

LAMPIRAN

Lampiran 1 Surat Kesediaan Pembimbing

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Muhammad Dani Haikal
NIM : 20090031
Program Studi : D4 Teknik Informatika

Pihak Kedua

Nama : M. Nishom, M.Kom.
Status : Dosen
NIDN : 0619048701
Jabatan Fungsional : Lektor
Pangkat/Golongan : Penata/IIIC

Pada hari ini Kamis tanggal Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat:

1. Berjanji akan melakukan bimbingan minimal 1 kali dalam 1 minggu, dan memberikan konfirmasi jika ada halangan/absen.
2. Berjanji akan menyelesaikan penelitian (produk dan laporan) sesuai jadwal penelitian.
3. Berjanji tidak akan meminta rekomendasi ujian/sidang skripsi jika tidak dapat memenuhi syarat pada poin 1 dan 2.

Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 21 Maret 2024

Pihak Pertama



Muhammad Dani Haikal

Pihak Kedua



M. Nishom, M.Kom.

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom.
NIPY. 09.015.225

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Muhammad Dani Haikal

NIM : 20090031

Program Studi : D4 Teknik Informatika

Pihak Kedua

Nama : Dairoh, M.Sc

Status : Dosen

NIDN : 0619048701

Jabatan Fungsional : Lektor

Pangkat/Golongan : Penata Tk I/IIID

Pada hari ini Kamis tanggal Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat:

1. Berjanji akan melakukan bimbingan minimal 1 kali dalam 1 minggu, dan memberikan konfirmasi jika ada halangan/absen.
2. Berjanji akan menyelesaikan penelitian (produk dan laporan) sesuai jadwal penelitian.
3. Berjanji tidak akan meminta rekomendasi ujian/sidang skripsi jika tidak dapat memenuhi syarat pada poin 1 dan 2.

Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 15 Maret 2024

Pihak Pertama



Muhammad Dani Haikal

Pihak Kedua



Dairoh, M.Sc

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom.
NIPY. 09.015.225

Lampiran 2 Surat Pernyataan Pengajuan HKI

B.

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. Nama : Muhammad Dani Haikal
Kewarganegaraan : Indonesia
Alamat : Jl. Kerti RT.01 RW.12 NO 17, Desa Suradadi, Kecamatan Suradadi, Kabupaten Tegal, Jawa Tengah
2. Nama : M. Nishom, S.Kom., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Jl. Jepara, Perum Griya Putri Land Blok A6, RT 03 RW 04, Margadana, Tegal, 52143
3. Nama : Dairoh, M.Sc.
Kewarganegaraan : Indonesia
Alamat : Jl. Garuda No 3 RT/RW 002/009 Randugunting Tegal Selatan Kota Tegal

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan:
Berupa : Program Komputer
Berjudul : Rancang Bangun Sistem Informasi Pemesanan Rental Motor Berbasis Website
 - Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
 - Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
 - Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
 - Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
 - Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
 - Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.
2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang saya mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas saya / kami melanggar, maka saya / kami bersedia secara sukarela bahwa:

- a. permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau
- b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
- c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini saya/kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Tegal, 23 Juli 2024



Muhammad Dani Haikal
Pemegang Hak Cipta *

M. Nishom, S.Kom., M.Kom.
Pemegang Hak Cipta *

Dairoh, M.Sc.
Pemegang Hak Cipta *

* Semua pemegang hak cipta agar menandatangani di atas materai.

Lampiran 3 Surat Pengalihan HKI

C.

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini:

1. Nama : Muhammad Dani Haikal
Kewarganegaraan : Indonesia
Alamat : Jl. Kerti RT.01 RW.12 NO 17, Desa Suradadi, Kecamatan Suradadi, Kabupaten Tegal, Jawa Tengah
2. Nama : M. Nishom, S.Kom., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Jl. Jepara, Perum Griya Putri Land Blok A6, RT 03 RW 04, Margadana, Tegal, 52143
2. Nama : Dairoh, M.Sc
Kewarganegaraan : Indonesia
Alamat : Jl. Garuda No 3 RT/RW 002/009 Randugunting Tegal Selatan Kota Tegal

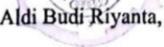
Adalah **Pihak I** selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada:

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Rancang Bangun Sistem Informasi Pemesanan Rental Motor Berbasis *Website*". untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Pemegang Hak Cipta
Ketua P3M

(Dr. Aldi Budi Riyanta, S.Si., M.T)

Tegal, 27 Juli 2024
Pencipta



(Muhammad Dani Haikal)

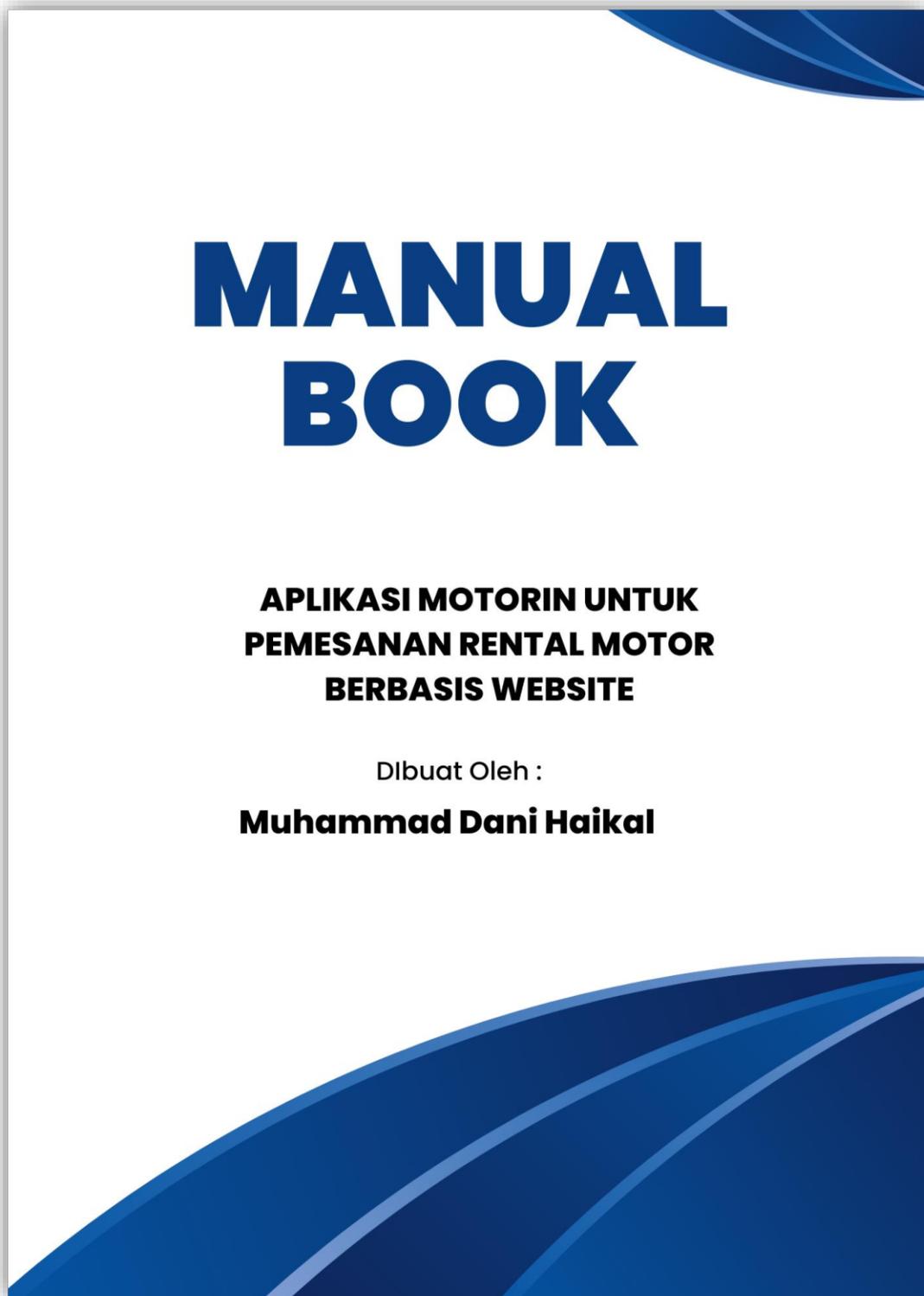


(M. Nishom, S.Kom., M.Kom.)



(Dairoh, M.Sc)

D.



1. PENDAHULUAN

1.1 Tujuan Pembuatan Dokumen

Dokumen user manual untuk "Aplikasi MotorIn Pemesanan Rental Motor Berbasis *Website*" dibuat dengan tujuan menggambarkan, menjelaskan, dan memberikan panduan penggunaan aplikasi MotorIn. Aplikasi ini dirancang untuk mempermudah pengguna dalam melakukan pemesanan motor untuk disewa. Panduan ini ditujukan untuk semua pengguna aplikasi, termasuk pelanggan dan penyedia rental motor, dengan tujuan untuk memastikan mereka dapat memanfaatkan semua fitur yang tersedia dengan maksimal.

1.2 Deskripsi Umum Sistem

Aplikasi "MotorIn" untuk Pemesanan Rental Motor di berbagai kota merupakan sistem berbasis *website* yang dirancang untuk mempermudah proses pemesanan motor. Aplikasi ini memungkinkan pengguna untuk mencari, membandingkan, dan memesan berbagai motor yang tersedia di kota pilihan mereka. Tujuan dibuatnya sistem ini adalah untuk menyediakan platform yang efisien dan user-friendly bagi pelanggan dan penyedia rental, sehingga dapat meningkatkan kenyamanan dan kemudahan dalam mengatur penyewaan motor secara profesional.

1.3 Deskripsi Dokumen

Dokumen ini dibuat untuk memberikan panduan penggunaan "Aplikasi MotorIn Pemesanan Rental Motor Berbasis *Website*". Dokumen ini berisikan informasi sebagai berikut:

1. BAB I.

Berisikan informasi umum yang merupakan bagian pendahuluan, yang meliputi tujuan pembuatan dokumen, deskripsi umum sistem, serta deskripsi dokumen.

2. BAB II.

Berisikan perangkat yang dibutuhkan untuk menjalankan Aplikasi MotorIn, termasuk perangkat lunak dan perangkat keras yang diperlukan untuk menjalankan aplikasi ini dengan optimal..

3. BAB III.

Berisikan user manual untuk menggunakan Aplikasi MotorIn, mencakup langkah-langkah pendaftaran, *login*, pencarian dan pemesanan motor, serta fitur-fitur lain yang tersedia bagi pengguna dan penyedia rental motor.

