

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

**SKRIPSI**

***E-MARKETPLACE* PENYEWAAN MOBIL  
(STUDI KASUS : KOTA PALEMBANG)**



**Diajukan Oleh :  
NABILA ELFA RIENI  
021190014**

**Untuk Memenuhi Sebagian dari Syarat  
Mencapai Gelar Sarjana Komputer**

**PALEMBANG  
2023**

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

**SKRIPSI**

***E-MARKETPLACE* PENYEWAAN MOBIL  
(STUDI KASUS : KOTA PALEMBANG)**



**Diajukan Oleh :  
NABILA ELFA RIENI  
021190014**

**Untuk Memenuhi Sebagian dari Syarat  
Mencapai Gelar Sarjana Komputer**

**PALEMBANG  
2023**

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

---

**HALAMAN PENGESAHAN PEMBIMBING SKRIPSI**

**NAMA : NABILA ELFA RIENI**  
**NOMOR POKOK : 021190014**  
**PROGRAM STUDI : SISTEM INFORMASI**  
**JENJANG PENDIDIKAN : STRATA SATU**  
**JUDUL : E-MARKETPLACE PENYEWAAN MOBIL  
(STUDI KASUS : KOTA PALEMBANG)**

**Tanggal : 23 Agustus 2023**  
**Pembimbing**

**Mengetahui,**  
**Rektor**

**Yarza Aprizal, S.Kom., M.Kom.**  
**NIDN : 0212049302**

**Benedictus Effendi, S.T., M.T.**  
**NIP : 09.PCT.13**

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI  
INSTITUT TEKNOLOGI DAN BISNIS PALCOMTECH**

---

**HALAMAN PENGESAHAN PENGUJI SKRIPSI**

**NAMA : NABILA ELFA RIENI**  
**NOMOR POKOK : 021190014**  
**PROGRAM STUDI : SISTEM INFORMASI**  
**JENJANG PENDIDIKAN : STRATA SATU**  
**JUDUL : E-MARKETPLACE PENYEWAAN MOBIL**  
**(STUDI KASUS : KOTA PALEMBANG)**

**Tanggal : 23 Agustus 2023**  
**Penguji 1**

**Tanggal : 21 Agustus 2023**  
**Penguji 2**

**Eka Hartati, S.Kom., M.Kom.**  
**NIDN : 0226119002**

**Jaka Purnama, S.Kom., M.Kom.**  
**NIDN : 0219089401**

**Menyetujui,**  
**Rektor**

**Benedictus Effendi, S.T., M.T.**  
**NIP : 09.PCT.13**

**MOTTO :**

- ✓ *“Keluarga adalah yang utama”*
- ✓ *“Perjuangkan dan pegang erat apa yang digenggam”*
- ✓ *“Jangan mengharapkan kebaikan orang lain, tetapi jadilah orang baik”*
- ✓ *“Berbuat baik kepada seluruh makhluk hidup tanpa mengharapkan balasan”*
- ✓ *“Ikhlaskan dalam menjalankan kehidupan sehari – hari”*

**“Allah tidak membebani seorang melainkan sesuai dengan kesanggupannya”  
( QS. Al-Baqarah:286 )**

**“Karena, sesungguhnya sesudah kesulitan itu ada kemudahan”  
( QS. Al-Insyirah:5 )**

**“ Sesungguhnya Allah tidak akan mengubah keadaan suatu kaum, sebelum kaum itu sendiri mengubah apa yang ada pada diri mereka”  
( QS. Ar-Ra’d:11 )**

**Kepersembahkan Kepada :**

- *Papa dan Mama Tercinta*
- *Saudara – Saudaraku Tersayang*
- *Temanku yang Baik Hati*
- *Para Pendidik yang Kuhormati*

## KATA PENGANTAR

Alhamdulillah puji dan syukur peneliti panjatkan atas kehadiran Allah SWT yang telah melimpahkan rahmat, hidayah, dan karunia-Nya sehingga peneliti dapat menyelesaikan Laporan Skripsi yang peneliti buat ini.

Laporan ini disusun untuk memenuhi salah satu persyaratan Menyelesaikan Program Studi Sistem Informasi Program Sarjana Institut Teknologi dan Bisnis Palcomtech.

Peneliti melakukan kegiatan Skripsi dengan judul “***E-MARKETPLACE PENYEWAAN MOBIL (STUDI KASUS : KOTA PALEMBANG)***”. Tentunya dalam proses penyusunan Laporan Skripsi ini, peneliti banyak menghadapi suatu kendala namun itu semua tidak berarti karena adanya bantuan, dukungan, bimbingan serta informasi dari banyak pihak sehingga pada akhirnya peneliti dapat menyelesaikan Laporan Skripsi ini dengan lancar. Untuk itu pada kesempatan ini peneliti ingin mengucapkan banyak terimakasih kepada :

- Kedua Orang Tuaku MGS. Umar dan Nyayu Sri Herlinda yang tidak pernah lelah memberikan kasih sayang, semangat, inspirasi, perhatian dan tentunya do'a yang tidak pernah putus selama ini.
- Kedua saudaraku Mgs. M. Aufa Lutfhi Mahdi, S.T. dan Mgs. Azhari Refani yang telah memberikan dukungan selama ini.
- Bapak Benedictus Effendi, S.T., M.T. selaku Rektor Institut Teknologi dan Bisnis Palcomtech.
- Ibu Adelin, S.T., M.Kom. selaku Wakil Ketua I Program Studi Sistem Informasi Institut Teknologi dan Bisnis Palcomtech.
- Ibu Dini Hari Pertiwi, S.Kom., M.Kom. selaku Ketua Program Studi Sistem Informasi Institut Teknologi dan Bisnis Palcomtech.
- Bapak Yarza Aprizal, S.Kom., M.Kom. selaku Dosen Pembimbing Skripsi.
- Ibu Meidyan Permata Putri, S.Kom., M.Kom. selaku Dosen Pembimbing Akademik.

- PT. Sekayu Sersan Sekate, CV. Palembang Berlian Abadi Jaya, CV. Sriwijaya XI, CV. Sriwijaya Maju Gemilang, CV. Anugrah Esa Utama yang telah bersedia menjadi tempat penelitian.
- Sahabatku Oktarina, Abdul Rahman Ariga, Yulita, A.Md, Farm. Dessy Erisa, S.Farm. dan Diah Arika Putri, S.Farm. yang telah banyak membantu memberikan semangat dan dukungan sekaligus menjadi inspirator dalam menjalani kehidupan.

Semoga Allah SWT membalas semua kebaikan yang telah kalian lakukan serta semoga Laporan Skripsi ini dapat memberikan kontribusi yang positif bagi perusahaan, pembaca dan juga peneliti sendiri.

Aamiin Yaa Rabbal ‘Alamin.

Palembang, 14 Agustus 2023

Peneliti

## DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	<b>ii</b>
<b>HALAMAN PENGESAHAN PEMBIMBING LAPORAN SKRIPSI</b> .....	<b>ii</b>
<b>HALAMAN PENGESAHAN PENGUJI LAPORAN SKRIPSI</b> .....	<b>iii</b>
<b>MOTTO DAN PERSEMBAHAN</b> .....	<b>iv</b>
<b>KATA PENGANTAR</b> .....	<b>v</b>
<b>DAFTAR ISI</b> .....	<b>vii</b>
<b>DAFTAR GAMBAR</b> .....	<b>x</b>
<b>DAFTAR TABEL</b> .....	<b>xiii</b>
<b>DAFTAR LAMPIRAN</b> .....	<b>xiv</b>
<b>ABSTRACT</b> .....	<b>xv</b>
<b>ABSTRAK</b> .....	<b>xvi</b>
<b>BAB I PENDAHULUAN</b>	
1.1. Latar Belakang Penelitian .....	1
1.2. Rumusan Masalah .....	3
1.3. Batasan Masalah .....	3
1.4. Tujuan Penelitian .....	4
1.5. Manfaat Penelitian .....	5
1.5.1. Manfaat Bagi Peneliti .....	5
1.5.2. Manfaat Bagi Perusahaan .....	5
1.5.3. Manfaat Bagi Akademik .....	5
1.6. Sistematika Penelitian .....	5
<b>BAB II GAMBARAN UMUM PERUSAHAAN</b>	
2.1. Profil Perusahaan PT. Sekayu Serasan Sekate .....	7
2.1.1. Sejarah Perusahaan .....	7
2.1.2. Visi dan Misi .....	7
2.1.3. Struktur Organisasi .....	8
2.1.4. Tugas dan Wewenang .....	9
2.2. Profil Perusahaan CV. Palembang Berlian Abadi Jaya .....	11
2.2.1. Sejarah Perusahaan .....	11



2.2.2.	Visi dan Misi .....	12
2.2.3.	Struktur Organisasi .....	12
2.2.4.	Tugas dan Wewenang .....	13
2.3.	Profil Perusahaan CV. Sriwijaya XI .....	15
2.3.1.	Sejarah Perusahaan .....	15
2.3.2.	Visi dan Misi .....	16
2.3.3.	Struktur Organisasi .....	16
2.3.4.	Tugas dan Wewenang .....	17
2.4.	Profil Perusahaan CV. Sriwijaya Maju Gemilang .....	20
2.4.1.	Sejarah Perusahaan .....	20
2.4.2.	Visi dan Misi .....	21
2.4.3.	Struktur Organisasi .....	21
2.4.4.	Tugas dan Wewenang .....	22
2.5.	Profil Perusahaan CV. Anugrah Esa Utama .....	24
2.5.1.	Sejarah Perusahaan .....	24
2.5.2.	Visi dan Misi .....	25
2.5.3.	Struktur Organisasi .....	25
2.5.4.	Tugas dan Wewenang .....	25
<b>BAB III TINJAUAN PUSTAKA</b>		
3.1.	Teori Pendukung .....	29
3.1.1.	Penyewaan .....	29
3.1.2.	<i>Marketplace</i> .....	29
3.1.3.	<i>Business to Business (B2B)</i> .....	30
3.1.4.	<i>Database</i> .....	31
3.2.	Penelitian Terdahulu .....	31
3.3.	Kerangka Pemikiran .....	34
<b>BAB IV METODE PENELITIAN</b>		
4.1.	Lokasi dan Waktu Penelitian .....	36
4.1.1.	Lokasi .....	36
4.1.3.	Waktu Penelitian .....	36
4.2.	Jenis Data .....	37

4.2.2.	Data Primer .....	37
4.2.3.	Data Sekunder .....	38
4.3.	Teknik Pengumpulan Data .....	38
4.3.1.	Wawancara .....	38
4.3.2.	Pengamatan (Observasi) .....	39
4.3.3.	Studi Pustaka .....	39
4.4.	Alat dan Teknik Pengembangan Sistem .....	39
4.4.1.	Alat Pengembangan Sistem .....	39
4.4.1.1.	<i>Unified Modeling Language (UML)</i> .....	39
4.5.2.	Teknik Pengembangan Sistem .....	44
4.5.	Alat dan Teknik Pengujian .....	46
<b>BAB V HASIL DAN PEMBAHASAN</b>		
5.1.	Hasil Penelitian .....	48
5.1.1.	Rencana Kebutuhan ( <i>Requirement Planning</i> ) .....	48
5.1.2.	Desain Sistem ( <i>Design System</i> ) .....	49
5.1.2.1.	<i>Use Case Diagram</i> .....	49
5.1.2.2.	<i>Activity Diagram</i> .....	53
5.1.2.3.	<i>Class Diagram</i> .....	56
5.1.2.4.	<i>Squence Diagram</i> .....	57
5.1.3.	Implementasi ( <i>Implementation</i> ) .....	60
5.1.3.1.	Desain Tabel .....	60
5.1.3.2.	Desain Website .....	67
5.1.3.3.	Desain Antarmuka .....	84
5.1.3.4.	Alat dan teknik pengujian .....	99
5.1.3.5.	Pengenalan <i>E-Marketplace</i> Penyewaan Mobil Riemera .....	158
<b>BAB VI KESIMPULAN DAN SARAN</b>		
6.1.	Kesimpulan .....	160
6.2.	Saran .....	160
<b>DAFTAR PUSTAKA .....</b>		<b>xvii</b>
<b>HALAMAN LAMPIRAN .....</b>		<b>xviii</b>

## DAFTAR GAMBAR

Gambar 3.1 Kerangka Pemikiran.....	13
Gambar 4.1 Tahapan Metode <i>Rapid Application Development (RAD)</i> .....	22
Gambar 5.1 <i>Use Case Diagram</i> Admin.....	29
Gambar 5.2 <i>Use Case Activity Diagram</i> Penyedia.....	30
Gambar 5.3 <i>Use Case Activity Diagram</i> Penyewa.....	31
Gambar 5.4 <i>Activity Diagram</i> Admin.....	32
Gambar 5.5 <i>Activity Diagram</i> Penyedia.....	33
Gambar 5.6 <i>Activity Diagram</i> Penyewa.....	34
Gambar 5.7 <i>Class Diagram E-Marketplace</i> Penyewaan Mobil.....	35
Gambar 5.8 <i>Squence Diagram</i> Admin.....	36
Gambar 5.9 <i>Squence Diagram</i> Penyedia.....	37
Gambar 5.10 <i>Squence Diagram</i> Penyewa.....	38
Gambar 5.11 Desain Halaman <i>Dashboard</i> Awal.....	47
Gambar 5.12 Desain Halaman Data Mobil.....	48
Gambar 5.13 Desain Halaman <i>Login</i> .....	49
Gambar 5.14 Desain Halaman Daftar Akun.....	49
Gambar 5.15 Desain Halaman Daftar Penyedia.....	50
Gambar 5.16 Desain Halaman Daftar Penyewa.....	51
Gambar 5.17 Desain Halaman <i>Dashboard</i> Admin.....	52
Gambar 5.18 Desain Halaman <i>Slider</i> .....	52
Gambar 5.19 Desain Halaman Kendaraan Pada Admin.....	53
Gambar 5.20 Desain Halaman Pembayaran Pada Admin.....	54
Gambar 5.21 Desain Halaman Penyewaan Pada Admin.....	54
Gambar 5.22 Desain Halaman Pencairan Pada Admin.....	55
Gambar 5.23 Desain Halaman Data Penyedia Pada Admin.....	55
Gambar 5.24 Desain Halaman Data Penyewa Pada Admin.....	56
Gambar 5.25 Desain Halaman <i>Dashboard</i> Penyedia.....	56
Gambar 5.26 Desain Halaman Kendaraan Pada Penyedia.....	57
Gambar 5.27 Desain Halaman <i>Input</i> Kendaraan Pada Penyedia.....	58

Gambar 5.28 Desain Halaman Data Pembayaran Pada Penyedia.....	59
Gambar 5.29 Desain Halaman Data Penyewaan Pada Penyedia.....	59
Gambar 5.30 Desain Halaman Data Pencairan Pada Penyedia.....	60
Gambar 5.31 Desain Halaman <i>Dashboard</i> Penyewa.....	61
Gambar 5.32 Desain Halaman Data Pembayaran Pada Penyewa.....	61
Gambar 5.33 Desain Halaman <i>Form</i> Pembayaran.....	62
Gambar 5.34 Desain Halaman Data Penyewaan.....	62
Gambar 5.35 Desain Ubah <i>Password</i> .....	63
Gambar 5.36 Desain <i>Logout</i> .....	63
Gambar 5.37 Hasil Desain Halaman <i>Dashboard</i> Awal.....	64
Gambar 5.38 Hasil Desain Halaman Data Mobil.....	65
Gambar 5.39 Hasil Desain Halaman <i>Login</i> .....	66
Gambar 5.40 Hasil Desain Halaman Daftar Akun.....	66
Gambar 5.41 Hasil Desain Halaman Daftar Penyedia.....	67
Gambar 5.42 Hasil Desain Halaman Daftar Penyewa.....	67
Gambar 5.43 Hasil Desain Halaman <i>Dashboard</i> Admin.....	68
Gambar 5.44 Hasil Desain Halaman <i>Slider</i> .....	68
Gambar 5.45 Hasil Desain Halaman Kendaraan Pada Admin.....	69
Gambar 5.46 Hasil Desain Halaman Pembayaran Pada Admin.....	69
Gambar 5.47 Hasil Desain Halaman Penyewaan Pada Admin.....	70
Gambar 5.48 Hasil Desain Halaman Pencairan Pada Admin.....	70
Gambar 5.49 Hasil Desain Halaman Data Penyedia.....	71
Gambar 5.50 Hasil Desain Halaman Data Penyewa.....	71
Gambar 5.51 Hasil Desain Halaman <i>Dashboard</i> Penyedia.....	72
Gambar 5.52 Hasil Desain Halaman Kendaraan Pada Penyedia.....	72
Gambar 5.53 Hasil Desain Halaman <i>Input</i> Kendaraan Pada Penyedia.....	73
Gambar 5.54 Hasil Desain Halaman Data Pembayaran Pada Penyedia.....	74
Gambar 5.55 Hasil Desain Halaman Data Pencairan Pada Penyedia.....	74
Gambar 5.56 Hasil Desain Halaman <i>Dashboard</i> Penyewa.....	75
Gambar 5.57 Hasil Desain Halaman Data Pembayaran Pada Penyewa.....	76
Gambar 5.58 Hasil Desain Halaman <i>Form</i> Pembayaran.....	77

Gambar 5.59 Hasil Desain Halaman Data Penyewaan.....	77
Gambar 5.60 Hasil Desain Ubah <i>Password</i> .....	78
Gambar 5.61 Hasil Desain <i>Logout</i> .....	78
Gambar 5.62 <i>Source Code</i> Fungsi Pencairan Kendaraan.....	80
Gambar 5.63 <i>Flowgraph</i> Fungsi Pencairan Kendaraan.....	80
Gambar 5.64 <i>Source Code</i> Pengujian <i>Test</i> Fungsi Pencairan Kendaraan.....	82
Gambar 5.65 Hasil Pengujian <i>Test</i> Fungsi Pencairan Kendaraan.....	82
Gambar 5.66 <i>Source Code</i> Fungsi Proses <i>Login</i> .....	83
Gambar 5.67 <i>Flowgraph</i> Fungsi Proses <i>Login</i> .....	83
Gambar 5.68 <i>Source Code</i> Pengujian <i>Test</i> Fungsi Proses <i>Login</i> .....	85
Gambar 5.69 Hasil Pengujian <i>Test</i> Fungsi Proses <i>Login</i> .....	85
Gambar 5.70 <i>Source Code</i> Fungsi Pengujian <i>Edit</i> Kendaraan.....	86
Gambar 5.71 <i>Flowgraph</i> Fungsi <i>Edit</i> Kendaraan.....	87
Gambar 5.72 <i>Source Code</i> Pengujian <i>Test Edit</i> Kendaraan.....	87
Gambar 5.73 Hasil Pengujian <i>Test</i> Fungsi <i>Edit</i> Kendaraan.....	88
Gambar 5.74 <i>Source Code</i> Fungsi <i>View</i> Kendaraan.....	88
Gambar 5.75 <i>Flowgraph</i> Fungsi <i>View</i> Kendaraan.....	89
Gambar 5.76 <i>Source Code</i> Pengujian <i>Test View</i> Kendaraan.....	90
Gambar 5.77 Hasil Pengujian <i>Test</i> Fungsi <i>View</i> Kendaraan.....	90
Gambar 5.78 <i>Source Code</i> Fungsi <i>Print</i> Pencairan.....	91
Gambar 5.79 <i>Flowgraph</i> Fungsi <i>Print</i> Pencairan.....	91
Gambar 5.80 <i>Source Code</i> Pengujian <i>Test Print</i> Pencairan.....	92
Gambar 5.81 Hasil Pengujian <i>Test</i> Fungsi <i>Print</i> Pencairan.....	93
Gambar 5.82 <i>Source Code</i> Fungsi Ubah Penyedia.....	93
Gambar 5.83 <i>Flowgraph</i> Fungsi Ubah Penyedia.....	94
Gambar 5.84 <i>Source Code</i> Pengujian <i>Test</i> Fungsi Ubah Penyedia.....	95
Gambar 5.85 Hasil Pengujian <i>Test</i> Fungsi Ubah Penyedia.....	95
Gambar 5.86 <i>Source Code</i> Fungsi <i>Edit Slider</i> .....	96
Gambar 5.87 <i>Flowgraph</i> Fungsi <i>Edit Slider</i> .....	93
Gambar 5.88 <i>Source Code</i> Pengujian <i>Test</i> Fungsi <i>Edit Slider</i> .....	97
Gambar 5.89 Hasil Pengujian <i>Test</i> Fungsi <i>Edit Slider</i> .....	98

## DAFTAR TABEL

Tabel 1.1 Rincian Data Mobil.....	2
Tabel 3.1 Jenis – Jenis Interaksi di Dunia Bisnis.....	8
Tabel 3.2 Penelitian Terdahulu.....	10
Tabel 4.1 Alamat Penyedia Jasa Penyewaan Mobil.....	14
Tabel 4.2 Jadwal Penelitian.....	15
Tabel 4.3 Simbol <i>Activity Diagram</i> .....	18
Tabel 4.4 Simbol <i>Use Case Diagram</i> .....	19
Tabel 4.5 Simbol <i>Class Diagram</i> .....	20
Tabel 5.6 Simbol <i>Squence Diagram</i> .....	21
Tabel 5.1 Desain Tabel Admin.....	39
Tabel 5.2 Desain Tabel Penyedia.....	40
Tabel 5.3 Desain Tabel Penyewa.....	41
Tabel 5.4 Desain Tabel Tersedia.....	42
Tabel 5.5 Desain Tabel Kendaraan.....	43
Tabel 5.6 Desain Tabel Transaksi.....	44
Tabel 5.7 Desain Tabel Pembayaran.....	45
Tabel 5.8 Desain Tabel Pencairan.....	45
Tabel 5.9 Desain Tabel Slider.....	46
Tabel 5.10 Daftar Tabel <i>Method</i> .....	79
Tabel 5.11 Dokumentasi Pengenalan <i>E-Marketplace</i> Penyewaan Mobil.....	158

## DAFTAR LAMPIRAN

- Lampiran 1. *Form* Topik dan Judul (*Fotocopy*)
- Lampiran 2. Surat Balasan dari (*Fotocopy*)
- Lampiran 3. *Form* Konsultasi (*Fotocopy*)
- Lampiran 4. Surat Pernyataan (*Fotocopy*)
- Lampiran 5. *Form* Revisi Ujian Pra Sidang (*Fotocopy*)
- Lampiran 6. *Form* Revisi Ujian Kompre (Asli)
- Lampiran 7. *Listing Code*

## ***ABSTRACT***

**NABILA ELFA RIENI.** *E-MARKETPLACE For Car Rental ( Case Study : Palembang City ).*

*The growth of Palembang city has led to the community's need for comfortable transportation, especially cars. Unfortunately, not everyone is capable of purchasing a car. Car rental services can become a solution for the Palembang city community. Challenges that occur in the car rental business include difficulties for renters to obtain rental price information, frequent payment delays, and the risk of cars being pawned. The purpose of this research is to develop a car rental e-marketplace. The system development technique used is Rapid Application Development (RAD), and the testing method is white box testing using statement coverage technique. The results obtained from this research is a car rental e-marketplace named "Rimera", which can assist renters in finding cars that suit their needs and enables providers to find serious renters for transactions, while also striving to reduce the risk of cars being pawned.*

**Keywords :** *Car Rental, E-marketplace, Rapid Application Development, White Box, Statement Coverage.*



## ABSTRAK

**NABILA ELFA RIENI.** *E-MARKETPLACE* Penyewaan Mobil ( Studi Kasus : Kota Palembang ).

Pertumbuhan kota Palembang membuat masyarakat membutuhkan alat transportasi yang nyaman khususnya mobil, sayangnya tidak semua orang mampu untuk membeli mobil. Jasa penyewaan mobil dapat menjadi solusi untuk masyarakat kota Palembang, Kendala yang terjadi pada bisnis penyewaan mobil seperti penyewa sulit mendapatkan informasi harga sewa, pembayaran sering tidak tepat waktu, dan resiko mobil digadaikan. Adapun tujuan dari penelitian ini adalah untuk membangun *E-marketplace* penyewaan mobil. Adapun teknik pengembangan sistem yang digunakan adalah *Rapid Application Development* (RAD) , dan metode pengujian *white box* dengan teknik *statement coverage*. Hasil yang didapat dari penelitian ini adalah *E-marketplace* penyewaan mobil telah peneliti beri nama Rimeria yang dapat membantu penyewa menemukan mobil yang sesuai dengan kebutuhannya dan memungkinkan penyedia menemukan penyewa yang serius untuk melakukan transaksi, serta berupaya untuk mengurangi resiko mobil digadaikan.

**Kata kunci :** *Penyewaan Mobil , E-marketplace, Rapid Application Development, White Box, Statement Coverage.*

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang Penelitian

Palembang adalah kota terbesar kedua setelah Medan di pulau Sumatera (Yuliati, Eka, dan Dian Ratnasari Yahya 2018:6). Pergerakan pertumbuhan kota Palembang membuat masyarakat membutuhkan mobil sebagai alat transportasi yang nyaman untuk melakukan aktivitas, akan tetapi tidak semua orang mampu untuk membeli atau memiliki sebuah mobil dikarenakan harganya yang cukup mahal. Jasa penyewaan mobil adalah solusi bagi masyarakat kota Palembang yang belum memiliki mobil, sehingga masyarakat tidak perlu memikirkan biaya pembelian, cicilan, perawatan, pajak mobil dll.

*Marketplace* adalah pasar virtual dimana para penjual dan pembeli melakukan transaksi menggunakan media internet (Rini Yustiani, dkk 2017:3). Di Indonesia banyak *marketplace* yang menyediakan berbagai produk penunjang kehidupan sehari – hari, tetapi penyedia *marketplace* berupa jasa penyewaan khususnya mobil masih terbatas, bahkan jarang diketahui oleh masyarakat. *Business to Business* menurut Mahir Pradana (2015:5) adalah transaksi bisnis antara pelaku bisnis dengan pelaku bisnis lainnya. Pada bisnis penyewaan mobil, transaksi ini terjadi karena mobil sewaan sedang tidak tersedia sehingga penyedia jasa penyewaan mobil menyewa mobil dari penyedia jasa penyewaan lain untuk disewakan, yang biasanya sering disebut dengan istilah *rent to rent*. Peneliti melakukan penelitian pada 5 perusahaan penyedia jasa penyewaan mobil, dengan rincian data mobil yang dapat dilihat pada tabel 1.1.

**Tabel 1.1** Rincian Data Mobil

No.	Nama Perusahaan	Jumlah Mobil yang disediakan	Harga Sewa Mulai dari (per hari)
1.	PT. Sekayu Serasan Sekate	7 unit	Rp 400.000
2.	CV. Palembang Berlian Abadi Jaya	11 unit	Rp 350.000
3.	CV. Sriwijaya XI	13 unit	Rp 600.000
4.	CV. Sriwijaya Maju Gemilang	16 unit	Rp 400.000
5.	CV. Anugrah Esa Utama	16 unit	Rp 1.000.000

Sumber : Pemilik Jasa Penyewaan Mobil

Hasil observasi yang peneliti lakukan didapatkan bahwa transaksi *rent to rent* rata – rata terjadi 5 kali dalam sebulan karena adanya permintaan kebutuhan pelanggan. Selain itu peneliti menemukan kendala pada bisnis penyewaan mobil seperti penyewa sulit menemukan informasi harga penyewaan mobil, pembayaran yang dilakukan oleh penyewa sering tidak tepat waktu, tidak adanya transparansi pada bisnis *rent to rent* sehingga dapat menjadi penyebab mobil digadaikan.

