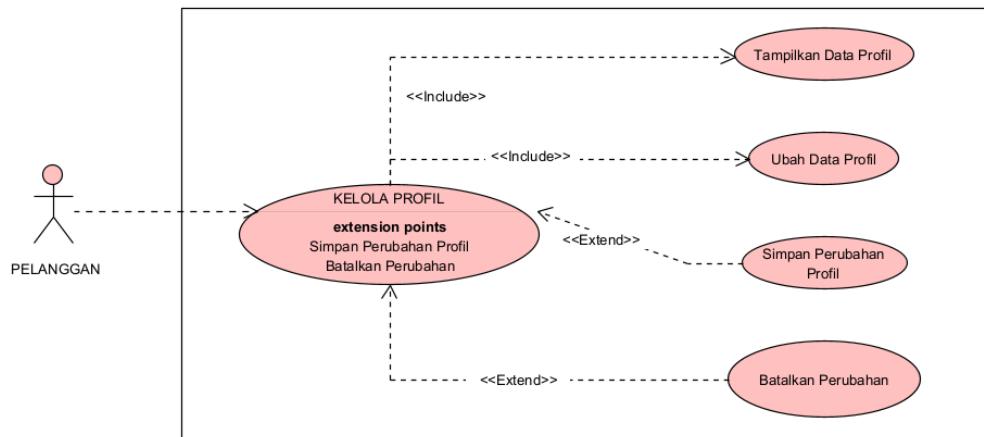
- Pelanggan Daftar



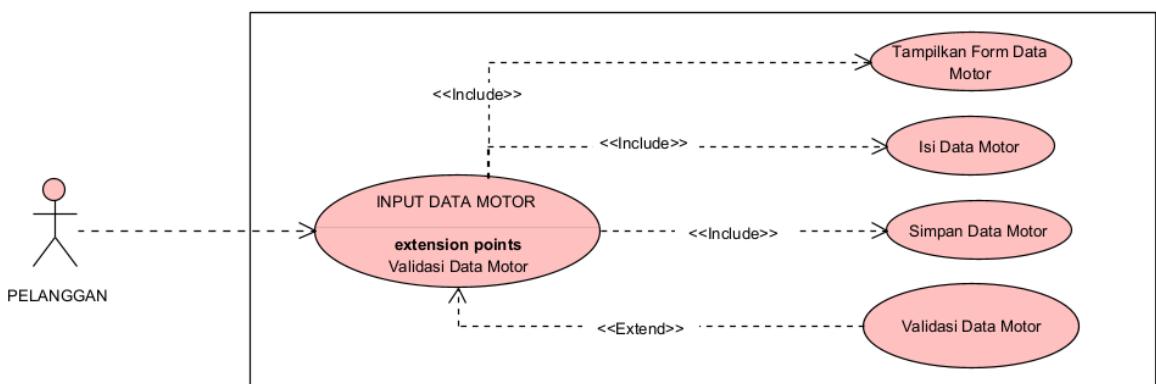
- **Pelanggan Login**



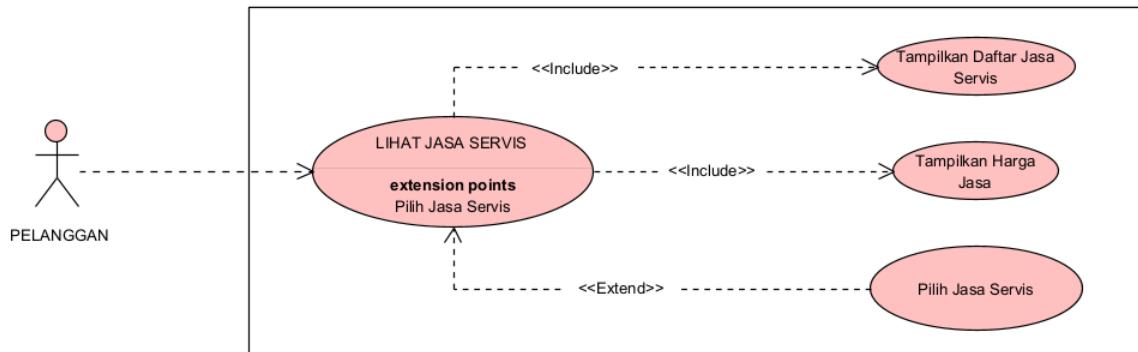
- **Pelanggan kelola Profil**



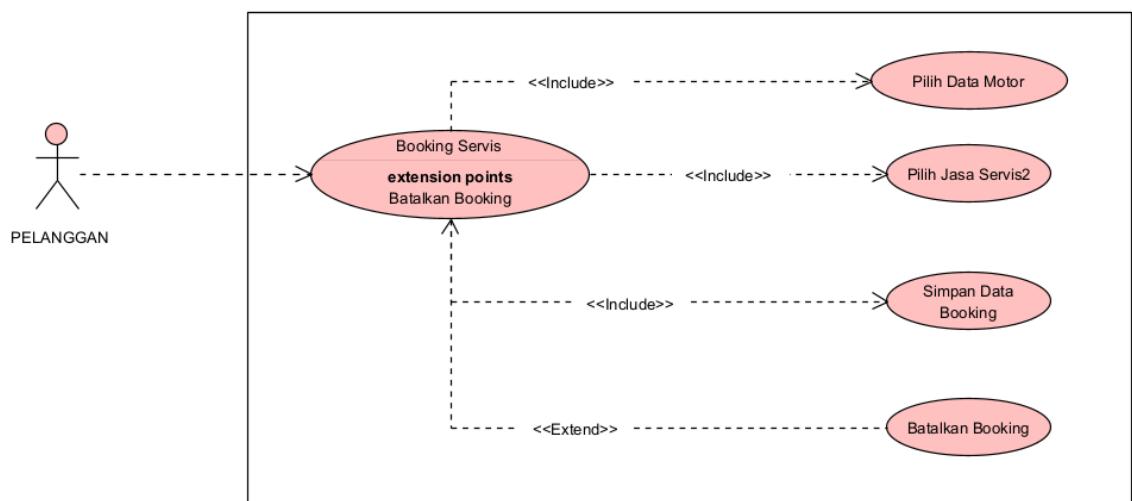
- **Pelanggan Input Data Motor**



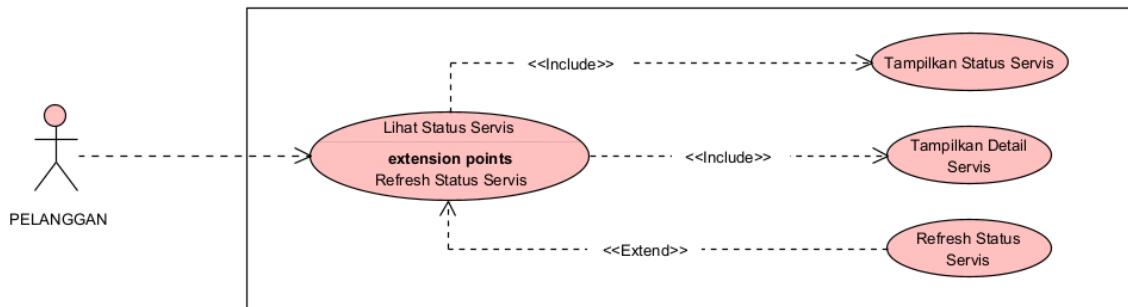
- **Pelanggan Lihat Jasa Servis**



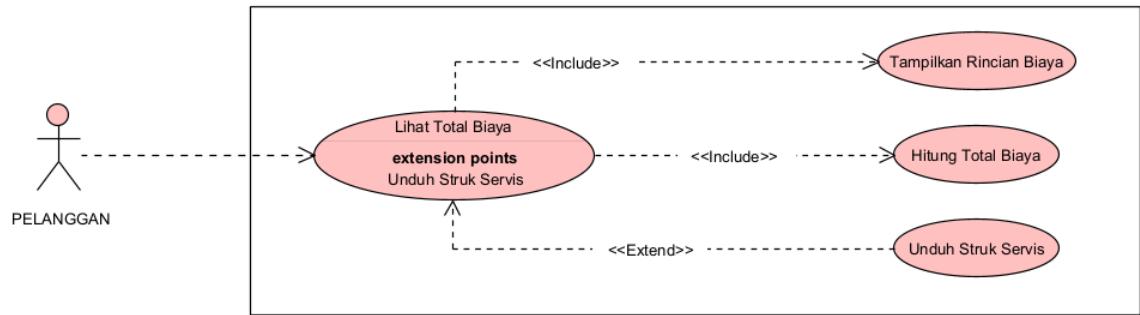
- **Pelanggan Booking Servis**



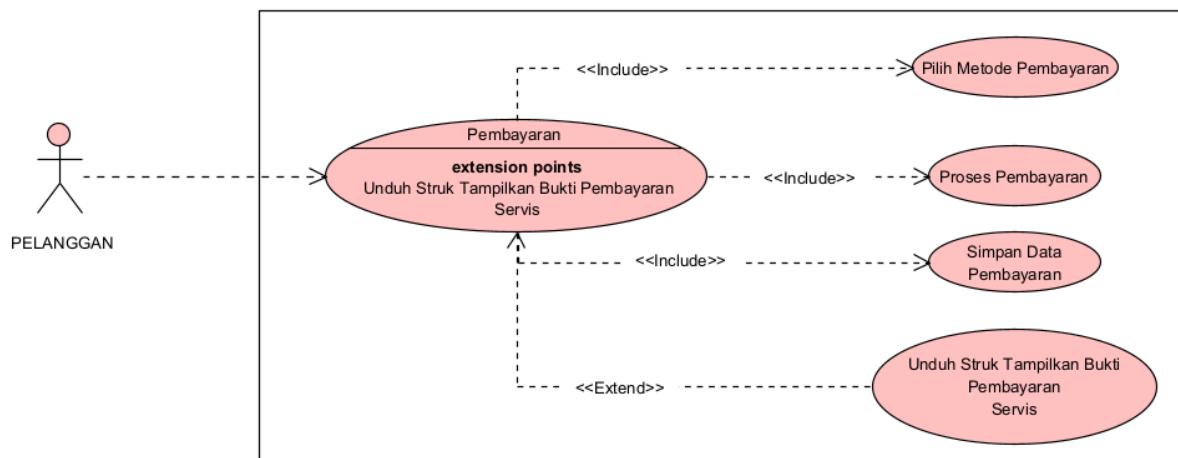
- **Pelanggan Lihat Status Servis**



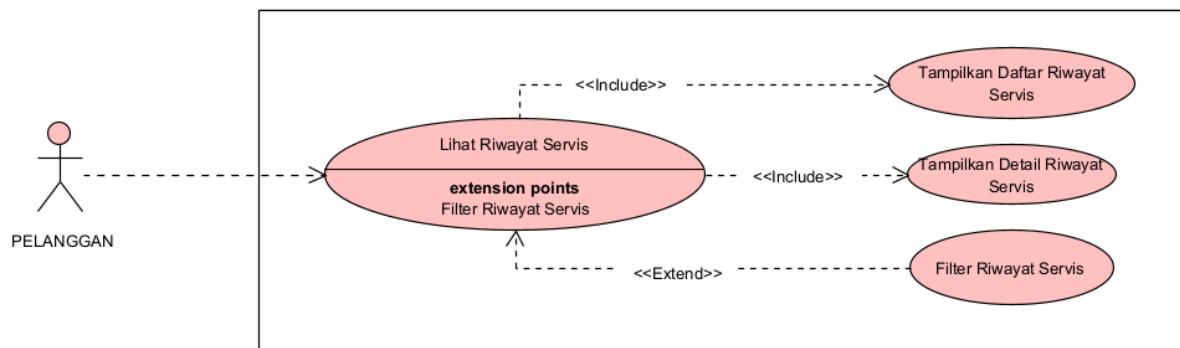
- **Pelanggan Lihat Total Biaya**



- **Pelanggan Pembayaran**

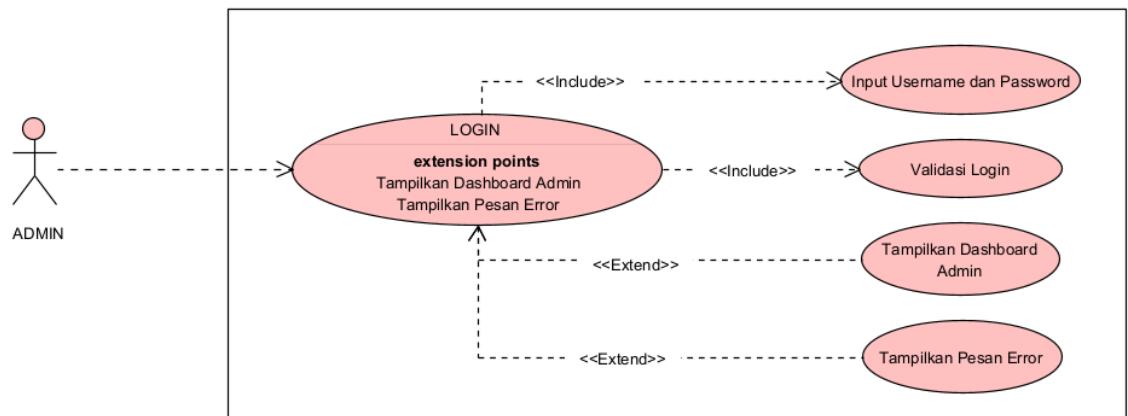


- **Pelanggan Lihat riwayat Servis**

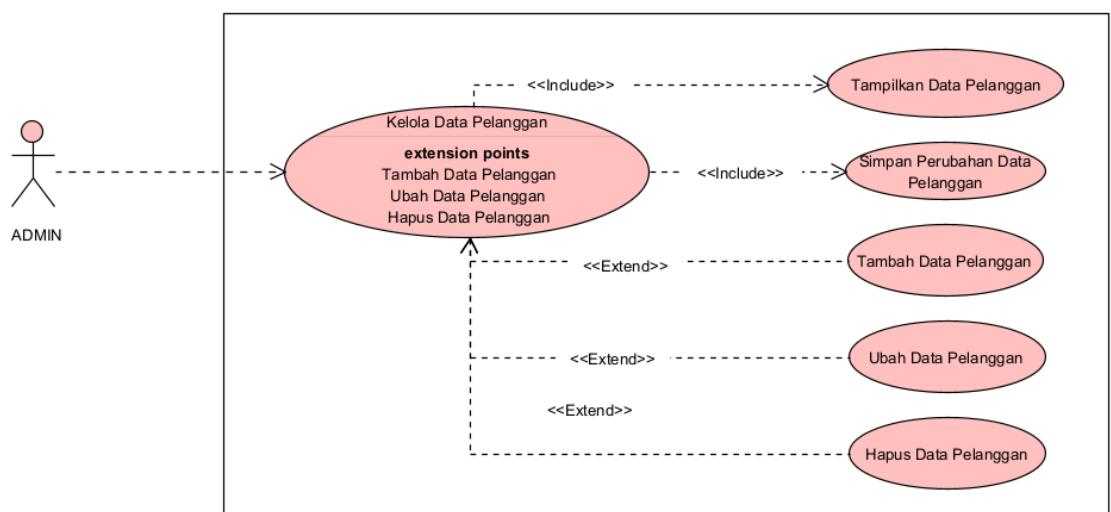


### 1.3.2 Admin

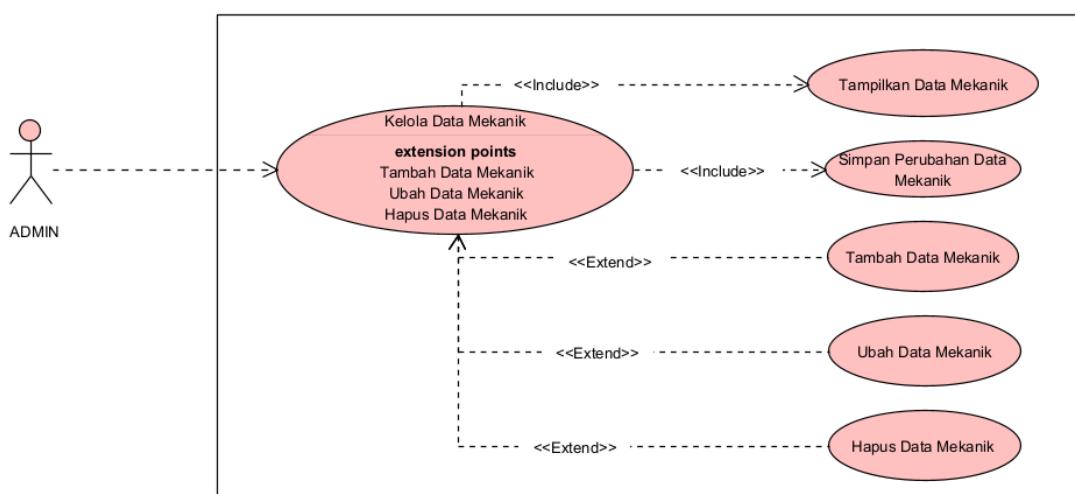
#### Admin Login



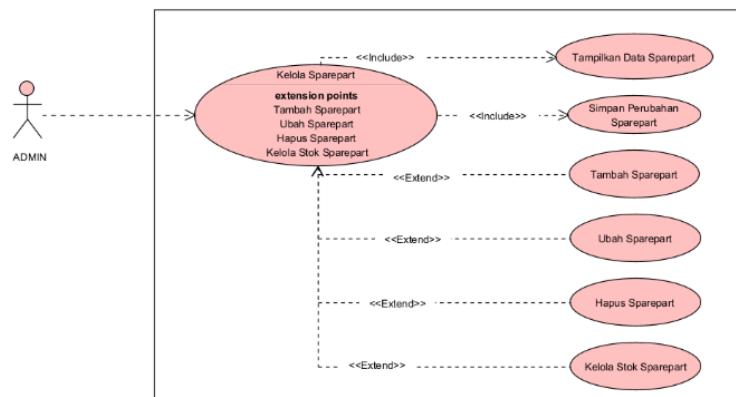
### • Admin Kelola Data Pelanggan



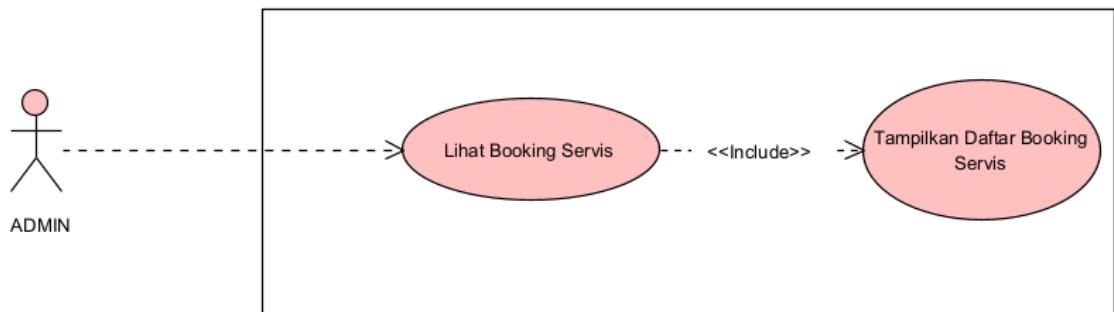
### • Admin Kelola Data Mekanik



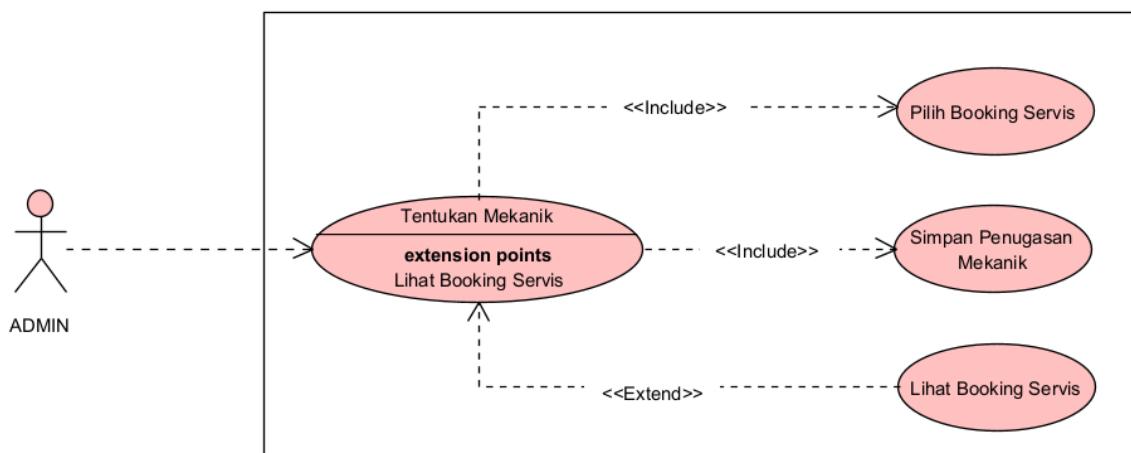
- **Admin Kelola Data Sparepart**



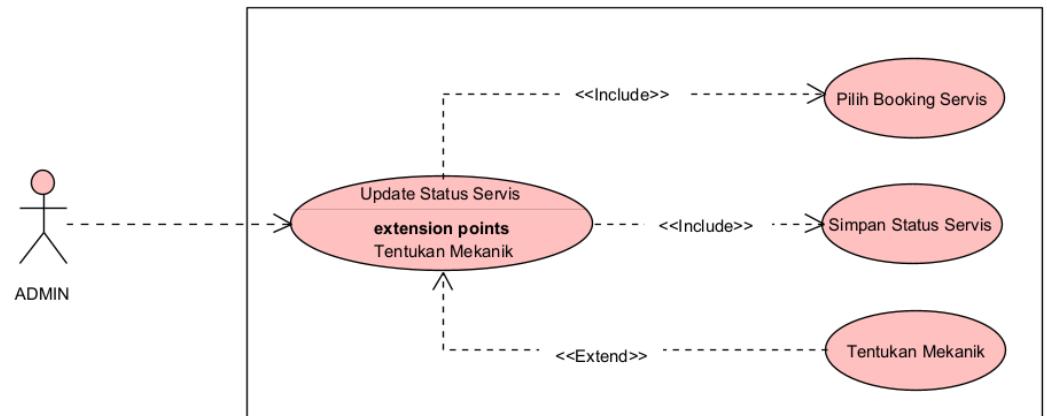
- **Lihat Booking Servis**



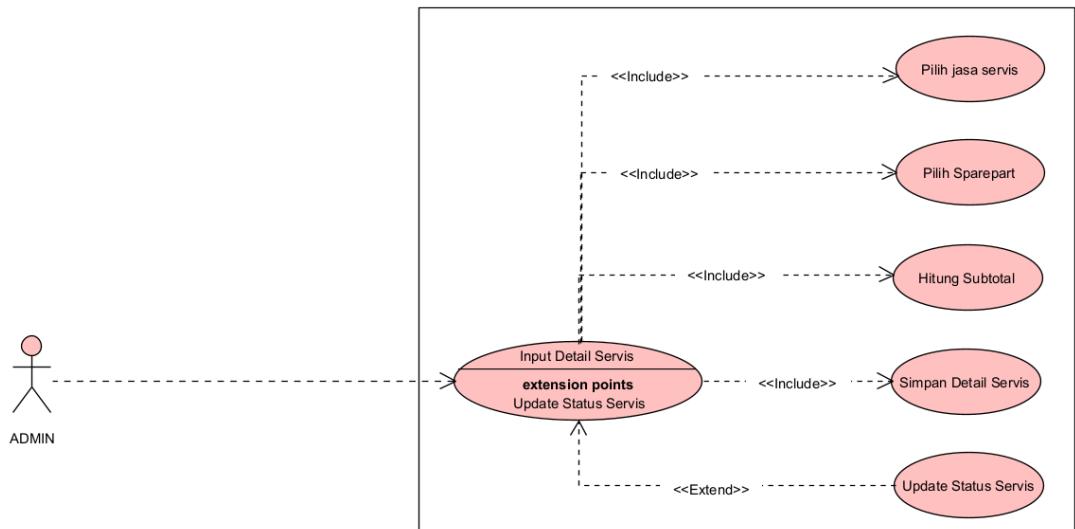
- **Tentukan Mekanik**



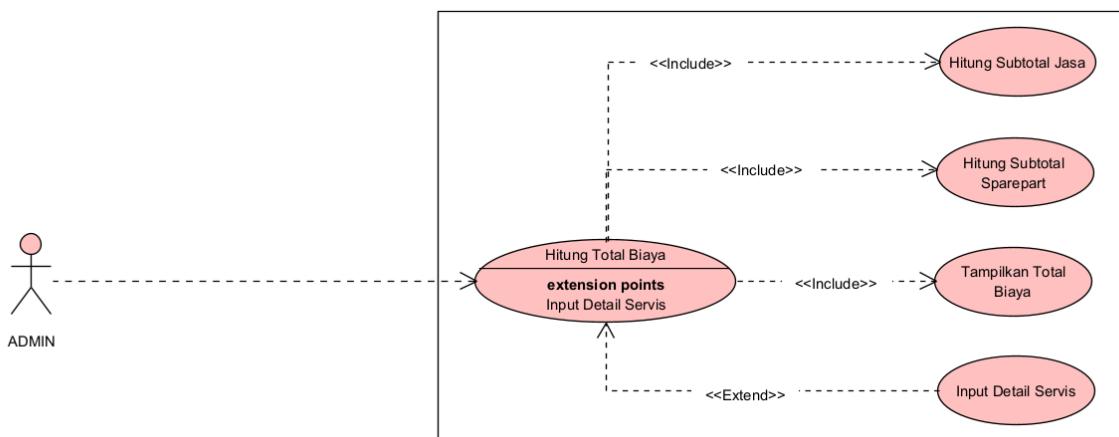
## Update Status Servis



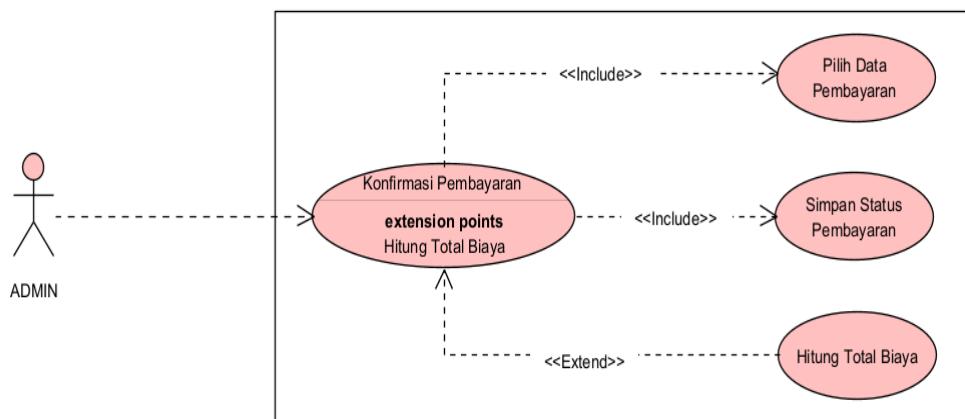
### • Input Detail Servis



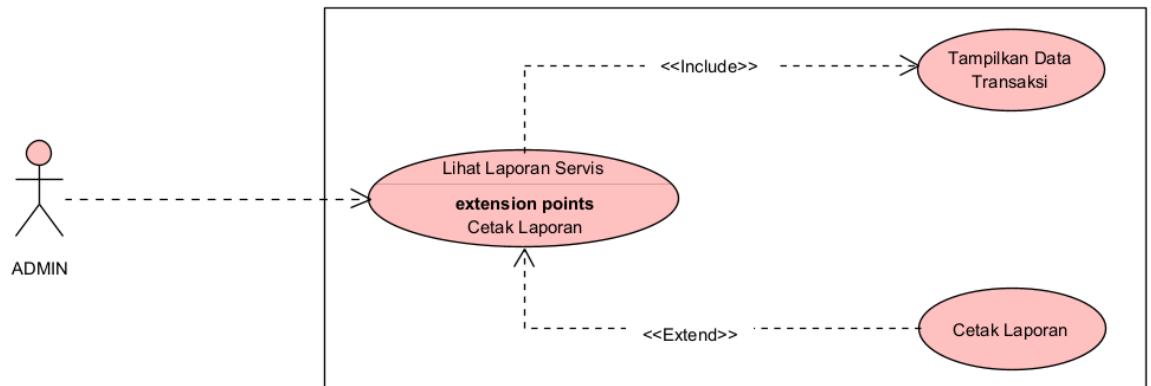
### • Hitung Total Biaya



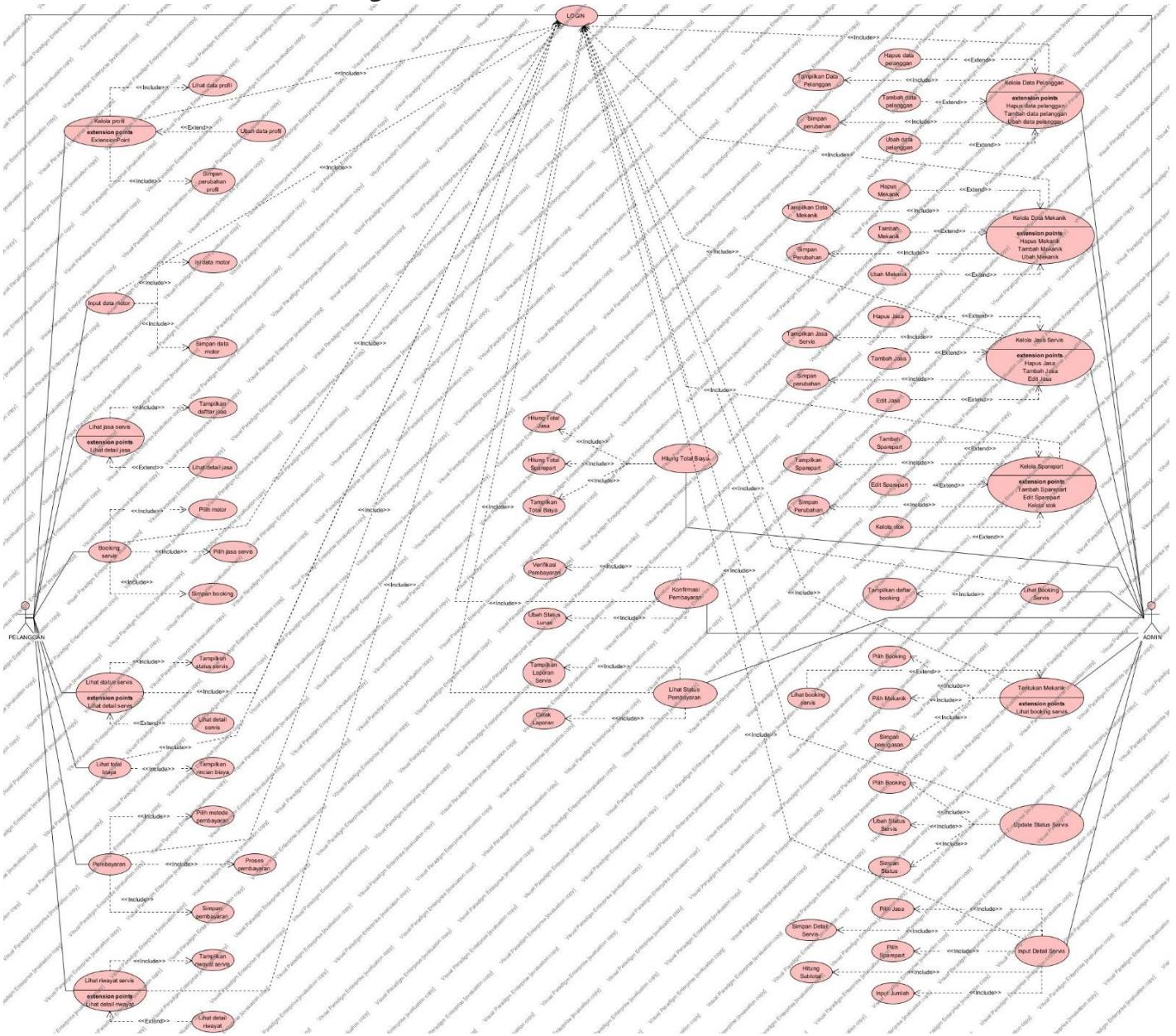
- **Konfirmasi Pembayaran**



- **Lihat Laporan Servis**



### 1.3.3 UseCase Diagram



## 2.4 Spesifikasi Kebutuhan Non Fungsional

#### 2.4.1 Spesifikasi User Interface

NO	DESKRIPSI	PRIORITAS	UKURAN
1	Sistem memiliki tampilan antarmuka yang sederhana dan mudah dipahami oleh pelanggan dan admin.	Must	Pengguna dapat memahami alur booking servis tanpa panduan khusus.