2. PERANGKAT YANG DIBUTUHKAN

2.1 Perangkat Lunak

Perangkat lunak yang dibutuhkan adalah sebagai berikut:

1. *System operasi seperti Windows*
2. *Web browser*
3. *Visual Studio Code*
4. *Node.js v.20*

2.2 Perangkat Keras

Perangkat keras (Hardware) yang dibutuhkan sebagai berikut:

1. Komputer / Laptop dengan spesifikasi minimal :
 - Prosesor Intel Core i3 atau setara
 - RAM Minimal 2GB
 - Penyimpanan 128GB

2.3 Penggunaan Aplikasi

Pengguna aplikasi yang akan menggunakan aplikasi adalah sebagai berikut:

1. *Penyewa* : Pengguna ini memiliki fitur untuk mencari rental motor berdasarkan kota yang dipilih, melihat dan membandingkan motor yang tersedia, melakukan pemesanan motor, dan melakukan pembayaran sesuai ketentuan yang ditentukan. Penyewa juga dapat mengubah profil mereka dan melihat status pesanan pada *dashboard*.

2. Rental : Pengguna ini memiliki fitur untuk mengelola rental motor, termasuk menambahkan, mengedit, dan menghapus data motor. Rental juga dapat melihat data pemesanan yang masuk serta melihat laporan penjualan bulanan. Selain itu, rental dapat mengubah profil mereka melalui *dashboard*.

3. MENU DAN CARA PENGGUNAAN

3.1 Struktur Menu

Adapun struktur halaman pada Aplikasi MotorIn Untuk Pemesanan Rental Motor *Website* adalah sebagai berikut:

1. Halaman Utama Home

Pada halaman home akan muncul tampilan awal pada saat mengakses Aplikasi MotorIn adapun menu *navbar* yang dapat digunakan untuk sumber informasi penggunaan aplikasi dan menu tombol *login* dan daftar yang dapat digunakan untuk masuk sebagai penyewa dan rental.

2. Halaman Navbar

Pada halaman navbar ada beberapa menu yang tersedia adalah sebagai berikut :

- a. Home berisi halaman awal dalam aplikasi.
- b. Pesan untuk memilih rental motor dan memesan/menyewa motor.
- c. Syarat dan Ketentuan berisi syarat dan ketentuan berlaku untuk penyewa motor.

3. Halaman *Dashboard*

Pada halaman *dashboard* ada beberapa menu yang tersedia sebagai berikut:

- a. Melihat dan merubah data profil pada penyewa atau rental.
- b. Melihat dan merubah data pesanan pada penyewa atau rental.
- c. Melihat, menambahkan, dan merubah data motor pada rental.

d. Melihat data laporan penjualan bulanan pada rental.

3.2 Pengguna

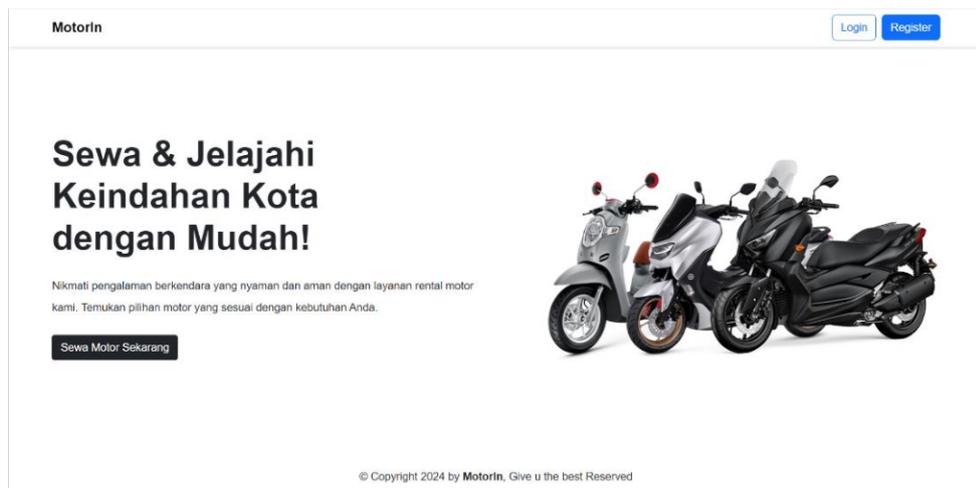
Pada bagian ini akan dijelaskan mengenai menu-menu yang tersedia disistem sebagai berikut:

3.2.1 Cara membuka Sistem

Untuk menjalankan *website* MotorIn adalah sebagai berikut:

1. Bukalah *website* sistem melalui *web browser* yang tersedia seperti (*Goggle Chrome* dll)
2. Tekan tombol *enter* pada *keyboard*
3. Setelah itu, akan muncul tampilan utama *website* sistem.

Pada halaman terdapat menu di *navbar*. Adapun tampilan utama dapat dilihat pada gambar 1.



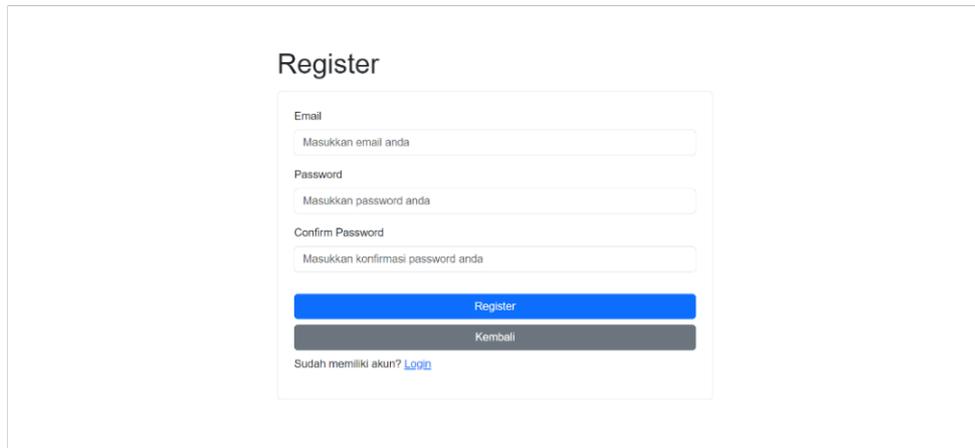
Gambar 1 Halaman Home Sebelum *Login*

3.2.2 Halaman *Autentikasi*

Pada sistem ini terdapat halaman untuk *autentikasi* yang terdiri dari berbagai menu sebagai berikut:

1. Halaman *Register*

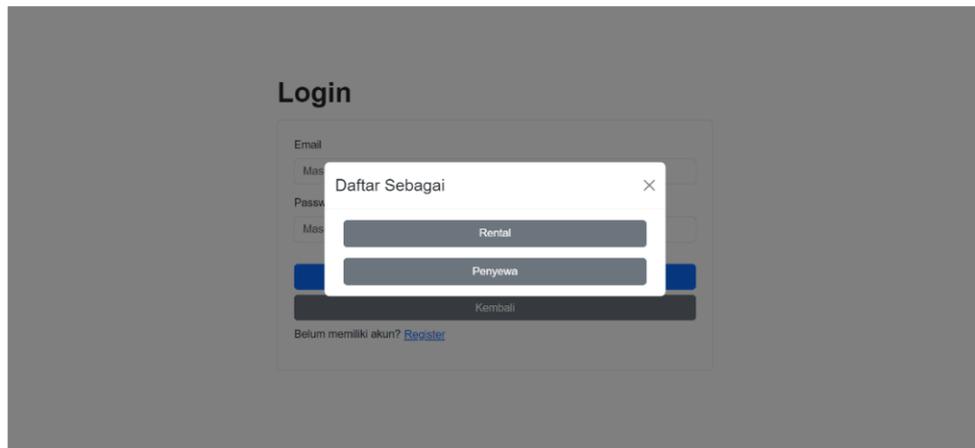
Halaman ini digunakan oleh pengguna baru untuk membuat akun dengan memasukkan data *register* seperti email, *password* dan *confirm password*, yang dapat dilihat pada gambar 2.



Gambar 2 Halmaan *Register*

2. Menu Pilihan *Register* User

Menu ini memungkinkan pengguna untuk memilih jenis akun yang ingin didaftarkan, yaitu sebagai penyewa atau rental (owner), yang dapat dilihat pada gambar 3.



Gambar 3 Menu Pilihan Daftar

3. Halaman *Login*

Halaman ini digunakan oleh pengguna yang sudah terdaftar untuk masuk ke dalam sistem dengan memasukkan data email dan *password*, yang dapat dilihat pada gambar 4.

Login

Email
Masukkan email anda

Password
Masukkan password anda

Login

Kembali

Belum memiliki akun? [Register](#)

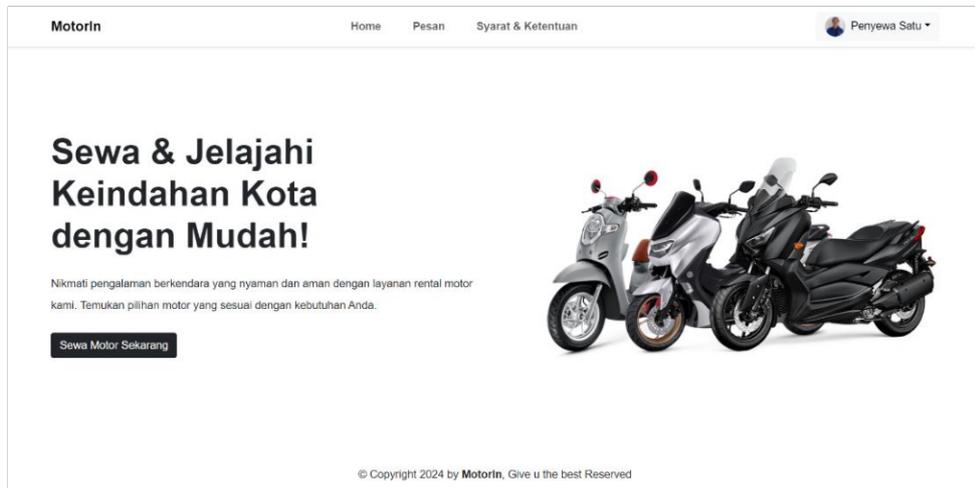
Gambar 4 Halaman *Login*

3.2.3 Halaman Utama

Pada sistem ini terdapat berbagai halaman utama sebagai berikut:

1. Halaman Home

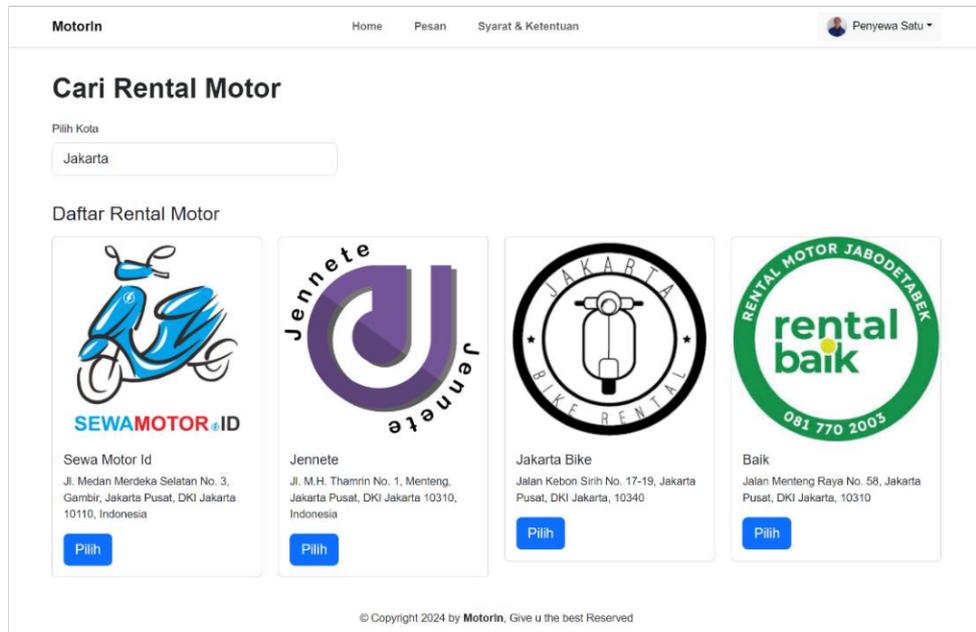
Halaman ini adalah halaman pertama yang dilihat pengguna setelah *login*, menampilkan informasi umum dan navigasi ke fitur utama, yang dapat dilihat pada gambar 5.