*E-marketplace* penyewaan mobil menjadi wadah untuk menampung penyedia jasa penyewaan dan penyewa mobil, peneliti bermaksud ingin membuat *e-marketplace* penyewaan mobil dengan menerapkan metode pengembangan sistem *Rapid Application Development (RAD)* yang terdiri dari 3 tahapan, yaitu : rencana kebutuhan, desain sistem, dan implementasi ini diharapkan dapat dibuat dengan waktu relatif lebih cepat, memberikan kemudahan bagi penyewa dalam menemukan mobil yang sesuai dengan kebutuhannya, dan bagi penyedia jasa penyewaan mobil dalam menemukan penyewa yang serius untuk melakukan transaksi, serta dengan adanya transparansi bisnis *rent to rent* diharapkan dapat mengurangi resiko mobil digadaikan. Maka dari itu peneliti memutuskan untuk

mengangkat judul “**E-MARKETPLACE PENYEWAAN MOBIL (STUDI KASUS : KOTA PALEMBANG)**).

### **1.2. Rumusan Masalah**

Berdasarkan latar belakang yang telah disampaikan diatas maka rumusan masalah yang diangkat adalah bagaimana membuat *e-marketplace* penyewaan mobil yang diharapkan dapat memberikan kemudahan bagi penyewa dalam menemukan mobil yang sesuai dengan kebutuhannya, dan bagi penyedia jasa penyewaan mobil dalam menemukan penyewa yang serius untuk melakukan transaksi, serta dengan adanya transparansi bisnis *rent to rent* diharapkan dapat mengurangi resiko mobil digadaikan menggunakan metode pengembangan sistem *Rapid Application Development (RAD)* berbasis *website*.

### **1.3. Batasan Masalah**

Peneliti membatasi masalah dalam penelitian ini, maka penelitian ini menuangkan ruang lingkup sebagai berikut :

- 1) Objek Penelitian ini merupakan PT. Sekayu Sersan Sekate, CV. Palembang Berlian Abadi Jaya, CV. Sriwijaya XI, CV. Sriwijaya Maju Gemilang, CV. Anugrah Esa Utama.
- 2) Metode pengembangan sistem menggunakan *Rapid Application Development (RAD)*.
- 3) Desain sistem menggunakan *Use Case Diagram, Activity Diagram, Class Diagram, dan Sequence Diagram*.

- 4) *Marketplace* penyewaan mobil dapat diakses oleh Admin, pemilik jasa penyewaan mobil, dan pelanggan.
  - a) Admin mengelola informasi yang ada di *marketplace*, menginput data mobil yang tersedia, mengkonfirmasi pesanan penyewaan mobil dari pelanggan, memilih penyedia yang menyediakan kebutuhan pelanggan, dan melakukan pencairan dana kepada penyedia jasa penyewaan mobil.
  - b) Pemilik jasa penyewaan mobil dapat menginput qty dan harga mobil, dan mengedit data mobil yang disewakan.
  - c) Pelanggan dapat melihat data mobil, memilih dan menyewa mobil yang sudah disediakan oleh pemilik jasa penyewaan mobil.
- 5) Teknik pengujian sistem menggunakan *white box (Statement Coverage)*.

#### **1.4. Tujuan Penelitian**

Tujuan dari penelitian ini adalah membangun *e-marketplace* penyewaan mobil yang bertujuan untuk memberi dukungan secara aktif dalam kelancaran usaha penyewaan mobil dengan cara mempertemukan penyedia jasa penyewaan dan penyewa mobil untuk melakukan transaksi kapanpun dan dimanapun berada, dan penyewa dapat memilih dan menemukan mobil yang sesuai dengan kebutuhannya tanpa menghabiskan waktu dan biaya.

## **1.5. Manfaat Penelitian**

### **1.5.1. Manfaat Bagi Peneliti**

Penelitian ini diharapkan dapat menjadi pengalaman sekaligus gambaran bagaimana nantinya bekerja dengan profesi sebagai programing, dan pada saat terjuan dalam dunia kerja peneliti sudah tahu tugas dan wewenang pada masing – masing devisi yang ada di dalam perusahaan.

### **1.5.2. Manfaat Bagi Perusahaan**

Perusahaan tempat peneliti melakukan kegiatan penelitian diharapkan memperluas jangkauan pemasarannya, dapat membantu dalam pencatatan transaksi dalam laporan transaksi.

### **1.5.3. Manfaat Bagi Akademik**

Dapat menjalin hubungan baik dengan perusahaan yang menjadi tempat penelitian, mendapat refrensi dalam penelitian skripsi di kemudian hari, selain itu juga bisa memberikan pengalaman bagi mahasiswa tentang bagaimana disiplin kerja, cara bersosialisasi, profesionalisme, dan etika kerja.

## **1.6. Sistematika Penelitian**

Untuk mempermudah pembahasan dan penyusunan penelitian laporan penelitian ini menjadi lebih mudah dipahami dan terarah, dimana penelitian menjabarkan dalam 6 bab permasalahan dan masing – masing bab akan diuraikan menjadi beberapa sub bab. Sistematika laporan penelitian ini diuraikab sebagai berikut :

**BAB I PENDAHULUAN**

Pada bab ini berisikan latar belakang, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penelitian.

**BAB II GAMBARAN UMUM PERUSAHAAN**

Pada bab ini akan membahas tentang profil tempat penelitian, visi dan misi, struktur organisasi, dan tugas wewenang.

**BAB III TINJAUAN PUSTAKA**

Pada bab ini akan diuraikan teori – teori yang mendukung yang terkait dengan penelitian.

**BAB IV METODE PENELITIAN**

Pada bab ini diuraikan jenis data, teknik pengumpulan data, jenis penelitian, alat dan teknik pengembangan sistem, serta alat data teknik pengujian.

**BAB V HASIL DAN PEMBAHASAN**

Pada bab ini dilaporkan hasil – hasil yang diperoleh dalam penelitian dan pembahasan terhadap hasil yang telah dicapai dan uji coba teknologi. Hasil dan pembahasan disesuaikan dengan teknik pengembangan sistem yang digunakan.

**BAB VI PENUTUP**

Pada bab ini memberikan kesimpulan secara umum dari apa yang telah dibahas.

## **BAB II**

### **GAMBARAN UMUM PERUSAHAAN**

#### **2.1. Profil Perusahaan PT. Sekayu Serasan Sekate**

##### **2.1.1. Sejarah Perusahaan**

PT. Sekayu Serasan Sekate berdiri pada tahun 2018. Awal mula berdirinya perusahaan ini dilatar belakangi pengalaman pemilik yang berkerja sebagai driver selama 2 tahun, kemudian dikarenakan melihat peluang bisnis yang baik sehingga memiliki inisiatif untuk menjadi pengusaha rental. Dengan modal awal yang dimiliki dibelilah mobil innova reborn, karena modal yang masih sedikit, mobil tersebut dititipkan ke pengusaha rental lainnya sedangkan pemilik bekerja di perusahaan lain dengan bidang yang berbeda. Selama 9 bulan bekerja dengan gaji sebesar Rp. 12.000.000,- per bulan. Kemudian ditahun 2019 dengan gaji tersebut dijadikan sebagai modal tambahan. Sampai saat ini PT. Sekayu Serasan Sekate terus berkembang dan dipercaya untuk menyediakan jasa penyewaan mobil.

##### **2.1.2. Visi dan Misi**

Adapun visi dan misi pada PT. Sekayu Serasan Sekate adalah sebagai berikut :

**Visi :** Menjadi penyedia layanan rental mobil terpercaya dan terkemuka yang memberikan pengalaman sewa mobil yang mudah, nyaman, dan terjangkau bagi pelanggan kami.

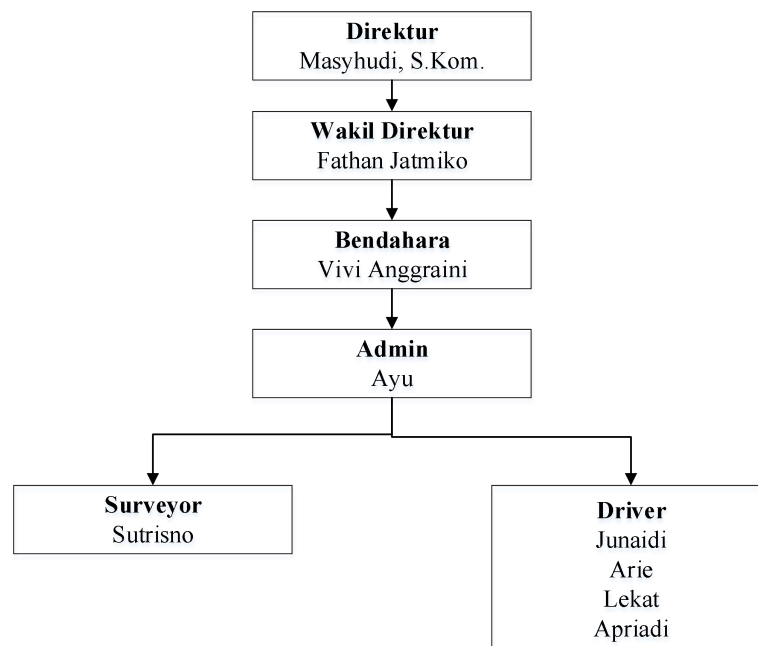
**Misi :** - Memberikan layanan penyewaan mobil yang berkualitas tinggi dengan armada kendaraan yang terawat, aman, dan terbaru.



- Menyediakan pilihan mobil yang beragam dan fleksibel untuk memenuhi kebutuhan pelanggan dalam perjalanan bisnis, liburan, atau acara khusus.
- Melakukan pemeliharaan dan perawatan teratur pada armada mobil kami untuk memastikan keamanan, kenyamanan, dan keandalan kendaraan yang disewakan.
- Mengurangi dampak lingkungan dengan memilih kendaraan yang ramah lingkungan dan menerapkan praktik bisnis yang berkelanjutan.
- Memberikan pengalaman sewa mobil yang terbaik bagi pelanggan kami.

### 2.1.3. Struktur Organisasi

Adapun struktur organisasi pada PT. Sekayu Serasan Sekate dapat dilihat pada gambar 2.1 :



Sumber : Penyedia Jasa Penyewaan Mobil

**Gambar 2.1** Struktur Organisasi PT. Sekayu Serasan Sekate

#### **2.1.4. Tugas dan Wewenang**

Adapun rincian tugas dan wewenang pengurus PT. Sekayu Serasan Sekate adalah sebagai berikut :

##### **1. Direktur**

- Bertanggung jawab penuh atas pengambilan keputusan strategis dan pengelolaan keseluruhan perusahaan.
- Menetapkan visi, misi, dan tujuan perusahaan.
- Mengawasi dan memberikan arahan kepada tim.
- Mengelola hubungan dengan mitra bisnis dan pihak-pihak eksternal lainnya.
- Mewakili perusahaan dalam pertemuan bisnis dan negosiasi penting.

##### **2. Wakil Direktur**

- Mendukung direktur dalam pengambilan keputusan strategis dan pengelolaan perusahaan.
- Menggantikan direktur dalam situasi ketidakhadiran atau keperluan bisnis tertentu.
- Mengkoordinasikan tugas-tugas dan proyek-proyek tertentu sesuai dengan arahan dari direktur.
- Melaksanakan tugas-tugas khusus yang diberikan oleh direktur.
- Mewakili direktur atau perusahaan dalam pertemuan atau acara tertentu.

##### **3. Bendahara**

- Bertanggung jawab atas pengelolaan keuangan perusahaan.
- Mempersiapkan dan mengawasi anggaran perusahaan.
- Menyusun laporan keuangan dan menganalisis kinerja keuangan.

- Mengelola pengeluaran dan pemasukan perusahaan.

#### **4. Admin**

- Melakukan tugas administratif, seperti pengarsipan, pengolahan data, dan penyusunan laporan.
- Mencatat semua hutang dan piutang;
- Mengelola inventaris perusahaan, seperti peralatan kantor dan kebutuhan operasional.
- Mencatat semua biaya perusahaan meliputi biaya promosi, biaya pembelian, biaya penjualan, dan biaya kantor;
- Mencatat semua pemasukan dan pengeluaran perusahaan;
- Mendukung staf lain dalam tugas-tugas sehari-hari.

#### **5. Surveyor**

- Tugas utama seorang surveyor dalam perusahaan rental mobil adalah melakukan peninjauan dan pengecekan data pelanggan, kondisi mobil sebelum disewakan kepada pelanggan.
- Surveyor bertanggung jawab untuk memastikan pelanggan yang akan melakukan penyewaan mobil tidak masuk dalam data blacklist, mobil dalam kondisi baik, aman, dan siap untuk disewakan.
- Mereka melakukan pemeriksaan terhadap kendaraan seperti mesin, rem, lampu, sistem pendingin, kondisi ban, dan komponen lainnya.
- Surveyor juga dapat melakukan pengecekan terhadap asuransi kendaraan serta dokumen dan perizinan yang diperlukan.

- Jika terdapat kerusakan atau masalah pada kendaraan, surveyor akan membuat laporan dan menentukan langkah perbaikan yang diperlukan sebelum mobil dapat disewakan.

## **6. Driver**

- Tugas utama seorang driver dalam perusahaan rental mobil adalah mengemudikan mobil untuk mengantarkan pelanggan ke tujuan yang diinginkan.
- Driver harus memiliki pengetahuan yang baik tentang jalan, rute, dan aturan lalu lintas.
- Mereka harus dapat mengemudikan kendaraan dengan aman, memperhatikan keselamatan penumpang dan mematuhi peraturan lalu lintas.
- Mereka harus bersikap ramah, sopan, dan memberikan layanan pelanggan yang baik.

## **2.2. Profil Perusahaan CV. Palembang Berlian Abadi Jaya**

### **2.2.1. Sejarah Perusahaan**

CV. Palembang Berlian Abadi jaya berdiri tanggal 04 April 2022, terletak di Jl. Soekarno hatta, Perumahan BSI Blok BI No. 1, Kel. Bukitbaru, Kec. Ilir Barat I, Kota Palembang, Provinsi Sumatera Selatan. Awal berdirinya perusahaan kami karena adanya kebutuhan pribadi untuk perjalanan keluar kota, seiring banyaknya kebutuhan maka didirikan perusahaan rental untuk memenuhi kebutuhan dari berbagai penyewa. Hingga saat ini CV. Palembang Berlian Abadi Jaya dipercaya

untuk menyediakan berbagai merk mobil untuk disewakan guna kepentingan pribadi maupun perusahaan.

### 2.2.2. Visi dan Misi

Adapun visi dan misi pada CV. Palembang Berlian Abadi Jaya adalah sebagai berikut :

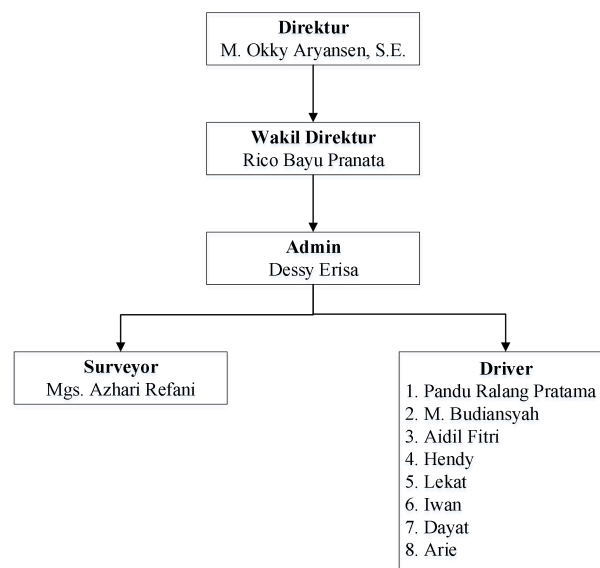
**Visi :** Menjadi perusahaan penyewaan kendaraan terpercaya dan memberikan solusi terbaik untuk kebutuhan pelanggan.

**Misi :**

- Membantu dan memenuhi kebutuhan penyewaan kendaraan dengan pelayanan dan harga terbaik.
- Mengedepankan kepuasan pelanggan dengan memaksimalkan *after sales* dan *fast response*.

### 2.2.3. Struktur Organisasi

Adapun struktur organisasi pada CV. Palembang Berlian Abadi Jaya dapat dilihat pada gambar 2.2 :



Sumber : Penyedia Jasa Penyewaan Mobil

**Gambar 2.2** Struktur Organisasi CV. Palembang Berlian Abadi Jaya

#### **2.2.4. Tugas dan Wewenang**

Adapun rincian tugas dan wewenang pengurus CV. Palembang Berlian Abadi Jaya adalah sebagai berikut :

##### **1. Direktur**

- Bertanggung jawab penuh atas pengambilan keputusan strategis dan pengelolaan keseluruhan perusahaan.
- Menetapkan visi, misi, dan tujuan perusahaan.
- Mengawasi dan memberikan arahan kepada tim.
- Mengelola hubungan dengan mitra bisnis dan pihak-pihak eksternal lainnya.
- Mewakili perusahaan dalam pertemuan bisnis dan negosiasi penting.

##### **2. Wakil Direktur**

- Mendukung direktur dalam pengambilan keputusan strategis dan pengelolaan perusahaan.
- Menggantikan direktur dalam situasi ketidakhadiran atau keperluan bisnis tertentu.
- Mengkoordinasikan tugas-tugas dan proyek-proyek tertentu sesuai dengan arahan dari direktur.
- Melaksanakan tugas-tugas khusus yang diberikan oleh direktur.
- Mewakili direktur atau perusahaan dalam pertemuan atau acara tertentu.
- Mengelola pengeluaran dan pemasukan perusahaan.

##### **3. Admin**

- Melakukan tugas administratif, seperti pengarsipan, pengolahan data, dan penyusunan laporan.

- Mencatat semua hutang dan piutang;
- Mengelola inventaris perusahaan, seperti peralatan kantor dan kebutuhan operasional.
- Mencatat semua biaya perusahaan meliputi biaya promosi, biaya pembelian, biaya penjualan, dan biaya kantor;
- Mencatat semua pemasokan dan pengeluaran perusahaan;
- Mendukung staf lain dalam tugas-tugas sehari-hari.

#### **4. Surveyor**

- Tugas utama seorang surveyor dalam perusahaan rental mobil adalah melakukan peninjauan dan pengecekan data pelanggan, kondisi mobil sebelum disewakan kepada pelanggan.
- Surveyor bertanggung jawab untuk memastikan pelanggan yang akan melakukan penyewaan mobil tidak masuk dalam data blacklist, mobil dalam kondisi baik, aman, dan siap untuk disewakan.
- Mereka melakukan pemeriksaan terhadap kendaraan seperti mesin, rem, lampu, sistem pendingin, kondisi ban, dan komponen lainnya.
- Surveyor juga dapat melakukan pengecekan terhadap asuransi kendaraan serta dokumen dan perizinan yang diperlukan.
- Jika terdapat kerusakan atau masalah pada kendaraan, surveyor akan membuat laporan dan menentukan langkah perbaikan yang diperlukan sebelum mobil dapat disewakan.

## **5. Driver**

- Tugas utama seorang driver dalam perusahaan rental mobil adalah mengemudikan mobil untuk mengantarkan pelanggan ke tujuan yang diinginkan.
- Mereka harus dapat mengemudikan kendaraan dengan aman, memperhatikan keselamatan penumpang dan mematuhi peraturan lalu lintas.
- Driver juga dapat memberikan informasi dan panduan mengenai tujuan wisata atau lokasi yang dikunjungi kepada pelanggan.
- Mereka harus bersikap ramah, sopan, dan memberikan layanan pelanggan yang baik.

## **2.3. Profil Perusahaan CV. Sriwijaya XI**

### **2.3.1. Sejarah Perusahaan**

CV. Sriwijaya XI berlokasi di Palembang sejak tahun 2018. Rental Mobil Sriwijaya XI merupakan salah satu perusahaan penyedia layanan Dimulai dengan satu mobil minibus Xenia yang digunakan untuk melayani klien kami baik dalam kota maupun luar kota Palembang. Selanjutnya untuk memenuhi permintaan pasar yang terus meningkat, Rental Mobil Sriwijaya XI mulai menambah armadanya dengan jenis yang berbeda sehingga dapat menjangkau semua klien kami, baik korporasi maupun individu. Namun demikian, target utama kami adalah 4x4. *Experiences are the best teacher* merupakan cerminan dari team kami. Team kami sangat berpengalaman di bidangnya. Selain didukung dengan armada yang mumpuni, team kami juga didukung oleh pengemudi yang handal. Pengemudi



kami dilengkapi dengan DDT 4x4 *certification (defensive driving training 4x4)* by ORD yang menunjukkan keseriusan kami dalam bidang jasa rental/sewa mobil 4x4 untuk menjaga keselamatan dan kenyamanan anda selama menggunakan jasa kami. Kami juga menyediakan *Recovery software* di setiap mobil 4x4 sesuai dengan kebutuhan anda.

### 2.3.2. Visi dan Misi

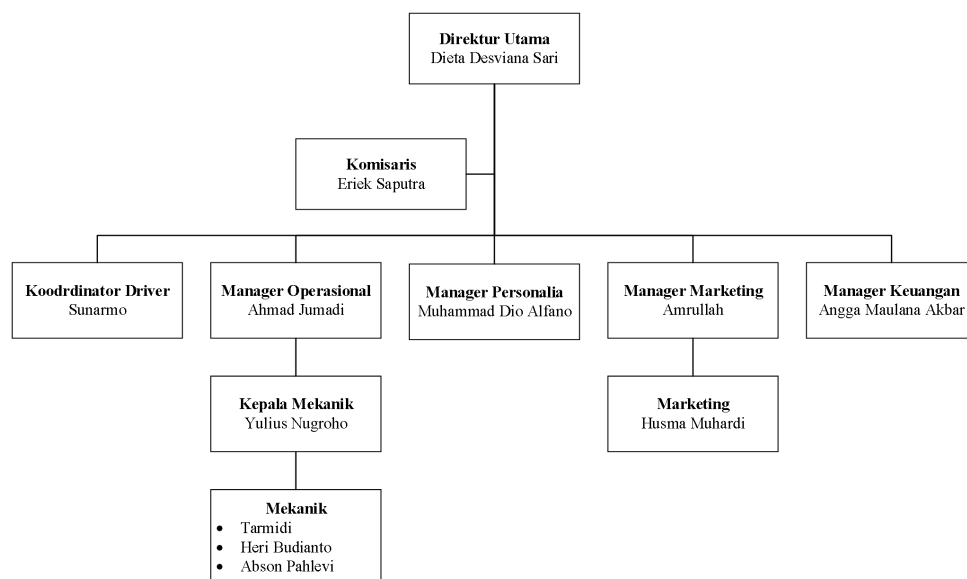
Adapun visi dan misi pada CV. Sriwijaya XI adalah sebagai berikut :

**Visi :** Menjadi salah satu perusahaan rental / sewa mobil terpercaya dan amanah di Sumatera Selatan.

**Misi :** Memberikan pelayanan terbaik dan menciptakan hubungan kerjasama secara berkesinambungan antara kami dan pihak penyewa.

### 2.3.3. Struktur Organisasi

Adapun struktur organisasi pada CV. Sriwijaya XI dapat dilihat pada gambar 2.3 :



Sumber : Penyedia Jasa Penyewaan Mobil

**Gambar 2.3** Struktur Organisasi CV. Sriwijaya XI

#### **2.3.4. Tugas dan Wewenang**

Adapun rincian tugas dan wewenang pengurus CV. Sriwijaya XI adalah sebagai berikut :

##### **1. Direktur Utama**

- Memutuskan dan menentukan peraturan dan kebijakan tertinggi perusahaan.
- Bertanggung jawab dalam memimpin dan menjalankan perusahaan bertanggung jawab atas kerugian termasuk juga keuntungan yang diperoleh perusahaan.
- Merencanakan serta mengembangkan sumber – sumber pendapatan dan pembelanjaan kekayaan perusahaan.
- Bertindak sebagai perwakilan perusahaan dalam hubungannya dengan dunia luar perusahaan.
- Menetapkan strategi – strategi strategis untuk mencapai visi dan misi perusahaan.
- Mengkoordinasikan dan mengawasi semua kegiatan di perusahaan.
- Memimpin perusahaan dengan membuat kebijakan – kebijakan perusahaan.
- Memilih, menentukan dan mengawasi pekerjaan karyawan.
- Menyetujui anggaran tahunan perusahaan dan melaporkan laporan pada komisaris.

##### **2. Komisaris**

- Melakukan pengawasan secara umum dan / atau khusus sesuai dengan anggaran dasar.

- Melakukan pengawasan terhadap pengelolaan perusahaan yang dilakukan oleh direksi.
- Memantau dan memberikan nasihat terkait kebijakan direksi dalam menjalankan perusahaan.
- Mengawasi bagaimana pelaksanaan strategi untuk memenuhi harapan para pemegang saham.

### **3. Koordinator Driver**

- Memutuskan jadwal kerja para pengemudi berdasarkan permintaan dan kebutuhan perusahaan.
- Menetapkan tugas dan rute yang akan diambil oleh pengemudi, dengan mempertimbangkan faktor-faktor seperti jarak, waktu tempuh, dan prioritas pengiriman.
- Mengoordinasikan aktivitas antara pengemudi dan departemen lain.
- Berwenang memberikan pelatihan kepada pengemudi mengenai standar keselamatan, prosedur perusahaan, dan keterampilan berkendara yang diperlukan.

### **4. Manager Operasional**

- Bertanggung jawab atas operasional harian perusahaan atau bagian tertentu.
- Membuat keputusan terkait perencanaan operasional, mengatur alokasi sumber daya, mengelola tim operasional, memastikan kualitas produk atau layanan.
- Berkolaborasi dengan departemen lain untuk memastikan tujuan operasional tercapai.

## **5. Kepala Mekanik**

- Bertanggung jawab atas pengelolaan dan pemeliharaan peralatan atau mesin dalam lingkungan kerja.
- Memimpin tim mekanik dalam melakukan perawatan, perbaikan, dan inspeksi rutin.
- Mengatur jadwal perawatan, mengambil keputusan terkait perbaikan darurat, mengawasi peralatan agar beroperasi dengan efisien.
- Berkoordinasi dengan manajemen untuk memastikan kelancaran produksi.

## **6. Mekanik**

- Melakukan perawatan, perbaikan, dan inspeksi peralatan atau mesin.
- Memastikan bahwa mobil berfungsi dengan baik dan melakukan tindakan perbaikan saat diperlukan.
- Bertanggung jawab atas perawatan dan perbaikan mobil sesuai dengan standar keselamatan dan kualitas yang ditetapkan.

## **7. Manager Personalia**

- Mengelola aspek sumber daya manusia dalam perusahaan. Termasuk perekrutan, pengembangan karyawan, pengelolaan kinerja, administrasi tenaga kerja, dan pematuhan hukum ketenagakerjaan.
- Merancang dan menerapkan kebijakan SDM, mengelola masalah karyawan, mengawasi pelatihan dan pengembangan.
- Memastikan perusahaan mematuhi regulasi ketenagakerjaan.

## **8. Manager Marketing**

- Bertanggung jawab atas strategi pemasaran dan promosi perusahaan.

- Mengidentifikasi peluang pasar, mengembangkan kampanye pemasaran, dan meningkatkan citra perusahaan.
- Mengatur anggaran pemasaran, mengarahkan tim pemasaran, mengambil keputusan terkait promosi.
- Menganalisis efektivitas strategi pemasaran.

## **9. Marketing**

- Melaksanakan strategi pemasaran yang telah ditetapkan.
- Terlibat dalam kampanye iklan, promosi produk atau layanan, riset pasar, dan interaksi dengan pelanggan untuk memahami kebutuhan dan preferensi mereka.