2	Sistem menampilkan menu booking servis, pembayaran, dan riwayat servis dengan navigasi yang jelas.	Must	Semua menu utama dapat diakses maksimal dalam 2–3 klik.

3	Sistem memiliki tampilan yang responsif pada berbagai ukuran layar.	Must	Tampilan menyesuaikan layar smartphone, tablet, dan desktop.
---	---	------	--

#### 2.4.2 Spesifikasi Kinerja

NO	DESKRIPSI	PRIORITAS	UKURAN
1	Sistem dapat menampilkan halaman booking servis dengan cepat.	Must	Waktu respon halaman tidak lebih dari 3 detik.
2	Sistem dapat memproses data booking dan pembayaran dengan baik.	Must	Data booking dan pembayaran tersimpan tanpa error.
3	Sistem dapat menghitung estimasi biaya servis secara otomatis.	Must	Perhitungan biaya jasa dan sparepart dilakukan secara akurat.
4	Sistem dapat dioperasikan sesuai fungsi yang dirancang.	Must	Seluruh fitur booking, pembayaran, dan laporan berjalan normal.

#### 2.4.3 Ketersediaan dan keandalan

NO	DESKRIPSI	PRIORITAS	UKURAN
1	Sistem dapat diakses menggunakan perangkat PC dan smartphone.	Must	Sistem berjalan pada browser desktop dan mobile.
2	Sistem dapat diakses melalui berbagai browser.	Must	Mendukung Google Chrome, Mozilla Firefox, Safari, dan Edge.
3	Sistem tersedia selama 24 jam.	Must	Waktu aktif sistem minimal 24 jam per hari.
4	Sistem memiliki tingkat kesalahan yang rendah.	Optional	Jumlah error sistem sangat minim dan tidak mengganggu proses servis.

#### 2.4.4 Spesifikasi Keamanan

NO	DESKRIPSI	PRIORITAS	UKURAN
1	Sistem memiliki mekanisme login untuk pelanggan dan admin.	Must	Akses sistem hanya untuk pengguna terdaftar.
2	Sistem menyimpan data akun pengguna dengan aman.	Must	Password disimpan dalam bentuk terenkripsi.
3	Sistem menjaga keamanan data booking dan pembayaran.	Must	Data transaksi tidak dapat diubah tanpa otorisasi.
4	Sistem membatasi akses fitur sesuai peran pengguna.	Must	Admin dan pelanggan memiliki hak akses berbeda.