Gambar 5 Halaman Home Setelah *Login*

2. Halaman Pesan

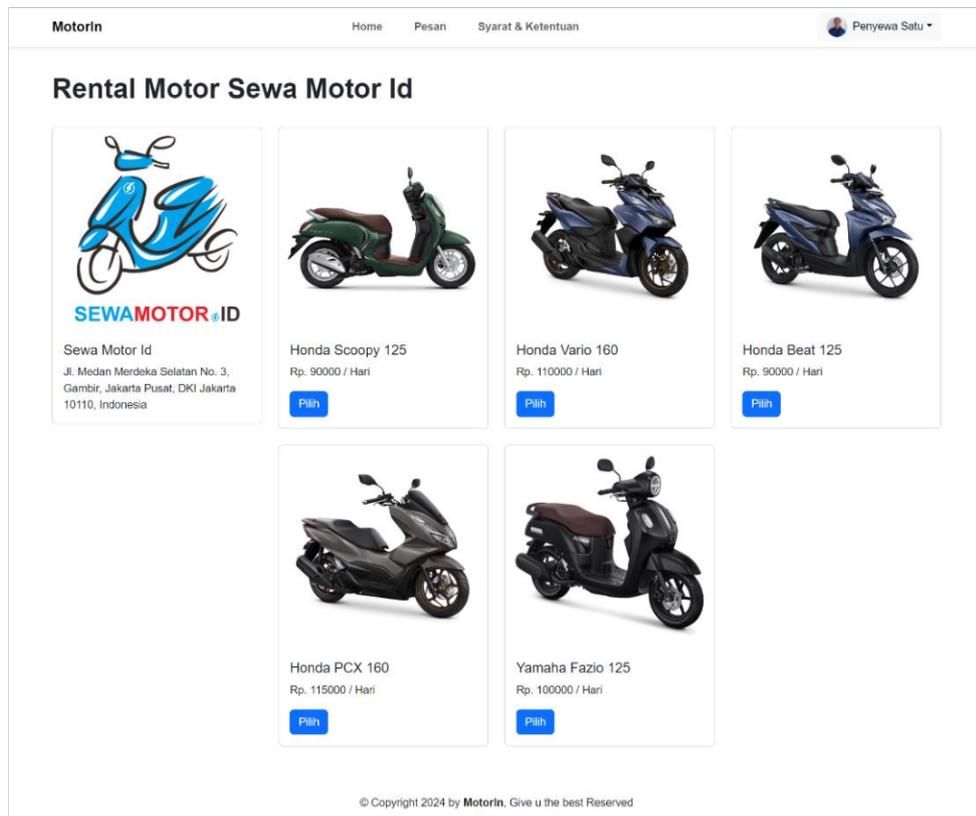
Halaman ini memungkinkan pengguna untuk mencari rental motor berdasarkan kota yang dipilih dan melihat daftar motor yang tersedia, yang dapat dilihat pada gambar 6.



Gambar 6 Halman Pesan

3. Halaman Dalam Rental

Halaman ini menampilkan detail rental, termasuk informasi mengenai motor yang disewakan dan profil rental, yang dapat dilihat pada gambar 7.



Gambar 7 Halaman Dalam Rental

4. Halaman Buat Pesanan

Halaman ini digunakan oleh pengguna untuk membuat pesanan dengan memilih motor dan mengisi informasi pemesanan, yang dapat dilihat pada gambar 8.

Motorin Home Pesan Syarat & Ketentuan  Penyewa Satu ▾

Pesanan

Tanggal Mulai
23/07/2024

Tanggal Selesai
23/07/2024

Lokasi Ambil Custom Waktu Ambil 06:00

stasiun gambir

Lokasi Kembali Rental Waktu Kembali 20:00

Jaminan 1
KTP (Wajib)

Jaminan 2
STNK

Buat Pesanan



Honda Scoopy 125

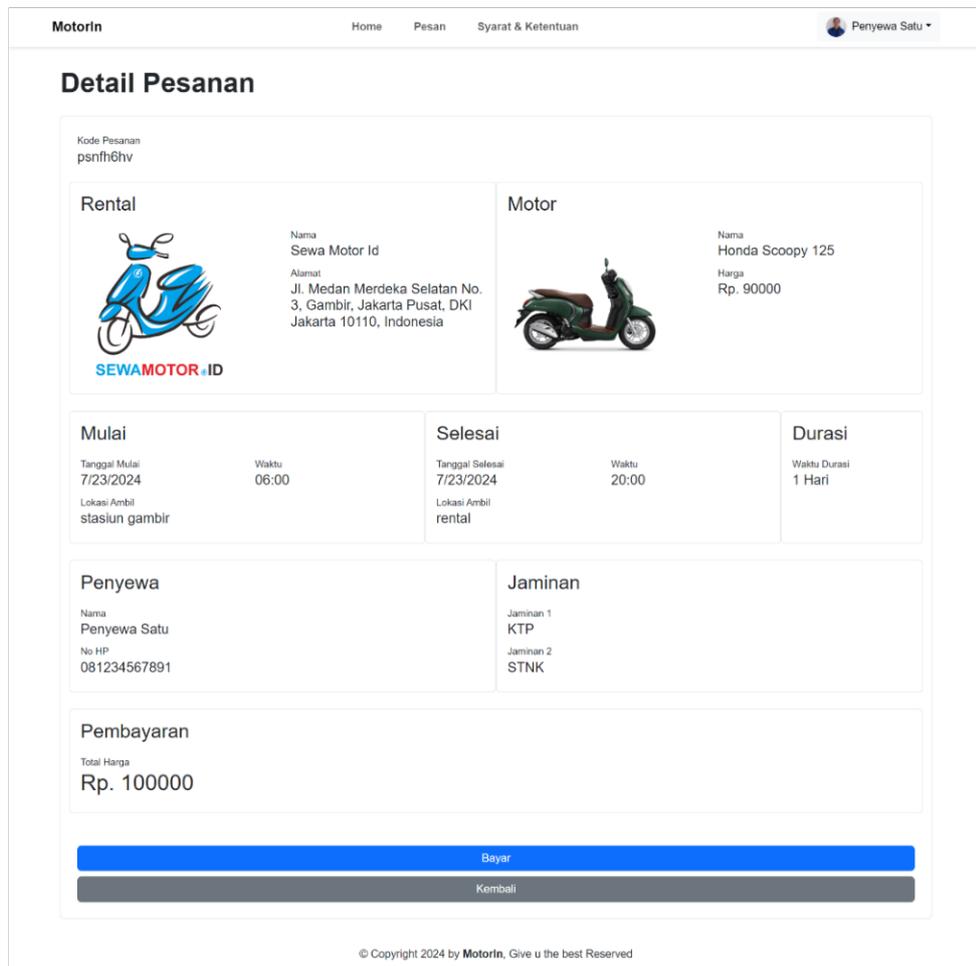
Nama Rental: Sewa Motor Id
Alamat Rental: Jl. Medan Merdeka Selatan No. 3, Gambir, Jakarta Pusat, DKI Jakarta 10110, Indonesia
Harga Per Hari: Rp. 90000
Jumlah Hari: 1 Hari
Total Harga: Rp. 100000

© Copyright 2024 by Motorin. Give u the best Reserved

Gambar 8 Halman Buat Pesanan

5. Halaman Detail Pesanan

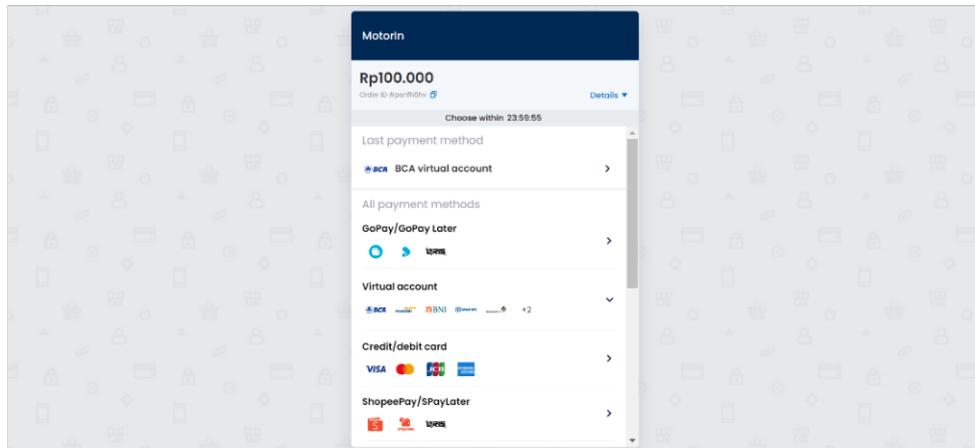
Halaman ini menampilkan detail dari pesanan yang telah dibuat, termasuk status dan informasi lainnya, yang dapat dilihat pada gambar 9.