## **10. Manager Keuangan**

- Mengelola aspek keuangan perusahaan, termasuk perencanaan anggaran, pelaporan keuangan, analisis keuangan, dan pengelolaan risiko keuangan.
- Menyusun kebijakan keuangan, mengawasi tim keuangan, membuat keputusan investasi, mengelola arus kas, dan menjaga kesehatan keuangan perusahaan.

### **2.4. Profil Perusahaan CV. Sriwijaya Maju Gemilang**

#### **2.4.1. Sejarah Perusahaan**

CV. Sriwijaya Maju Gemilang berlokasi di Jl. Tanjung Api – Api, Komplek Tanjung Harapan indah No. 3, Palembang dan berdiri sejak tahun 2016 hingga sekarang. Pada awal berdirinya perusahaan ini adalah untuk keperluan bisnis travel, kemudian dikarenakan adanya permintaan dan kebutuhan dari pelanggan

sehingga berubah menjadi bisnis penyewaan mobil. Pada awal pendiriannya CV. Sriwijaya Maju Gemilang mengalami kesulitan promosi dikarenakan belum adanya sosial media, namun hal itu dapat diatasi dikarenakan adanya perkembangan teknologi di Indonesia. Hingga Saat ini unit kendaraan CV. Sriwijaya Maju Gemilang terus bertambah dan dipercaya oleh pelanggan untuk menyediakan mobil sewa.

#### **2.4.2. Visi dan Misi**

Adapun visi dan misi pada CV. Sriwijaya Maju Gemilang adalah sebagai berikut :

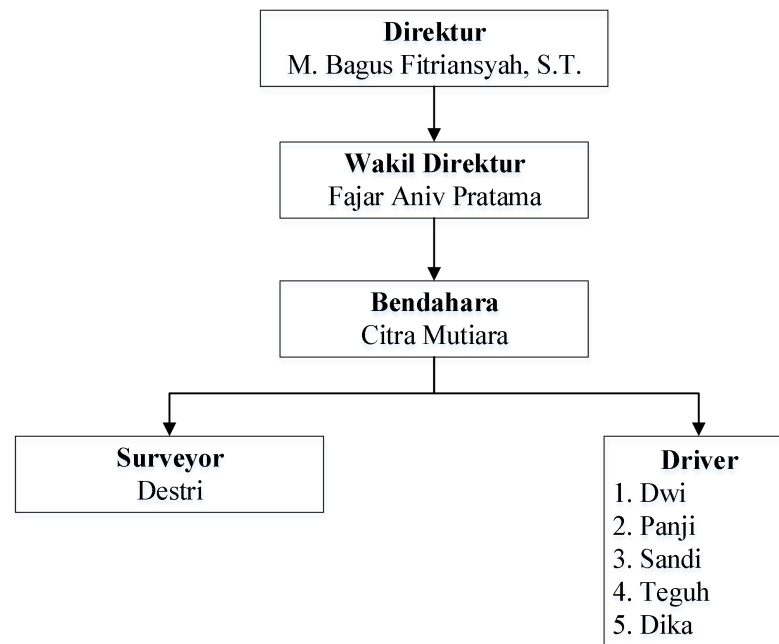
**Visi :** Menjadi mitra pilihan yang terpercaya dalam penyediaan layanan rental mobil, dengan komitmen untuk memberikan pengalaman sewa yang mudah, nyaman dan terjangkau bagi pelanggan kami.

**Misi :**

- Menyediakan armada mobil berkualitas tinggi yang terawat baik, dan memastikan beragam pilihan mobil yang fleksibel dan sesuai dengan kebutuhan dan referensi pelanggan kami.
- Menawarkan harga yang kompetitif, transparan, dan adil, agar pelanggan merasa puas dengan nilai yang mereka dapatkan.
- Mengutamakan integritas, profesionalisme, dan etika bisnis dalam semua aspek operasional demi membangun kepercayaan pelanggan dan citra positing perusahaan.

#### **2.4.3. Struktur Organisasi**

Adapun struktur organisasi pada CV. Sriwijaya Maju Gemilang dapat dilihat pada gambar 2.4 :



Sumber : Penyedia Jasa Penyewaan Mobil

**Gambar 2.4** Struktur Organisasi CV. Sriwijaya Maju Gemilang

#### 2.4.4. Tugas dan Wewenang

Adapun rincian tugas dan wewenang pengurus CV. Sriwijaya Maju Gemilang adalah sebagai berikut :

##### 1. **Direktur**

- Bertanggung jawab penuh atas pengambilan keputusan strategis dan pengelolaan keseluruhan perusahaan.
- Menetapkan visi, misi, dan tujuan perusahaan.
- Mengawasi dan memberikan arahan kepada tim.
- Mengelola hubungan dengan mitra bisnis dan pihak-pihak eksternal lainnya.
- Mewakili perusahaan dalam pertemuan bisnis dan negosiasi penting.

##### 2. **Wakil Direktur**

- Mendukung direktur dalam pengambilan keputusan strategis dan pengelolaan perusahaan.

- Menggantikan direktur dalam situasi ketidakhadiran atau keperluan bisnis tertentu.
- Mengkoordinasikan tugas-tugas dan proyek-proyek tertentu sesuai dengan arahan dari direktur.
- Melaksanakan tugas-tugas khusus yang diberikan oleh direktur.
- Mewakili direktur atau perusahaan dalam pertemuan atau acara tertentu.
- Mengelola pengeluaran dan pemasukan perusahaan.

### **3. Surveyor**

- Tugas utama seorang surveyor dalam perusahaan rental mobil adalah melakukan peninjauan dan pengecekan data pelanggan, kondisi mobil sebelum disewakan kepada pelanggan.
- Surveyor bertanggung jawab untuk memastikan pelanggan yang akan melakukan penyewaan mobil tidak masuk dalam data blacklist, mobil dalam kondisi baik, aman, dan siap untuk disewakan.
- Mereka melakukan pemeriksaan terhadap kendaraan seperti mesin, rem, lampu, sistem pendingin, kondisi ban, dan komponen lainnya.
- Surveyor juga dapat melakukan pengecekan terhadap asuransi kendaraan serta dokumen dan perizinan yang diperlukan.
- Jika terdapat kerusakan atau masalah pada kendaraan, surveyor akan membuat laporan dan menentukan langkah perbaikan yang diperlukan sebelum mobil dapat disewakan.



#### **4. Driver**

- Tugas utama seorang driver dalam perusahaan rental mobil adalah mengemudikan mobil untuk mengantarkan pelanggan ke tujuan yang diinginkan.
- Mereka harus dapat mengemudikan kendaraan dengan aman, memperhatikan keselamatan penumpang dan mematuhi peraturan lalu lintas.
- Driver juga dapat memberikan informasi dan panduan mengenai tujuan wisata atau lokasi yang dikunjungi kepada pelanggan.
- Mereka harus bersikap ramah, sopan, dan memberikan layanan pelanggan yang baik.

### **2.5. Profil Perusahaan CV. Anugrah Esa Utama**

#### **2.5.1. Sejarah Perusahaan**

CV. Anugrah Esa Utama Didirikan oleh bapak Ikhsan Yuratama pada tanggal 10 Januari 2011 bersama nona Ariyani Astuti, CV ini bergerak dalam bidang jasa, kecuali dalam bidang hukum dan pajak. Melakukan usaha dalam bidang perdagangan umum, termasuk ekspor impor. Mengusahakan perusahaan perencana, pelaksana dan pemborong pendirian bangunan, jalan, jembatan, irigasi dan pekerjaan teknikimunya. Pada tanggal 25 April 2012 CV ini mengalami pembaruan struktur yang baru, dan mereka juga memperbarui bidang yang bergerak di ATK (alat tulis kantor), rental, alat berat, pembangunan jalan. Pada tahun 2019 CV ini lebih mengutamakan rental mobil, tetapi pada bidang lainnya tetap berjalan dengan biasanya.

### 2.5.2. Visi dan Misi

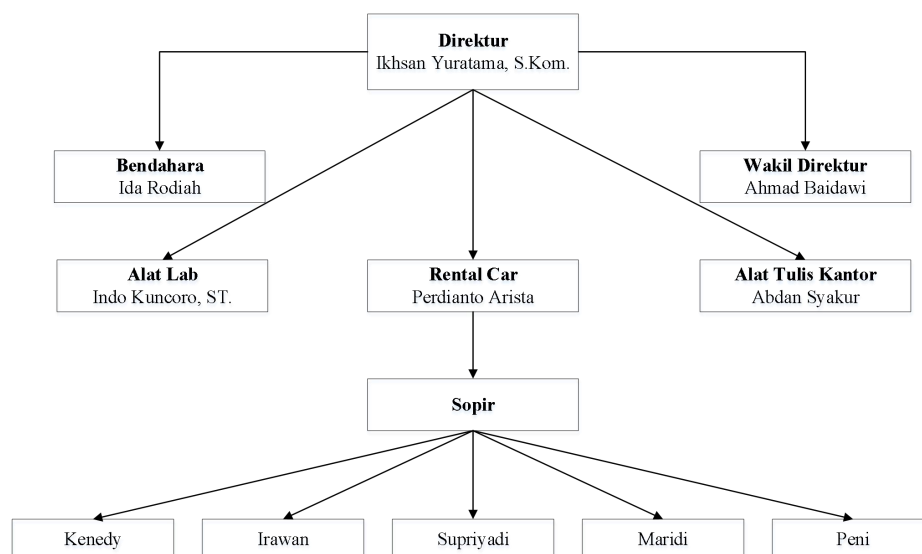
Adapun visi dan misi pada CV. Anugerah Esa Utama adalah sebagai berikut :

**Visi :** Terciptanya hubungan kerja yang benar dengan pihak Swasta dan Pemerintah, melalui kegiatan pengadaan barang dan jasa..

**Misi :** Melaksanakan komunikasi dan interaksi sesuai prosedur dan peraturan yang ada mempertahankan dan meningkatkan kualitas pelayanan, agar terciptanya hasil kerja yang baik meningkatkan kemampuan dan kesejahteraan

### 2.5.3. Struktur Organisasi

Adapun struktur organisasi pada CV. Anugerah Esa Utama dapat dilihat pada gambar 2.5 :



Sumber : Penyedia Jasa Penyewaan Mobil

**Gambar 2.5** Struktur Organisasi CV. Anugerah Esa Utama

### 2.5.4. Tugas dan Wewenang

Adapun rincian tugas dan wewenang pengurus CV. SANugerah Esa Utama adalah sebagai berikut :

### **1. Direktur**

- Memimpin perusahaan dengan menerbitkan kebijakan-kebijakan perusahaan atau institusi.
- Memilih, menetapkan, mengawasi tugas dari karyawan kepala bagian (manajer) atau wakil direktur rmenyetujui anggaran tahunan perusahaan atau institusi.
- Menyampaikan laporan kepada pemegang saham atas kinerja perusahaan atau institusi.

### **2. Wakil Direktur**

- Membantu Direktur dalam menyusun rencana kerjasama anggaran untuk mencapai tujuan perusahaan.
- Membantu Direktur dalam memimpin dammengkoordinir seluruh aktivitas perusahaan.

### **3. Sekretaris**

- Menjalankan Perintah Pimpinan.
- Membantu Pekerjaan Pimpinan.
- Penghubung antara Pemimpin dan Anggota.
- Penghubung Pemimpin dengan Orang- Orang di Luar Organisasi.
- Melakukan Korespondensi.
- Pengarsipan.
- Mengatur Jadwal.
- Kreatif.

**4. Alat Lab**

- Menyelesaikan kontrak dengan rekanan.
- Pembelian, pemesanan alat lab, Pengiriman alat lab.
- Laporan

**5. Rental Car**

- Penyelesaian kontrak.
- Penyediaan mobil.
- Pengiriman .
- Pengadaan sopir kontrak.
- Perbaikan, servis Seluruh permasalahan yang terjadi di lapangan(kecelakaan,mogok,dll) .
- Penurusan BPJS TK/kesehatan
- Laporan data keluar/masuk (pengeluaran tak terduga dan terduga).

**6. ATK (alat tulis kantor)**

- Menyelesaikan kontrak dengan rekanan.
- Pembelian, pemesanan alat tulis kantor.
- Pengiriman alat tulis kantor.
- Laporan

**7. Sopir**

- Mengantarkan pelanggan ke tujuan yang diinginkan.
- Mengemudikan kendaraan dengan aman, memperhatikan keselamatan penumpang dan mematuhi peraturan lalu lintas.

- Memberikan informasi dan panduan mengenai tujuan wisata atau lokasi yang dikunjungi kepada pelanggan.
- Bersikap ramah, sopan, dan memberikan layanan pelanggan yang baik.

## BAB III

### TINJAUAN PUSTAKA

#### 3.1. Teori Pendukung

##### 3.1.1. Penyewaan

Menurut Sandra Christiana (2022:18) Penyewaan adalah meminjamkan suatu barang atau jasa, namun harus dibayar terlebih dahulu sebelum bisa digunakan sesuai dengan kontrak perjanjian yang tertera pada peraturan penyedia jasa penyewaan yang ada.

##### 3.1.2. Marketplace

Menurut Rini Yustiani, dkk (2017:3) *Marketplace* adalah pasar virtual dimana para pelaku bisnis dan pembeli melakukan transaksi menggunakan media internet. jenis – jenis interaksi antara pelaku bisnis didalam dunia bisnis dapat dilihat pada tabel 3.1.

**Tabel 3.1** Jenis - Jenis Interaksi di Dunia Bisnis

Nomor	Nama	Penjelasan
1.	B2B ( <i>Business to Business</i> )	Transaksi bisnis antara pelaku bisnis dengan pelaku bisnis lainnya. Dapat berupa kesepakatan spesifik yang mendukung kelancaran bisnis.
2.	B2C ( <i>Business to Consumer</i> )	Aktivitas yang dilakukan produsen kepada konsumen secara langsung.
3.	C2C ( <i>Consumer to Consumer</i> )	Aktivitas bisnis (penjualan) yang dilakukan oleh individu (konsumen) kepada individu (konsumen) lainnya.
4.	C2B ( <i>Consumer to Business</i> )	C2B merupakan model bisnis dimana konsumen (individu) menciptakan dan membentuk nilai akan proses bisnis.

Nomor	Nama	Penjelasan
5.	B2G (Business to Government)	Merupakan turunan dari B2B, perbedaannya proses ini terjadi antara pelaku bisnis dan instansi pemerintah
6.	G2C (Government to Consumer)	Merupakan hubungan atau interaksi antara pemerintah dengan masyarakat. Konsumen, dalam hal ini masyarakat, dapat dengan mudah menjangkau pemerintah sehingga memperoleh kemudahan dalam pelayanan sehari-hari.

Sumber : Mahir Pradana (2015:5)

Menurut Alfiah, Damayanti (2020:113) *Marketplace* adalah wadah komunitas bisnis interaktif secara elektronik yang menyediakan pasar dimana perusahaan dapat mengambil andil dalam B2B *e-commerce* dan atau kegiatan *e-business* lain. Dengan adanya marketplace sebagai wadah bertemunya pemilik bisnis dan calon pelanggan sehingga dapat melakukan transaksi menggunakan media internet. Kemudian menurut Mahir Pradana (2015:5).

### 3.1.3. Business to Business (B2B)

Menurut Mahir Pradana (2015:5) *Business to Business (B2B)* adalah Transaksi bisnis antara pelaku bisnis dengan pelaku bisnis lainnya. Dapat berupa kesepakatan spesifik yang mendukung kelancaran bisnis. Menurut Febri Nova Lenti (2017:42) (B2B) merupakan interaksi perdagangan dengan menggunakan jaringan internet antar pelaku bisnis, yaitu pembeli dari produk yang dijual oleh penjual bukan konsumen akhir namun pabrikasi atau usaha per orangan di mana produk yang dibeli akan diproses lagi atau dijual langsung ke konsumen akhir.

### 3.1.4. Database

Menurut Connoly dan Begg (2015:63) *Database* adalah kumpulan data yang saling terhubung secara logis dan deskripsi dari data tersebut, dirancang untuk menemukan informasi yang dibutuhkan oleh sebuah organisasi. *Database* yang digunakan peneliti adalah MySQL. Menurut Jubille Enterprise (2014:2) MySQL adalah *Relational Database Management System (RDBMS)* yang cepat dan mudah digunakan, serta sudah banyak digunakan untuk berbagai kebutuhan. MySQL bersifat *open source*, merupakan program yang *powerful* dan menyediakan fitur yang lengkap, menggunakan bahasa SQL, dapat bekerja dengan banyak sistem operasi dan bahasa pemrograman lain seperti PHP, C, C++, Java, dll.

### 3.2. Penelitian Terdahulu

Adapun beberapa jurnal terkait dengan penelitian sebagai referensi digunakan peneliti dalam mengkaji penelitian yang dilakukan. Berikut merupakan beberapa penelitian terdahulu dapat dilihat pada tabel 3.2.

**Tabel 3.2** Penelitian Terdahulu

No.	Judul	Peneliti dan Tahun	Hasil
1.	Perancangan Sistem Informasi Marketplace dan Reseller pada PT. XYZ dengan Metode RAD ( <i>Rapid Application Development</i> ). Vol. 14 P-ISSN : 2406-9841.	Dhian Yusuf Al Afghani (2022)	Sistem informasi marketplace dan reseller berbasis web yang dapat mengembangkan bisnis pada PT. XYZ. Dengan sistem yang dirancang memenuhi kebutuhan B2C, B2B, dan C2C, serta sistem yang dirancang dapat diakses oleh 3 level user yang berbeda.



No.	Judul	Peneliti dan Tahun	Hasil
2.	Sistem Informasi E-Marketplace penyedia Jasa Sewa Dekorasi Berbasis Web. Vol.10 E-ISSN : 2354-6654. P-ISSN : 2302-5700.	Uchti Nur Rachman, dkk (2021)	Memudahkan pencari jasa dalam mendapatkan informasi harga dari berbagai penyedia jasa, pencari jasa dapat membandingkan harga penyewaan dekor antara penyedia jasa satu dan yang lainnya, serta menjadi media pemasaran untuk penyedia jasa dekorasi.
3.	Perancangan E-Marketplace Sewa - Menyewa Sebagai Alternatif Bisnis di Era Digital. Vol. 4 E-ISSN : 2443-2210. P-ISSN : 2443-2229.	David Christian Aditya Gunadi, Teddy Marcus Zakaria (2018)	Membuat sebuah aplikasi e-market berbasis website sewa menyewa, dapat menjadi alternatif yang menerapkan proses bisnis yang baik agar keamanan dan transaksi sewa dapat terjamin, dan memudahkan proses sewa menyewa.
4.	Implementasi Metode Rapid Application Development dalam Membangun E-Commerce di Bidang UKM. Vol.13 E-ISSN : 2527-6514. P-ISSN : 1978-1946.	Verry Riyanto (2017)	Dengan metode RAD sangat membantu dalam membangun aplikasi berbasis web, memungkinkan programmer untuk bekerja lebih efisien dalam waktu pekerjaan, E-commerce yang dibuat dapat diakses kapanpun dan dimanapun melalui jaringan internet, memperkecil kemungkinan kesalahan data, pembuatan laporan menjadi lebih mudah.

Sumber : Peneliti

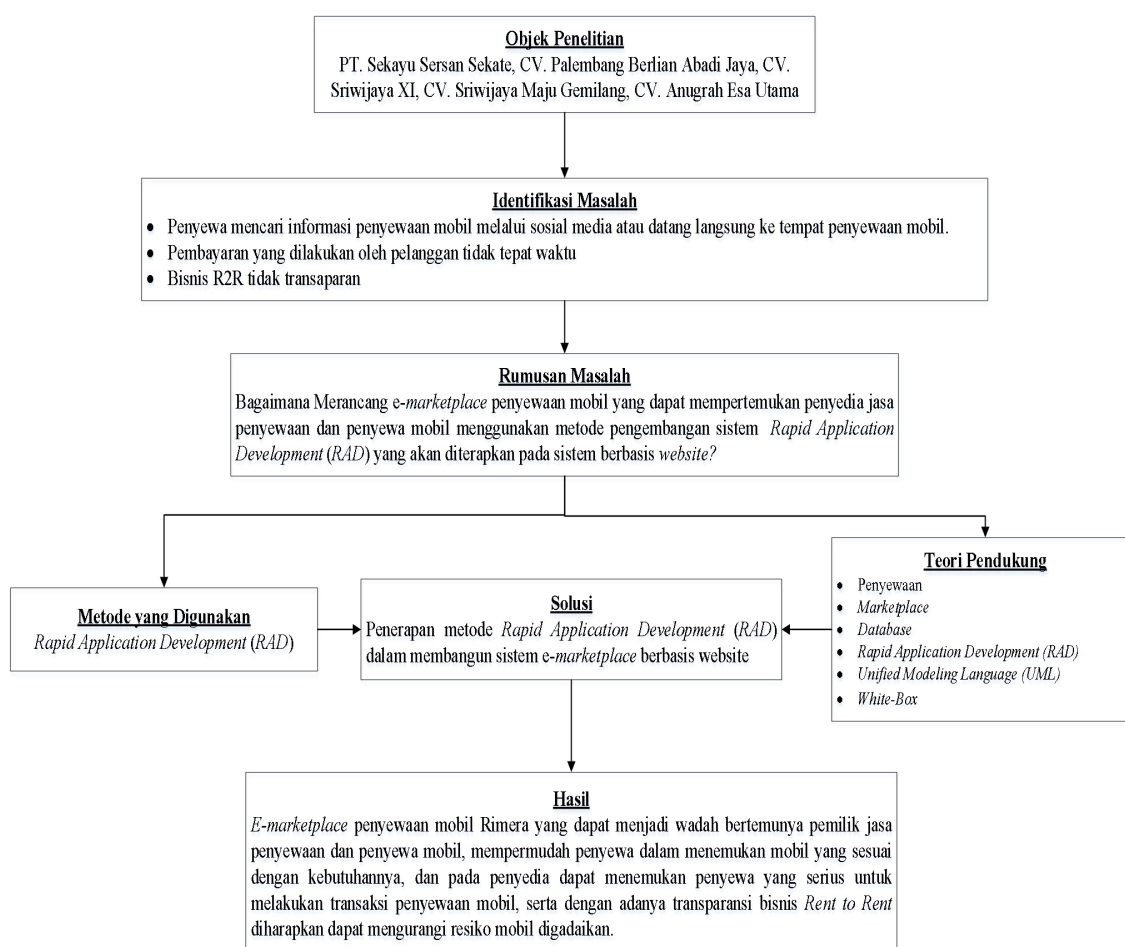
Penelitian terdahulu ini menjadi pembandingan antara penelitian yang pernah ada dan yang akan peneliti lakukan penelitian tersebut sebagai berikut :

- 1) Pada penelitian Dhian Yusuf Al Afghani (2022) yang berjudul Perancangan Sistem Informasi Marketplace dan Reseller pada PT. XYZ dengan Metode RAD (*Rapid Application Development*), tujuan dari penelitian ini adalah untuk menghasilkan *marketplace* berbasis web yang dapat digunakan oleh PT. XYZ dalam mengembangkan bisnisnya. Aplikasi ini memiliki kesamaan dengan web yang sedang peneliti rancang yaitu menggunakan metode RAD sebagai teknik pengembangan sistem, memiliki 3 user, digunakan untuk kebutuhan bisnis B2B, dan B2C sehingga dapat menjadi acuan bagi peneliti dalam pembuatan *e-marketplace*.
- 2) Pada penelitian Uhti Nur Rachman, dkk (2021) yang berjudul Sistem Informasi *E-Marketplace* penyedia Jasa Sewa Dekorasi Berbasis Web, aplikasi ini bertujuan untuk memberikan solusi untuk membantu pencari jasa dalam mendapatkan informasi harga dari berbagai penyedia jasa yang dapat menjadi acuan bagi peneliti dalam pembuatan *e-marketplace*.
- 3) Pada penelitian David Christian Aditya Gunadi, Teddy Marcus Zakaria (2018) yang berjudul Perancangan *E-Marketplace* Sewa - Menyewa Sebagai Alternatif Bisnis di Era Digital, pada penelitian ini bertujuan untuk menjadi wadah dimana transaksi sewa dapat berjalan secara online. Aplikasi ini memiliki kesamaan dengan web yang sedang peneliti rancang yaitu adanya pengelompokan barang per kategori, keterangan barang yang akan disewa agar sesuai dengan keinginan pelanggan.

- 4) Pada penelitian Verry Riyanto (2017) yang berjudul *Implementasi Metode Rapid Application Development dalam Membangun E-Commerce di Bidang UKM*, pada penelitian ini dapat diketahui bahwa metode RAD sangat membantu dalam membangun sebuah aplikasi berbasis web sehingga dapat menjadi acuan peneliti dalam membangun *e-marketplace*.

### 3.3. Kerangka Pemikiran

Berikut merupakan hasil dari kerangka pemikiran yang dapat dilihat pada gambar 3.1.



Sumber : Peneliti

**Gambar 3.1** Kerangka Pemikiran

Adapun penjelasan dari kerangka pemikiran diatas adaalah sebagai berikut :

1. Peneliti melakukan penelitian pada 5 jasa penyewaan mobil di kota Palembang, yaitu : PT. Sekayu Sersan Sekate, CV. Palembang Berlian Abadi Jaya, CV. Sriwijaya XI, CV. Sriwijaya Maju Gemilang, CV. Anugrah Esa Utama.
2. Peneliti menemukan kendala pada bisnsi penyewaan mobil seperti : penyewa sulit menemukan informasi harga penyewaan mobil, pembayaran yang tidak tepat waktu, tidak adanya transparansi pada bisni *Rent to Rent*.
3. Adapun rumusan masalah penelitian ini adalah Bagaimana Merancang *e-marketplace* penyewaan mobil yang dapat mempertemukan penyedia jasa penyewaan dan penyewa mobil menggunakan metode pengembangan sistem *Rapid Application Development (RAD)* yang akan diterapkan pada sistem berbasis *website*.
4. Penulis membangun *E-Marketplace* penyewaan mobil Rimera dengan metode pengembangan sistem *Rappid Application Development (RAD)*, dan dengan beberapa teori pendukung.
5. Hasil yang didapatkan dari penelitian ini adalah *E-Marketplace* penyewaan mobil rimera.

## BAB IV

### METODE PENELITIAN

#### 4.1. Lokasi dan Waktu Penelitian

##### 4.1.1. Lokasi

Penelitian Ini dilakukan pada 5 perusahaan penyedia jasa penyewaan mobil dengan rincian alamat yang dapat dilihat pada tabel 4.1.

**Tabel 4.1** Alamat Penyedia Jasa Penyewaan Mobil

No.	Nama Perusahaan	Alamat
1	PT. Sekayu Serasan Sekate	Jl. Naskah RSS Bukit Sederhana, RT. 034, RW. 012, Blok D1 No. 1, Sukarami, Palembang, Sumatera Selatan.
2	CV. Palembang Berlian Abadi Jaya	Jl. Pesantren SMB II, Perumahan Dinda Residence Blok A2/B1, RT. 009, RW. 005, Palembang, Sumatera Selatan.
3	CV. Sriwijaya XI	Jl. Tembusan Soak Simpur, RT. 072, RW. 001, No. 2167, Sukajaya, Sukarame, Palembang, Sumatera Selatan.
4	CV. Sriwijaya Maju Gemilang	Jl. Tanjung Api – Api, Komplek Tanjung Harapan indah No. 3, Palembang, Sumatera Selatan.
5	CV. Anugrah Esa Utama	Lr. Sepakat, RT. 032, RW. 011, Kec. Alang – alang lebar, Palembang, Sumatera Selatan.