## 2.5 Karakteristik Pengguna

Website bengkel motor ini dirancang untuk digunakan oleh beberapa jenis pengguna dengan karakteristik sebagai berikut:

### 1. Admin Bengkel

Admin merupakan pengguna yang memiliki hak akses penuh terhadap sistem.

Admin bertugas untuk:

- Mengelola data master seperti data jasa servis, sparepart, dan mekanik.
- Mengelola data pelanggan.
- Memproses dan memverifikasi booking servis dari pelanggan.
- Mengelola status servis dan pembayaran.
- Melihat serta mencetak laporan servis dan transaksi.

### 2. Pelanggan

Pelanggan merupakan pengguna yang menggunakan sistem untuk melakukan booking servis. Pelanggan dapat:

- Melakukan pendaftaran akun dan login ke sistem.
- Mengelola data profil pribadi.
- Melakukan booking servis motor secara online.
- Melakukan pembayaran servis.
- Melihat status booking dan riwayat servis.

## 2.6 Batasan-batasan Sistem

Adapun batasan-batasan dalam pengembangan website bengkel-in ini adalah sebagai berikut:

- Website hanya dapat diakses menggunakan perangkat yang terhubung dengan koneksi internet.
- Pelanggan wajib melakukan pendaftaran akun terlebih dahulu sebelum dapat login dan menggunakan fitur booking servis.
- Password akun harus terdiri dari kombinasi huruf dan angka.
- Jika pengguna salah memasukkan password sebanyak 3 kali, maka akun akan terkunci dan hanya dapat dibuka kembali oleh admin.
- Sistem pembayaran dilakukan sesuai dengan metode pembayaran yang telah disediakan oleh sistem.
- Website hanya berfungsi untuk pengelolaan data servis dan booking, bukan untuk penjualan produk secara langsung.

## 3. Catatan Khusus

Dokumen Software Requirements Specification (SRS) pada website bengkel motor ini wajib memuat beberapa diagram sebagai berikut:

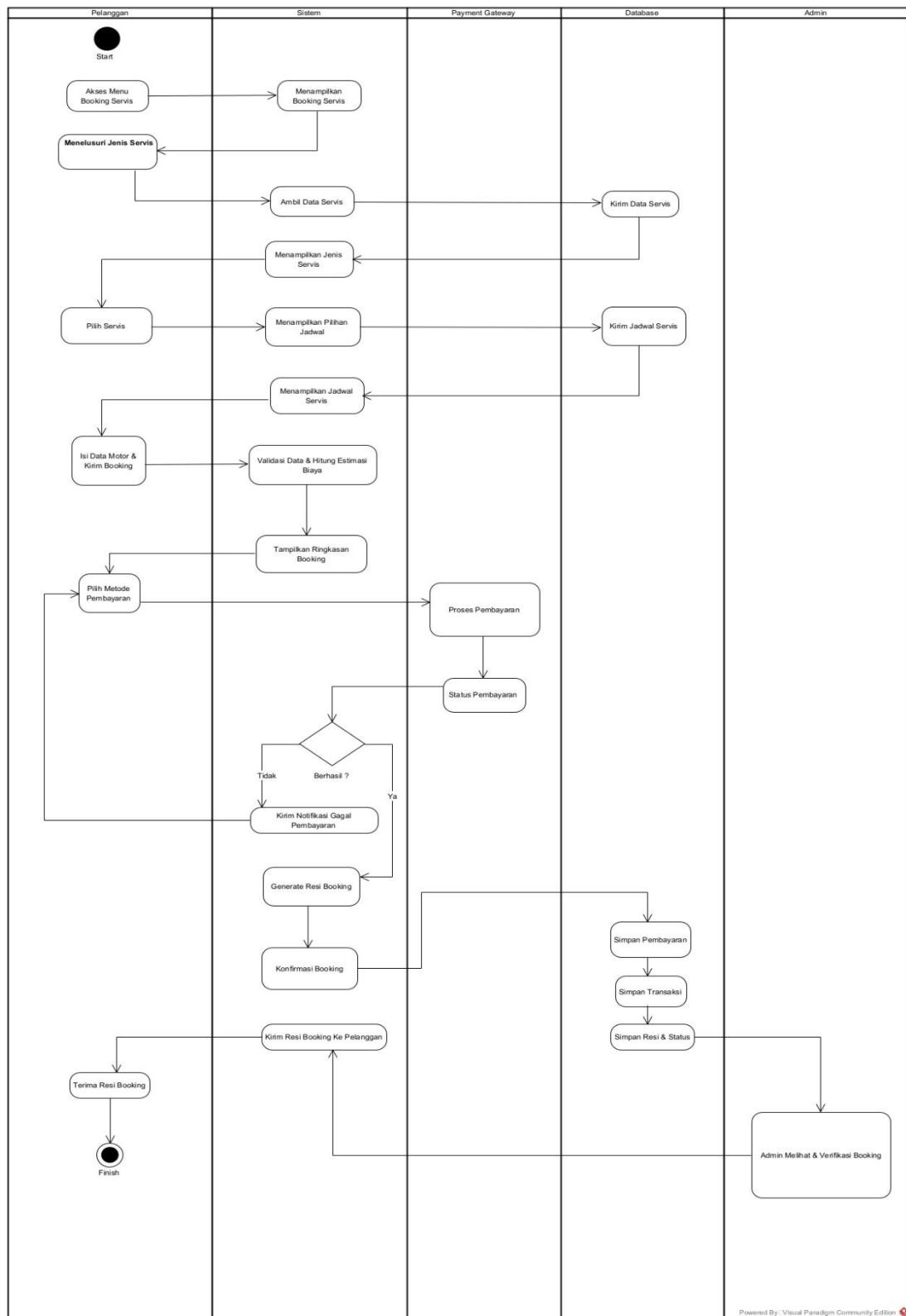
### **3.1 UML (Unified Modeling Language)**