Gambar 9 Halaman Detail Pesanan

6. Halaman Pembayaran

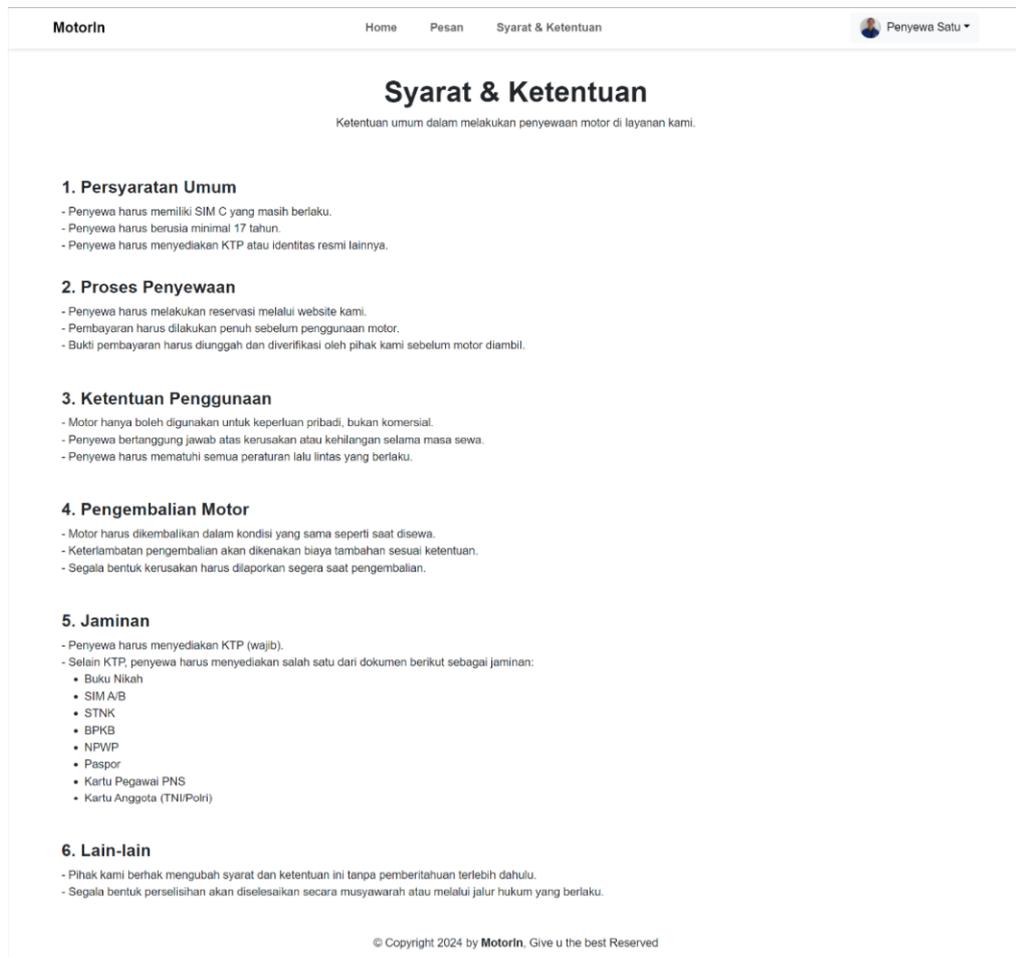
Halaman ini digunakan untuk melakukan proses pembayaran pesanan sesuai dengan metode pembayaran yang tersedia, yang dapat dilihat pada gambar 10.



Gambar 10 Halaman Pembayaran

7. Halaman Syarat & Ketentuan

Halaman ini menampilkan informasi mengenai syarat dan ketentuan penggunaan aplikasi dan layanan rental motor, yang dapat dilihat pada gambar 11.



Gambar 11 Halaman Syarat dan Ketentuan

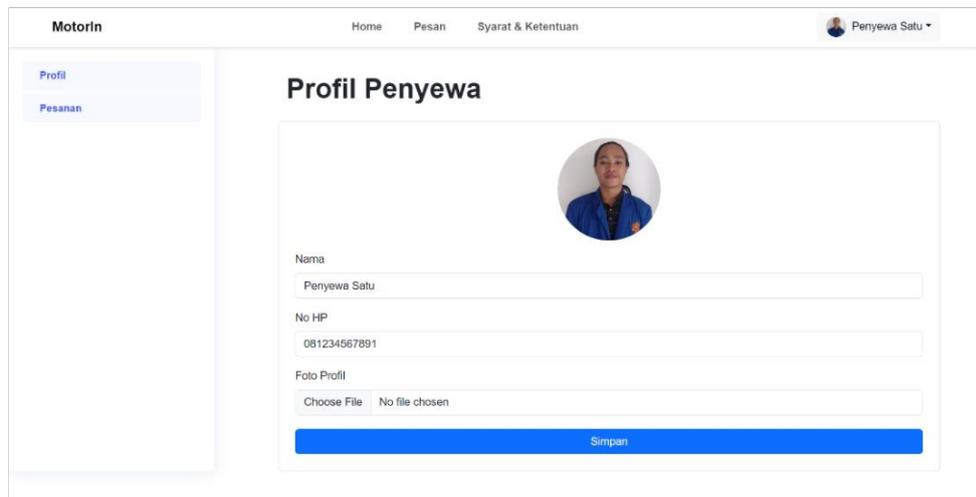
Menu *logout* digunakan untuk keluar dari halaman admin dan jika diklik akan kembali kehalaman utama. Adapun tampilan halaman *logout* dapat dilihat pada gambar.

7.2.4 Halaman *Dashboard*

Pada sistem ini terdapat halaman *dashboard* sebagai berikut:

1. Halaman *Dashboard* Profil Penyewa

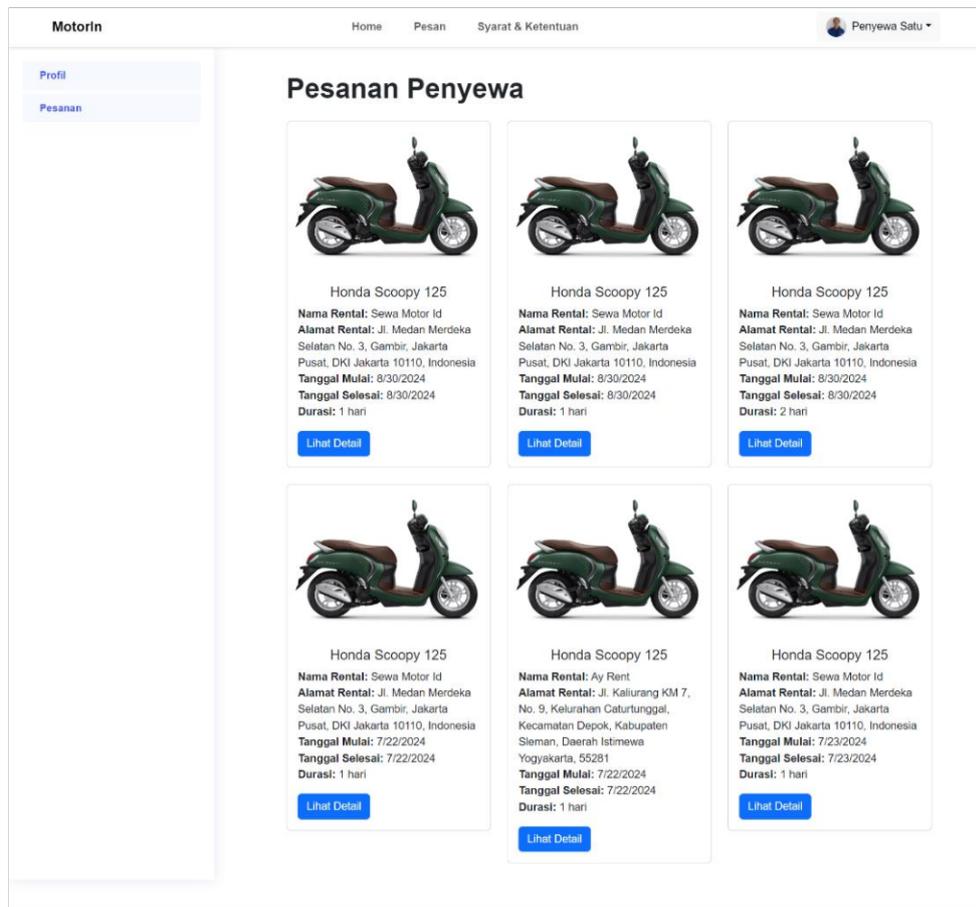
Halaman ini memungkinkan penyewa untuk melihat dan mengedit profil mereka, yang dapat dilihat pada gambar 12.



Gambar 12 Halaman *Dashboard* Profil Penyewa

2. Halaman *Dashboard* Pesanan Penyewa

Halaman ini menampilkan daftar pesanan yang telah dibuat oleh penyewa beserta statusnya, yang dapat dilihat pada gambar 13.



Gambar 13 Halaman *Dashboard* Pesanan Penyewa

3. Halaman *Dashboard* Profil Rental

Halaman ini memungkinkan rental untuk melihat dan mengedit profil mereka, yang dapat dilihat pada gambar 14.

The image shows a web application interface for 'Motorin'. At the top left, the word 'Motorin' is displayed. At the top right, there is a user profile icon and the text 'Sewa Motor Id'. On the left side, there is a vertical navigation menu with four items: 'Profil', 'Pesanan', 'Motor', and 'Laporan'. The main content area is titled 'Profil Rental' and contains a form with the following fields:

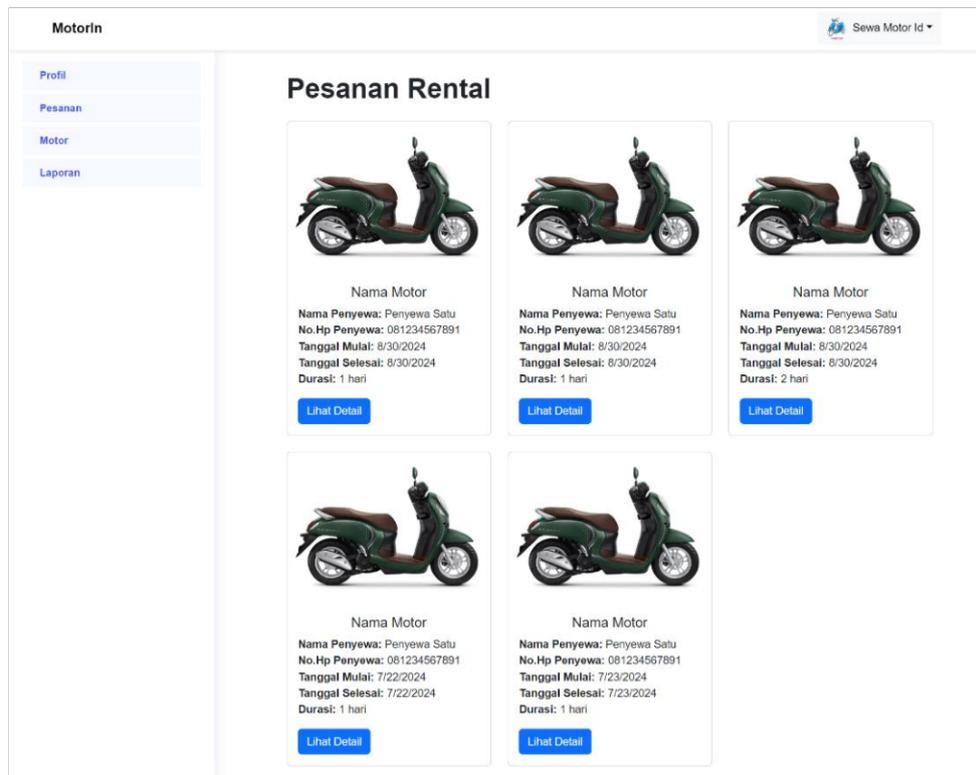
- Profil**
 - Nama: Sewa Motor Id
 - No HP: 081234567891
 - Kota: Jakarta
 - Alamat: Jl. Medan Merdeka Selatan No. 3, Gambir, Jakarta Pusat, DKI Jakarta 10110, Indonesia
 - Foto Profil: Choose File (No file chosen)
- Bank**
 - Bank: BCA
 - No Rekening: 1234567890

A blue 'Simpan' button is located at the bottom of the form.