*Sumber : Pemilik Jasa Penyewaan Mobil*

##### 4.1.3. Waktu Penelitian

Dalam penelitian ini peneliti melaksanakan penelitian dalam 5 bulan dimulai pada bulan Maret 2023 sampai bulan Juli 2023. Adapun jadwal penelitian dapat dilihat pada tabel 4.2:



mobil yang sedang berjalan, dan kendala yang ada yang dapat digunakan untuk bahan materi pembuatan *e-marketplace* penyewaan mobil.

#### **4.2.3. Data Sekunder**

Menurut Massie (2017:4489), data sekunder adalah data yang diperoleh secara tidak langsung melalui media perantara yaitu melalui hasil-hasil penelitian, buku-buku, artikel dan berbagai publikasi serta instansi terkait yang relevan dengan masalah yang diangkat. Peneliti mendapatkan data sekunder dengan cara membaca jurnal dan laporan yang berkaitan dengan *e-marketplace* dan sistem informasi penyewaan mobil. Contoh data sekunder dapat kita ambil referensi untuk peneliti yaitu data dari berbagai jurnal terdahulu, buku, dokumen, internet. Pada penelitian ini peneliti mendapatkan data sekunder dari beberapa judul yang dijadikan referensi dan kiat – kiat untuk menyelesaikan penelitian.

### **4.3. Teknik Pengumpulan Data**

Dalam penelitian laporan ini peneliti menggunakan beberapa metode teknik pengumpulan data yang terdiri dari :

#### **4.3.1. Wawancara**

Menurut Surya Ramadhan (2019:94) wawancara adalah sebuah teknik pengumpulan data dengan melakukan tanya jawab langsung kepada narasumber sehingga menghasilkan data, dan informasi yang dibutuhkan.

Adapun kegiatan wawancara dalam laporan ini adalah dengan penyedia jasa penyewaan mobil mengenai masalah dan kendala yang terjadi pada badan usaha tersebut.

### **4.3.2. Pengamatan (Observasi)**

Menurut Fahrival, Sentosa Pohan, dan Marnis Nasution (2018:20) pengamatan adalah sebuah teknik pengumpulan data dengan melakukan penelitian secara langsung dengan tujuan untuk mengetahui dan memahami cara kerja sistem yang sedang berjalan.

Adapun kegiatan pengamatan dalam laporan ini adalah pengamatan secara langsung mengenai sistem penyewaan mobil pada perusahaan penyedia jasa penyewaan mobil.

### **4.3.3. Studi Pustaka**

Menurut Milla Tunnah Imah dan Budi Purwoko (2018:13) studi pustaka adalah suatu usaha mengumpulkan data dengan penelaahan terhadap buku, jurnal, catatan, dan laporan yang relevan dengan fokus penelitian.

Adapun kegiatan studi pustaka ini adalah dengan cara membaca jurnal, dan laporan yang berkaitan dengan *e-marketplace* dan sistem informasi penyewaan mobil.

## **4.4. Alat dan Teknik Pengembangan Sistem**

Alat – alat pengembangan sistem yang digunakan dalam suatu metodologi umumnya berupa gambar atau diagram alir atau grafik. Alat pengembangan sistem yang peneliti gunakan adalah sebagai berikut :

### **4.4.1. Alat Pengembangan Sistem**

#### **4.4.1.1. *Unified Modeling Language (UML)***

Menurut Aditama (2017:23), *Unified Modeling Language (UML)* merupakan sebuah bahasa spesifikasi standar yang memvisualisasikan




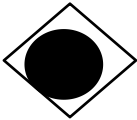

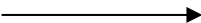
berdasarkan gambar untuk menspesifikasikan, membangun dan mendokumentasikan dari sebuah sistem pengembang *software*. *UML* memiliki 2 jenis diagram. *UML* memiliki beberapa tahapan sebagai berikut :

### 1) *Activity Diagram*

Menurut Indah Kusuma Dewi, dkk (2018:30), *activity diagram* menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis.

Simbol *activity diagram* dapat dilihat pada tabel 4.3 berikut :

**Tabel 4.3** Simbol *Activity Diagram*

No	Simbol	Keterangan
1.		<i>Activities</i> , menggambar kan suatu proses/kegiatan bisnis.
2.		<i>Decision Points</i> , menggambar kanpilihan untuk pengambilan keputusan, true atau false.
3.		<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
4.		<i>End Point</i> , akhir aktivitas.
5.		Kondisi transisi Menunjukkan kondisi transisi antar aktivitas.

Sumber : Indah Kusuma Dewi, dkk (2018:30)

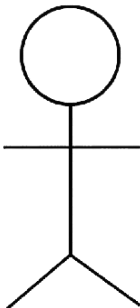
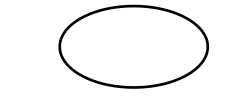

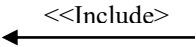
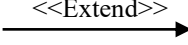
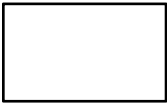
### 2) *Use Case Diagram*

Menurut Indah Kusuma Dewi, dkk (2018:30), *Use case diagram* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dengan menggunakan *use case* dapat diketahui apa

saja fungsi dan siapa saja yang berhak menggunakan sistem yang telah dibuat.

Simbol *use case* diagram dapat dilihat pada tabel 4.4 berikut :

**Tabel 4.4** Simbol *Use Case Diagram*

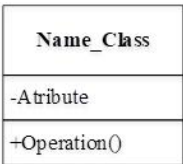
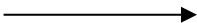

No	Simbol	Keterangan
1.		<i>Actor</i> atau aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran.
2.		<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor.
3.		<i>Association</i> menunjukkan hubungan antara aktor dengan dan <i>use case</i> atau antar <i>use case</i> .
4.		Generalisasi menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> .
5.		Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya.
6.		Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi.
7.		Sistem menspesifikasikan paket yang menampilkan sistem secara terbatas.

Sumber : Indah Kusuma Dewi, dkk (2018:30),

### 3) *Class Diagram*

Menurut Indah Kusuma Dewi, dkk (2018:31), *class diagram* adalah sebuah spesifikasi yang jika diinisiasati akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan-hubungan objek. Simbol *class diagram* dapat dilihat pada tabel 4.5 berikut :

**Tabel 4.5** Simbol *Class Diagram*

No	Simbol	Keterangan
1.	1	Satu dan hanya satu.
2.	0..*	Boleh tidak ada atau 1 atau lebih.
3.	1..*	1 atau lebih.
4.	0..1	Boleh tidak ada, maksimal 1.
5.	n..n	Batasan antara, contoh 2..4 mempunyai arti minimal 2 maksimal 4.
6.		<i>Class</i> Menunjukkan <i>class-class</i> yang dibangun berdasarkan proses-proses sebelumnya.
7.		<i>Unidirectional Association</i> Menunjukkan hubungan antara class pada diagram class.
8.		<i>Aggregation Relasi</i> antar kelas dengan makna semua-bagian ( <i>whole-part</i> ).

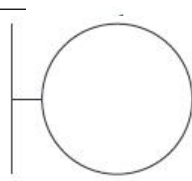

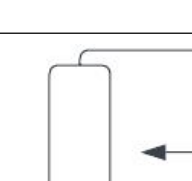
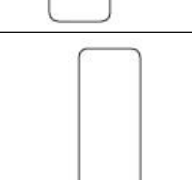
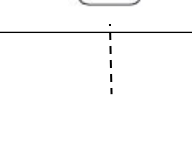


Sumber : Indah Kusuma Dewi, dkk (2018:31)

### 4) *Sequence Diagram*

Menurut Indah Kusuma Dewi, dkk (2018:32) *sequence diagram* menggambarkan kelakuan objek pada use case dengan

mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol *sequence diagram* dapat dilihat pada tabel 4.6 berikut :

**Tabel 4.6** Simbol *Sequence Diagram*

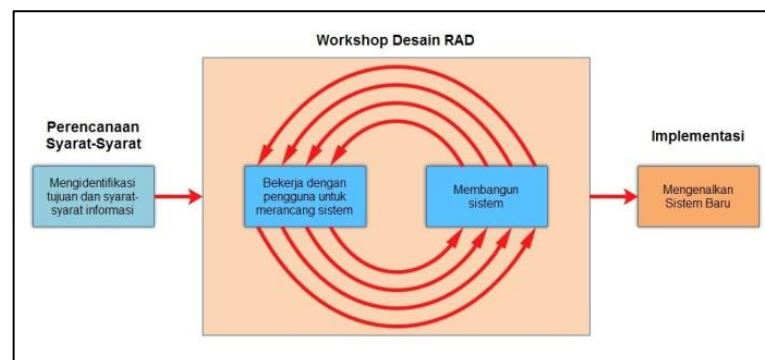
No	Simbol	Keterangan
1.		<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
2.		<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi interface atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan form cetak.
3.		<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
4.		<i>Message</i> , simbol mengirim pesan antar class.
5.		<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untukdirinya sendiri.
6.		<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
7.		<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i> .

Sumber : Indah Kusuma Dewi, dkk (2018:32)

#### 4.5.2. Teknik Pengembangan Sistem

Menurut Darma Setiawan Putra, dan Ami Fauziah (2018:168) *Rapid Application Development (RAD)* merupakan model adaptasi dari model air terjun untuk pengembangan setiap komponen perangkat lunak. Menurut Yayuk Ike Melani dan Mahmud (2020:5) metode RAD adalah metode pengembangan sistem yang bertujuan untuk mempersingkat waktu dalam perencanaan, perancangan dan penerapan suatu sistem. Sehingga dengan menerapkan metode ini diharapkan dapat menghasilkan suatu sistem yang cepat karena dikembangkan sesuai dengan keinginan pengguna.

Adapun tahapan metode RAD menurut Darma Setiawan Putra, dan Ami Fauziah (2018:168) dapat dilihat pada gambar 4.1 sebagai berikut :



Sumber : Darma Setiawan Putra, dan Ami Fauziah (2018:168)

**Gambar 4.1** Tahapan Metode *Rapid Application Development (RAD)*

Adapun beberapa penjelasan tahapan dari metode *RAD* menurut Darma Setiawan Putra, dan Ami Fauziah (2018:168) sebagai berikut :

##### 1) Rencana Kebutuhan (*Requirement Planning*)

Pada tahap ini pengguna dan penganalisis bertemu untuk mengidentifikasi tujuan-tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat informasi. Pada fase ini peneliti melakukan

pengumpulan data dengan cara wawancara dan observasi sebagai langkah awal penelitian. Dalam langkah ini peneliti melakukan analisa dan wawancara langsung dengan penyedia jasa penyewaan mobil untuk menentukan kebutuhan sistem. Peneliti menganalisa kegiatan penyewaan mobil untuk mengetahui masalah yang ada pada badan usaha tersebut sehingga bisa menentukan akan membuat sistem seperti apa.

## **2) Proses Desain Sistem (*Design System*)**

Pada tahap ini keaktifan *user* yang terlibat menentukan untuk mencapai tujuan karena pada proses ini melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan *analyst*. Seorang *user* dapat langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain, merancang sistem dengan mengacu pada dokumentasi kebutuhan *user* yang dibuat pada tahap sebelumnya. Keluaran dari tahapan ini adalah spesifikasi *software* yang meliputi organisasi sistem secara umum, struktur data dan yang lain. Pada proses ini peneliti menentukan perancangan desain sistem yang akan dibangun berdasarkan analisa kebutuhan yang telah didapatkan. Mengenai pemodelan yang dilakukan peneliti berupa *use case diagram*, *activity diagram*, *class diagram* dan *sequence diagram*.

## **3) Implementasi (*Implementation*)**

Tahapan ini adalah tahapan programmer yang mengembangkan desain suatu program yang telah disetujui oleh *user* dan *analyst*. Sebelum diaplikasikan pada suatu organisasi terlebih dahulu dilakukan proses pengujian terhadap program tersebut apakah ada kesalahan atau tidak. Pada tahap ini *user* biasa

memberikan tanggapan akan sistem yang sudah dibuat serta mendapat persetujuan mengenai sistem tersebut. Pada tahap ini peneliti menerapkan perancangan model sistem yang telah dibuat dalam kode program. Peneliti menggunakan pemrograman menggunakan laravel versi 10.18.0 dan *database* menggunakan MySQL. Pada tahap ini peneliti juga akan melakukan testing menggunakan *white box* untuk memeriksa dan menguji kode program yang dibuat, serta dilakukan pengenalan dan training sistem *marketplace* kepada pemilik jasa penyewaan mobil.

#### **4.5. Alat dan Teknik Pengujian**

Dalam penelitian laporan ini peneliti menggunakan teknik pengujian sistem berupa *white box*. Menurut Supriyanto, Iskandar Fitri dan Nurhayati (2022:230) *White-box testing* adalah suatu metode pengujian sistem dengan cara menganalisis apakah ada yang salah atau tidak pada kode suatu program. Menurut Rahmaria Yunisa, Hanson Prihantoro Putro (2019:197) *White-box testing* adalah teknik pengujian yang digunakan untuk memeriksa struktur program dan mengambil data uji dari logika kode program yang dibuat.

Pada penelitian ini, peneliti menggunakan metode pengujian perangkat lunak *white box* dengan teknik *Statement Coverage*. Menurut Rahmaria Yunisa dan Hanson Prihantoro Putro (2020:205) metode *statement coverage* adalah teknik pengujian dengan melakukan pengujian berdasarkan eksekusi pada setiap pernyataan kode program, yang dilakukan pengujian pada *function – function* atau *method* dalam *class*. Pengujian dilakukan dengan harapan dapat memastikan

setiap pernyataan telah diuji dan mencakup pada semua pernyataan ketika dijalankan pengujian dengan setidaknya satu kali test.

Pengujian dilakukan dengan cara melihat *source code* yang ada pada sistem dan menentukan fungsi *use case* yang memiliki *if* untuk memeriksa kondisi telah dieksekusi dengan tepat sehingga menghasilkan keluaran yang valid. Menurut Yunisa (2018:103) tahapan yang dilakukan dalam pengujian ini adalah :

- 1) Menentukan atau memilih fungsi dalam kelas-kelas yang tersedia.
- 2) Menghitung jumlah statement (baris kode) dalam setiap fungsi.
- 3) Menghitung jumlah kondisi dalam setiap fungsi.
- 4) Menentukan kode program yang akan diuji.
- 5) Mengkonversikan *source code* ke dalam bentuk *flowgraph* (notasi lingkaran) untuk menggambarkan statement.
- 6) Menghitung nilai *cyclomatic complexity* untuk menentukan jumlah jalur yang melewati pada statement.
- 7) Membuat rancangan data uji dengan mengidentifikasi setiap jalur.

Setelah semua tahap melewati selanjutnya adalah mengukur tingkat keberhasilan pengujian dengan teknik *statement coverage*. Teknik ini dilakukan dengan menjalankan data uji yang mencakup semua *statement* yang dijalankan. Untuk mencari nilai 100% pada *statement coverage* maka dilakukan perhitungan dengan rumus sebagai berikut:

$$V(G) = R = E - N + 2 = \dots$$

Dimana  $V(G)$  adalah *cyclomatic complexity graph*, R adalah jumlah *region* dalam program *flowgraph*, E adalah jumlah *edge*, dan N adalah jumlah *node*.



## **BAB V**

### **HASIL DAN PEMBAHASAN**

#### **5.1. Hasil Penelitian**

Selama melakukan penelitian mengenai perancangan *Marketplace* Penyewaan Mobil (Studi Kasus : Kota Palembang) dengan data dari 5 penyedia jasa penyewaan mobil, didapatkan hasil yang akan dijelaskan dengan menggunakan metode pengembangan sistem yaitu *Rapid Application Development (RAD)*. Adapun tahapan – tahapan dari metode RAD adalah sebagai berikut :

##### **5.1.1. Rencana Kebutuhan (*Requirement Planning*)**

Pada tahap ini peneliti dan penyedia jasa penyewaan mobil melakukan pertemuan untuk mengidentifikasi tujuan dari sistem dan kebutuhan informasi. Adapun langkah yang dilakukan adalah :

- 1) Melakukan wawancara langsung dengan penyedia jasa penyewaan mobil mengenai masalah yang ada pada perusahaan mereka, khususnya website untuk penyewaan mobil, didapatkan hasil 1 penyedia jasa penyewaan mobil sudah memiliki website penyewaan mobil namun hanya sebatas informasi mobil yang mereka sediakan, untuk sistem penyewaan tetap harus melalui WhatsApp yang tersedia pada *website*, dan 4 lainnya belum memiliki website. Didapatkan pula masalah pada sistem pembayaran yang sering tidak tepat waktu, sistem penyewaan R2R merahasiakan penyewa yang melakukan transaksi sehingga tidak mengetahui data pelanggan yang melakukan penyewaan.

- 2) Melakukan observasi dengan mengamati secara langsung alur kerja dan proses penyewaan mobil pada penyedia jasa penyewaan mobil.
- 3) Melakukan dokumentasi dalam penelitian ini yang dapat digunakan untuk memperoleh data berbentuk dokumentasi.
- 4) Melakukan studi pustaka tentang pembuatan sistem berbasis *website*.

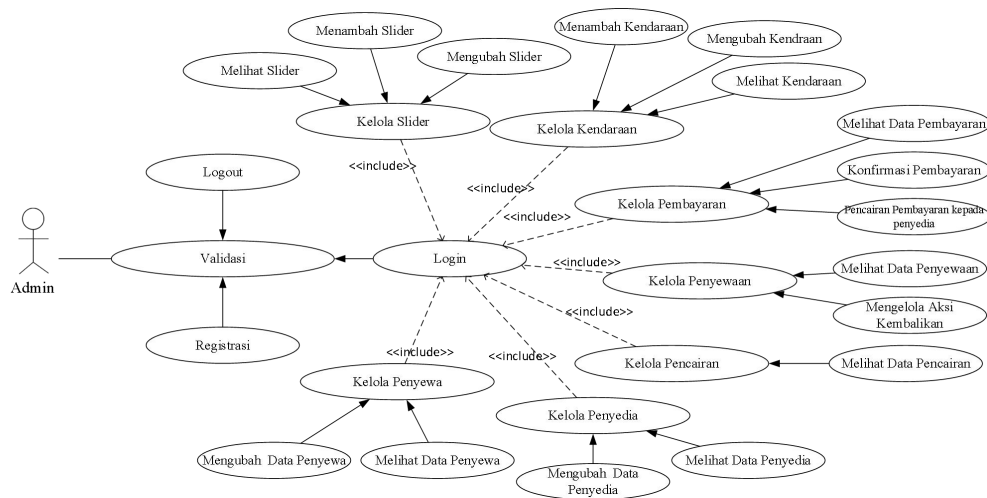
### **5.1.2. Desain Sistem (*Design System*)**

Adapun desain sistem yang peneliti gunakan adalah dengan menggunakan UML. Peneliti menggunakan UML untuk menggambarkan dan mendeskripsikan interaksi yang terjadi antara pengguna dan sistem. Menurut Djodi Maulana (2022:3) dengan menggunakan pemodelan UML sebagai permodelan visual yang mudah dipahami serta dilengkapi dengan pemodelan efektif mudah dikomunikasikan untuk berbagi. UML terdiri dari 13 pemodelan visual perancangan aplikasi ini peneliti hanya menggunakan 4 perancangan UML sebagai berikut :

#### **5.1.2.1. Use Case Diagram**

*Use case diagram E-Marketplace* penyewaan mobil yang terdiri dari admin, penyedia, dan penyewa. Adapun use case diagram dapat dilihat pada 5.1, 5.2, dan 5.3.

Berdasarkan *use case diagram* admin gambar 5.1 dibawah dijelaskan bahwa admin setelah membuka e-marketplace penyewaan mobil akan melakukan akses *login* dengan gambaran berikut :



Sumber : Peneliti

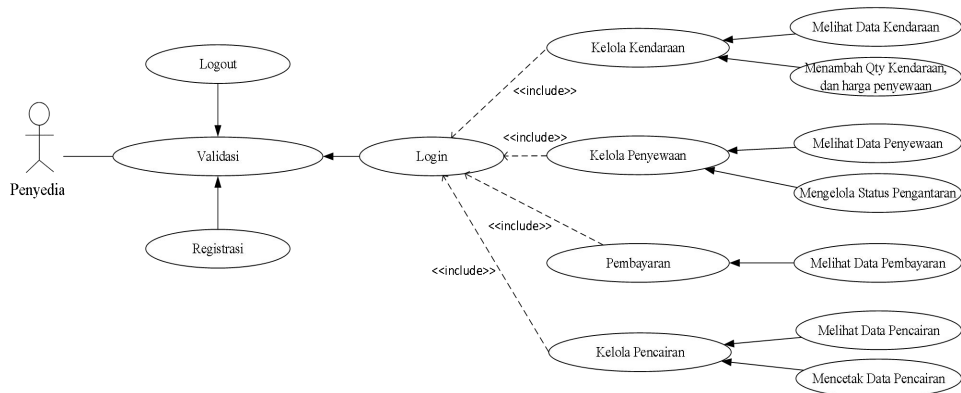
**Gambar 5.1** Use Case Diagram Admin

- 1) Mengelola slider yang berisi aktivitas menginput, mengubah, dan menghapus gambar yang akan berganti secara otomatis pada *e-marketplace* penyewaan mobil.
- 2) Mengelola kendaraan yang berisi aktivitas melihat, dan menginput data mobil yang tersedia untuk disewa. Pada halaman ini admin menginput gambar mobil, merk/type, transmisi, tahun pembuatan, dan harga penyewaan mobil pada *e-marketplace* penyewaan mobil sedangkan qty yang tersedia pada halaman kendaraan adalah hasil dari penginputan yang dilakukan oleh penyedia.
- 3) Mengelola pembayaran yang berisi aktivitas melihat data pembayaran, pengecekan dan konfirmasi pembayaran yang telah dilakukan oleh penyewa, pencairan dana kepada penyedia.
- 4) Mengelola penyewaan yang berisi aktifitas melihat data penyewaan, pada beberapa kasus seperti penyewa lupa mengklik aksi kembalikan mobil padahal mobil telah dikembalikan ke penyedia dalam keadaan baik, maka

admin dapat langsung mengkonfirmasi pengembalian mobil pada halaman ini.

- 5) Mengelola pencairan yang berisi aktivitas melihat data pencairan dana hasil penyewaan mobil kepada penyedia.
- 6) Mengelola penyedia yang berisi aktivitas melihat data penyedia, dan mengubah data penyedia.
- 7) Mengelola penyewa yang berisi aktivitas melihat data penyewa, dan mengubah data penyewa.

Berdasarkan *use case diagram* penyedia gambar 5.2 dibawah dijelaskan bahwa penyedia setelah membuka e-marketplace penyewaan mobil akan melakukan akses *login* dengan gambaran berikut :

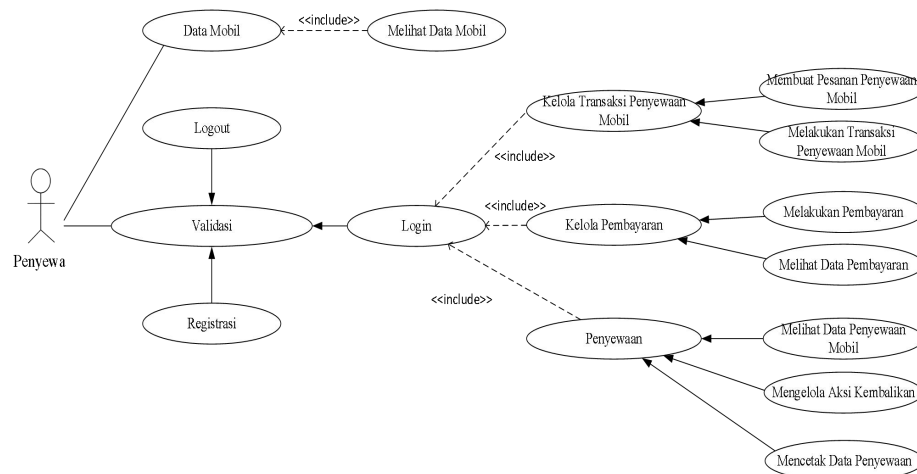


Sumber : Peneliti

**Gambar 5.2 Use Case Diagram Penyedia**

- 1) Mengelola kendaraan yang berisi aktivitas melihat data kendaraan, menambah qty kendaraan, dan menginput harga penyewaan mobil.
- 2) Mengelola penyewaan yang berisi aktivitas melihat data penyewaan, dan mengubah status pengantaran.

- 3) Mengelola pembayaran yang berisi aktivitas melihat data pembayaran yang telah dilakukan oleh penyewa.
- 4) Mengelola pencairan yang berisi aktivitas melihat data pencairan, dan mencetak data pencairan.



Sumber : Peneliti

**Gambar 5.3** Use Case Diagram Penyewa

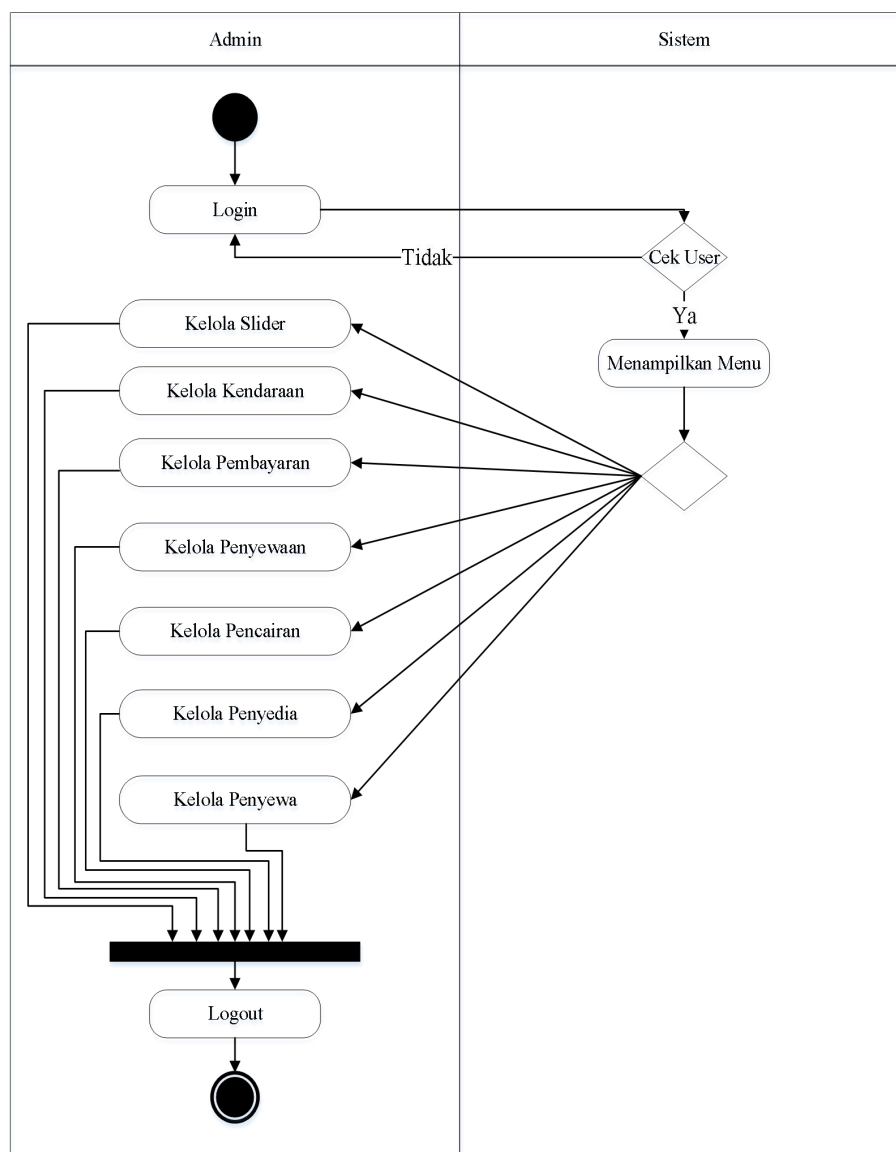
Berdasarkan *use case diagram* penyewa gambar 5.3 dibawah dijelaskan bahwa penyewa setelah membuka e-marketplace penyewaan mobil akan melakukan akses *login* dengan gambaran berikut :

- 1) Mengelola data mobil yang berisi aktivitas melihat data mobil yang tersedia untuk disewa.
- 2) Mengelola transaksi yang berisi aktivitas membuat pesanan penyewaan mobil, melakukan transaksi penyewaan mobil.
- 3) Mengelola pembayaran yang berisi aktivitas melihat data pembayaran, dan membayar transaksi penyewaan mobil.
- 4) Mengelola penyewaan, melihat data penyewaan, mengelola aksi kembalikan mobil sewaan, mencetak data penyewaan mobil.