Diagram UML digunakan untuk menggambarkan alur dan struktur sistem secara visual, yang terdiri dari:

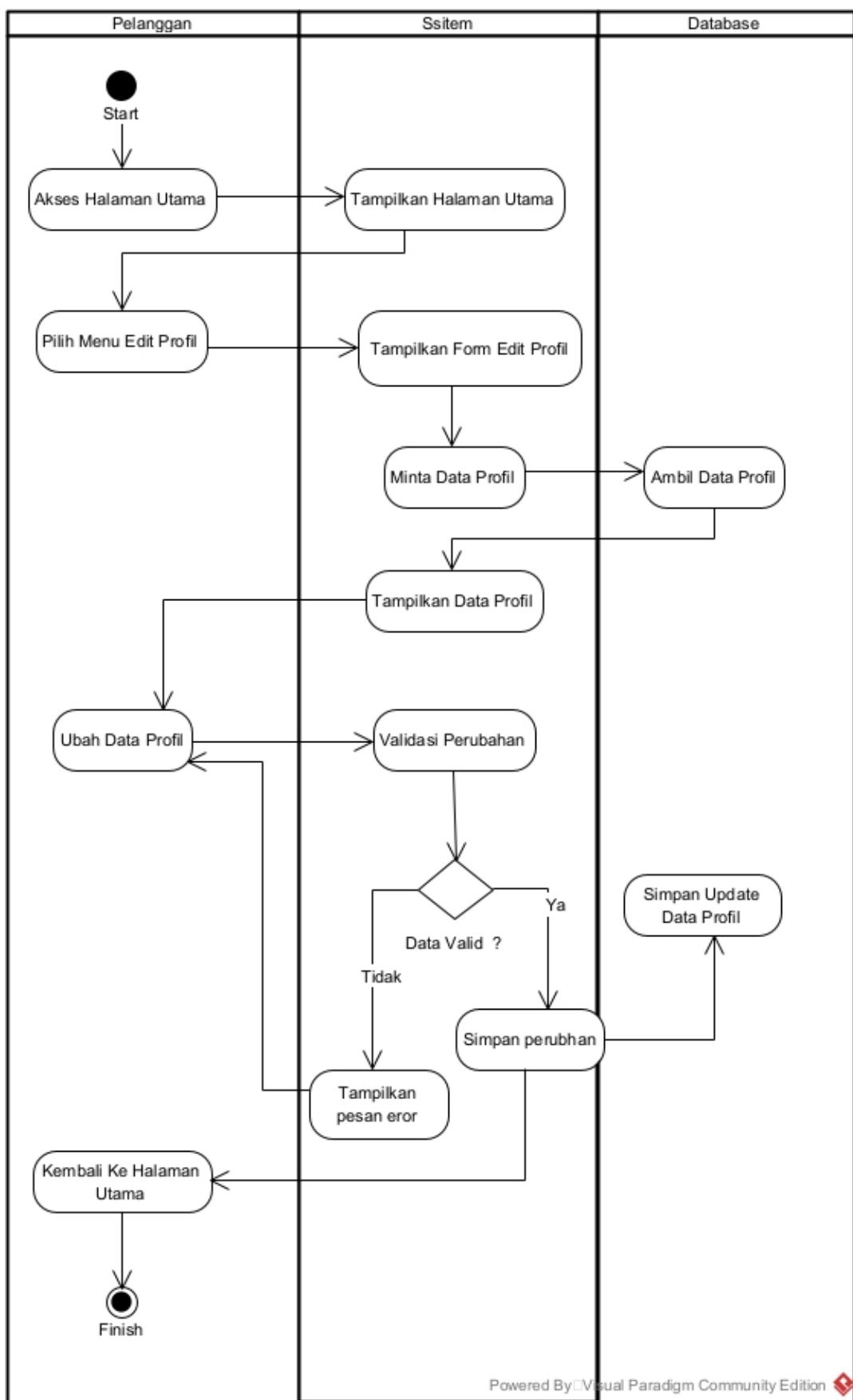
- Use Case Diagram, untuk menggambarkan interaksi antara aktor (Admin dan Pelanggan) dengan sistem.
- Activity Diagram, untuk menggambarkan alur proses bisnis seperti booking servis, pembayaran, dan pengelolaan data.
- Sequence Diagram, untuk menggambarkan urutan interaksi antara pengguna, sistem, dan database.
- Class Diagram, untuk menggambarkan struktur kelas, atribut, metode, serta relasi antar kelas.