Gambar 14 Halaman *Dashboard* Profil Rental

4. Halaman *Dashboard* Pesanan Rental

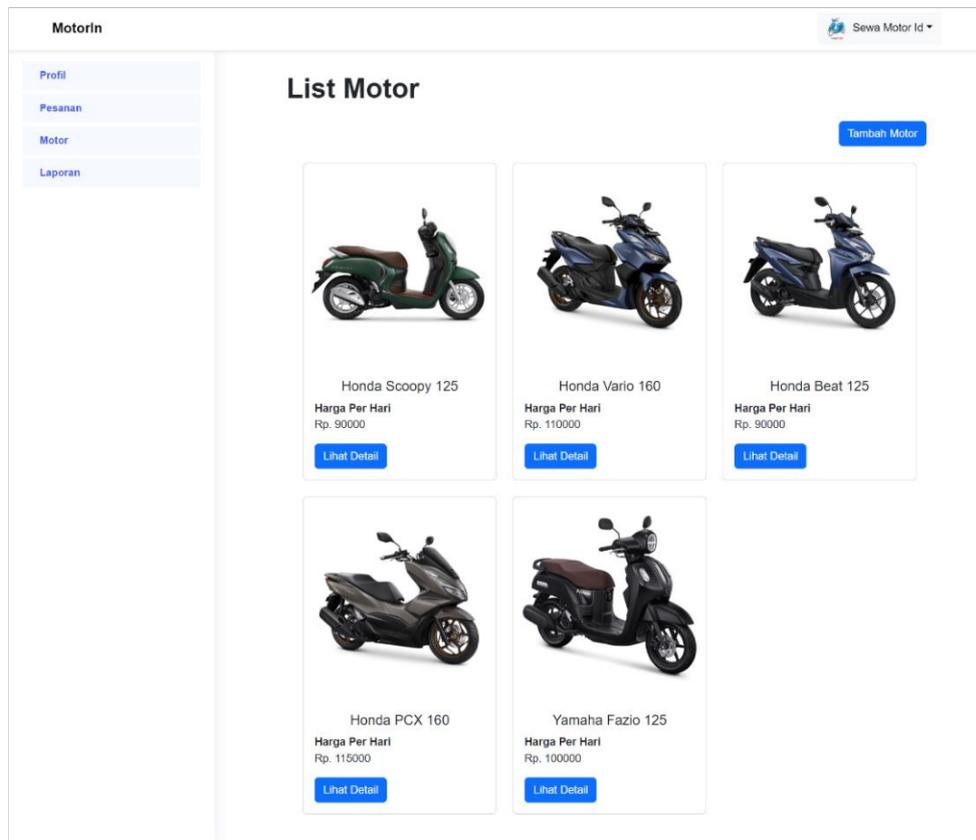
Halaman ini menampilkan daftar pesanan yang masuk ke rental beserta statusnya, yang dapat dilihat pada gambar 15.



Gambar 15 Halaman *Dashboard* Pesanan Rental

5. Halaman *Dashboard* List Motor Rental

Halaman ini menampilkan daftar motor yang disewakan oleh rental, yang dapat dilihat pada gambar 16.



Gambar 16 Halaman *Dashboard* List Motor Rental

6. Halaman *Dashboard* Lihat Motor Rental

Halaman ini menampilkan detail dari motor yang disewakan, termasuk informasi spesifik mengenai motor tersebut, yang dapat dilihat pada gambar 17.

Motorin Sewa Motor Id ▾

Motor Honda Scoopy 125



Nama

Harga (Rp.)

Foto Motor
 No file chosen

Gambar 17 Halaman *Dashboard* Lihat Motor Rental

7. Halaman *Dashboard* Tambah Motor Rental

Halaman ini memungkinkan rental untuk menambahkan motor baru ke dalam daftar motor yang disewakan, yang dapat dilihat pada gambar 18.

Motorin Sewa Motor Id ▾

Tambah Motor



Nama

Harga (Rp.)

Foto Motor
 No file chosen

Gambar 18 Halaman *Dashboard* Tambah Motor Rental

8. Halaman *Dashboard* Laporan Penjualan Rental

Halaman ini menampilkan laporan penjualan bulanan dari rental, termasuk statistik dan informasi penjualan lainnya, yang dapat dilihat pada gambar 19.

Motorin Sewa Motor Id

Profil
Pesanan
Motor
Laporan

Laporan Penjualan

Cari Laporan

July 2024

No	Nama Motor	Total Pendapatan
1	Honda Scoopy 125	Rp.90000
Jumlah Total Pendapatan		Rp.90000

Gambar 19 Halaman *Dashboard* Laporan Penjualan Rental



TECHNICAL BOOK

**APLIKASI MOTORIN UNTUK
PEMESANAN RENTAL MOTOR
BERBASIS WEBSITE**

Dibuat Oleh :

Muhammad Dani Haikal



A. Pendahuluan

MotorIn adalah sebuah aplikasi *web* yang didesain untuk memudahkan pengguna dalam melakukan pemesanan rental motor. Aplikasi ini menggunakan teknologi modern untuk memastikan pengalaman pengguna yang optimal dan efisien. Dibangun menggunakan framework *Express.js* untuk *backend* dan *React.js* untuk *frontend*, aplikasi ini mencakup peran pengguna seperti penyewa dan rental (owner), masing-masing dengan fungsionalitas yang spesifik..

B. Spesifikasi

1. Spesifikasi *Hardware*

Untuk menjalankan aplikasi MotorIn dengan lancar, berikut adalah spesifikasi *hardware* yang disarankan:

- RAM: Minimal 4 GB (disarankan 8 GB)
- *Storage*: SSD 128 GB atau lebih
- *Network*: Koneksi internet yang stabil

2. Spesifikasi *Software*

Untuk pengembangan dan *deployment* aplikasi MotorIn, berikut adalah spesifikasi *software* yang diperlukan:

a) Sistem Operasi

- Windows: Windows 10 atau lebih baru

b) Bahasa Pemrograman dan *Framework*

- *Node.js*: Versi 20 atau lebih tinggi

c) *Database*

- *Mysql*: Versi 5.7 atau lebih tinggi

d) Alat

- Visual Studio *Code*: Editor untuk pengembangan kode

C. *Source Code Frontend*

Berikut penjelasan mengenai *Source Code frontend website* MotorIn, yang meliputi komponen dan fitur utama:

1. Halaman *Register*

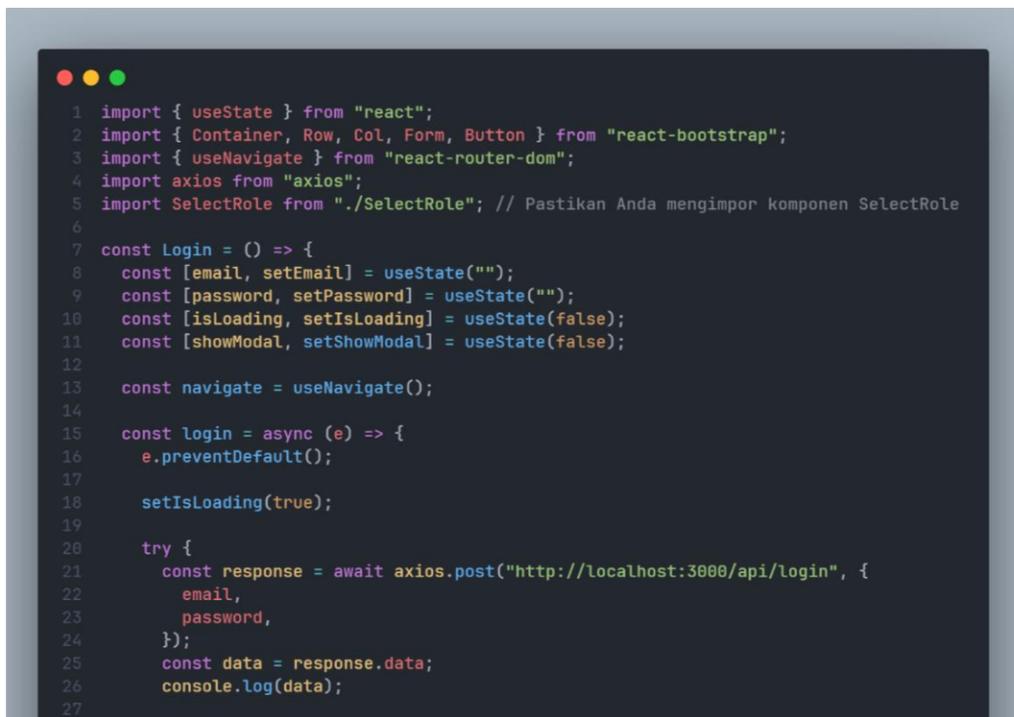
Kode di dibawah merupakan komponen "Register" dalam aplikasi *React* yang menggunakan *React Bootstrap* dan *axios*, memungkinkan pengguna untuk mendaftar dengan memasukkan email, *password*, dan konfirmasi *password*. Komponen ini menangani pemilihan peran pengguna berdasarkan query parameter dari *URL*, dan mencakup validasi form untuk memastikan *password* dan konfirmasi *password* cocok. Setelah pendaftaran berhasil, data dikirimkan ke *server*, dan pengguna diarahkan ke halaman login, yang dapat dilihat pada gambar 1 berikut.

```
1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Form, Button } from "react-bootstrap";
3 import { Link, useLocation, useNavigate } from "react-router-dom";
4 import axios from "axios";
5
6 const Register = () => {
7   const [email, setEmail] = useState("");
8   const [password, setPassword] = useState("");
9   const [confPassword, setConfPassword] = useState("");
10  const [isLoading, setIsLoading] = useState(false);
11  const [roleId, setRoleId] = useState("");
12
13  const navigate = useNavigate();
14  const location = useLocation();
15
16  useEffect(() => {
17    const queryParams = new URLSearchParams(location.search);
18    const roleIdFromQuery = queryParams.get("roleId");
19    if (roleIdFromQuery) {
20      setRoleId(roleIdFromQuery);
21    } else {
22      // Jika roleId tidak ditemukan, arahkan kembali ke halaman pemilihan
23      navigate("/select-role");
24    }
25  }, [location.search, navigate]);
26
27  const register = async (e) => {
28    e.preventDefault();
29
30    if (password !== confPassword) {
31      alert("Password dan Konfirmasi Password tidak cocok");
32      return;
33    }
34
35    setIsLoading(true);
36
37    try {
38      console.log({ email, password, confPassword, roleId });
39      const response = await axios.post("http://localhost:3000/api/register", {
40        email,
```

Gambar 20 Source Code Halaman Register

2. Halaman *Login*

Kode di bawah merupakan komponen "*Login*" dalam aplikasi *React* yang menggunakan *React Bootstrap* dan *axios*, memungkinkan pengguna untuk masuk dengan memasukkan *email* dan *password*. Komponen ini menangani autentikasi pengguna dengan mengirimkan data login ke *server*, menyimpan ``userId`` dan ``roleId`` di *localStorage* jika berhasil, dan mengarahkan pengguna ke halaman profil rental atau halaman utama berdasarkan ``roleId`` mereka. Komponen ini juga memiliki fitur untuk menampilkan modal ``SelectRole`` bagi pengguna yang belum memiliki akun dan ingin mendaftar, yang dapat dilihat pada gambar 2 berikut.