### 5.1.2.2. Activity Diagram

Activity Diagram adalah diagram yang menggambarkan aktivitas kerja pada *E-Marketplace* penyewaan mobil Rimera. Terdapat 3 *activity diagram* yang dapat dilihat pada gambar 5.4, 5.5, dan 5.6.

Berdasarkan *activity diagram* 5.4 menggambarkan alur dari proses sistem yang dilakukan oleh admin pada e-marketplace penyewaan mobil.

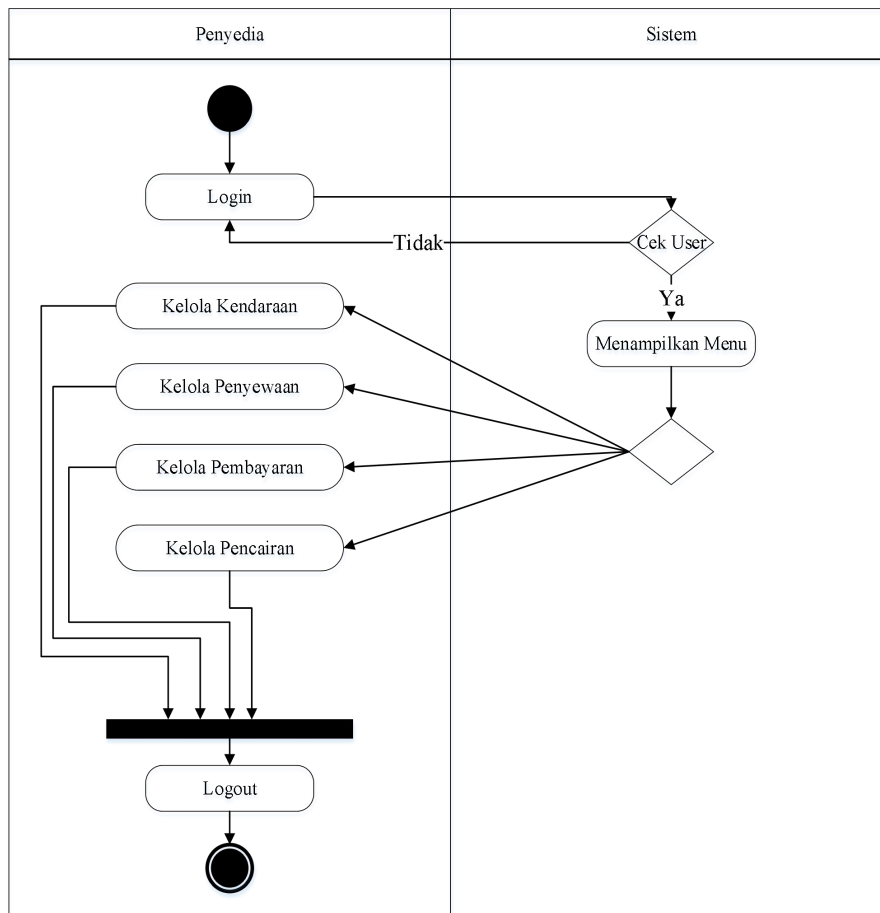


Sumber : Peneliti

**Gambar 5.4** Activity Diagram Admin

Adapun dari gambar diatas adalah sebagai berikut :

- 1) Admin melakukan *login* pada *e-marketplace* penyewaan mobil.
- 2) Apabila tidak berhasil maka akan kembali ke halaman *login*, apabila berhasil maka admin dapat menampilkan menu kelola slider, kelola kendaraan, kelola pembayaran, kelola penyewaan, kelola pencairan, kelola penyedia, dan kelola penyewa.
- 3) Kemudian dari aktivitas tersebut admin dapat melakukan *logout* dari akun yang telah *login*.

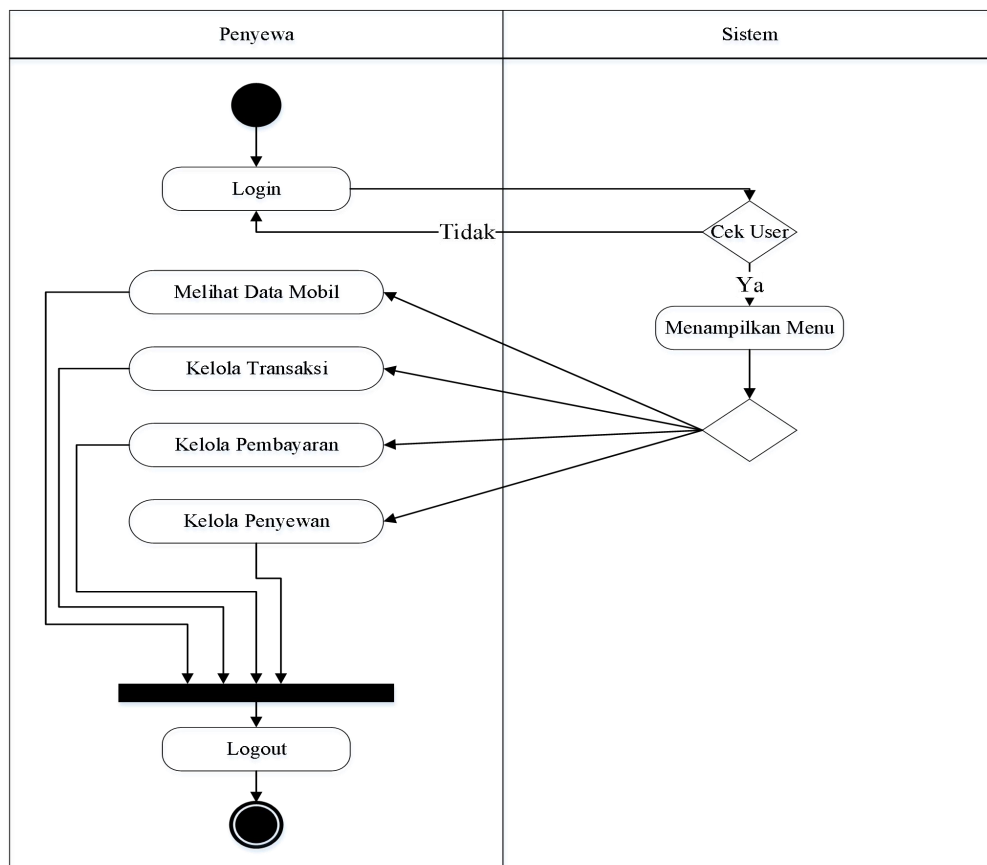


Sumber : Peneliti

**Gambar 5.5** Activity Diagram Penyedia

Berdasarkan *activity diagram* penyedia gambar 5.5 diatas menggambarkan alur dari proses sistem yang dilakukan oleh penyedia pada e-marketplace penyewaan mobil. Adapun penjelasannya adalah sebagai berikut :

- 1) Penyedia melakukan *login* pada *e-marketplace* penyewaan mobil.
- 2) Apabila tidak berhasil maka akan kembali ke halaman *login*, apabila berhasil maka penyedia dapat menampilkan menu kelola kendaraan, kelola pembayaran, kelola penyewaan, dan kelola pencairan.
- 3) Kemudian dari aktivitas tersebut penyedia dapat melakukan *logout* dari akun yang telah *login*.



Sumber : Peneliti

**Gambar 5.6** *Activity Diagram* Penyewa

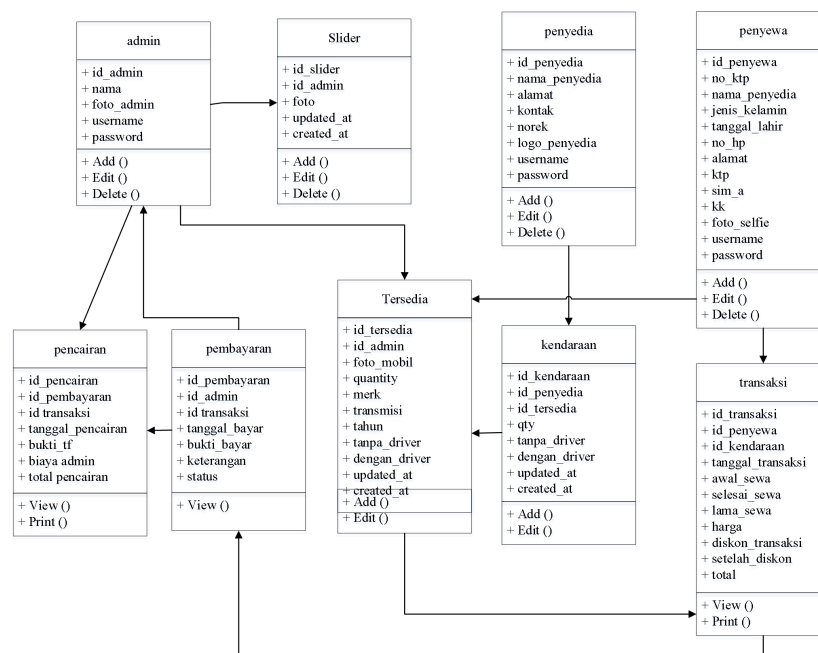


Berdasarkan *activity diagram* penyewa gambar 5.6 diatas digambarkan alur dari proses sistem yang dilakukan oleh penyewa pada e-marketplace penyewaan mobil. Adapun penjelasannya adalah sebagai berikut :

- 1) Penyewa melakukan *login* pada *e-marketplace* penyewaan mobil.
- 2) Apabila tidak berhasil maka akan kembali ke halaman *login*, apabila berhasil maka penyedia dapat menampilkan menu melihat data mobil, kelola transaksi, kelola pembayaran, dan kelola penyewaan.
- 3) Kemudian dari aktivitas tersebut penyewa dapat melakukan logout dari akun yang telah login.

### 5.1.2.3. Class Diagram

*Class diagram* menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan E-Marketplace Penyewaan mobil Rimera. Adapun *class diagram* dapat dilihat pada gambar 5.7 dibawah ini :



Sumber : Peneliti

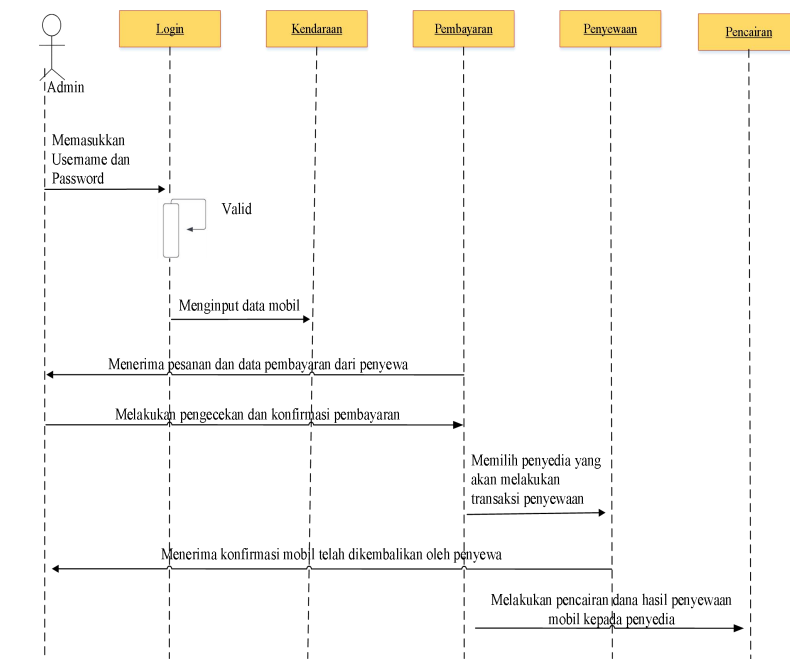
**Gambar 5.7** Class Diagram E- Marketplace Penyewaan mobil

Berdasarkan *class diagram* gambar 5.7 diatas digambarkan isi dari class yang ada pada *e-marketplace* penyewaan mobil, berikut adalah penjelasannya :

1. Admin berhubungan dengan tabel slider, tersedia, pembayaran, dan pencairan.
2. Penyedia berhubungan dengan tabel kendaraan.
3. Penyewa berhubungan dengan tabel tersedia, dan tabel transaksi.

#### 5.1.2.4. *Sequence Diagram*

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

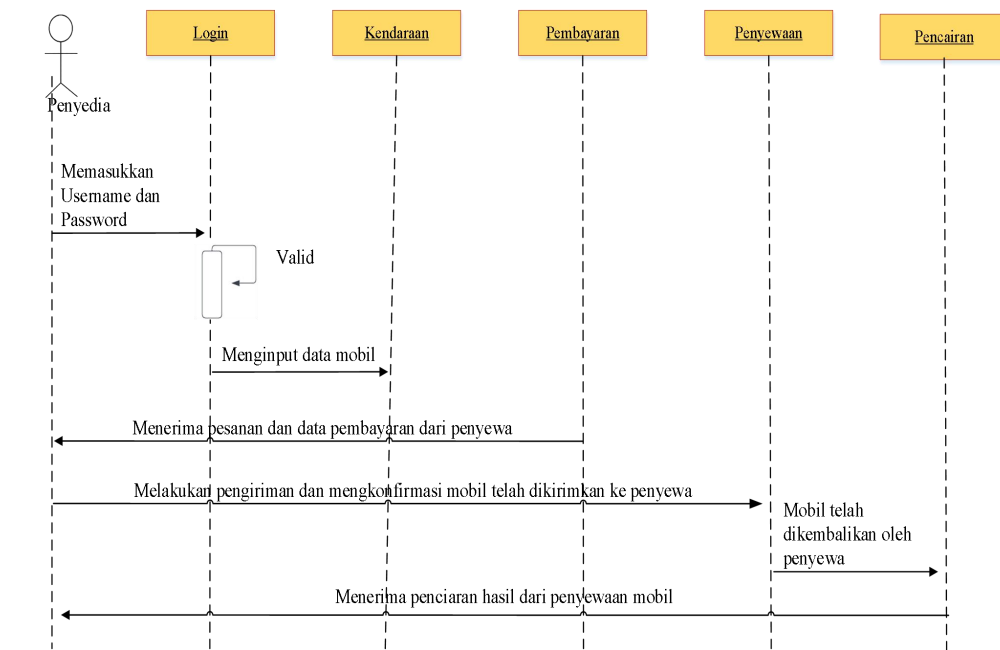


Sumber : Pnulis

**Gambar 5.8** *Sequence Diagram Admin*

Berdasarkan *sequence diagram* pada gambar 5.8 diatas digambarkan alur dari proses sistem login yang dilakukan oleh aktor admin pada *e-marketplace* penyewaan mobil, adapun penjelasannya dalah sebagai berikut :

- 1) Aktor admin harus *login* untuk dapat melakukan aktivitas pada *e-marketplace* penyewaan mobil.
- 2) Admin menginput data mobil.
- 3) Admin menerima pesanan penyewaan dan pembayaran dari penyewa.
- 4) Admin mengecek dan mengkonfirmasi pembayaran.
- 5) Admin memilih penyedia yang akan melakukan transaksi penyewaan mobil.
- 6) Admin menerima konfirmasi mobil telah dikirimkan kepada penyewa.
- 7) Admin menerima konfirmasi mobil telah dikembalikan oleh penyewa.
- 8) Admin melakukan pencairan dana hasil penyewaan mobil kepada penyedia.

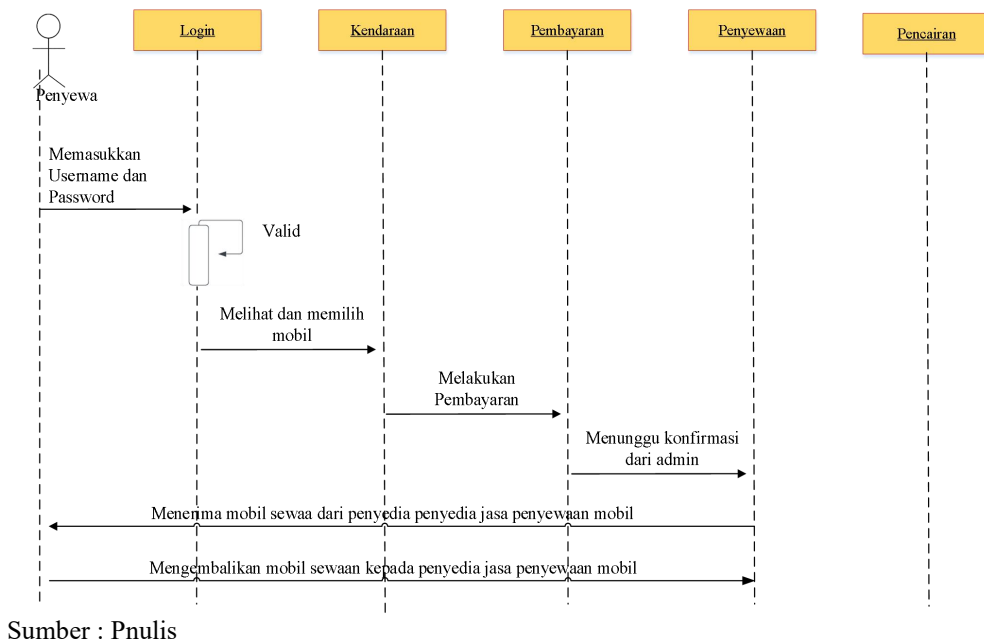


Sumber : Pnulis

**Gambar 5.9** *Sequence Diagram* Penyedia

Berdasarkan *sequence diagram* pada gambar 5.9 diatas digambarkan alur dari proses sistem login yang dilakukan oleh aktor penyedia pada e-marketplace penyewaan mobil, adapun penjelasannya dalah sebagai berikut :

- 1) Aktor penyedia harus *login* untuk dapat melakukan aktivitas pada *e-marketplace* penyewaan mobil.
- 2) Penyedia menginput data mobil.
- 3) Penyedia menerima data penyewaan dari penyedia.
- 4) Penyedia mengirimkan mobil sewaan sesuai dengan alamat pengiriman.
- 5) Penyedia mengkonfirmasi pengembalian mobil sewaan.
- 6) Penyedia menerima pencairan hasil penyewaan mobil.



**Gambar 5.10** *Sequence Diagram* Penyewa

Berdasarkan *sequence diagram* pada gambar 5.10 diatas digambarkan alur dari proses sistem login yang dilakukan oleh aktor penyewa pada *e-marketplace* penyewaan mobil, adapun penjelasannya dalah sebagai berikut :

- 1) Aktor penyewa harus *login* untuk dapat melakukan aktivitas pada *e-marketplace* penyewaan mobil.
- 2) Penyewa melihat dan memilih mobil yang akan disewa.
- 3) Penyewa melakukan pesanan penyewaan dan melakukan pembayaran.

- 4) Penyewa menunggu konfirmasi pembayaran dari admin.
- 5) Penyewa menerima mobil sewaan dari penyedia jasa penyewaan mobil.
- 6) Penyewa mengembalikan mobil dan mengkonfirmasi pengembalian.

### 5.1.3. Implementasi (*Implementation*)

Pada tahap implementasi melakukan tahap selanjutnya, adapun implementasi yang diterapkan berupa sistem yang dibangun seperti tampilan berikut :

#### 5.1.3.1. Desain Tabel

*Database* merupakan tempat untuk menampung data dari *server*, data tersebut nanti akan diproses oleh program yang kita gunakan dalam pembuatan website, database terdiri dari tabel yang dibuat dengan *MySQL*

Adapun database yang dibuat oleh peneliti dengan tabel – tabel sebagai berikut :

#### 1. Admin

Tabel ini digunakan untuk menampung informasi data admin pada *e-Marketplace* Rimera. Adapun struktur tabel admin dapat dilihat pada tabel 5.1 :

Nama Tabel : admin

*Primery Key* : id\_admin

Foreign Key :-

**Tabel 5.1** Desain Tabel Admin

No.	Nama Field	Tipe Data	Ukuran
1	id_admin	int	11

No.	Nama Field	Tipe Data	Ukuran
2	nama	varchar	50
3	username	varchar	30
4	password	varchar	255
5	foto_admin	text	
6	updated_at	timestamp	
7	created_at	timestamp	

Sumber : peneliti

## 2. Penyedia

Tabel ini digunakan untuk menampung informasi data penyedia pada *e-Marketplace* Rimera. Adapun struktur tabel penyedia dapat dilihat pada tabel 5.2 :

Nama Tabel : penyedia

Primary Key : id\_penyedia

Foreign Key : -

**Tabel 5.2** Desain Tabel Penyedia

No.	Nama Field	Tipe Data	Ukuran
1	id_penyedia	int	11
2	nama_penyedia	varchar	50
3	alamat	text	
4	kontak	varchar	25
5	username	varchar	30
6	password	varchar	255
7	no_rek	varchar	16
8	logo_penyedia	text	
9	akte_pendirian	text	

No.	Nama Field	Tipe Data	Ukuran
10	updated_at	timestamp	
11	created_at	timestamp	

Sumber : peneliti

### 3. Penyewa

Tabel ini digunakan untuk menampung informasi data penyewa pada *e-Marketplace* Rimer. Adapun struktur tabel penyewa dapat dilihat pada tabel 5.3 :

Nama Tabel : penyewa

Primary Key : id\_penyewa

Foreign Key : -

**Tabel 5.3** Desain Tabel Penyewa

No.	Nama Field	Tipe Data	Ukuran
1	id_penyewa	int	11
2	no_ktp	char	16
3	nama_penyewa	varchar	50
4	jenis_kelamin	enum('Laki-laki','Perempuan')	
5	tanggal_lahir	date	
6	no_hp	varchar	15
7	alamat	text	
8	username	varchar	30
9	password	varchar	255
10	ktp	text	
11	sim_a	text	
12	kk	text	
13	foto_selfie	text	
14	updated_at	timestamp	

No.	Nama Field	Tipe Data	Ukuran
15	created_at	timestamp	

Sumber : peneliti

#### 4. Tersedia

Tabel ini digunakan untuk menampung informasi data mobil yang tersedia pada *e-Marketplace* Rimera untuk disewa. Adapun struktur tabel tersedia dapat dilihat pada tabel 5.4 :

Nama Tabel : tersedia

Primary Key : id\_tersedia

Foreign Key : id\_admin

**Tabel 5.4** Desain Tabel Tersedia

No.	Nama Field	Tipe Data	Ukuran
1	id_tersedia	int	11
2	id_admin	int	11
3	foto_mobil	text	
4	quantity	int	11
5	merk	varchar	50
6	transmisi	enum('A/T','M/T')	
7	tahun	char	4
8	tanpa_driver	int	11
9	dengan_driver	int	11
10	updated_at	timestamp	
11	created_at	timestamp	

Sumber : peneliti

#### 5. Kendaraan

Tabel ini digunakan untuk menampung informasi data kendaraan yang tersedia untuk disewakan pada *e-Marketplace* Rimera, dimana foto, jenis/merk,



transmisi, dan tahun sudah diinputkan lebih dulu oleh admin, penyedia hanya perlu mengklik data mobil dan mengisi quantity serta harga. Dari quantity yang telah diinputkan oleh penyedia akan otomatis menambahkan quantity yang ada pada tabel tersedia. Adapun struktur tabel kendaraan dapat dilihat pada tabel 5.5:

Nama Tabel : kendaraan

*Primary Key* : id\_kendaraan

Foreign Key : id\_penyedia, dan id\_tersedia

**Tabel 5.5** Desain Tabel Kendaraan

No.	Nama Field	Tipe Data	Ukuran
1	id_kendaraan	int	11
2	id_penyedia	int	11
3	id_tersedia	int	11
4	qty	int	11
5	tanpa_driver	int	11
6	dengan_driver	int	11
7	updated_at	timestamp	
8	created_at	timestamp	

Sumber : peneliti

## 6. Transaksi

Tabel ini digunakan untuk menampung data transaksi penyewaan mobil pada *e-Marketplace* Rimera yang dilakukan oleh penyewa. Adapun struktur tabel transaksi dapat dilihat pada tabel 5.6 :

Nama Tabel : transaksi

*Primary Key* : id\_transaksi

Foreign Key : id\_tersedia, dan id\_kendaraan

**Tabel 5.6** Desain Tabel Transaksi

No.	Nama Field	Tipe Data	Ukuran
1	id_transaksi	int	11
2	id penyewa	int	11
3	id_tersedia	int	11
4	id_kendaraan	int	11
5	no_invoice	varchar	25
6	quantity_sewa	int	11
7	tanggal_transaksi	date	
8	awal_sewa	date	
9	selesai_sewa	date	
10	lama_sewa	int	11
11	kondisi_harga	enum('Tanpa Driver','Dengan Driver')	
12	harga	int	11
13	total	int	11
14	status	enum('Belum dikembalikan','Sudah Dikembalikan')	
15	updated_at	timestamp	
16	created_at	timestamp	

Sumber : peneliti

## 7. Pembayaran

Tabel ini digunakan untuk menampung data pembayaran penyewaan mobil pada *e-Marketplace* Rimera yang telah dilakukan oleh penyewa, pada tabel ini admin dapat mengecek transaksi melalui data pembayaran yang selanjutnya akan melakukan konfirmasi pembayaran dengan melakukan aksi pengecekan atau pelunasan pada pembayaran tersebut.. Adapun struktur tabel pembauaran dapat dilihat pada tabel 5.7 :

Nama Tabel : pembayaran

*Primery Key* : id\_pembayaran

Foreign Key : id\_admin, dan id\_transaksi

**Tabel 5.7** Desain Tabel Pembayaran

No.	Nama Field	Tipe Data	Ukuran
1	id_pembayaran	int	11
2	id_admin	int	11
3	id_transaksi	int	11
4	tanggal_bayar	date	
5	bukti_bayar	text	
6	keterangan	varchar	100
7	alamat	varchar	150
8	status	enum('Pengecekan','Lunas')	
9	updated_at	timestamp	
10	created_at	timestamp	

Sumber : peneliti

## 8. Pencairan

Tabel pencairan digunakan untuk menampung data transaksi dana pencairan yang dilakukan oleh admin. Adapun struktur tabel pencairan dapat dilihat pada tabel 5.8:

Nama Tabel : pencairan

Primary Key : id\_pencairan

Foreign Key : id\_pembayaran

**Tabel 5.8** Desain Tabel Pencairan

No.	Nama Field	Tipe Data	Ukuran
1	id_pencairan	int	11
2	id_pembayaran	int	11
3	tanggal_pencairan	date	
4	bukti_tf	text	

No.	Nama Field	Tipe Data	Ukuran
5	biaya_admin	double	
6	total_pencairan	double	
7	updated_at	timestamp	
8	created_at	timestamp	

Sumber : peneliti

## 9. Slider

Tabel ini digunakan untuk menampung data slider yang dilakukan oleh admin. Adapun struktur tabel slider dapat dilihat pada tabel 5.9 :

Nama Tabel : slider

Primary Key : id\_slider

Foreign Key : id\_admin

**Tabel 5.9** Desain Tabel Slider

No.	Nama Field	Tipe Data	Ukuran
1	id_slider	int	11
2	id_admin	int	11
3	foto	text	
4	updated_at	timestamp	
5	created_at	timestamp	

Sumber : Peneliti

### 5.1.3.2. Desain Website

Untuk mempermudah dalam perancangan sistem, maka peneliti menuliskan antarmuka halaman aplikasi yang dibangun sehingga dapat memberikan gambaran dari aplikasi yang dihasilkan.