### 3.1.1 Activity Diagram

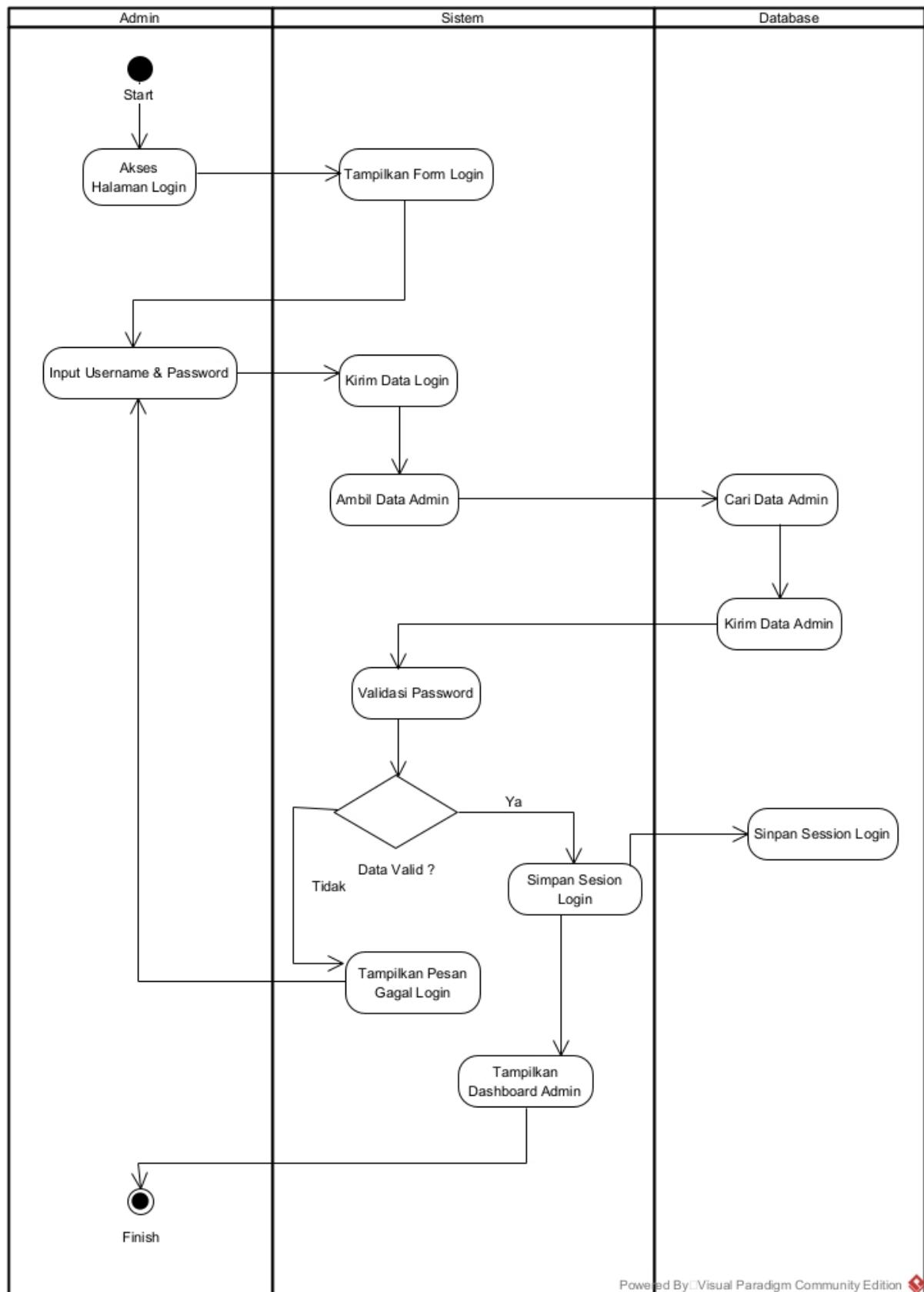
- Pelanggan Booking



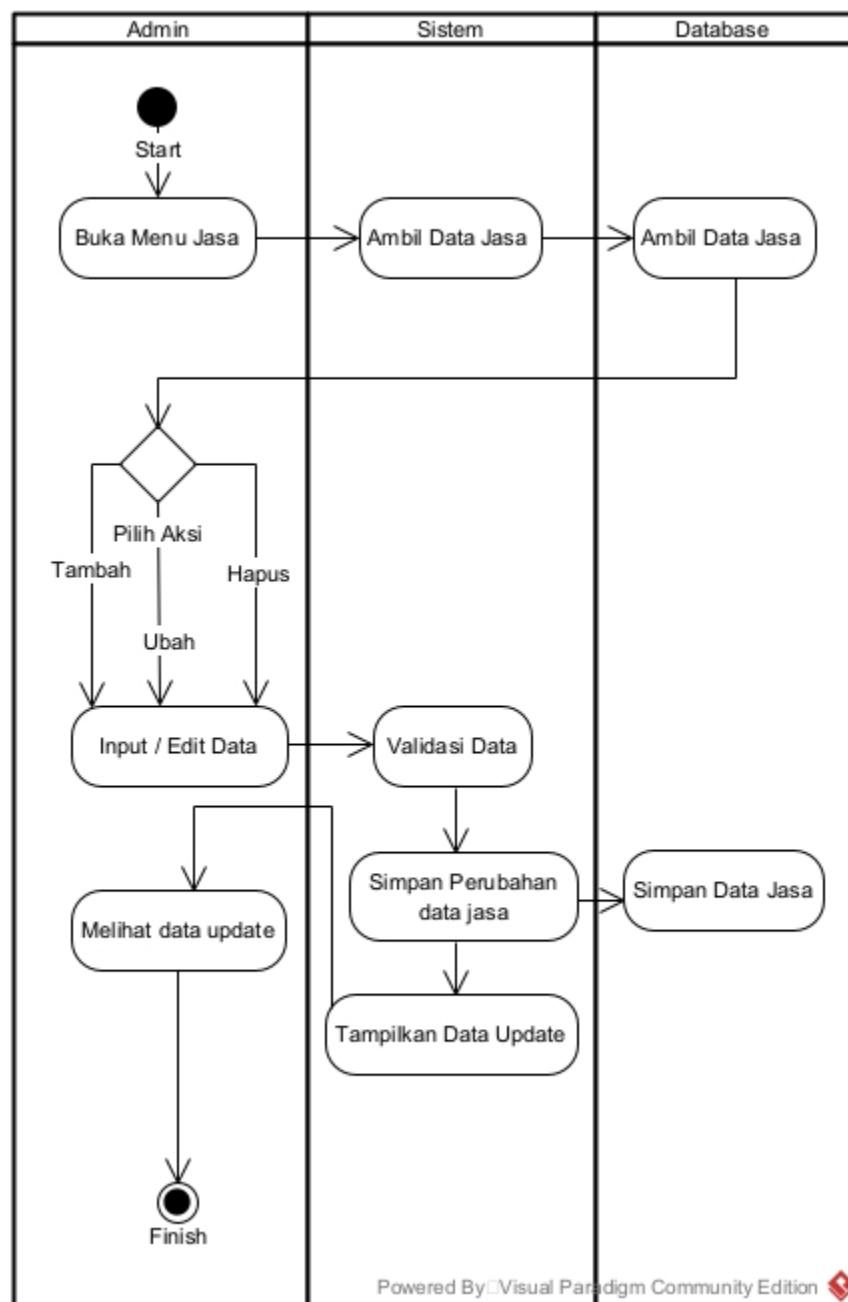
- Pelanggan Edit Profil



- Admin Login

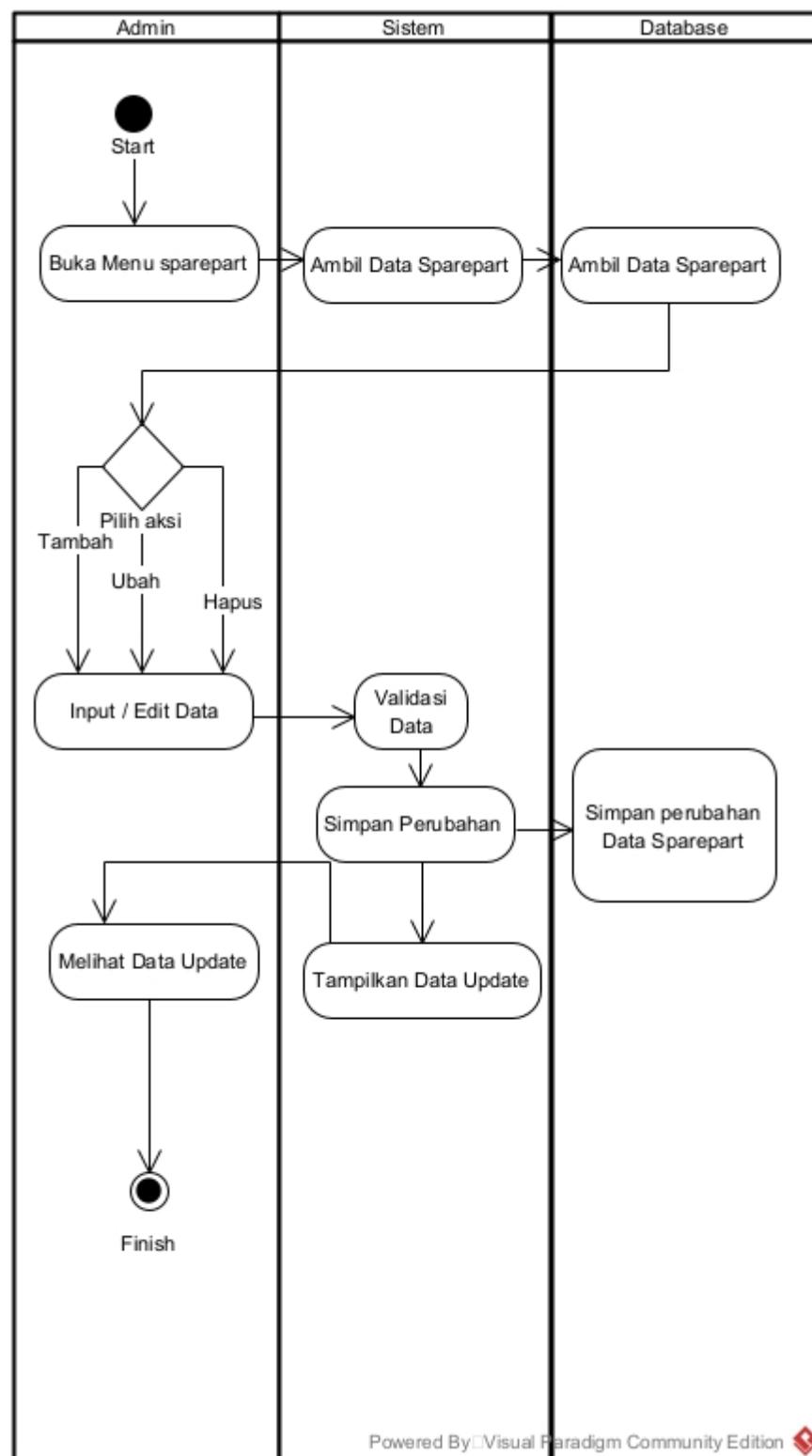


- Admin Kelola data jasa servis

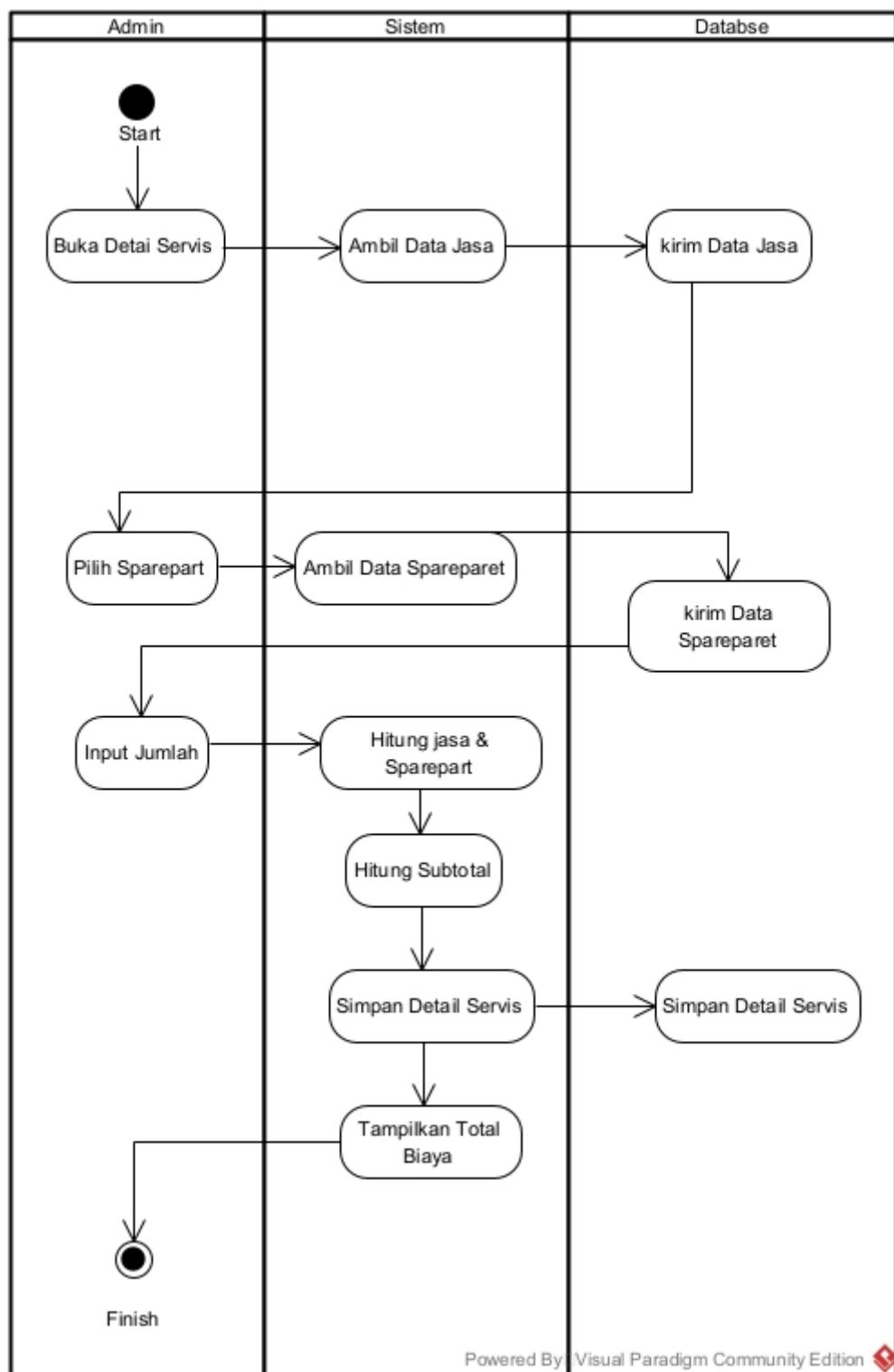


Powered By: Visual Paradigm Community Edition

- Admin Kelola Data Sparepart

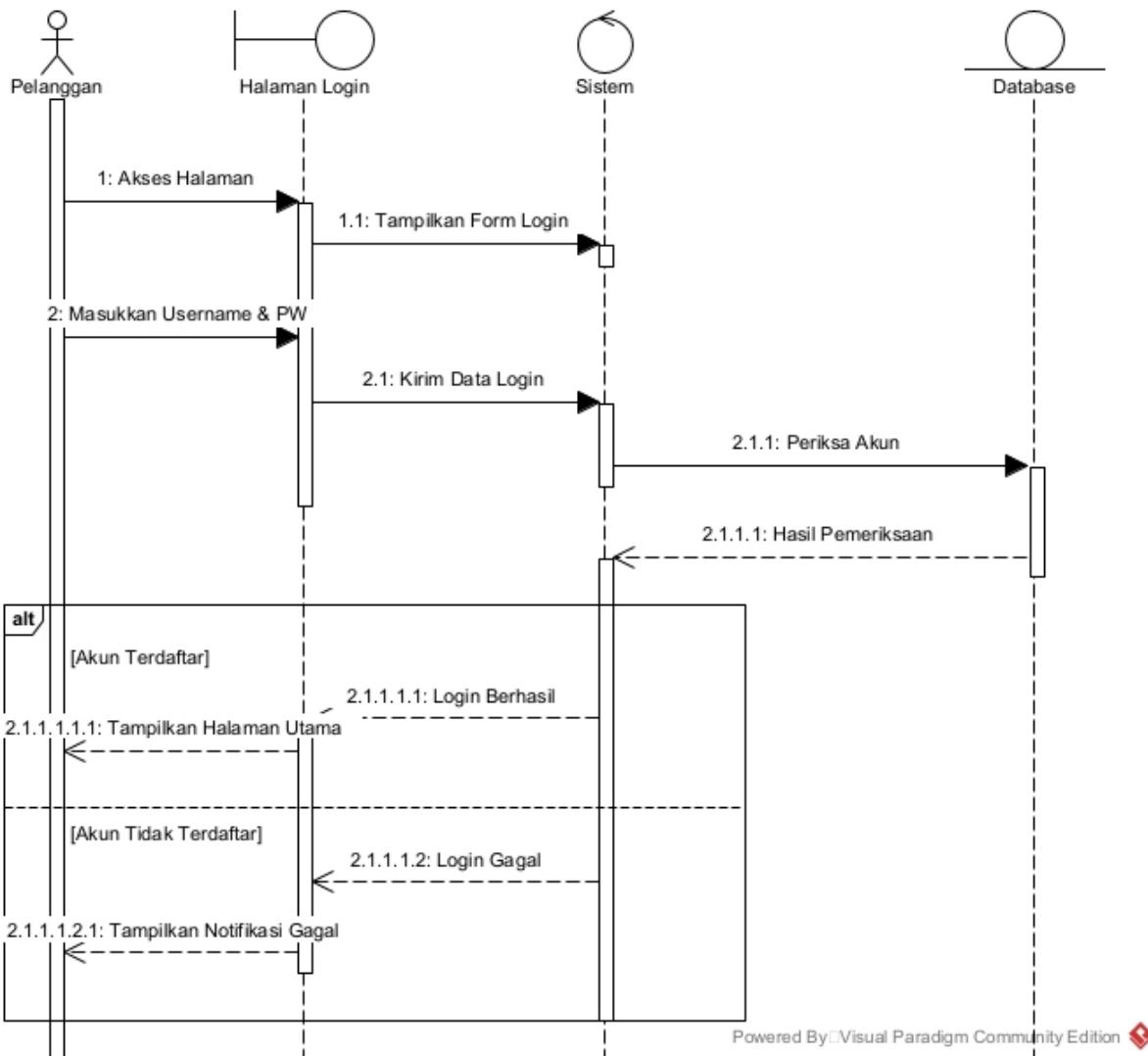


- Admin Input Detail & Hitung Biaya Servis

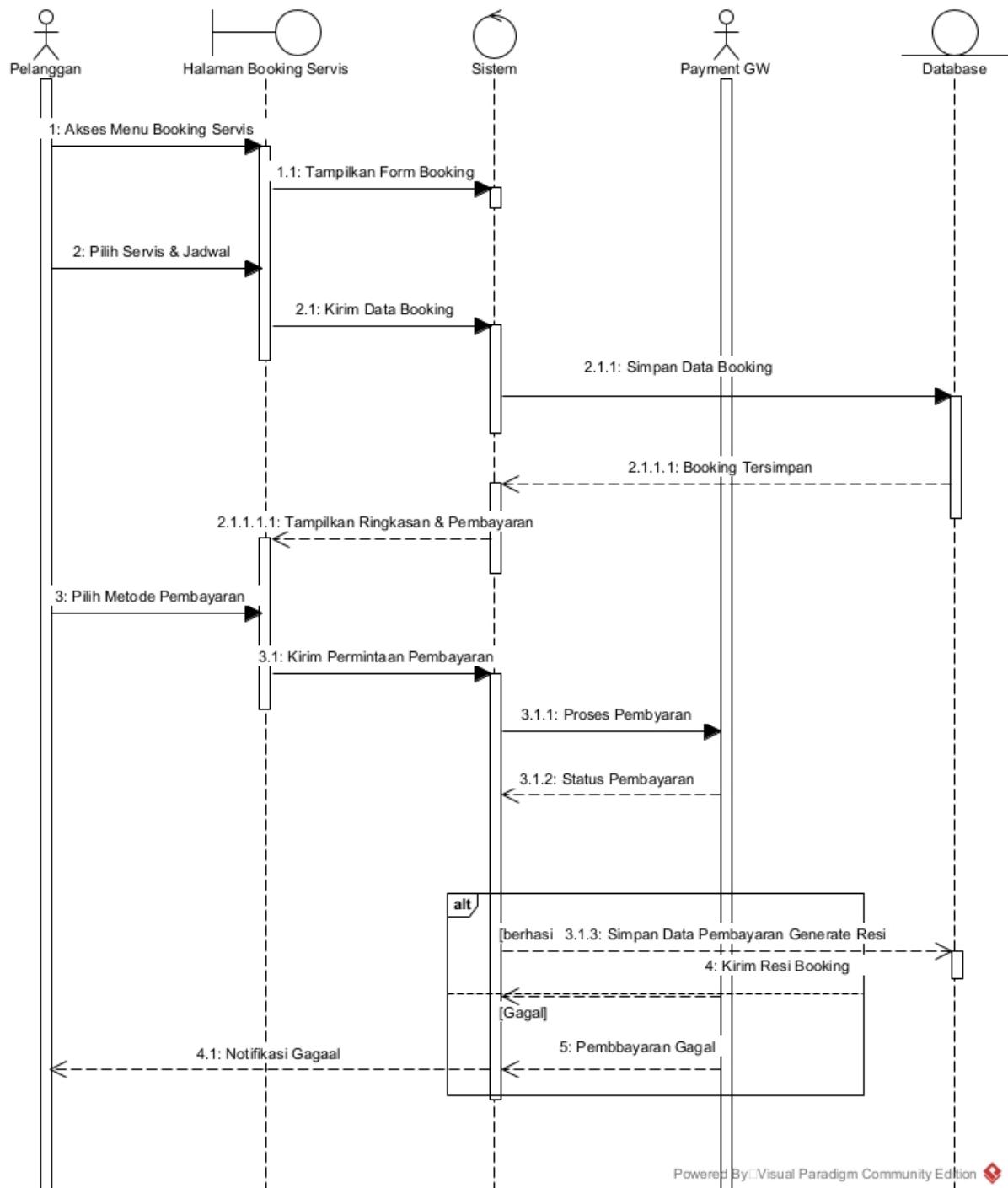


### 3.1.2 Sequence Diagram

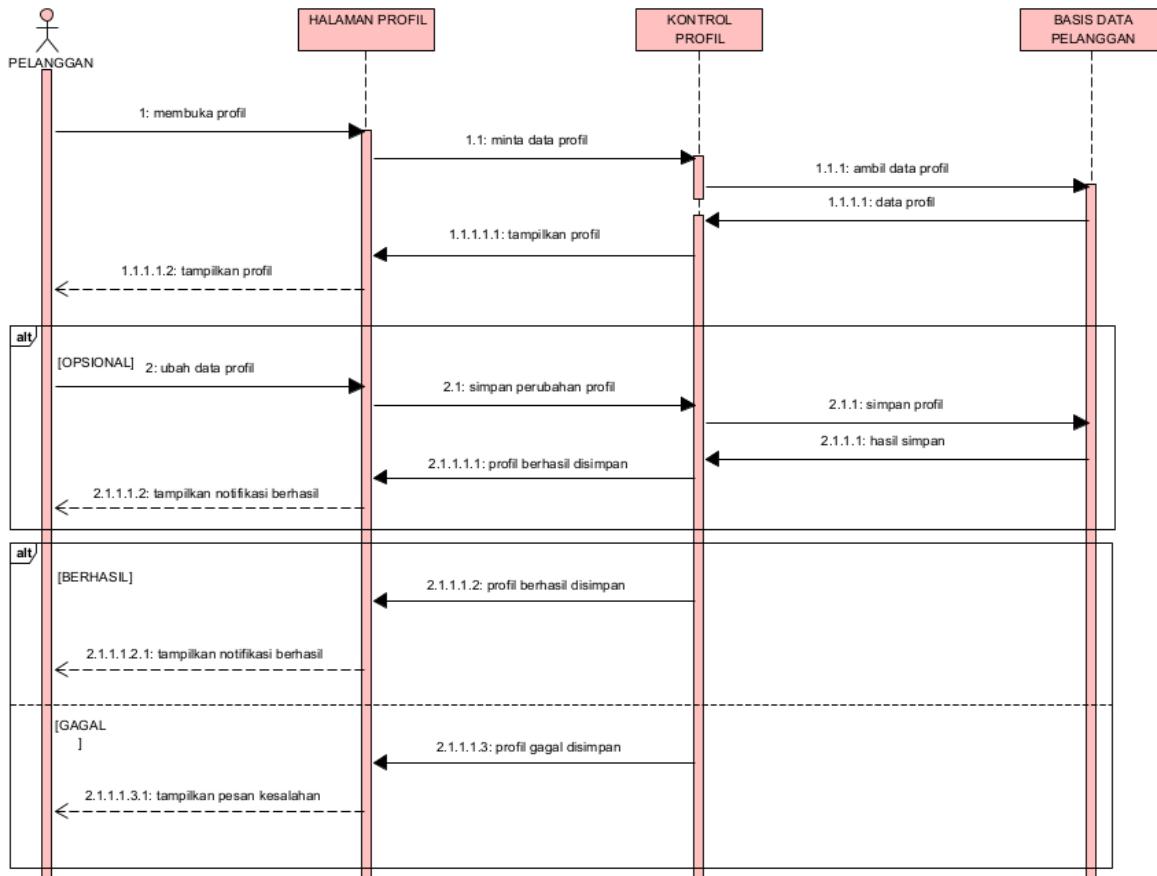
- Pelanggan Login



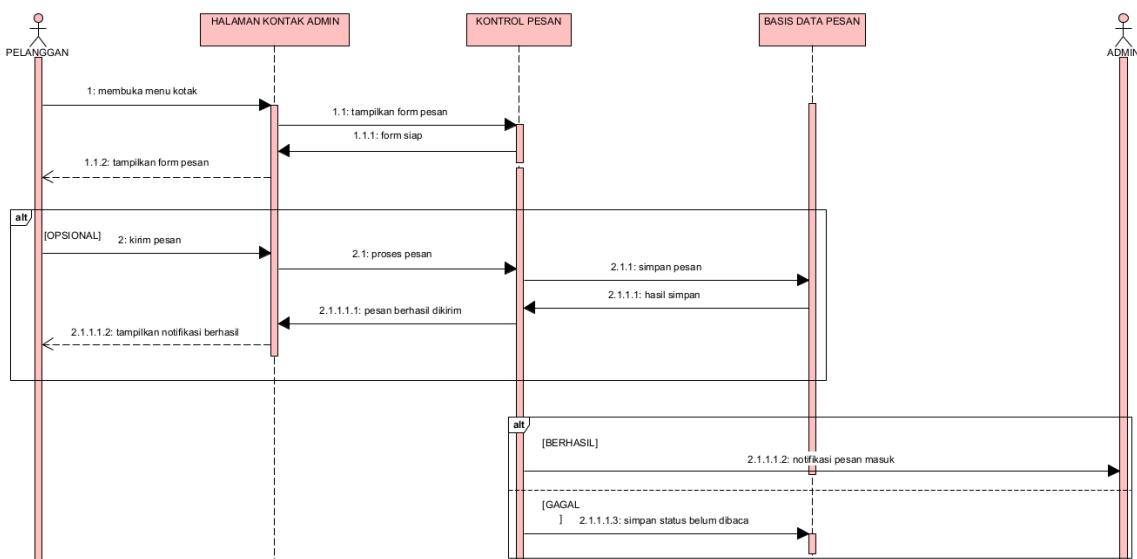
- Pelanggan Booking Servis