A screenshot of a code editor showing the source code for a Login component. The code is written in JavaScript and uses React hooks and axios. It includes imports for useState, useNavigate, and SelectRole. The Login component is a functional component that uses useState to manage email, password, isLoading, and showModal states. It uses useNavigate to handle navigation. The login function is an async function that prevents default behavior, sets isLoading to true, and sends a POST request to the server. The response is logged to the console.

```
1 import { useState } from "react";
2 import { Container, Row, Col, Form, Button } from "react-bootstrap";
3 import { useNavigate } from "react-router-dom";
4 import axios from "axios";
5 import SelectRole from "../SelectRole"; // Pastikan Anda mengimpor komponen SelectRole
6
7 const Login = () => {
8   const [email, setEmail] = useState("");
9   const [password, setPassword] = useState("");
10  const [isLoading, setIsLoading] = useState(false);
11  const [showModal, setShowModal] = useState(false);
12
13  const navigate = useNavigate();
14
15  const login = async (e) => {
16    e.preventDefault();
17
18    setIsLoading(true);
19
20    try {
21      const response = await axios.post("http://localhost:3000/api/login", {
22        email,
23        password,
24      });
25      const data = response.data;
26      console.log(data);
27    }
28  }
29 }
```

Gambar 21 *Source Code* Halaman *Login*

3. Halaman *Pesan*

Kode di bawah merupakan komponen "*CariRental*" dalam aplikasi *React* yang menggunakan *React Bootstrap* dan *axios*, memungkinkan pengguna untuk mencari rental motor berdasarkan kota yang dipilih. Komponen ini memuat daftar kota dan rental motor dari *server*, serta memperbarui daftar rental motor secara dinamis sesuai dengan kota yang

dipilih pengguna. Pengguna dapat melihat daftar rental motor dalam bentuk kartu (card) yang berisi foto, nama, dan alamat rental, serta tombol untuk memilih rental tersebut, yang dapat dilihat pada gambar 3 berikut.

```
1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Form, Card, Button } from "react-bootstrap";
3 import { Link, useParams, useNavigate } from "react-router-dom";
4 import axios from "axios";
5
6 const Rental = () => {
7   const [rental, setRental] = useState(null);
8   const [motors, setMotors] = useState([]);
9   const { rentalId } = useParams();
10
11   const userId = localStorage.getItem("userId");
12   const roleId = localStorage.getItem("roleId");
13
14   const navigate = useNavigate();
15
16   useEffect(() => {
17     // cek role
18     if (roleId !== 3) {
19       navigate("/");
20     }
21   }, []);
22
23   useEffect(() => {
24     getRentalByRentalId();
25     getMotorsByRentalId();
26   }, [rental, motors]);
27
28   const getRentalByRentalId = async () => {
29     try {
30       const response = await axios.get(
31         `http://localhost:3000/api/rental?rentalId=${rentalId}`
32       );
33       const data = response.data;
34       // console.log(data);
35
36       setRental(data.data);
37     } catch (error) {
38       console.error("Error :" + error.message);
39     }
40   };
41 }
```

Gambar 22 *Source Code* Halaman Pesan

4. Halaman Buat Pesanan

Komponen "Pesanan" ini memungkinkan pengguna untuk membuat pesanan rental motor dengan mengisi formulir yang mencakup detail seperti

tanggal, lokasi pengambilan dan pengembalian, serta jaminan. Komponen ini juga menampilkan ringkasan pesanan, termasuk informasi motor dan total harga berdasarkan durasi sewa yang dapat dilihat pada gambar 4 berikut.

```
1 import { useState, useEffect } from "react";
2 import {
3   Container,
4   Row,
5   Col,
6   Form,
7   Card,
8   Button,
9   FormControl,
10 } from "react-bootstrap";
11 import { Link, useNavigate } from "react-router-dom";
12 import axios from "axios";
13
14 const Pesanan = () => {
15   const [motor, setMotor] = useState(null);
16   const [rental, setRental] = useState(null);
17   const [rentalId, setRentalId] = useState("");
18   const [motorId, setMotorId] = useState("");
19   const [jaminans, setJaminans] = useState([]);
20   const [jaminanId, setJaminanId] = useState("1");
21   const [tanggalMulai, setTanggalMulai] = useState("");
22   const [tanggalSelesai, setTanggalSelesai] = useState("");
23   const [pilihanLokasiAmbil, setPilihanLokasiAmbil] = useState("rental");
24   const [pilihanLokasiKembali, setPilihanLokasiKembali] = useState("rental");
25   const [lokasiAmbil, setLokasiAmbil] = useState("rental");
26   const [lokasiKembali, setLokasiKembali] = useState("rental");
27   const [waktuAmbil, setWaktuAmbil] = useState("");
28   const [waktuKembali, setWaktuKembali] = useState("");
29   const [durasi, setDurasi] = useState("0");
30   const [totalHarga, setTotalHarga] = useState("0");
31   const [penyewaId, setPenyewaId] = useState("");
32
33   const userId = localStorage.getItem("userId");
34   const roleId = localStorage.getItem("roleId");
35
36   const navigate = useNavigate();
```

Gambar 23 *Source Code* Halaman Buat Pesanan

5. Halaman Detail Pesanan

Kode di bawah merupakan komponen "DetailPesanan" dalam aplikasi *React* yang menampilkan detail pesanan berdasarkan `pesananId` dari *URL*. Komponen ini menggunakan *React Bootstrap* untuk tata letak dan styling, serta *axios* untuk melakukan permintaan HTTP. Komponen ini

mengambil data pesanan dari *server* dan menampilkannya dalam berbagai format, termasuk informasi tentang rental, motor, jadwal, penyewa, jaminan, dan pembayaran. Jika pembayaran belum selesai, pengguna dapat membuat transaksi baru dengan menekan tombol "Bayar", yang akan membuka *URL* pembayaran yang relevan. Selain itu, komponen ini juga menyediakan tombol "Kembali" untuk navigasi kembali ke halaman sebelumnya, yang dapat dilihat pada gambar 5 berikut.

```
1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Button } from "react-bootstrap";
3 import { useParams, useNavigate } from "react-router-dom";
4 import axios from "axios";
5
6 const DetailPesanan = () => {
7   const { pesananId } = useParams();
8   const [pesanan, setPesanan] = useState(null);
9   const [urlPembayaran, setUrlPembayaran] = useState("");
10  const [isLoading, setIsLoading] = useState(false);
11
12  const userId = localStorage.getItem("userId");
13  const roleId = localStorage.getItem("roleId");
14
15  const navigate = useNavigate();
16
17  const getPesananByPesananId = async () => {
18    try {
19      const response = await axios.get(
20        `http://localhost:3000/api/pesanan?pesananId=${pesananId}`
21      );
22      const data = response.data;
23      console.log(data);
24
25      if (response.status === 200) {
26        setPesanan(data.data);
27        setUrlPembayaran(data.data.pembayaran.url);
28      }
29    } catch (error) {
30      console.error("Error : " + error.message);
31    }
32  };
```

Gambar 24 *Source Code* Detail Pesanan

6. Halaman *Dashboard* Profil

Kode ini merupakan komponen "Profil" yang memungkinkan pengguna untuk melihat dan memperbarui profil mereka, termasuk nama, nomor HP, dan lain lain. Data diambil dari *server* dan diperbarui menggunakan *FormData*, dengan validasi untuk memastikan hanya pengguna dengan `roleId` yang sesuai yang dapat mengakses halaman ini, yang dapat dilihat pada gambar 6-7 berikut.

```
1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Form, Button } from "react-bootstrap";
3 import axios from "axios";
4 import { useNavigate } from "react-router-dom";
5 import Sidebar from "../components/Sidebar";
6
7 const ProfilPenyewa = () => {
8   const [nama, setNama] = useState("");
9   const [noHp, setNoHp] = useState("");
10  const [foto, setFoto] = useState(null); // untuk kirim data
11  const [preview, setPreview] = useState(""); // untuk preview saja
12  const [isLoading, setIsLoading] = useState(false);
13  const userId = localStorage.getItem("userId");
14  const roleId = localStorage.getItem("roleId");
15
16  const navigate = useNavigate();
17
18  const getPenyewaById = async () => {
19    try {
20      // ambil data profil penyewa
21      const response = await axios.get(
22        `http://localhost:3000/api/penyewa?userId=${userId}`
23      );
24      const data = response.data;
25      console.log(data);
26
27      if (response.status === 200) {
28        setNama(data.data.nama || "");
29        setNoHp(data.data.noHp || "");

```

Gambar 25 Source Code Profil Penyewa

```
1 import { useState, useEffect } from "react";
2 import { Container, Col, Row, Form, Button } from "react-bootstrap";
3 import axios from "axios";
4 import { useNavigate } from "react-router-dom";
5 import Sidebar from "../components/Sidebar";
6
7 const ProfilRental = () => {
8   const [nama, setNama] = useState("");
9   const [noHp, setNoHp] = useState("");
10  const [kotas, setKotas] = useState([]);
11  const [kotaId, setKotaId] = useState("");
12  const [alamat, setAlamat] = useState("");
13  const [foto, setFoto] = useState(null); // untuk kirim data
14  const [preview, setPreview] = useState(""); // untuk preview saja
15
16  const [banks, setBanks] = useState([]);
17  const [bankId, setBankId] = useState("");
18  const [noRekening, setNoRekening] = useState("");
19
20  const [isLoading, setIsLoading] = useState(false);
21  const userId = localStorage.getItem("userId");
22  const roleId = localStorage.getItem("roleId");
23
24  const navigate = useNavigate();
25
```