## 1. Desain halaman *dashboard* awal

Desain halaman *dashboard* awal merupakan tampilan awal sebelum *user* melakukan *login* pada *e-marketplace* Rimer. Adapun desain halaman *dashboard* awal dapat dilihat pada gambar 5.11.



Sumber : Peneliti

**Gambar 5.11** Desain Halaman *Dashboard* Awal

## 2. Desain halaman data mobil

Desain halaman data mobil merupakan tampilan data mobil yang tersedia pada *e-marketplace* Rimer. Adapun desain halaman halaman data mobil dapat dilihat pada gambar 5.12.

Logo RIMERA Home Tentang Kami Sewa RIMERA

SLIDER

Ketikkan merk atau jenis mobil yang ingin anda sewa Cari

RIMERA Foto Mobil

Quantity : 1

Merk/Type : Daihatsu Sigra R

Transmisi : M/T

Tahun Pembuatan : 2023

Harga Tanpa Driver : Rp. xxx

Harga Dengan Driver : Rp. xxx

Sewa Sekarang !

RIMERA Foto Mobil

Quantity : 1

Merk/Type : Daihatsu Sigra R

Transmisi : M/T

Tahun Pembuatan : 2023

Harga Tanpa Driver : Rp. xxx

Harga Dengan Driver : Rp. xxx

Sewa Sekarang !

RIMERA Foto Mobil

Quantity : 1

Merk/Type : Daihatsu Sigra R

Transmisi : M/T

Tahun Pembuatan : 2023

Harga Tanpa Driver : Rp. xxx

Harga Dengan Driver : Rp. xxx

Sewa Sekarang !

Footer

Sumber : Peneliti

**Gambar 5.12** Desain Halaman Data Mobil

### 3. Desain halaman *login*

Desain halaman *login* merupakan tampilan *form* halaman *login* yang tersedia pada *e-marketplace* Rimer. *User* harus melakukan *login* untuk dapat masuk ke halaman selanjutnya sesuai dengan tugas dan wewenang masing – masing *user*, jika *username* atau *password* yang dimasukkan salah maka akan kembali ke halaman *form login* lagi. Adapun desain halaman halaman *login* dapat dilihat pada gambar 5.13.

Daftar', and a 'Login' button. At the bottom is a 'Footer' bar."/>

Sumber : Peneliti

**Gambar 5.13** Desain Halaman *Login*

#### 4. Sesain Halaman Daftar akun

Desain halaman daftar akun merupakan halaman yang ada sebelum melakukan pendaftaran. Adapun desain halaman daftar akun dapat dilihat pada gambar 5.14.

Sumber : Peneliti

**Gambar 5.14** Desain Halaman Daftar Akun

## 5. Desain halaman daftar penyedia

Desain halaman daftar penyedia merupakan halaman form pendaftaran *user* sebagai penyedia. Adapun desain halaman daftar penyedia dilihat pada gambar 5.15.

The wireframe shows a registration form for a provider. At the top is a navigation bar with a 'Logo' button, a 'RIMERA' button, a 'Home' button, a 'Tentang Kami' button, a 'Sewa' button, and another 'RIMERA' button. Below the navigation bar is a 'SLIDER' area. The main form contains the following fields and elements:

- Nama Usaha :** Input field with placeholder text 'Ketikan nama'.
- Alamat Usaha :** Input field with placeholder text 'Ketikan alamat'.
- Kontak Usaha :** Input field with placeholder text 'Ketikan kontak'.
- Username**: Input field with the value 'CV. PBAJ'.
- Password**: Input field with the value 'xxxxxxxx'.
- No Rekening :** Input field with placeholder text 'Ketikan no rekening'.
- Logo Usaha:** File upload area with a 'Choose File' button and the text 'No file chosen'.
- Akte Pendirian**: File upload area with a 'Choose File' button and the text 'No file chosen'.
- Daftar**: A dark grey button at the bottom of the form.

At the bottom of the page is a 'Footer' area.

Sumber : Peneliti

**Gambar 5.15** Desain Halaman Daftar Penyedia

## 6. Desain halaman daftar penyewa

Desain halaman daftar penyewa merupakan halaman form pendaftaran *user* sebagai penyewa. Adapun desain halaman daftar penyewa dilihat pada gambar 5.16.



Logo RIMERA Home Tentang Kami Sewa RIMERA

SLIDER

No KTP :  
Ketikan no KTP

Nama Lengkap :  
Ketikan nama lengkap

Jenis Kelamin :  
-- Pilih Jenis Kelamin --

Tanggal Lahir :  
MM / DD / YY

Alamat :  
Ketikan alamat

Username :  
Nabila Elfa Rieni

Password :  
xxxxxxxx

KTP :  
Choose File No file chosen

SIM A :  
Choose File No file chosen

KK :  
Choose File No file chosen

Foto Selfie :  
Choose File No file chosen

Daftar

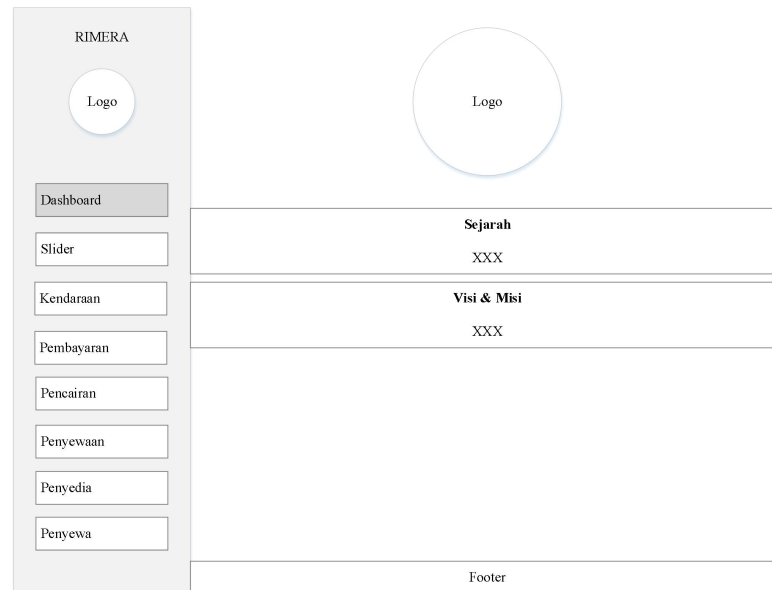
Footer

Sumber : Peneliti

**Gambar 5.16** Desain Halaman Daftar Akun

## 7. Desain halaman *dashboard* admin

Desain halaman *dashboard* admin merupakan tampilan awal ketika admin berhasil melakukan *login*, pada halaman ini terdapat informasi *e-marketplace* Rimera berupa logo, sejarah, serta visi dan misi. Adapun desain halaman *dashboard* admin dilihat pada gambar 5.17.

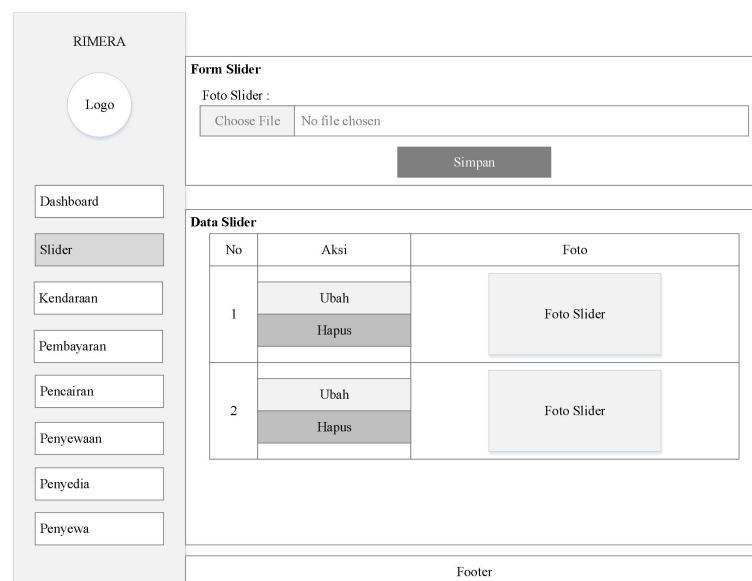


Sumber : Peneliti

**Gambar 5.17** Desain *Dashboard Admin*

## 8. Desain halaman *Slider* pada admin

Desain halaman *slider* berisi form input slider yang nantinya akan muncul pada dashboard awal, halaman penyedia dan penyewa. Adapun desain halaman *slider* dilihat pada gambar 5.18.



Sumber : Peneliti

**Gambar 5.18** Desain Halaman *Slider*

## 9. Desain halaman kendaraan pada admin

Desain halaman kendaraan pada admin berisi form input kendaraan yang tersedia untuk disewa. Adapun desain halaman kendaraan dilihat pada gambar 5.19.

**Form Kendaraan**

Foto Mobil :  No file chosen

Merk :

Transmisi :

Tahun :

Harga Tanpa Driver :

Harga Dengan Driver :

**Data Kendaraan**

No	Aksi	Foto Mobil	Quantity	Merk / Type	Transmisi	Tahun Pembuatan	Tanpa Driver	Dengan Driver
1	Lihat Ubah Hapus	Lihat	1	Daihatsu Sigra R	M/T	M/T	RP. xxx	Rp. xxx
2	Lihat Ubah Hapus	Lihat	1	Daihatsu Sigra R	M/T	M/T	RP. xxx	Rp. xxx
3	Lihat Ubah Hapus	Lihat	1	Daihatsu Sigra R	M/T	M/T	RP. xxx	Rp. xxx

Footer

Sumber : Peneliti

**Gambar 5.19** Desain Halaman Kendaraan pada Admin

## 10. Desain halaman pembayaran pada admin

Desain halaman pembayaran pada admin berisi informasi data pembayaran yang telah dilakukan oleh penyewa, pada halaman ini admin dapat mengkonfirmasi pembayaran dan memilih penyedia mana yang akan melakukan transaksi pada pesanan penyewaan mobil tersebut, pada halaman ini admin juga dapat melakukan pencairan dana kepada penyedia. Aksi pencairan dana dapat dilakukan apabila penyewa telah mengembalikan mobil kepada penyedia. Adapun desain halaman pembayaran pada admin dilihat pada gambar 5.20.

RIMERA									
Data Pembayaran									
No	Aksi	Status Pembayaran	Status Penyewaan	Nama Penyedia	Status Penyewa	Status Kendaraan	Status Transaksi	Tanggal Bayar	Bukti Bayar
1	✓	Pengecekan	Belum dikembalikan	Belum ditunaskan	Lihat	Lihat	Lihat	2023-07-25	Lihat Download
2	-	Lunas	Belum dikembalikan	CV. XXX	Lihat	Lihat	Lihat	2023-07-25	Lihat Download
3	-	Lunas	Sudah dikembalikan	CV. XXX	Lihat	Lihat	Lihat	2023-07-25	Lihat Download

Sumber : Peneliti

**Gambar 5.20** Desain Halaman Pembayaran pada Admin

## 11. Desain halaman penyewaan pada admin

Desain halaman penyewaan pada admin berisi data penyewaan yang sedang berlangsung, pada saat tanggal penyewaan telah selesai, maka akan muncul aksi kembalikan, aksi ini juga akan muncul pada penyewa. Adapun desain halaman penyewaan admin dilihat pada gambar 5.21.

RIMERA											
Data Penyewaan											
No	Aksi	Transaksi	Nama Penyedia	Status Kendaraan	No Transaksi	No Invoice	QTY Sewa	Tanggal Penesman	Tanggal Awal Sewa	Tanggal Selesai Sewa	
1	Kembalikan!	Belum dikembalikan	CV. XXX	Lihat	xxx	xxx	xxx	2023-07-25	2023-07-25	2023-07-25	
2	-	Sudah dikembalikan	CV. XXX	Lihat	xxx	xxx	xxx	2023-07-25	2023-07-25	2023-07-25	
3	-	Sudah dikembalikan	CV. XXX	Lihat	xxx	xxx	xxx	2023-07-25	2023-07-25	2023-07-25	

Sumber : Peneliti

**Gambar 5.21** Desain Halaman Penyewaan pada Admin

## 12. Desain halaman pencairan pada admin

Desain halaman pencairan berisi data pencairan dana kepada penyedia.

Adapun desain halaman data pencairan pada admin dilihat pada gambar 5.22.

RIMERA									
Data Pembayaran									
No	Nama Penyedia	Status Penyewa	Status Kendaraan	Status Transaksi	Status Pembayaran	Bukti Transfer	Biaya Admin	Total Pencairan	
1	CV. XXX	Lihat	Lihat	Lihat	Lihat	Lihat Download	XXX	XXX	
2	CV. XXX	Lihat	Lihat	Lihat	Lihat	Lihat Download	XXX	XXX	
3	CV. XXX	Lihat	Lihat	Lihat	Lihat	Lihat Download	XXX	XXX	

Sumber : Peneliti

**Gambar 5.22** Desain Halaman Pencairan pada Admin

## 13. Desain halaman data penyedia pada admin

Desain halaman penyedia berisi informasi data penyedia pada *e-marketplace*

Rimera. Adapun desain data penyedia dilihat pada gambar 5.23.

RIMERA							
Data Penyedia							
No	Aksi	Nama Usaha	Alamat	Kontak	Username	No Rekening	Detail
1	Ubah Hapus	CV. xxx	Jl. xxx	0895xxx	abcd	xxxxxx	Logo, Akte Pendirian Lihat / Download
2	Ubah Hapus	CV. xxx	Jl. xxx	0895xxx	abcd	xxxxxx	Logo, Akte Pendirian Lihat / Download
3	Ubah Hapus	CV. xxx	Jl. xxx	0895xxx	abcd	xxxxxx	Logo, Akte Pendirian Lihat / Download
4	Ubah Hapus	CV. xxx	Jl. xxx	0895xxx	abcd	xxxxxx	Logo, Akte Pendirian Lihat / Download
5	Ubah Hapus	CV. xxx	Jl. xxx	0895xxx	abcd	xxxxxx	Logo, Akte Pendirian Lihat / Download

Sumber : Peneliti

**Gambar 5.23** Desain Halaman Penyedia pada Admin

#### 14. Desain halaman data penyewa pada admin

Desain halaman penyewa berisi informasi data penyewa pada *e-marketplace* Rimera. Adapun desain data penyewa dilihat pada gambar 5.24.

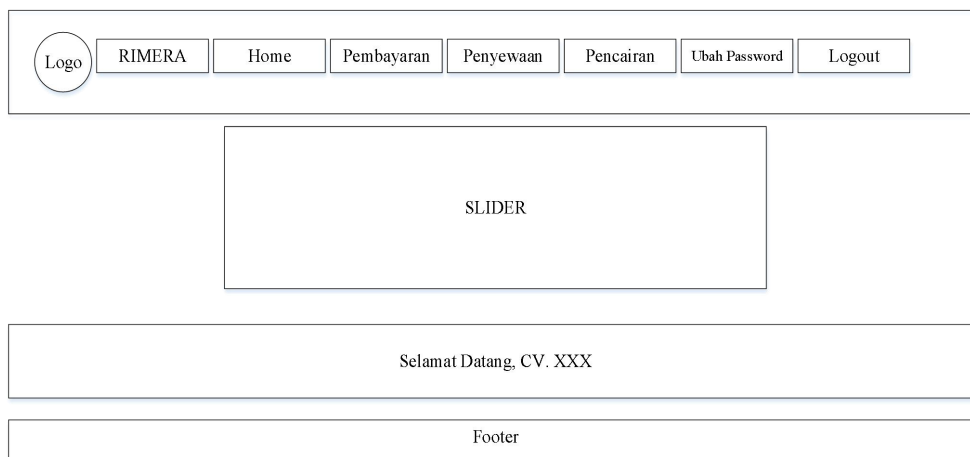
RIMERA												
Data Penyedia												
No	Aksi	No. KTP	Nama Penyewa	Jenis Kelamin	Tanggal Lahir	No. Hp	Alamat	Username	KTP	SIM A	KK	Foto Selfie
1	Ubah Hapus	xxxxxxxxxxxxxxxx	Nabila Elfa Rieni	Perempuan	Xx/xx/xxxx	0895xxxx	xxx	Lihat Download	Lihat Download	Lihat Download	Lihat Download	Lihat Download
2	Ubah Hapus	xxxxxxxxxxxxxxxx	Nabila Elfa Rieni	Perempuan	Xx/xx/xxxx	0895xxxx	xxx	Lihat Download	Lihat Download	Lihat Download	Lihat Download	Lihat Download

Sumber : Peneliti

**Gambar 5.24** Desain Halaman Penyewa pada Admin

#### 15. Desain halaman dashboard pada penyedia

Desain halaman dashboard penyedia berisi tampilan awal setelah penyedia melakukan login pada *e-marketplace* Rimera. Adapun desain dashboard penyedia dilihat pada gambar 5.25



Sumber : Peneliti

**Gambar 5.25** Desain Halaman Dashboard Penyedia

## 16. Desain halaman data kendaraan pada penyedia

Desain halaman kendaraan pada penyedia berisi informasi kendaraan, dan form menuju penginputan kendaraan pada *e-marketplace* Rimera. Adapun desain halaman data kendaraan pada penyedia dilihat pada gambar 5.26

Logo RIMERA Home Pembayaran Penyewaan Pencairan Ubah Password Logout

SLIDER

Form Cari Kendaraan

Cari

**Data Kendaraan**

**Data Kendaraan CV. XXX**

No	Aksi	Qty	Merk / Type	Transmisi	Tahun Pembuatan	Harga Lepas Kunci	Harga Dengan Driver
1	Ubah	1	Toyota Avanza	M/T	XXXX	Rp. XXX	Rp. XXX
2	Ubah	1	Toyota Fortuner	M/T	XXXX	Rp. XXX	Rp. XXX
3	Ubah	1	Daihatsu Xenia	M/T	XXXX	Rp. XXX	Rp. XXX

Footer

Sumber : Peneliti

**Gambar 5.26** Desain Halaman data Kendaraan Penyedia

## 17. Desain halaman *form input* kendaraan pada penyedia

Desain halaman *form input* kendaraan pada penyedia berisi informasi penginputan kendaraan pada *e-marketplace* Rimera. Adapun desain halaman *form input* kendaraan pada penyedia dilihat pada gambar 5.27

The image shows a wireframe of a website interface. At the top, there is a navigation bar with a circular 'Logo' button and several menu items: 'RIMERA', 'Home', 'Pembayaran', 'Penyewaan', 'Pencairan', 'Ubah Password', and 'Logout'. Below the navigation bar is a large rectangular area labeled 'SLIDER'. Underneath the slider is a section titled 'Form Input Kendaraan'. This form contains several input fields: 'Foto Mobil : [Lihat](#)', 'Qty' (with a placeholder 'Ketikan qty'), 'Merk / Type' (with a placeholder 'Toyota Fortuner VRZ Diesel'), 'Tahun Pembuatan' (with a placeholder '2018'), 'Harga Tanpa Driver' (with a placeholder 'Ketikan Harga'), and 'Harga Dengan Driver' (with a placeholder 'Ketikan Harga'). A 'Simpan' button is located at the bottom of the form. At the very bottom of the page is a 'Footer' area.

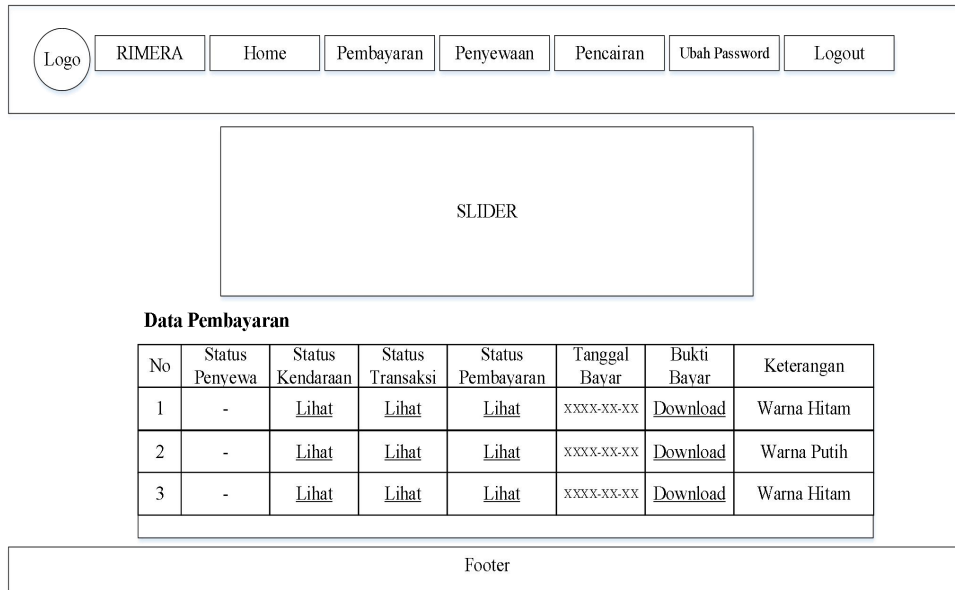
Sumber : Peneliti

**Gambar 5.27** Desain Halaman Input Kendaraan Penyedia

## 18. Desain halaman data pembayaran Penyedia

Desain halaman data pembayaran penyedia berisi informasi data pembayaran pada *e-marketplace* Rimera. Adapun desain halaman data pembayaran pada penyedia dilihat pada gambar 5.28



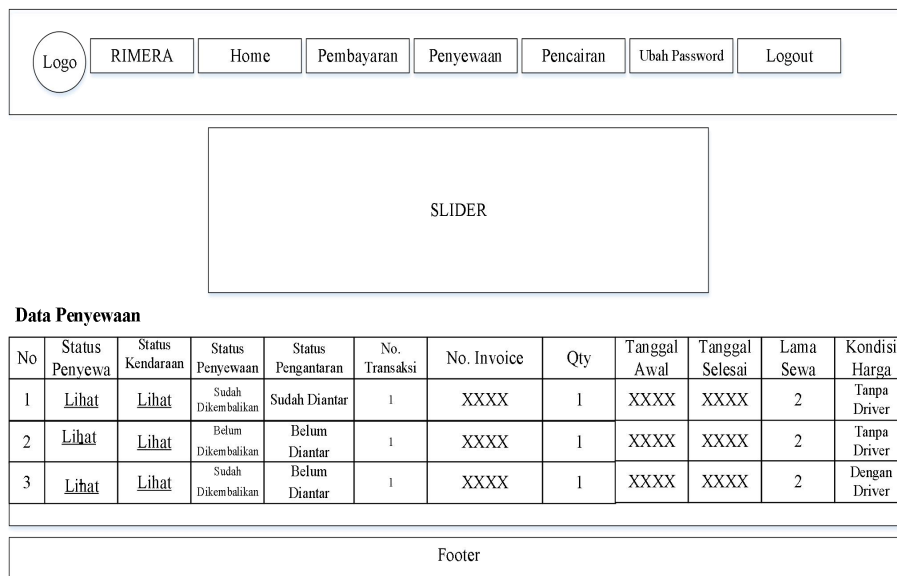


Sumber : Peneliti

**Gambar 5.28** Desain Halaman Data Pembayaran Penyedia

## 19. Desain halaman data penyewaan penyedia

Desain halaman data penyewaan penyedia berisi informasi data transaksi penyewaan mobil pada *e-marketplace* Rimera. Adapun desain halaman data penyewaan penyedia dilihat pada gambar 5.29

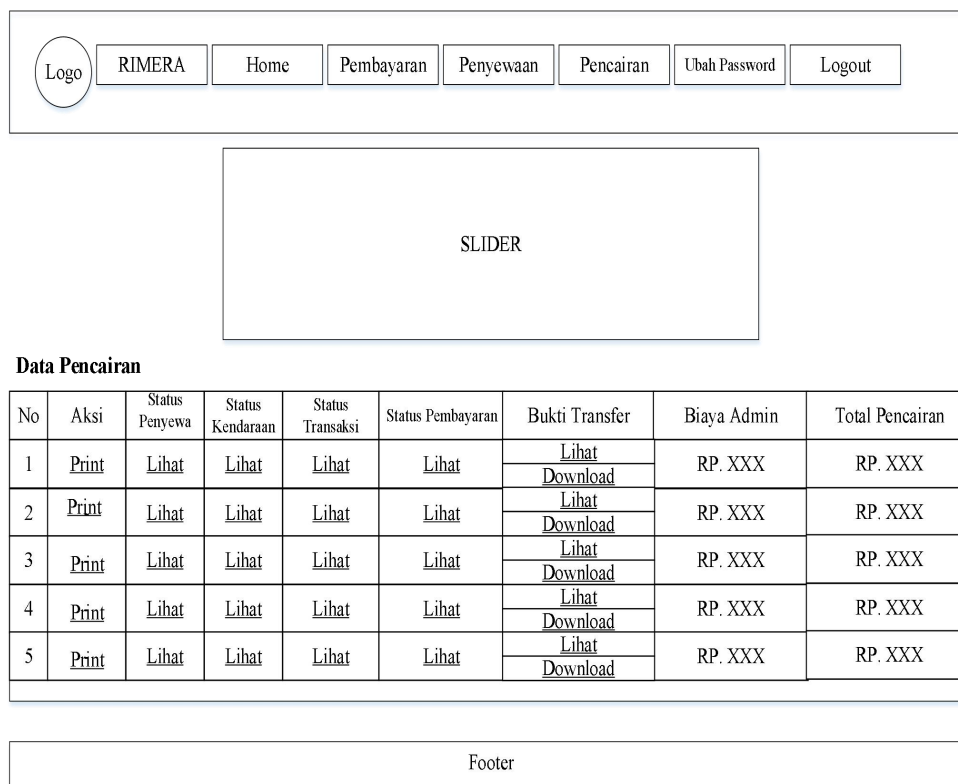


Sumber : Peneliti

**Gambar 5.29** Desain Halaman Data Penyewaan Penyedia

## 20. Desain halaman data pencairan pada penyedia

Desain halaman data pencairan pada penyedia berisi informasi data pencairan hasil penyewaan mobil pada *e-marketplace* Rimer. Adapun desain halaman data pencairan penyedia dilihat pada gambar 5.30

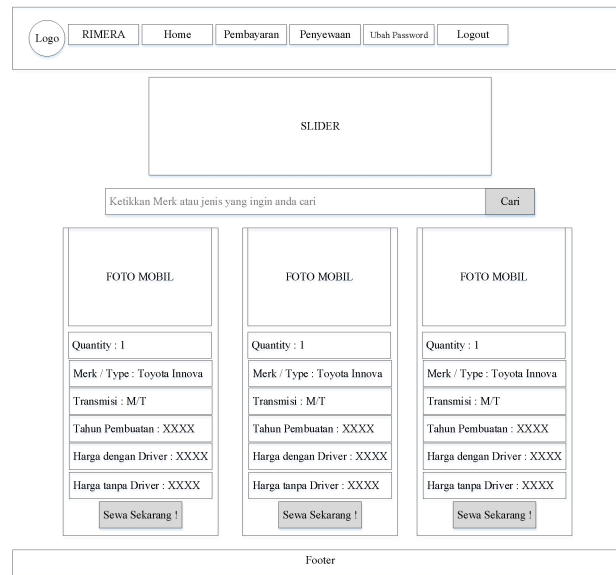


Sumber : Peneliti

**Gambar 5.30** Desain Halaman Data Pencairan Penyedia

## 21. Desain halaman *dashboard* penyewa

Desain halaman *dashboard* penyewa berisi tampilan awal setelah penyedia melakukan *login* pada *e-marketplace* Rimer. Pada halaman ini berisi informasi data mobil yang tersedia untuk disewa. Adapun desain *dashboard* penyewa dilihat pada gambar 5.31.