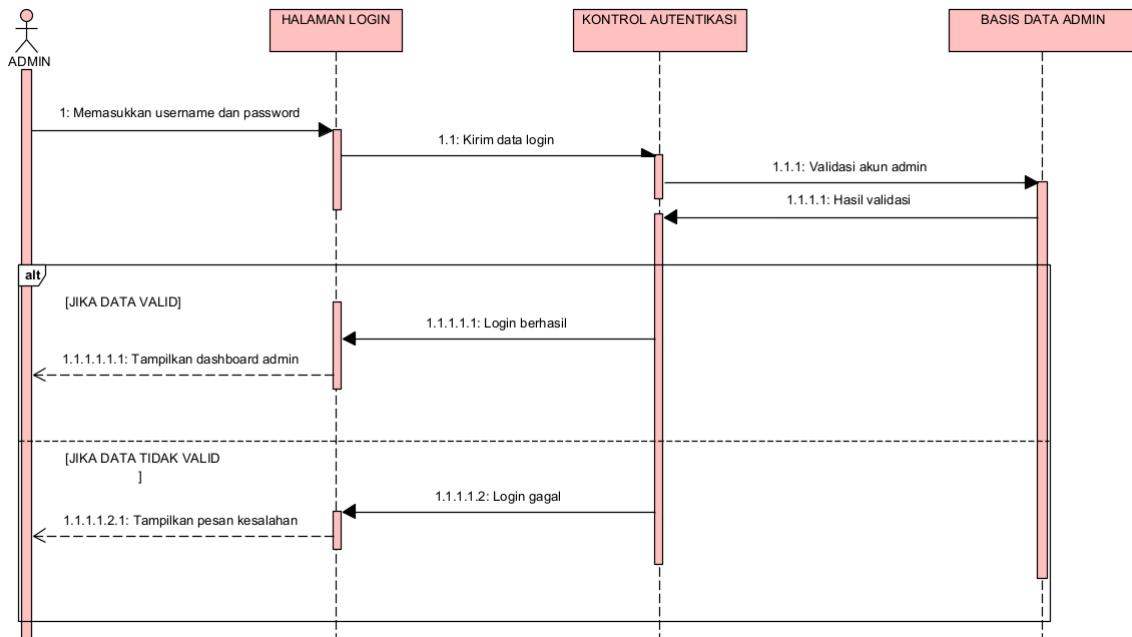
- Pelanggan Edit Profil



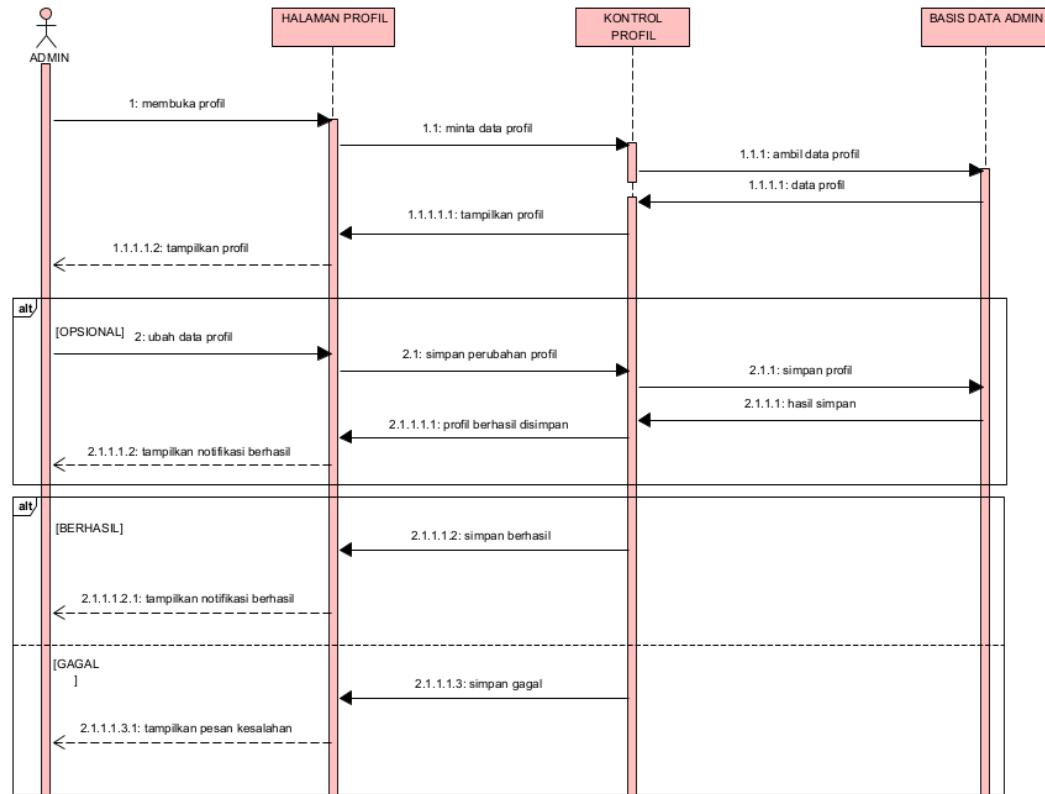
- Pelanggan Menghubungi Admin



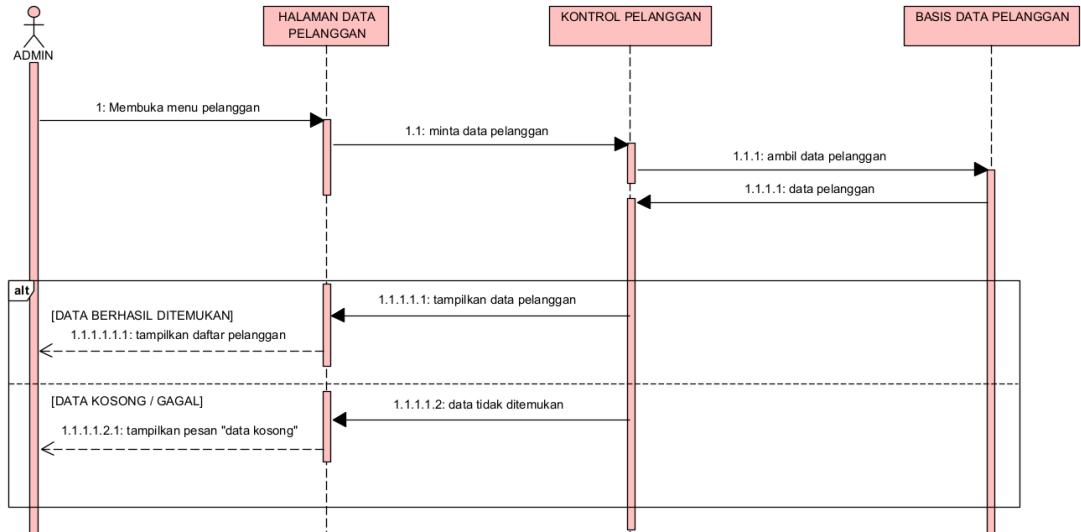
- **Admin Login**



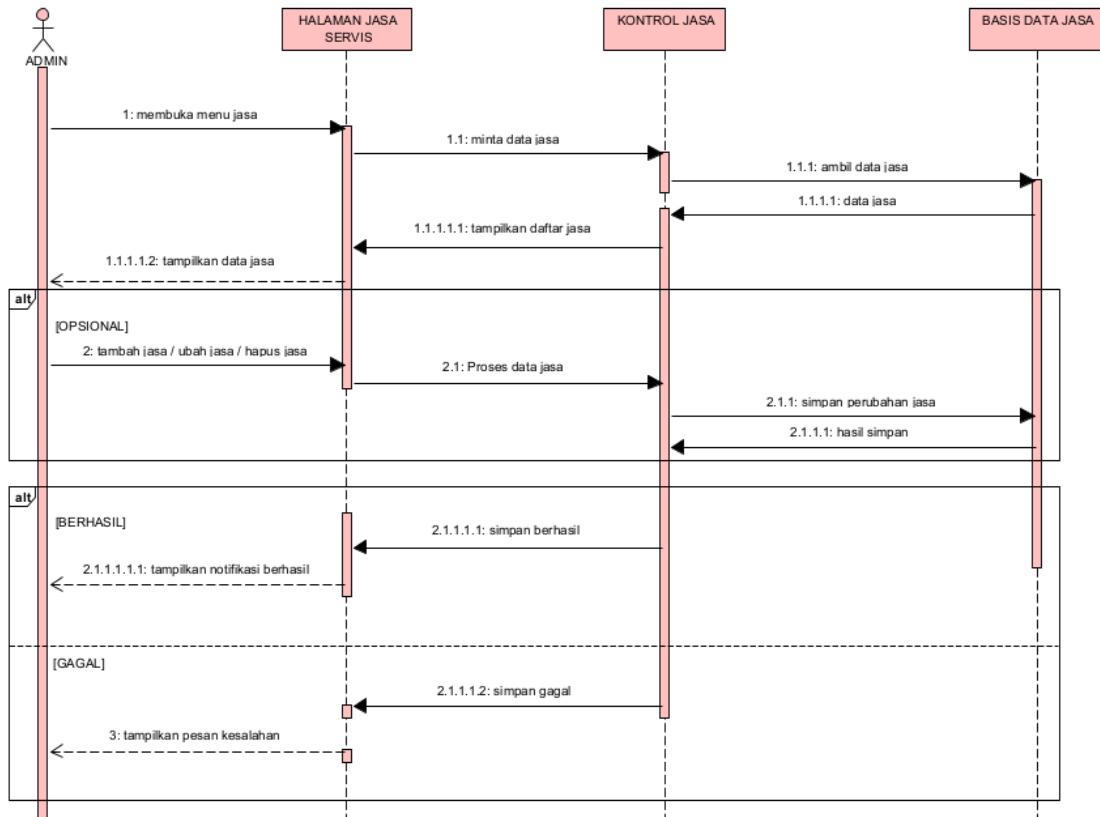
### Admin Profil Admin



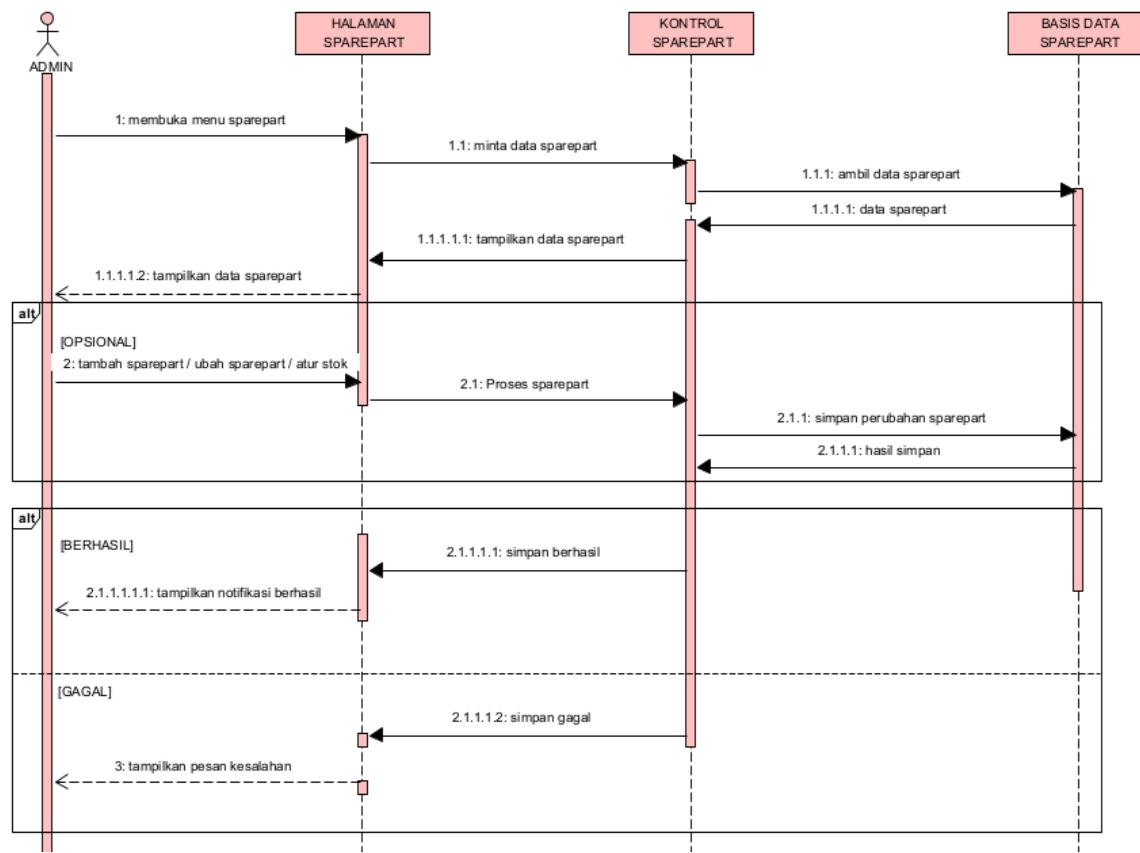
- Admin Kelola Data Pelanggan



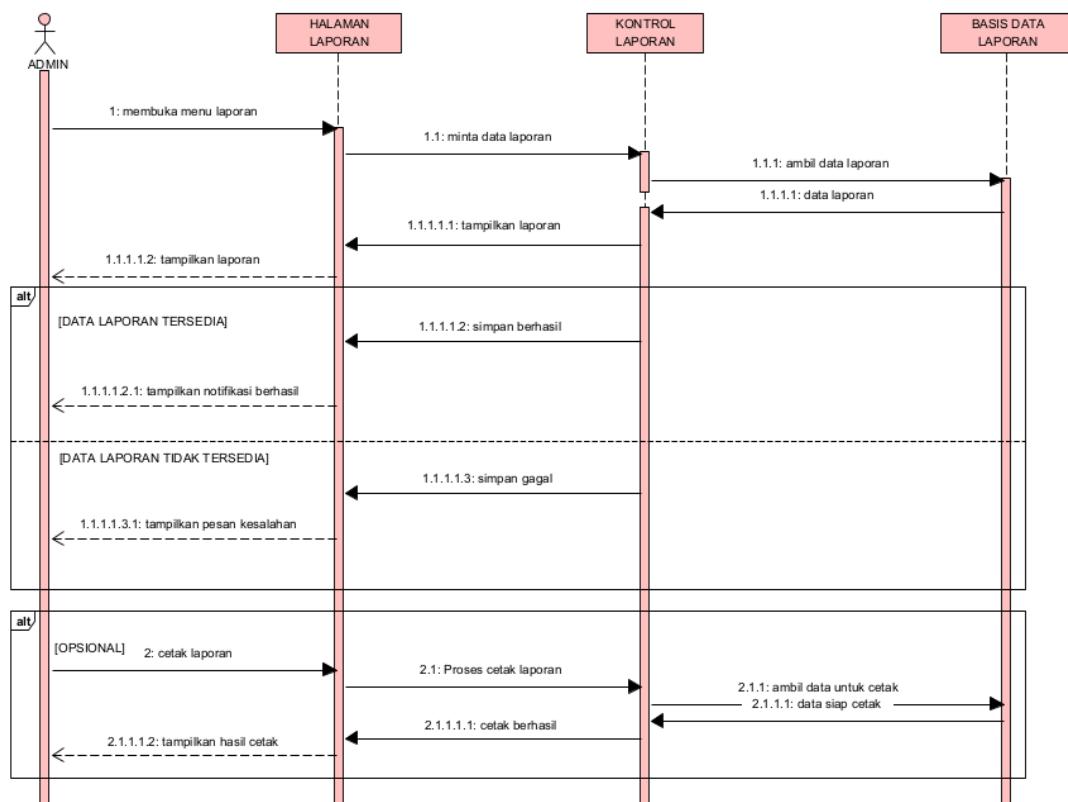
- Admin Kelola Jasa Servis



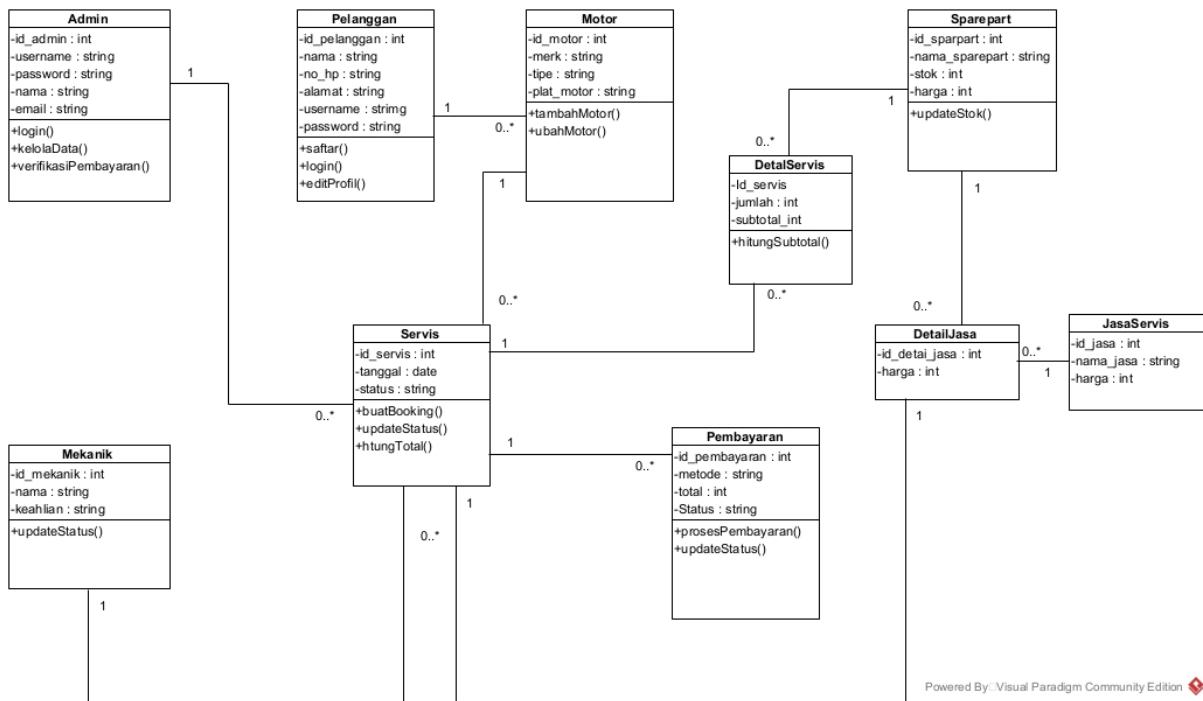
- Admin Kelola Data Sparepart



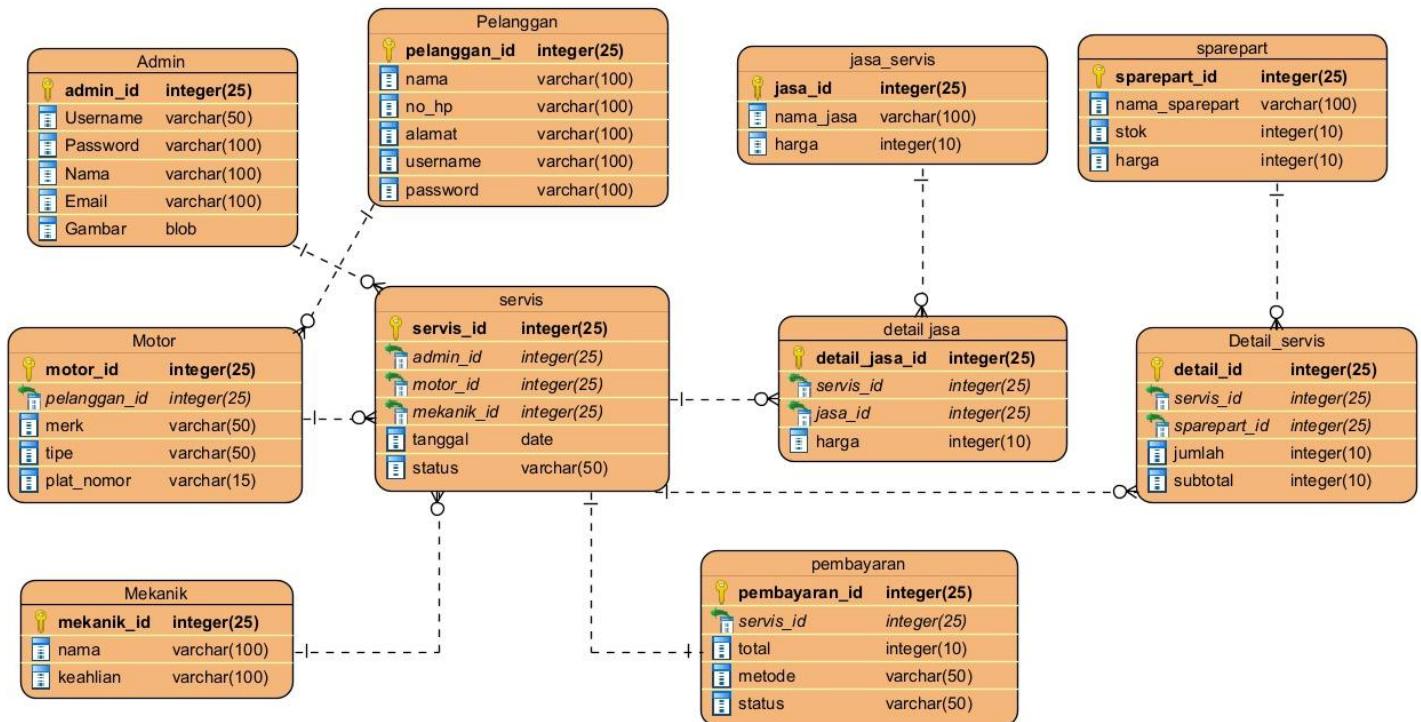
- Admin Kelola Hasil Laporan



### 3.1.3 Class Diagram



### 3.2 Diagram ERD (Entity Relationship Diagram)



## SOFTWARE TESTING DOCUMENT

### 1. PENDAHULUAN

#### 1.1 Tujuan Pembuatan Dokumen

Dokumen ini digunakan sebagai panduan dalam melakukan pengujian terhadap **Aplikasi Bengkel-In**, yaitu aplikasi bengkel berbasis mobile yang dirancang untuk memudahkan proses booking servis, pengelolaan kendaraan, serta layanan bengkel. Dokumen ini bertujuan untuk mengevaluasi kemampuan dan kinerja sistem yang telah dirancang agar sesuai dengan kebutuhan dan harapan pengguna. Penyusunan dokumen ini merupakan bagian dari tugas mata kuliah **Analisis dan Perancangan Sistem** dan ditujukan untuk memastikan bahwa aplikasi Bengkel-In berfungsi dengan baik, berjalan sesuai kebutuhan fungsional, serta memberikan pengalaman pengguna yang optimal.