Gambar 26 *Source Code* Profil Rental

7. Halaman *Dashboard* Pesanan

Kode ini adalah komponen "Pesanan" yang menampilkan daftar pesanan penyewa/rental dengan informasi motor, rental, tanggal, dan durasi. Data pesanan diambil dari *server* dan ditampilkan dalam format kartu, dengan opsi untuk melihat detail pesanan lebih lanjut jika ada, yang dapat dilihat pada gambar 8-9 berikut.

```

1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Card, Button } from "react-bootstrap";
3 import { Link, useParams } from "react-router-dom";
4 import axios from "axios";
5 import { useNavigate } from "react-router-dom";
6 import Sidebar from "../components/Sidebar";
7
8 const PesananPenyewa = () => {
9   const [pesanans, setPesanans] = useState([]);
10  const userId = localStorage.getItem("userId");
11  const roleId = localStorage.getItem("roleId");
12
13  const navigate = useNavigate();
14
15  const getPesanansByUserId = async () => {
16    try {
17      const response = await axios.get(
18        `http://localhost:3000/api/pesanan/penyewa?userId=${userId}`
19      );
20      const data = response.data;
21      console.log(data);
22
23      setPesanans(data.data);
24    } catch (error) {
25      console.error("Error :" + error.message);
26    }
27  };

```

Gambar 27 *Source Code* Halaman *Dashboard* Pesanan Penyewa

```

1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Card, Button } from "react-bootstrap";
3 import { Link, useNavigate } from "react-router-dom";
4 import axios from "axios";
5 import Sidebar from "../components/Sidebar";
6
7 const PesananRental = () => {
8   const [pesanans, setPesanans] = useState([]);
9   const userId = localStorage.getItem("userId");
10  const roleId = localStorage.getItem("roleId");
11
12  const navigate = useNavigate();
13
14  const getPesanansByRentalId = async () => {
15    try {
16      const response = await axios.get(
17        `http://localhost:3000/api/pesanan/rental?userId=${userId}`
18      );
19      const data = response.data;
20      console.log(data);
21
22      setPesanans(data.data);
23    } catch (error) {
24      console.error("Error : " + error.message);
25      console.log(error.response.data.message);
26    }
27  };

```

Gambar 28 *Source Code* Halaman *Dashboard* Pesanan Rental

8. Halaman *Dashboard* Motor

Komponen "Motor" menampilkan daftar motor yang tersedia dengan informasi nama, harga, dan foto dalam format kartu. Pengguna dengan `roleId` 2 dapat melihat daftar motor dan menambah motor baru, serta melihat detail setiap motor jika ada, yang dapat dilihat pada gambar 10 berikut.

```

1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Card, Button } from "react-bootstrap";
3 import { Link, useParams } from "react-router-dom";
4 import axios from "axios";
5 import { useNavigate } from "react-router-dom";
6 import Sidebar from "../components/Sidebar";
7
8 const ListMotor = () => {
9   const [motors, setMotors] = useState([]);
10  const userId = localStorage.getItem("userId");
11  const roleId = localStorage.getItem("roleId");
12
13  const navigate = useNavigate();
14
15  const getMotorsByUserId = async () => {
16    try {
17      const response = await axios.get(
18        `http://localhost:3000/api/motors?userId=${userId}`
19      );
20      const data = response.data;
21      console.log(data);
22
23      setMotors(data.data);
24    } catch (error) {
25      console.error("Error : " + error.message);
26    }
27  };

```

Gambar 29 Halaman *Dashboard* Motor

9. Halaman *Dashboard* Laporan Penjualan

Kode di atas merupakan komponen "LaporanPenjualan" dalam aplikasi *React* yang menampilkan laporan penjualan motor berdasarkan bulan dan tahun yang dipilih pengguna. Komponen ini menggunakan *React Bootstrap* untuk tata letak dan styling, *axios* untuk mengambil data laporan dari *server*, dan memeriksa peran pengguna untuk memastikan hanya pengguna dengan `roleId` tertentu yang dapat mengakses halaman ini, yang dapat dilihat pada gambar 11 berikut.

```

1 import { useState, useEffect } from "react";
2 import { Container, Row, Col, Button, Form, Table } from "react-bootstrap";
3 import axios from "axios";
4 import { useNavigate } from "react-router-dom";
5 import Sidebar from "../components/Sidebar";
6
7 const LaporanPenjualan = () => {
8   // Mendapatkan bulan dan tahun saat ini
9   const now = new Date();
10  const currentMonth = now.getMonth() + 1; // getMonth() returns month from 0-11
11  const currentYear = now.getFullYear();
12
13  const [bulan, setBulan] = useState(currentMonth);
14  const [tahun, setTahun] = useState(currentYear);
15  const [motors, setMotors] = useState([]);
16  const [totalKeseluruhan, setTotalKeseluruhan] = useState(0);
17  const userId = localStorage.getItem("userId");
18  const roleId = localStorage.getItem("roleId");
19
20  const navigate = useNavigate();
21
22  const getLaporanPenjualan = async () => {
23    try {
24      const response = await axios.get(
25        `http://localhost:3000/api/pesanan/laporan?userId=${userId}&bulan=${bulan}&tahun=${tahun}`
26      );
27      const data = response.data;
28      console.log(data);
29
30      setMotors(data.data.penjualanPerMotor);
31      setTotalKeseluruhan(data.data.totalKeseluruhan);
32    } catch (error) {
33      console.error("Error : " + error.message);
34      alert(error.response.data.message);
35    }
36  };

```

Gambar 30 Halaman *Dashboard* Laporan Penjualan

D. Source Code Backend

Berikut penjelasan mengenai *Source Code backend website* MotorIn, yang meliputi komponen dan fitur utama:

1. Model Database

Kode dibawah merupakan *prisma* schema yang mendefinisikan struktur database untuk aplikasi MotorIn, mencakup model untuk pengguna, peran, rental, motor, pesanan, dan pembayaran dengan relasi antar tabel yang sesuai. Setiap model memiliki atribut dan relasi yang dirancang untuk mengelola informasi pengguna, rental motor, transaksi, serta detail lainnya dalam aplikasi pemesanan rental motor, yang dapat dilihat pada gambar 12 berikut.

```

1 generator client {
2   provider = "prisma-client-js"
3 }
4
5 datasource db {
6   provider = "postgresql"
7   url      = env("DATABASE_URL")
8 }
9
10 model Role {
11   id    Int    @id
12   nama  String
13
14   // relasi : One-to-Many
15   user  User[]
16
17   @@map("role")
18 }
19
20 model User {
21   id      String @id
22   email   String @unique
23   password String
24   roleId  Int    @map("role_id")
25
26   // relasi : Many-to-One
27   role Role @relation(fields: [roleId], references: [id])
28
29   // relasi: One-to-One
30   admin  Admin?
31   rental Rental?
32   penyewa Penyewa?
33
34   @@map("user")
35 }

```

Gambar 31 *Source Code Model Database*

2. Controller Register

Kode di bawah adalah fungsi `register` yang menangani pendaftaran pengguna baru dengan memvalidasi data masukan, memeriksa eksistensi pengguna, dan membuat entitas pengguna serta profil terkait berdasarkan peran. Fungsi ini mengenkripsi *password*, membuat ID unik untuk pengguna dan profil, dan menyimpan data ke database menggunakan *Prisma*, kemudian mengirimkan respons sukses atau error kepada client, yang dapat dilihat pada gambar 13 berikut.

```

1 export const register = async (req, res) => {
2   try {
3     // ambil data dari req.body
4     let { email, password, confPassword, roleId } = req.body;
5
6     // validasi: jika data tidak ada
7     if (!email || !password || !confPassword || !roleId) {
8       return res.status(400).json({
9         message: "email, password, confirm password dan roleId harus diisi",
10      });
11    }
12
13    // parsing data
14    roleId = parseInt(roleId);
15
16    // validasi: jika data yang diparsing tidak valid
17    if (isNaN(roleId)) {
18      return res.status(400).json({
19        message: "data (roleId) tidak valid",
20      });
21    }
22
23    // cek role berdasarkan roleId
24    const role = await prisma.role.findUnique({
25      where: {
26        id: roleId,
27      },
28    });

```

Gambar 32 *Source Code* Controller Register

3. Controller Login

Kode di bawah adalah fungsi `login` yang menangani proses autentikasi pengguna dengan memvalidasi data masuk, memeriksa keberadaan pengguna, dan membandingkan *password* dengan yang tersimpan di database. Jika autentikasi berhasil, fungsi ini membuat token akses JWT dan mengirimkan respons sukses bersama dengan data pengguna, yang dapat dilihat pada gambar 14 berikut.

```
1 export const login = async (req, res) => {
2   try {
3     // ambil data dari req.body
4     let { email, password } = req.body;
5
6     // validasi: jika data yang dikirim kurang
7     if (!email || !password) {
8       return res.status(400).json({
9         message: "data (email, password) tidak ada",
10      });
11    }
12
13    // cek user berdasarkan email dan roleId
14    const user = await prisma.user.findUnique({
15      where: {
16        email,
17      },
18    });
19
20    // validasi: jika user tidak ada
21    if (!user) {
22      return res.status(404).json({
23        message: "user tidak ada",
24      });
25    }
26  }
27 }
```

Gambar 33 *Source Code* Controller Login

4. Controller Pesanan

Kode di bawah adalah fungsi `createPesanan` yang menangani pembuatan pesanan baru dengan memvalidasi data input, menghitung harga berdasarkan durasi sewa dan lokasi, serta memastikan semua entitas terkait seperti motor, penyewa, dan jaminan valid. Fungsi ini juga membuat entitas pesanan dan pembayaran di database, kemudian mengirimkan respons sukses atau error sesuai hasil operasional, yang dapat dilihat pada gambar 15 berikut.