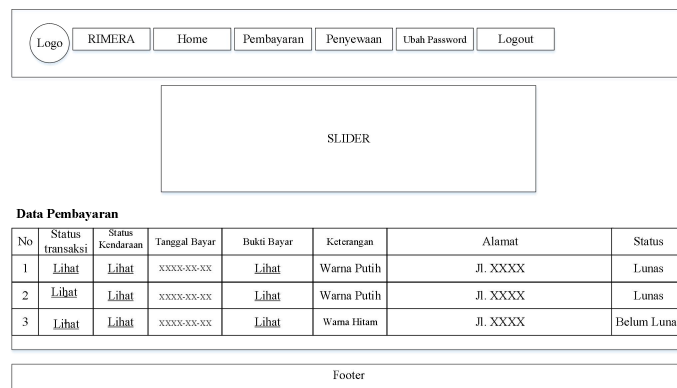


Sumber : Peneliti

**Gambar 5.31** Desain Halaman *Dashboard* Penyewa

## 22. Desain halaman data pembayaran pada penyewa

Desain halaman data pembayaran penyewa berisi informasi data pembayaran yang telah dilakukan oleh penyewa pada *e-marketplace* Rimera. Adapun desain halaman data pembayaran pada penyewa dilihat pada gambar 5.32.



Sumber : Peneliti

**Gambar 5.32** Desain Halaman Data Pembayaran Penyewa

## 23. Desain halaman *form* pembayaran

Desain halaman *form* pembayaran berisi informasi penginputan bukti pembayaran penyewaan mobil yang dilakukan oleh penyewa, pada form ini juga penyewa

dapat memberikan informasi keterangan tambahan dan alamat pengantaran mobil.

Adapun desain halaman *form* pembayaran dapat dilihat pada gambar 5.33.

**Form Pembayaran**

Bukti Bayar :

Choose File No. Invoice : XXXXX

\*Mohon pembayaran dapat dikakukan ke rekening : XXXXXXXX

Keterangan :

Ketikan keterangan : (tolong dikirimkan mobil warna putih)

Alamat Pengantaran:

Ketikan alamat pengantaran

Sumber : Peneliti

**Gambar 5.33** Desain Halaman *Form* Pembayaran

## 24. Desain Halaman penyewaan pada penyedia

Desain halaman data penyewaan penyedia berisi informasi data transaksi penyewaan mobil pada *e-marketplace* Rimera. Adapun desain halaman data penyewaan pada penyedia dilihat pada gambar 5.34.

Logo
RIMERA
Home
Pembayaran
Penyewaan
Ubah Password
Logout

SLIDER

Penyewaan Sedang Berlangsung											
No	Aksi	Status Kendaraan	Status Penyewaan	Status Pengantaran	No. Transaksi	No. Invoice	Qty	Tanggal Awal	Tanggal Selesai	Lama Sewa	Kondisi Harga
1	<a href="#">Print</a>	<a href="#">Lihat</a>	Sudah Dikembalikan	Sudah Diantar	1	XXXX	1	XXXX	XXXX	2	Tanpa Driver
2	<a href="#">Print</a>	<a href="#">Lihat</a>	Belum Dikembalikan	Belum Diantar	1	XXXX	1	XXXX	XXXX	2	Tanpa Driver
3	<a href="#">Print</a>	<a href="#">Lihat</a>	Sudah Dikembalikan	Belum Diantar	1	XXXX	1	XXXX	XXXX	2	Dengan Driver

Footer

Sumber : Peneliti

**Gambar 5.34** Desain Halaman Data penyewaan penyewa

## 25. Desain halaman ubah *password*

Desain halaman ubah *password* berisi *form* penginputan *password* yang dapat diubah oleh masing – masing *user*. Adapun desain halaman ubah *password* dilihat pada gambar 5.35.

Sumber : Peneliti

**Gambar 5.35** Desain Halaman Ubah Password

## 26. Desain *logout*

Desain halaman *logout* adalah informasi pemberitahuan kepada masing – masing *user*, apakah *user* yakin untuk menghentikan sesi. Adapun desain *logout* dilihat pada gambar 5.36.

Sumber : Peneliti

**Gambar 5.36** Desain Halaman *Logout*

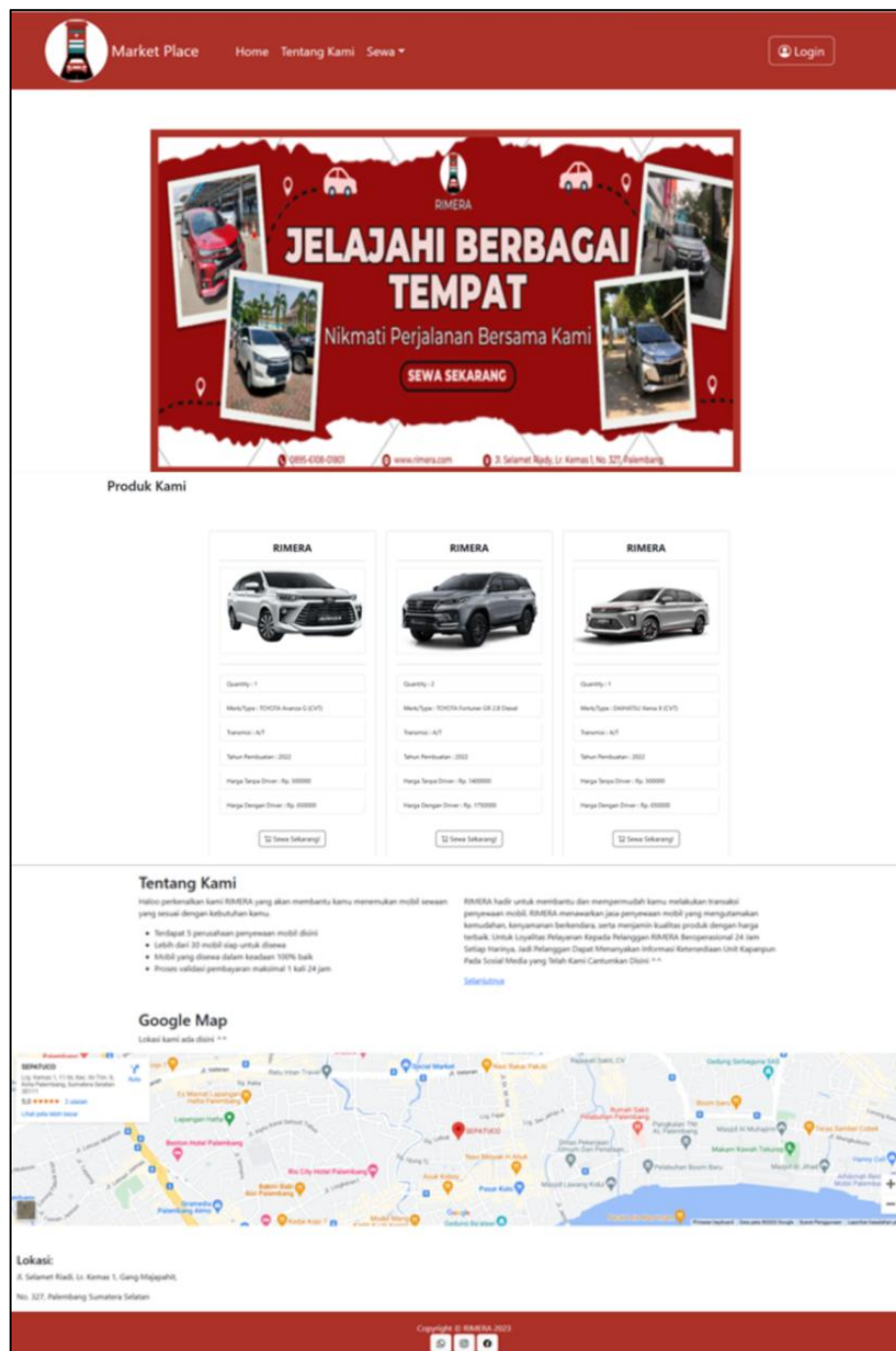
### 5.1.3.3. Desain Antarmuka

Desain antarmuka merupakan rancangan desain tampilan E-Marketplace penyewaan mobil (Studi kasus : Kota Palembang) sebagai berikut :

#### 1. Hasil desain halaman *dashboard* awal

Hasil desain halaman *dashboard* awal merupakan tampilan awal sebelum *user* melakukan *login* pada *e-marketplace* Rimera. Halaman ini tersedia informasi

kendaraan yang tersedia untuk disewa, tentang kami, alamat, google map, dan sosial media. Adapun hasil desain halaman *dashboard* awal dapat dilihat pada gambar 5.37.

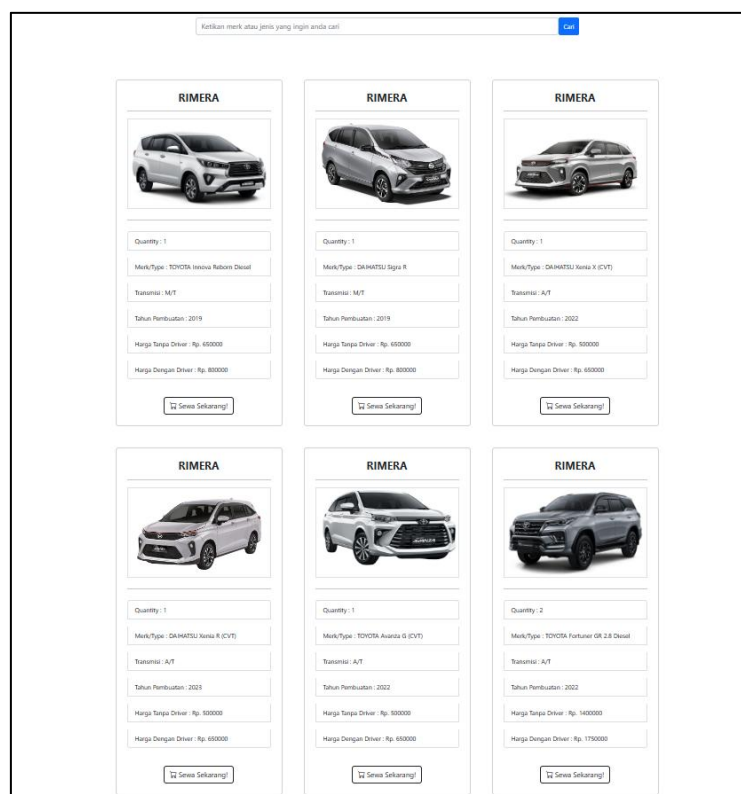


Sumber : Peneliti

**Gambar 5.37** Hasil Desain Halaman *Dashboard* Awal

## 2. Hasil desain halaman data mobil

Hasil desain halaman data mobil merupakan tampilan data mobil yang tersedia pada *e-marketplace* Rimera. Adapun hasil desain halaman halaman data mobil dapat dilihat pada gambar 5.37.



Sumber : Peneliti

**Gambar 5.38** Hasil Desain Halaman Data mobil

## 3. Hasil desain halaman login

Hasil desain halaman *login* merupakan tampilan *form* halaman *login* yang tersedia pada *e-marketplace* Rimera. *User* harus melakukan *login* untuk dapat masuk ke halaman selanjutnya sesuai dengan tugas dan wewenang masing – masing *user*, jika *username* atau *password* yang dimasukkan salah maka akan kembali ke halaman *form login* lagi. Adapun hasil desain halaman halaman *login* dapat dilihat pada gambar 5.38.

Sumber : Peneliti

**Gambar 5.39** Hasil Desain Halaman *Login*

#### 4. Hasil desain halaman daftar akun

Hasil desain halaman daftar akun merupakan halaman yang ada sebelum melakukan pendaftaran. Adapun hasil desain halaman daftar akun dapat dilihat pada gambar 5.40.

Sumber : Peneliti

**Gambar 5.40** Hasil Desain Daftar Akun



## 5. Hasil desain halaman daftar penyedia

Hasil desain halaman daftar penyedia merupakan halaman form pendaftaran *user* sebagai penyedia. Adapun hasil desain halaman daftar penyedia dilihat pada gambar 5.41.

Sumber : Peneliti

**Gambar 5.41** Hasil Desain Halaman Daftar Penyedia

## 6. Hasil desain halaman form daftar penyewa

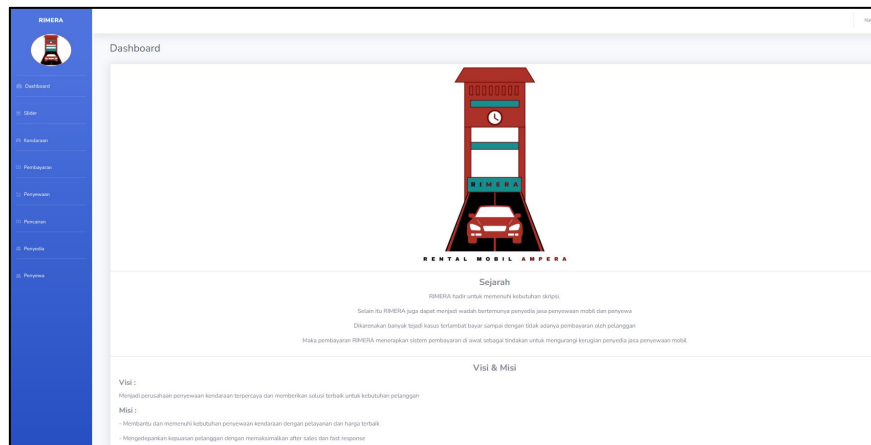
Hasil desain halaman daftar penyewa merupakan halaman form pendaftaran *user* sebagai penyewa. Adapun hasil desain halaman daftar penyewa dilihat pada gambar 5.42

Sumber : Peneliti

**Gambar 5.42** Hasil Desain Halaman Daftar Penyewa

## 7. Hasil desain halaman dashboard pada admin

Hasil desain halaman *dashboard* admin merupakan tampilan awal ketika admin berhasil melakukan *login*. Adapun hasil desain halaman *dashboard* admin dilihat pada gambar 5.43.

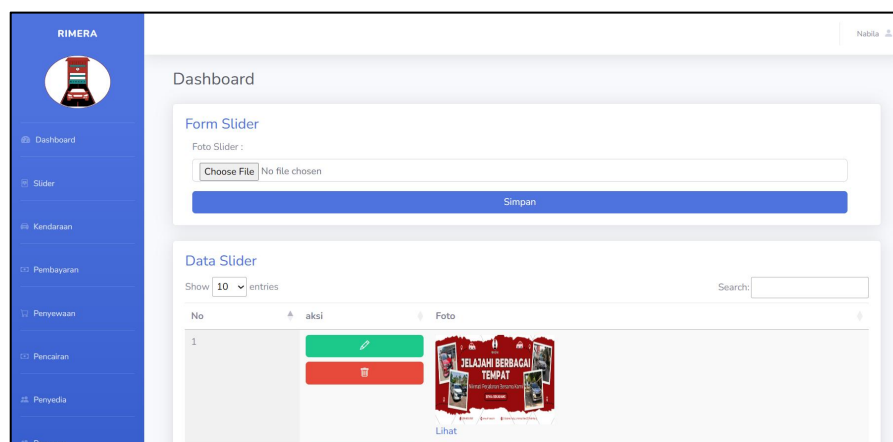


Sumber : Peneliti

**Gambar 5.43** Hasil Desain *Dashboard* Admin

## 8. Hasil desain halaman Slider pada admin

Hasil desain halaman *slider* berisi *slider* yang nantinya akan muncul pada dashboard awal, penyedia dan penyewa. Adapun hasil desain halaman *slider* dilihat pada gambar 5.44.



Sumber : Peneliti

**Gambar 5.44** Hasil Desain *Slider*

## 9. Hasil desain halaman kendaraan pada admin

Hasil desain halaman kendaraan pada admin berisi *form input* kendaraan yang tersedia untuk disewa. Adapun desain halaman kendaraan dilihat pada gambar 5.45.

The screenshot shows the RIMERA admin dashboard. On the left is a blue sidebar with navigation links: Dashboard, Sider, Kendaraan, Pembayaran, Penyewaan, Pencarian, Penyedia, and Penyewa. The main content area is titled 'Dashboard' and contains two sections:

**Form Kendaraan**

This form includes the following fields:

- Foto Mobil:  No file chosen
- Merik:
- Transmisi:
- Ketikan tahun:
- Harga Tanpa Driver:
- Harga Dengan Driver:

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

**Data Kendaraan**

This section displays a table of vehicle data with the following columns: No, aksi, Foto Mobil, Quantity, Merk/Type, Transmisi, Tahun Pembuatan, Tanpa Driver, and Dengan Driver. The table contains two entries:

No	aksi	Foto Mobil	Quantity	Merk/Type	Transmisi	Tahun Pembuatan	Tanpa Driver	Dengan Driver
1	<a href="#">Lihat</a>		1	TOYOTA Innova Reborn Diesel	M/T	2019	Rp. 650000	Rp. 800000
2	<a href="#">Lihat</a>		1	DAIHATSU Sigra R	M/T	2019	Rp. 650000	Rp. 800000

Sumber : Peneliti

**Gambar 5.45** Hasil Desain Halaman Kendaraan Pada Admin

## 10. Hasil desain halaman pembayaran pada admin

Hasil desain halaman pembayaran pada admin berisi informasi data pembayaran yang telah dilakukan oleh penyewa. Adapun hasil desain halaman pembayaran pada admin dilihat pada gambar 5.46.

The screenshot shows the RIMERA admin dashboard with the 'Data Pembayaran' section. The table displays payment records with the following columns: No, Aksi, Status Pembayaran, Status Penyewaan, Nama Penyedia, Status Penyewa, Status Kendaraan, Status Transaksi, Tanggal Bayar, Bukti Bayar, Keterangan, and Alamat. The table contains three entries:

No	Aksi	Status Pembayaran	Status Penyewaan	Nama Penyedia	Status Penyewa	Status Kendaraan	Status Transaksi	Tanggal Bayar	Bukti Bayar	Keterangan	Alamat
1	<a href="#">Lihat</a>	Pengajuan	Belum Dibayarkan	Belum Di lunaskan	Lihat	Lihat	Lihat	2023-07-25	Lihat Download	1	1
2	<a href="#">Lihat</a>	Lunas	Sudah Dibayarkan	CV, Palembang Berlian Abadi Jaya	Lihat	Lihat	Lihat	2023-07-25	Lihat Download	dikirim jam 12 siang	sesuai alamat ktp
3	<a href="#">Lihat</a>	Lunas	Sudah Dibayarkan	PT. Sekayu Serasan Sekate	Lihat	Lihat	Lihat	2023-07-25	Lihat Download	dikirim jam 1 siang	dikirim sesuai alamat ktp

Showing 1 to 3 of 3 entries

Copyright © RIMERA 2023

Sumber : Peneliti

**Gambar 5.46** Hasil Desain Halaman Kendaraan Pada Admin

## 11. Hasil desain halaman penyewaan pada admin

Hasil desain halaman penyewaan pada admin berisi data penyewaan yang sedang berlangsung, pada saat tanggal penyewaan telah selesai, maka akan muncul aksi kembalikan, aksi ini juga akan muncul pada penyewa. Adapun hasil desain halaman penyewaan admin dilihat pada gambar 5.47.

No	Aksi	Status Transaksi	Status Pengantaran	Nama Usaha	Status Penyewa	Status Kendaraan	No Transaksi	No Invoice	Quantity Sewa	Tanggal Pemesanan
1	Peringatan! Kembalikan	Sudah Diambil	Sudah Diantar	CV. Palembang Berlian Abadi Jaya	Lihat	Lihat	1	PY202307251	1	2023-07-25
2		Sudah Diambil	Sudah Diantar	CV. Palembang Berlian Abadi Jaya	Lihat	Lihat	1	PY202307251	1	2023-07-23
3		Sudah Diambil	Sudah Diantar	PT. Sekayu Serasan Sekate	Lihat	Lihat	2	PY202307252	1	2023-07-25

Sumber : Peneliti

**Gambar 5.47** Hasil Desain Halaman Penyewaan Pada Admin

## 12. Hasil desain halaman pencairan pada admin

Hasil desain halaman pencairan berisi data pencairan dana kepada penyedia. Adapun hasil desain halaman data pencairan pada admin dilihat pada gambar 5.48.

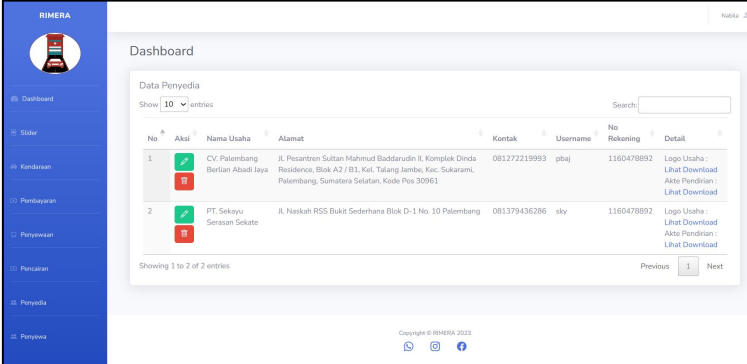
No	Nama Penyedia	Status Penyewa	Status Kendaraan	Status Transaksi	Status Pembayaran	Bukti Transfer	Biaya Admin	Total Pencairan
1	CV. Palembang Berlian Abadi Jaya	Lihat	Lihat	Lihat	Lihat	Lihat Download	100000	1200000





Sumber : Peneliti

**Gambar 5.48** Hasil Desain Halaman Pencairan Pada Admin

## 13. Hasil desain halaman data penyedia pada admin

Hasil desain halaman penyedia berisi informasi data penyedia pada *e-marketplace* Rimera. Adapun hasil desain data penyedia dilihat pada gambar 5.49.



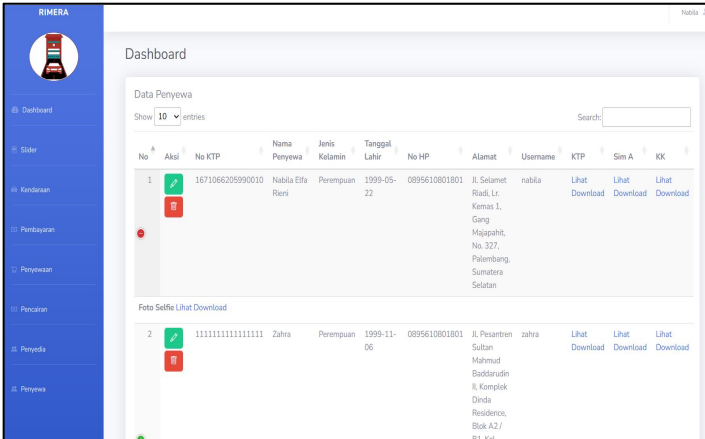
No	Aksi	Nama Usaha	Alamat	Kontak	Username	No Rekening	Detail
1	 	CV Palembang Berlian Abadi Jaya	Jl. Pesantren Sultan Mahmud Bakdarudin II, Komplek Dinda Residence, Blok A2 / 01, Kot. Talang Jambi, Kec. Sukarame, Palembang, Sumatera Selatan, Kode Pos 30961	081272219993	pbaj	1160478892	Logo Usaha : Lihat Download Akte Pendirian : Lihat Download
2	 	PT Setayu Sersan Sekate	Jl. Naskah RSS Bukit Sederhana Blok D-1 No. 10 Palembang	081379436286	sky	1160478892	Logo Usaha : Lihat Download Akte Pendirian : Lihat Download





Sumber : Peneliti

**Gambar 5.49** Hasil Desain Halaman Penyedia Pada Admin

#### 14. Hasil desain halaman data penyewa pada admin

Hasil desain halaman penyewa berisi informasi data penyewa pada *e-marketplace* Rimera. Adapun hasil desain data penyewa dilihat pada gambar 5.50.



No	Aksi	No KTP	Nama Penyewa	Jenis Kelamin	Tanggal Lahir	No HP	Alamat	Username	KTP	Sim A	KK
1	 	167106620590010	Nabila Efa Rendi	Perempuan	1999-05-22	0895610801801	Jl. Selamat Riadi, Lt. Kemas 1, Gang Majapahit, No. 327, Palembang, Sumatera Selatan	rabla	Lihat Download	Lihat Download	Lihat Download
2	 	1111111111111111111	Zahra	Perempuan	1999-11-06	0895610801801	Jl. Pesantren Sultan Mahmud Bakdarudin II, Komplek Dinda Residence, Blok A2 / 01, Kot.	zahra	Lihat Download	Lihat Download	Lihat Download

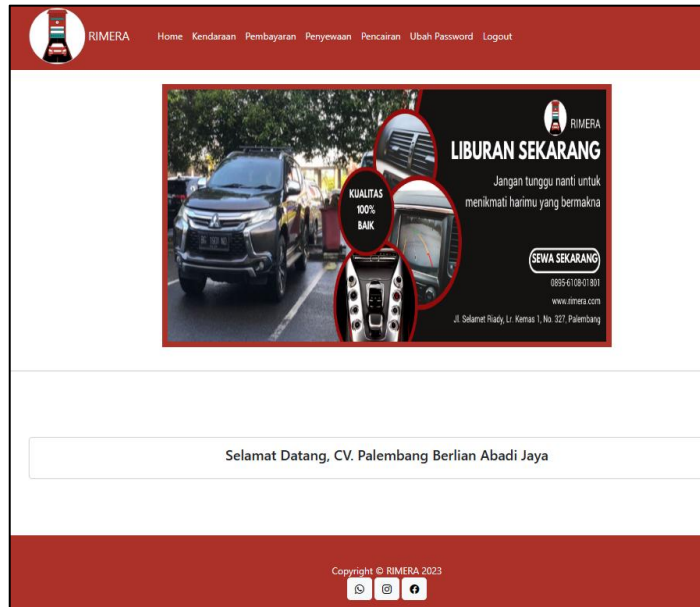
Sumber : Peneliti

**Gambar 5.50** Hasil Desain Halaman Penyewa Pada Admin

#### 15. Hasil desain halaman *dashboard* pada penyedia

Hasil desain halaman *dashboard* penyedia berisi tampilan awal setelah penyedia melakukan *login* pada *e-marketplace* Rimera.

Adapun hasil desain *dashboard* penyedia dilihat pada gambar 5.51.