#### 1.2 Deskripsi Umum Sistem

Perangkat lunak yang akan diuji adalah **aplikasi Bengkel-In**. Aplikasi ini merupakan platform berbasis **mobile** yang dirancang untuk memfasilitasi layanan bengkel, khususnya dalam proses booking servis, pengelolaan kendaraan, serta pemantauan status servis. Aplikasi Bengkel-In ditujukan bagi pelanggan bengkel yang ingin memperoleh layanan servis kendaraan secara lebih mudah dan terstruktur. Aplikasi ini menyediakan fitur untuk melakukan registrasi dan login pengguna, melihat daftar layanan servis, melakukan booking servis, serta mengelola data kendaraan dengan antarmuka yang mudah diakses.

#### 1.3 Deskripsi Dokumen (Ikhtisar)

Dalam dokumen ini berisi 5 bagian yaitu Pendahuluan, Lingkungan Pengujian Perangkat Lunak, Identifikasi, dan Kesimpulan, Rencana Pengujian, Deskripsi dan Uji Hasil.

#### 1.4 Definisi dan Singkatan

SRS	(Software Requirements Specification) merupakan dokumen yang dibuat untuk menjelaskan tentang spesifikasi kebutuhan apa saja yang digunakan dalam aplikasi
SDD	(Software Design Document) merupakan dokumen yang dibuat untuk menjelaskan langkah-langkah desain, spesifikasi kebutuhan fungsional, dan proses-proses dalam pembuatan sistem aplikasi yang akan diterapkan pada aplikasi.

DFD	(Data Flow Diagram) adalah diagram dan notasi yang digunakan untuk menunjukkan aliran data pada perangkat lunak.
ERD	(Entity Relationship Diagram) adalah diagram dan notasi yang digunakan untuk merepresentasikan struktur data statis pada perangkat lunak.
Bengkel-In	Aplikasi Bengkel-In: Sistem Layanan Bengkel Berbasis Mobile

## 1.5 Dokumen Referensi

- Sistem Informasi Layanan Bengkel. 2025. *Perencanaan, Deskripsi, dan Hasil Uji Perangkat Lunak*.
- Aplikasi Bengkel-In. 2025. *Software Requirements Specification (SRS)*.
- Aplikasi Bengkel-In. 2025. *Software Design Document (SDD)*.
- Dokumen Standar Pengujian Perangkat Lunak (Software Testing Documentation).

## 2. LINGKUNGAN PENGUJIAN PERANGKAT LUNAK

### 2.1 Perangkat Lunak Pengujian

Aplikasi Bengkel-In diujikan dengan menggunakan beberapa perangkat lunak pendukung, yaitu:

Sistem Operasi	Windows 11
Perangkat Lunak	Figma
Perangkat Uji	Emulator / Smartphone Android
Platform Aplikasi	Mobile

### 2.2 Perangkat Keras Pengujian

Perangkat keras yang diperlukan untuk menguji aplikasi Bengkel-In adalah satu set komputer dengan spesifikasi sebagai berikut:

Processor	Intel® Core i5-9300H @2.40Ghz
Memory	16GB DDR4
SSD	1 TB

### 2.3 Material Pengujian

Pada aplikasi Bengkel-In, pelanggan dapat mengakses berbagai informasi terkait layanan bengkel melalui aplikasi berbasis mobile. Pelanggan dapat menggunakan aplikasi ini dengan melakukan pendaftaran akun menggunakan username, nomor telepon, dan kata sandi. Setelah memiliki akun, pengguna dapat melihat daftar layanan servis yang tersedia, melakukan pencarian layanan berdasarkan jenis servis, serta melakukan booking servis sesuai jadwal yang diinginkan. Selain itu, pelanggan juga dapat memantau status servis kendaraan yang sedang atau telah dilakukan. Seluruh data booking, servis, pembayaran, serta laporan operasional bengkel dikelola oleh Admin melalui fitur khusus pada sistem admin.

## **2.4 Sumber Daya Manusia**

Persyaratan sumber daya manusia yang akan terlibat dalam proses pengujian Prototyping aplikasi ini adalah :

- Memahami Konsep Design dengan menggunakan Canva.
- Memahami Konsep prototyping dengan menggunakan perangkat lunak figma.

## **2.5 Prosedur Umum Pengujian**

### **2.5.1 Pengalaman dan Latihan**

Pengujian diberikan pelatihan singkat sebagai penyegaran kembali terhadap teknologi yang digunakan dalam pengembangan aplikasi sebelum proses pengujian dilakukan.

### **2.5.2 Persiapan Awal**

Persiapan awal untuk menguji aplikasi adalah mempersiapkan perangkat keras dan perangkat lunak yang akan digunakan.

#### **2.5.2.1 Persiapan Prosedural**

Pengujian ini dilakukan diluar jam perkuliahan dan diluar lingkungan kampus.

#### **2.5.2.2 Persiapan Perangkat Keras**

Perangkat keras yang perlu dipersiapkan adalah satu set laptop dengan spesifikasi:

- Web

Processor	Intel® Core i5-5350U
Memory	8GB DDR3
SSD	256GB

#### **2.5.2.3 Persiapan Perangkat Lunak**

Persiapan yang harus dilakukan adalah :

- Web

1. Aplikasi disimpan dalam bentuk prototipe aplikasi web di Figma
2. Hubungkan perangkat keras dengan jaringan internet agar bisa membuka Figma
3. Buka browser dan telusuri :

<https://www.figma.com/design/encFcdSZjH3G9s6vlh9z2a/KELOMPOK-4-ANSI-Bengkel-in?node-id=651-9361&t=FtDjgXQP7zjkuwCo-1>

atau dengan aplikasi Figma dan membuka file mockup yang sudah dibagikan.

### **2.5.3 Pelaksanaan**

Pelaksanaan pengujian dilakukan sesuai dengan persiapan sebelumnya. Dan pengujian dilaksanakan berdasarkan skenario yang telah disediakan.

### **2.5.4 Pelaporan Hasil Dokumen**

Setelah aplikasi diuji, laporan hasil dari pengujian tersebut akan diberikan kepada dosen pengampu Analisis dan Perancangan Sistem.

### 3. IDENTIFIKASI DAN RENCANA PENGUJIAN

Kelas Uji	Butir Uji	Identifikasi		Jenis Pengujian	Teknik Pengujian	Penguji
		SRS/SDD	SPMP			
Pengujian Login Pelanggan	Pengisian username dan password dengan benar	SRS-3.2.3	—	Sistem	Blackbox	Daffa
	Pengisian username atau password salah	STD-1.1	—	Sistem	Blackbox	Daffa
	Tidak mengisi username dan password	STD-1.2	—	Sistem	Blackbox	Daffa
Pengujian Menelusuri Layanan Servis	Menampilkan layanan servis berdasarkan kategori	SRS-3.2.3	—	Sistem	Blackbox	Daffa
	Halaman layanan tidak muncul karena gangguan jaringan	STD-2.1	—	Sistem	Blackbox	Daffa
	Tidak ada koneksi internet	STD-2.2	—	Sistem	Blackbox	Daffa
Pengujian Booking Servis	Menambahkan layanan ke booking	SRS-3.2.3	—	Sistem	Blackbox	Daffa
	Menghapus layanan dari booking	STD-3.1	—	Sistem	Blackbox	Daffa
	Data booking sesuai layanan yang dipilih	STD-3.2	—	Sistem	Blackbox	Daffa
Pengujian Konfirmasi Booking	Mengisi data kendaraan dan jadwal servis	SRS-3.2.3	—	Sistem	Blackbox	Daffa

### 4. DESKRIPSI DAN HASIL UJI

#### 4.1 Fungsional Admin

Fungsional	Sesuai	Tidak	Sebelum Revisi	Sesudah Revisi
Sistem dapat memvalidasi admin saat login dan berfungsi.	✓	—	—	—

Fungsional	Sesuai	Tidak	Sebelum Revisi	Sesudah Revisi
Sistem dapat menampilkan halaman Dashboard Admin dan berfungsi.	✓		–	–
Sistem dapat menampilkan halaman Kelola Layanan Servis dan berfungsi.	✓		–	–
Sistem dapat menampilkan halaman Kelola Pelanggan dan berfungsi.	✓		–	–
Sistem dapat menampilkan halaman Kelola Laporan Servis dan berfungsi.		✗	Tampilan laporan belum menampilkan data secara lengkap	Menambahkan kolom data laporan dan tombol simpan
Sistem dapat menampilkan halaman Edit Profil Admin dan berfungsi.	✓		–	–
Sistem dapat melakukan logout untuk mengakhiri sesi admin dan berfungsi.	✓		–	–

#### 4.2 Fungsional Pelanggan

Fungsional	Sesuai	Tidak	Sebelum Revisi	Sesudah Revisi
Sistem dapat menampilkan halaman login pelanggan dan berfungsi.	✓		–	–
Sistem dapat menampilkan halaman pendaftaran pelanggan dan berfungsi.		✗	Form pendaftaran belum menyediakan field nomor telepon dan mengganti name	Menambahkan field nomor telepon dan mengganti name

Fungsional	Sesuai	Tidak	Sebelum Revisi	Sesudah Revisi
			masih menggunakan field <i>name</i> .	menjadi username.
Sistem dapat menampilkan halaman daftar layanan servis dan berfungsi.	✓		—	—
Sistem dapat menampilkan detail layanan servis dan berfungsi.	✓		—	—
Sistem dapat menampilkan halaman booking servis dan berfungsi.	✓		—	—
Sistem dapat menampilkan halaman konfirmasi booking dan berfungsi.		✗	Sistem belum memberikan peringatan ketika data kendaraan belum lengkap.	Menambahkan validasi dan notifikasi pengisian data kendaraan.
Sistem dapat memberikan notifikasi kepada pelanggan untuk melengkapi data kendaraan dan jadwal servis.	✓		—	—

Fungsional	Sesuai	Tidak	Sebelum Revisi	Sesudah Revisi
Sistem dapat menghubungkan ke gateway pembayaran dan berfungsi.	✓		—	—
Sistem dapat menampilkan halaman detail booking yang berisi status servis dan konfirmasi pembayaran.	✓		—	—

## 5. Kesimpulan

Kesimpulan dari proyek Bengkel-In adalah bahwa aplikasi ini telah dirancang dan diuji sebagai sistem layanan bengkel berbasis mobile yang bertujuan untuk memudahkan pelanggan dalam melakukan booking servis kendaraan secara online. Aplikasi Bengkel-In dikembangkan untuk memberikan pengalaman pengguna yang mudah dan efisien, mulai dari proses login dan pendaftaran pengguna, pemilihan layanan servis, pengisian data kendaraan, hingga proses konfirmasi dan pembayaran. Berdasarkan hasil pengujian fungsional yang telah dilakukan, sebagian besar fitur aplikasi dapat berjalan sesuai dengan kebutuhan sistem. Beberapa kendala yang ditemukan telah diperbaiki melalui proses revisi sehingga aplikasi diharapkan mampu mendukung operasional bengkel secara lebih terstruktur serta meningkatkan kenyamanan dan kepuasan pelanggan.

## Link GitHub :

- *Saeful Iman\_* <https://github.com/saeful-iman>
- *Neng Rista Nur Rahmawati* <https://github.com/nengristanurr>
- *Azka Putra P\_* <https://github.com/ajekptr>
- *Muhamad Daffa Fasya R\_* <https://github.com/dafasshh>
- *Dwi zahwa Abdillah\_* <https://github.com/dwizahwasi24>
- *Nurlaila\_* <https://github.com/nurlailanurlailasi24-sketch>
- *Muhamad Nazar Z\_* <https://github.com/zulnazar>