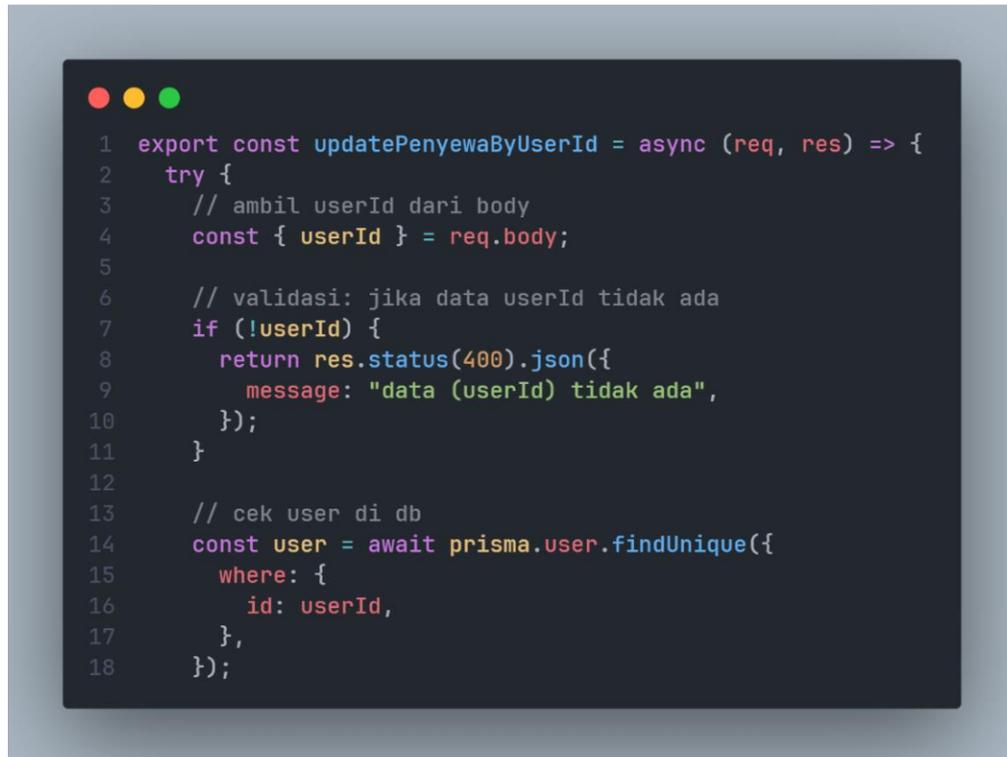
```
1 export const createPesanan = async (req, res) => {
2   try {
3     // ambil data pesanan
4     const {
5       tanggalMulai,
6       tanggalSelesai,
7       lokasiAmbil,
8       lokasiKembali,
9       waktuAmbil,
10      waktuKembali,
11      motorId,
12      penyewaId,
13    } = req.body;
14    const jaminanId = parseInt(req.body.jaminanId);
15
16    // validasi: jika data yang diparsing tidak valid
17    if (isNaN(jaminanId)) {
18      return res.status(400).json({
19        message: "data (jaminanId) tidak valid",
20      });
21    }
22
23    // validasi: jika data tidak ada
24    if (
25      !tanggalMulai ||
26      !tanggalSelesai ||
27      !lokasiAmbil ||
28      !lokasiKembali ||
29      !waktuAmbil ||
30      !waktuKembali ||
31      !jaminanId ||
```

Gambar 34 *Source Code* Controller Pesan

5. Controller Profil

Kode di bawah adalah fungsi `profil` yang memperbarui profil penyewa berdasarkan `userId`. Fungsi ini memvalidasi data input, memeriksa eksistensi user dan penyewa di database, serta menangani pembaruan informasi seperti nama, nomor HP, dan foto profil dengan mengupload file baru jika ada, dan menghapus foto lama dari penyimpanan. Fungsi ini

mengirimkan respons sukses atau error sesuai hasil operasional, yang dapat dilihat pada gambar 16-17 berikut.



```
1 export const updatePenyewaByUserId = async (req, res) => {
2   try {
3     // ambil userId dari body
4     const { userId } = req.body;
5
6     // validasi: jika data userId tidak ada
7     if (!userId) {
8       return res.status(400).json({
9         message: "data (userId) tidak ada",
10      });
11    }
12
13    // cek user di db
14    const user = await prisma.user.findUnique({
15      where: {
16        id: userId,
17      },
18    });
```

Gambar 35 *Source Code* Controller Profil Penyewa

```
1 export const updateRentalByUserId = async (req, res) => {
2   try {
3     // ambil userId dari body
4     const { userId } = req.body;
5
6     // validasi: jika data userId tidak ada
7     if (!userId) {
8       return res.status(400).json({
9         message: "data (userId) tidak ada",
10      });
11    }
12
13    // cek user di db
14    const user = await prisma.user.findUnique({
15      where: {
16        id: userId,
17      },
18    });
```

Gambar 36 *Source Code* Controller Profil Rental

6. Controller Motor

Kode di bawah terdiri dari tiga fungsi untuk mengelola data motor dalam aplikasi: `addMotorByUserId` menambahkan motor baru setelah memvalidasi data pengguna dan unggahan foto, `getMotors` mengambil daftar motor berdasarkan `userId` atau `rentalId` untuk menampilkan motor yang terkait dengan rental tertentu, dan `getMotorByMotorId` mengambil detail motor berdasarkan `motorId`, memastikan keberadaan motor sebelum mengirimkan data atau respons error. Ketiga fungsi ini menangani validasi, interaksi dengan database menggunakan *Prisma*, dan pengelolaan file dengan utilitas penyimpanan Supabase, yang dapat dilihat pada gambar 18 berikut.



```
1 export const addMotorByUserId = async (req, res) => {
2   try {
3     // ambil data userId
4     const { userId } = req.body;
5
6     // validasi: jika data userId tidak ada
7     if (!userId) {
8       return res.status(400).json({
9         message: "data (userId) tidak ada",
10      });
11    }
12
13    // cek rental di db
14    const rental = await prisma.rental.findUnique({
15      where: {
16        userId,
17      },
18    });
```

Gambar 37 *Source Code* Controller Motor

7. Controller Laporan Penjualan

Kode dibawah merupakan fungsi `getPesananByStatusPembayaran` mengambil data pesanan berdasarkan `userId`, bulan, dan tahun yang diberikan melalui query parameter. Fungsi ini memvalidasi input, menghitung rentang tanggal untuk bulan yang ditentukan, dan mencari data pesanan yang telah dibayar dengan status "settlement". Selanjutnya, fungsi ini menghitung total pendapatan per motor dan total keseluruhan, lalu mengirimkan data tersebut dalam respons sukses atau respons error jika terjadi masalah, yang dapat dilihat pada gambar 19 berikut.

```
1 export const getPesananByStatusPembayaran = async (req, res) => {
2   try {
3     // Ambil data (userId, bulan, tahun) dari query
4     let { userId, bulan, tahun } = req.query;
5
6     // Validasi input
7     if (!userId || !bulan || !tahun) {
8       return res.status(400).json({
9         message: "data (userId, bulan, tahun) tidak ada",
10      });
11    }
12
13    // Parsing bulan dan tahun ke integer
14    bulan = parseInt(bulan);
15    tahun = parseInt(tahun);
16
17    // Hitung tanggal awal dan akhir bulan
18    const startDate = new Date(tahun, bulan - 1, 1);
19    const endDate = new Date(tahun, bulan, 0, 23, 59, 59);
```

Gambar 38 *Source Code* Controller Laporan Penjualan

Lampiran 5 Sertifikat HKI yang Terbit

E.

REPUBLIC INDONESIA KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA	
SURAT PENCATATAN CIPTAAN	
Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:	
Nomor dan tanggal permohonan	: EC00202471355, 25 Juli 2024
Pencipta	
Nama	: Muhammad Dani Haikal, M. Nishom, S.Kom., M.Kom. dkk
Alamat	: Jalan Kerti, RT/RW 01/12, No 17, Desa Suradadi, Kecamatan Suradadi, Kabupaten Tegal, Propinsi Jawa Tengah 52182, Surodadi, Tegal, Jawa Tengah, 52182
Kewarganegaraan	: Indonesia
Pemegang Hak Cipta	
Nama	: Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama
Alamat	: Jalan Mataram No. 9, Pesurungan Lor, Kecamatan Margadana 52142, Margadana, Tegal, Jawa Tengah 52142
Kewarganegaraan	: Indonesia
Jenis Ciptaan	: Program Komputer
Judul Ciptaan	: Rancang Bangun Sistem Informasi Pemesanan Rental Motor Berbasis Website
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	: 25 Juli 2024, di Tegal
Jangka waktu perlindungan	: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan	: 000646706
adalah benar berdasarkan keterangan yang diberikan oleh Pemohon. Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.	
	a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL u.b. Direktur Hak Cipta dan Desain Industri
	 IGNATIUS M.T. SILALAH NIP. 196812301996031001
Disclaimer: Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.	

Lampiran 6 Lembar Bimbingan

F.



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Muhammad Dani Haikal
 Nim : 20090031
 No. Ponsel : 08984182517
 Judul TA : Rancang Bangun Sistem Informasi Pemesanan Rental Motor Berbasis Website Menggunakan Metode Haversine Formula
 Dosen Pembimbing I : M. Nishom, M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	26/03 - 2024	Judul, Konsep	- Ok. Lanjutkan pembuatan use case Diagram	
2.	3/04 - 2024	- Metode	- Tentukan metode pengembangan sistem yang akan digunakan. - Pelajari tentang metode Haversine. Simulasikan dengan perhitungan manual contoh: excel.	
3.	22/04 - 2024	- Perancangan dan pembangunan sistem	- Mulai rancang sistem - Selanjutnya, bangun sistem sesuai perancangan.	
4.	15/05 - 2024	- Back-end	- Selesaikan pembuatan service untuk seluruh fungsi aplikasi - Mulai integrasikan dengan elemen.	



SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
5.	4/07-2024	- sistem	- Mulai bangun UI dan implementasi proses bisnis ke dalam sistem. pastikan semua UI telah dibangun sesuai kebutuhan!	
6.	18/07-2024	- sistem	- Selesaikan semua fungsi utama aplikasi, pastikan semua fungsi telah diuji dengan baik.	
		- Laporan	- Mulai susun Laporan BAB I-II	
7	23/07-2024	- Ujian - Laporan	- Daftar kan Hki - Susun Laporan BAB III	
8.	24/07-2024	- Laporan	- Tujuan penelitian = Dibuat lebih singkat dan jelas. - Daftar pustaka: Min 15 pustaka - Format Laporan disesuaikan dengan pedoman penulisan skripsi.	



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing

Tegal, 29 Juli 2024
Dosen Pembimbing I

M. Nishom, M.Kom.
NIPY. 09.017.337



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Muhammad Dani Haikal
Nim : 20090031
No. Ponsel : 08984182517
Judul TA : Rancang Bangun Sistem Informasi Pemesanan Rental Motor Berbasis Website Menggunakan Metode Haversine Formula
Dosen Pembimbing II : Dairoh, M.Sc.

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1	17/04/2021	Pemeriksaan Usecase pada aplikasi.	Memperbaiki re usability case	f
2	6/5/2021	Usecase	Usecase Aze sequence diteliti	f
3	20/5/2021	Usecase	Aze sequence program Aplikasi selesai	f



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
4	02 Juni 2024	Aplikasi	- menyelesaikan fitur pesanan - menambahkan dashboard	
5	17 Juni 2024	Aplikasi	- menambahkan history pesanan - menambahkan laporan	
6	10 Juli 2024	Aplikasi	pengecekan seluruh fungsi aplikasi (selesaikan aplikasi)	
7	19 Juli 2024	Laporan	ACC Aplikasi Perbaiki Laporan BAB 2	
8	21 Juli 2024	Laporan	ACC Laporan	



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing

Tegal, 2024
Dosen Pembimbing II

Dairoh, M.Sc.
NIPY. 04.014.178