Sumber : Peneliti

**Gambar 5.51** Hasil Desain Dashboard Penyedia

## 16. Hasil desain halaman kendaraan pada penyedia

Hasil desain halaman kendaraan pada penyedia berisi informasi kendaraan, dan form menuju penginputan kendaraan pada *e-marketplace* Rimera. Adapun hasil desain halaman data kendaraan pada penyedia dilihat pada gambar 5.52

**Form Cari Kendaraan**

Cari Kendaraan

Mark

Ketikkan mark

**Data Kendaraan**

Data Kendaraan CV. Palembang Berlian Abadi Jaya

Show 10 entries

No	Aksi	Foto Mobil	Quantity	Merak/Type	Transmisi	Tahun Pembuatan	Dalam Kota - Kunci	Dalam Kota - Lepas Driver
1			1	TOYOTA Innova Reborn Diesel	M/T	2019	Rp. 600000	Rp. 750000
2			0	DAIHATSU Sigra R	M/T	2019	Rp. 350000	Rp. 450000
3			0	TOYOTA Avanza G Facelift	M/T	2021	Rp. 350000	Rp. 450000
4			1	TOYOTA Avanza G (CVT)	A/T	2022	Rp. 400000	Rp. 550000
5			1	DAIHATSU Xenia X (CVT)	A/T	2022	Rp. 400000	Rp. 550000
6			1	DAIHATSU Xenia R (CVT)	A/T	2023	Rp. 400000	Rp. 550000
7			1	TOYOTA Fortuner GR 2.8 Diesel	A/T	2022	Rp. 1300000	Rp. 1500000

Showing 1 to 7 of 7 entries

Previous 1 Next

Sumber : Peneliti

**Gambar 5.52** Hasil Desain Kendaraan Pada Penyedia

## 17. Hasil desain halaman form input kendaraan pada penyedia

Hasil desain halaman *form input* kendaraan pada penyedia berisi informasi penginputan kendaraan pada *e-marketplace* Rimera. Adapun hasil desain halaman *form input* kendaraan pada penyedia dilihat pada gambar 5.53.

The image shows a web form titled "Form Input Kendaraan". It has a header "Input Kendaraan" and a link "Foto Mobil : Lihat". Below are several input fields: "Quantity" (placeholder: "Ketikan quantity"), "Merk/Type" (value: "TOYOTA Avanza G Facelift"), "Transmisi" (value: "M/T"), "Tahun Pembuatan" (value: "2022"), "Tanpa Driver" (placeholder: "Ketikan harga"), and "Dengan Driver" (placeholder: "Ketikan harga"). A blue "Simpan" button is at the bottom.

Sumber : Peneliti

**Gambar 5.53** Hasil Desain Form Kendaraan Pada Penyedia

### **18. Hasil desain halaman data pembayaran pada penyedia**

Hasil desain halaman data pembayaran pada penyedia berisi informasi data pembayaran pada *e-marketplace* Rimera, pada halaman ini penyedia melakukan konfirmasi pengantaran, apabila mobil telah diantarkan ke penyewa maka penyedia dapat mengklik aksi “sudah diantarkan”. Adapun hasil desain halaman data pembayaran pada penyedia dilihat pada gambar 5.54.

The screenshot displays the RIMERA website interface. At the top, there is a navigation bar with the RIMERA logo and links for Home, Kendaraan, Pembayaran, Penyewaan, Pencairan, Ubah Password, and Logout. Below the navigation bar is a large advertisement for car rental services. The ad features a car and text in Indonesian: "LIBURAN SEKARANG" (Vacation Now), "Jangan tunggu nanti untuk menikmati harimu yang bermakna" (Don't wait later to enjoy your meaningful day), and "SEWA SEKARANG" (Rent Now). It also includes the phone number 0895-6108-01801 and the website www.rimera.com. Below the advertisement is a section titled "Data Pembayaran" (Payment Data). It includes a note: "Note : Apabila pembayaran telah lunas, segera antar mobil sesuai alamat tersebut!" (Note: If payment has been completed, please return the car to the specified address!). There is a search bar and a dropdown menu set to "10" entries. The table below shows two payment entries:

No	Aksi	Status Penyewa	Status Kendaraan	Status Transaksi	Status Pembayaran	Status Penyewaan	Status Pengantaran	Tanggal Bayar	Bukti Bayar	Keterangan
1	<a href="#">Antarkan</a>	<a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Pengecekan</a>	<a href="#">Pengecekan</a>	<a href="#">Belum Diantar</a>	2023-08-07	<a href="#">Download</a>	warna putih
2	-	<a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Lunas</a>	<a href="#">Sudah Dikembalikan</a>	<a href="#">Sudah Diantar</a>	2023-07-25	<a href="#">Download</a>	dikirimkan jam 12 siang

At the bottom of the table, it says "Showing 1 to 2 of 2 entries" and includes "Previous" and "Next" navigation buttons.

Sumber : Peneliti

**Gambar 5.54** Hasil Desain Halaman Pembayaran Pada Penyedia

## 19. Hasil desain halaman data pencairan

Hasil desain halaman data pencairan pada penyedia berisi informasi data pencairan hasil penyewaan mobil pada *e-marketplace* Rimera. Adapun hasil desain halaman data pencairan penyedia dilihat pada gambar 5.55.

The screenshot displays the RIMERA website interface. At the top, there is a navigation bar with the RIMERA logo and links for Home, Kendaraan, Pembayaran, Penyewaan, Pencairan, Ubah Password, and Logout. Below the navigation bar is a large advertisement for car rental services. The ad features a car and text in Indonesian: "CAPEK SETIR SENDIRI ?" (Tired of driving yourself?), "SEWA MOBIL + DRIVER AJA" (Rent car + driver), and "BUKA 24 JAM" (Open 24 hours). It also includes the phone number 0895-6108-01801 and the website www.rimera.com. Below the advertisement is a section titled "Data Pencairan" (Payment Data). It includes a search bar and a dropdown menu set to "10" entries. The table below shows one payment entry:

No	Aksi	Status Penyewa	Status Kendaraan	Status Transaksi	Status Pembayaran	Bukti Transfer	Biaya Admin	Total Pencairan
1	<a href="#">Print</a> <a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Lihat</a>	<a href="#">Lihat</a>	100000	1200000

At the bottom of the table, it says "Showing 1 to 1 of 1 entries" and includes "Previous" and "Next" navigation buttons.

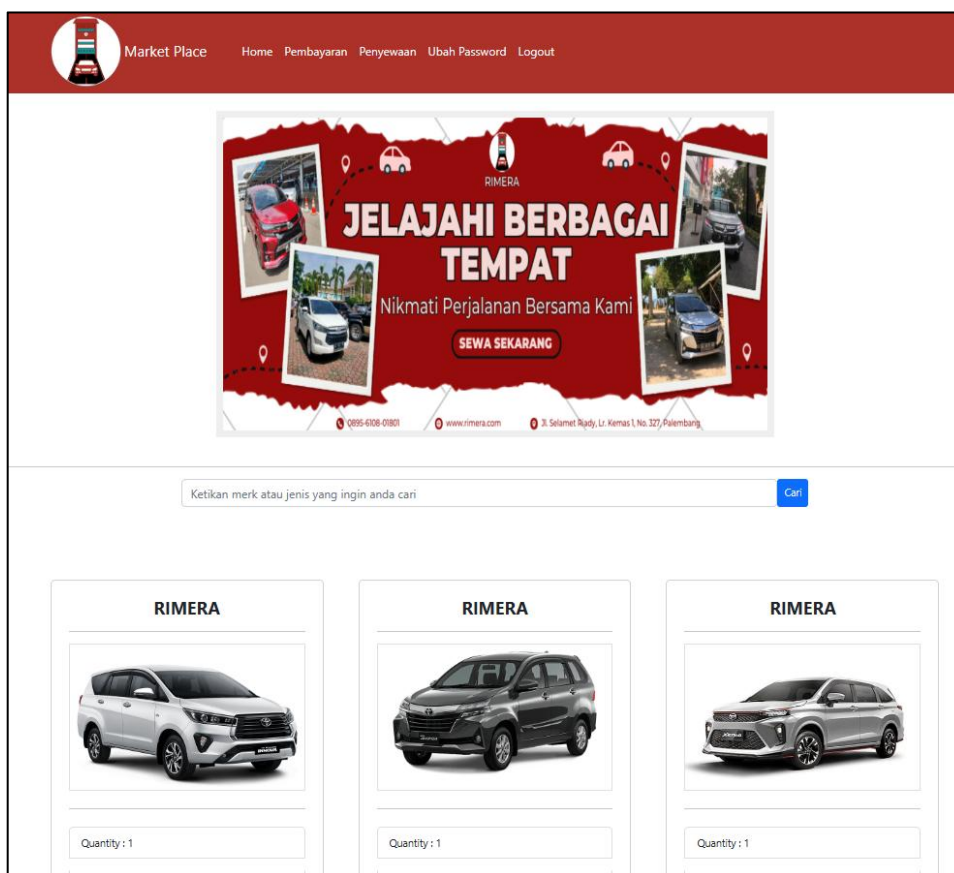
Sumber : Peneliti

**Gambar 5.55** Hasil Desain Halaman Pencairan Pada Penyedia



## 20. Hasil desain halaman *dashboard* penyewa

Hasil desain halaman *dashboard* penyewa berisi tampilan awal setelah penyedia melakukan *login* pada *e-marketplace* Rimera. Pada halaman ini berisi informasi data mobil yang tersedia untuk disewa. Adapun hasil desain *dashboard* penyewa dilihat pada gambar 5.56.



Sumber : Peneliti

**Gambar 5.56** Hasil Desain Halaman *Dashboard* Pada Penyewa

## 21. Hasil desain halaman data pembayaran pada penyewa

Hasil desain halaman data pembayaran penyewa berisi informasi data pembayaran yang telah dilakukan oleh penyewa pada *e-marketplace* Rimera. Adapun hasil desain halaman data pembayaran pada penyewa dilihat pada gambar 5.57.

**Data Pembayaran**

Note : Apabila status telah berubah menjadi lunas, mobil akan di antar ke alamat yang anda ajukan!

Show  entries Search:

No	Status Transaksi	Status Kendaraan	Tanggal Bayar	Bukti Bayar	Keterangan	Alamat	Status
1	<a href="#">Lihat</a>	<a href="#">Lihat</a>	2023-08-07	<a href="#">Download</a>	warna putih	Jl. Selamat Riady, Lr. Kemas 1, Gang Majapahit, No. 327, Palembang, Sumatera Selatan	<span>Pengecekan</span>
2	<a href="#">Lihat</a>	<a href="#">Lihat</a>	2023-07-25	<a href="#">Download</a>	dikirimkan jam 12 siang	sesuai alamat ktp	<span>Lunas</span>
3	<a href="#">Lihat</a>	<a href="#">Lihat</a>	2023-07-25	<a href="#">Download</a>	1	1	<span>Pengecekan</span>

Showing 1 to 3 of 3 entries Previous  Next

Sumber : Peneliti

**Gambar 5.57** Hasil Desain Halaman Pembayaran Pada Penyewa

## 22. Hasil desain halaman form pembayaran pada penyewa

Hasil desain halaman *form* pembayaran berisi informasi penginputan bukti pembayaran penyewaan mobil yang dilakukan oleh penyewa, pada *form* ini juga penyewa dapat memberikan informasi keterangan tambahan dan alamat pengantaran mobil. Adapun hasil desain halaman *form* pembayaran dapat dilihat pada gambar 5.58.

Sumber : Peneliti

**Gambar 5.58** Hasil Desain *Form* Pembayaran Pada Penyewa

### 23. Hasil desain halaman penyewaan pada penyewa

Hasil desain halaman data penyewaan penyedia berisi informasi data transaksi penyewaan mobil pada *e-marketplace* Rimera. Adapun hasil desain halaman data penyewaan pada penyedia dilihat pada gambar 5.59.

**JELAJAHI BERBAGAI TEMPAT**  
Nikmati Perjalanan Bersama Kami  
**SEWA SEKARANG**

Penyewaan Sedang Berlangsung

No	Aksi	Status Kendaraan	Status Penyewaan	Status Pengantaran	No Transaksi	No Invoice	Quantity Sewa	Tanggal Awal Sewa	Tanggal Selesai Sewa	Lama Sewa	Kondisi
1	<a href="#">Print</a> <a href="#">Lihat</a>	Prepocokan	Sudah Diantar		1	PY202308071	1	2023-08-08	2023-08-12	5 Hari	Tanpa Driver
2	<a href="#">Print</a> <a href="#">Lihat</a>	Sudah Diembalikan	Sudah Diantar		3	PY202307253	1	2023-07-26	2023-07-27	2 Hari	Tanpa Driver
3	<a href="#">Print</a> <a href="#">Lihat</a>	Sudah Diembalikan	Sudah Diantar		1	PY202307251	1	2023-07-23	2023-07-24	2 Hari	Tanpa Driver

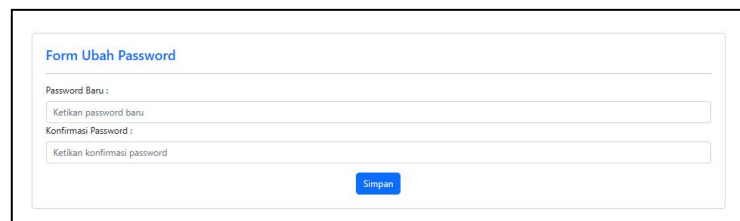
Showing 1 to 3 of 3 entries

Sumber : Peneliti

**Gambar 5.59** Hasil Desain Halaman Penyewaan Pada Penyewa

#### 24. Hasil desain halaman ubah *password*

Hasil desain halaman ubah *password* berisi *form* penginputan *password* yang dapat diubah oleh masing – masing *user*. Adapun hasil desain halaman ubah *password* dilihat pada gambar 5.60.



Sumber : Peneliti

**Gambar 5.60** Hasil Desain *Form Ubah Password*

#### 25. Hasil desain *logout*

Hasil desain halaman *logout* adalah informasi pemberitahuan kepada masing – masing *user*, apakah *user* yakin untuk menghentikan sesi. Adapun desain *logout* dilihat pada gambar 5.61.



Sumber : Peneliti

**Gambar 5.61** Hasil Desain *Logout*

#### 5.1.3.4. Alat dan teknik pengujian

Peneliti menggunakan metode pengujian *white box* dengan teknik *Statement Coverage*. Pengujian dilakukan dengan memilih fungsi yang memiliki percabangan kemudian dihitung jumlah cakupan dan kondisi yang telah ditentukan disetiap fungsi *use case*. Adapun beberapa pemilihan pengujian

berdasarkan daftar *use case* dan fungsi yang terdapat dalam setiap kelas dapat dilihat pada tabel 5.10

**Tabel 5.10** Daftar Tabel *Method*

No. Use Case	Nama File Controller	Nama Fungsi	Jumlah Statement	Jumlah Kondisi
UC-01	Home	+ index	15	5
		+ pencarian	12	3
		+ sewa	4	0
UC-02	Login	+ index	9	4
		+ proses	13	3
		+ logout	4	0
UC-03	Kendaraan	+ index	13	3
		+ edit	8	3
		+ detail	3	0
		+ cari	2	0
		+ hapus	10	2
UC-04	Pembayaran	+ index	5	2
		+ view	18	4
		+ pilih	3	0
		+ lunas	7	0
UC-05	Pencairan	+ index	11	3
		+ print	4	2
UC-06	Pengguna	+ penyedia	6	2

No. Use Case	Nama File Controller	Nama Fungsi	Jumlah Statement	Jumlah Kondisi
		+ penyewa	6	2
		+ penyedia_edit	4	2
		+ penyedia_hapus	4	0
		+ penyewa_edit	4	2
		+ penyewa_hapus	6	0
UC-07	Slider	+ index	6	2
		+ edit	4	2
		+ hapus	3	0

Sumber : Peneliti

## 1. Pengujian UC-01

Pengujian pada UC-01 dengan nama *controller Home*, terdapat fungsi *index*, pencarian, dan sewa. Adapun pengujian yang dilakukan adalah sebagai berikut.

### a. Index

Fungsi *index* terdapat terdapat 15 *statement*, dan 5 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.62.

```

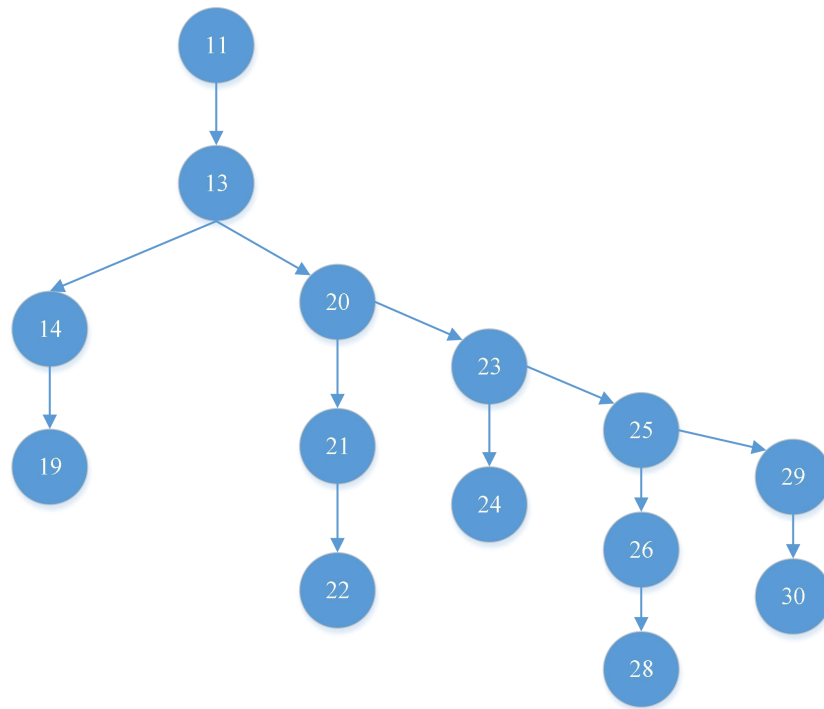
11 public function index()
12 {
13     if (Auth::guard('penyewa')->check()) {
14         $slider = DB::select('SELECT * from sliders');
15         $tersedia = DB::select(
16             "SELECT
17             * from tersedias where tersedias.quantity != 0"
18         );
19         return view('penyewa.home', ['tersedia' => $tersedia, 'slider' => $slider]);
20     } elseif (Auth::guard('penyedia')->check()) {
21         $slider = DB::select('SELECT * from sliders');
22         return view('penyedia.home', ['slider' => $slider]);
23     } elseif (Auth::guard('admin')->check()) {
24         return view('admin.home');
25     } elseif (Route::is('home')) {
26         $tersedia = DB::select('SELECT * from tersedias where tersedias.quantity != 0 order by rand() limit 3');
27         $slider = DB::select('SELECT * from sliders');
28         return view('depan.home', ['slider' => $slider, 'tersedia' => $tersedia]);
29     } else {
30         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
31     }
32 }
33

```

Sumber : Peneliti

**Gambar 5.62** Source Code Fungsi Index Pada Controller Home

Pengubahan *source code* menjadi *flowgraph* pada fungsi *index* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.63.



Sumber : Peneliti

**Gambar 5.63** *Flowgraph* Fungsi *Index* Pada *Controller Home*

Berdasarkan *flowgraph* pada fungsi *index* suspect total 15 *statement*, 14 *node* (N), dan 13 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 14 - 13 + 2 = 3 + 2 = 5$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 5. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *index* berdasarkan perolehan CC yaitu ada 5 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-011 = 11, 13, 14, 19
- 2) Jalur P-012 = 11, 13, 20, 21, 22

- 3) Jalur P-013 = 11, 13, 23, 24
- 4) Jalur P-014 = 11, 13, 25, 26
- 5) Jalur P-013 = 11, 13, 29, 30

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 5 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *index* dapat dilihat pada gambar 5.64.

```

14     public function test_succrssfull_response(): void
15     {
16         $response = $this->get('/penyedia/home');
17
18         $response->assertStatus(200);
19     }
20 }

```

Sumber : Peneliti

**Gambar 5.64** Source Code Pengujian Test Fungsi Index Pada Controller Home

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.65.

```

• PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ example

Tests:    1 passed (1 assertions)
Duration: 13.34s

```

Sumber : Peneliti

**Gambar 5.65** Hasil Pengujian Test Fungsi Index Pada Controller Home



## b. Pencarian

Fungsi pencarian terdapat terdapat 12 *statement*, dan 3 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.66.

```

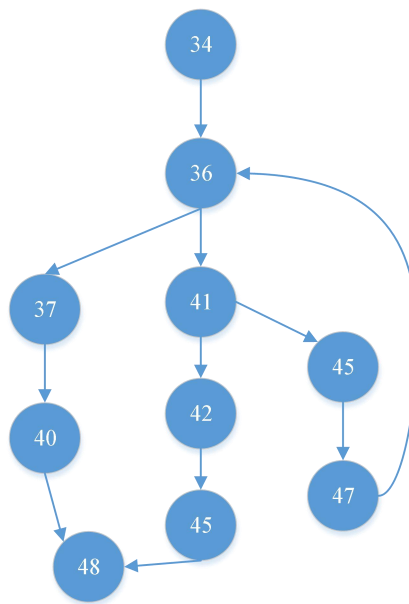
34 public function pencarian(Request $request)
35 {
36     if (Auth::guard('penyewa')->check()) {
37         $slider = DB::select('SELECT * from sliders');
38         $merk = str($request->merk);
39         $tersedia = DB::select("SELECT * from tersedias where tersedias.merk LIKE '%$merk%' and tersedias.quantity != 0");
40         return view('penyewa.pencarian', ['tersedia' => $tersedia, 'slider' => $slider]);
41     } elseif (url('/pencarian')) {
42         $slider = DB::select('SELECT * from sliders');
43         $merk = str($request->merk);
44         $tersedia = DB::select("SELECT * from tersedias where tersedias.merk LIKE '%$merk%' and tersedias.quantity != 0");
45         return view('depan.pencarian', ['tersedia' => $tersedia, 'slider' => $slider]);
46     } else {
47         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
48     }
49 }
50 }

```

Sumber : Peneliti

**Gambar 5.66** Source Code Fungsi Pencarian Pada Controller Home

Pengubahan *source code* menjadi *flowgraph* pada fungsi pencarian untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.67.



Sumber : Peneliti

**Gambar 5.67** Flowgraph Fungsi Pencarian Pada Controller Home

Berdasarkan *flowgraph* pada fungsi pencarian suspect total 12 *statement*, 10 *node* (N), dan 11 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 10 - 11 + 2 = 1 + 2 = 3$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 3. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi pencarian berdasarkan perolehan CC yaitu ada 3 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-021 = 34, 35, 37, 40, 48
- 2) Jalur P-022 = 34, 41, 42, 45, 48
- 3) Jalur P-023 = 34, 41, 46, 47, 36

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 3 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test -filter*. Adapun gambar *source code* pengujian pada fungsi pencarian dapat dilihat pada gambar 5.68.

```

26     public function testRead()
27     {
28         $tersedia= tersedia::find(1);
29         $this->assertEquals('DAIHATSU Sibra R',$tersedia->merk);
30     }
31 }

```

Sumber : Peneliti

**Gambar 5.68** *Source Code* Pengujian *Test* Fungsi Pencarian Pada *Controller Home*

Berdasarkan tahapan yang telah dilakukan, diperoleh 2 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.69.

```

PS C:\xampp\htdocs\nabila> php artisan test --filter AdminTest
Warning: TTY mode is not supported on Windows platform.

PASS Tests\Feature\AdminTest
✓ example
✓ read

Tests: 2 passed
Time: 0.66s

```

Sumber : Peneliti

**Gambar 5.69** Hasil Pengujian *Test* Pencarian Pada *Controller Home*

c. Sewa

Fungsi sewa terdapat terdapat 3 *stetment*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.70.

```

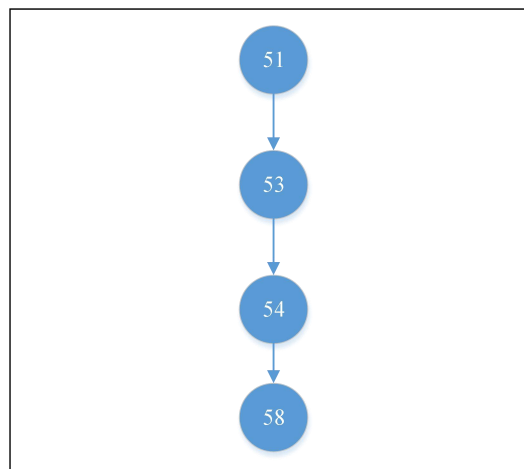
51 public function sewa()
52 {
53     $slider = DB::select('SELECT * from sliders');
54     $tersedia = DB::select(
55         "SELECT
56         * from tersedias where tersedias.quantity != 0"
57     );
58     return view('depan.sewa', ['tersedia' => $tersedia, 'slider' => $slider]);
59 }
60 }
61

```

Sumber : Peneliti

**Gambar 5.70** *Source Code* Fungsi Sewa Pada *Controller Home*

Pengubahan *source code* menjadi *flowgraph* pada fungsi sewa untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.71.



Sumber : Peneliti

**Gambar 5.71** *Flowgraph* Fungsi Sewa Pada *Controller Home*

Berdasarkan *flowgraph* pada fungsi sewa suspect total 4 *statement*, 4 *node* (N), dan 3 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 3 - 4 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi sewa berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-031 = 51, 53, 54, 58

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi sewa dapat dilihat pada gambar 5.72.

```

25     public function test_succrssl_response(): void
26     {
27         $response = $this->get('/sewa');
28
29         $response->assertStatus(200);
30     }
31 }
32

```

Sumber : Peneliti

**Gambar 5.72** *Source Code* Pengujian *Test* Fungsi Sewa Pada *Controller Home*

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.73.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 0.44s

```

Sumber : Peneliti

**Gambar 5.73** Hasil Pengujian *Test* Fungsi Sewa Pada *Controller Home*

## 2. Pengujian UC-02

Pengujian pada UC-02 dengan nama *Controller Login*, terdapat fungsi *index*, proses, dan *logout*. Adapun pengujian yang dilakukan adalah sebagai berikut.

### a. *Index*

Fungsi *index* terdapat terdapat 9 *stetment*, dan 4 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.74.

```

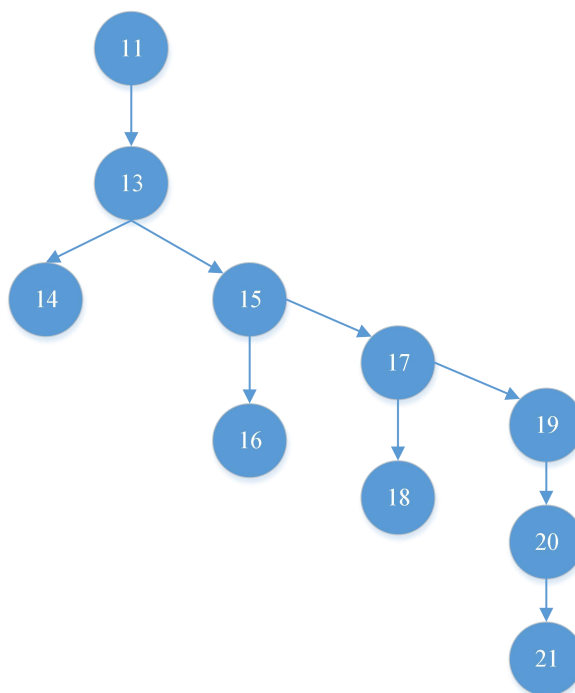
11 public function index()
12 {
13     if (Auth::guard('admin')->check()) {
14         return redirect('/admin/home');
15     } elseif (Auth::guard('penyedia')->check()) {
16         return redirect('/penyedia/home');
17     } elseif (Auth::guard('penyewa')->check()) {
18         return redirect('/penyewa/home');
19     } else {
20         $slider = DB::select('SELECT * from sliders');
21         return view('depan.login', ['slider' => $slider]);
22     }
23 }

```

Sumber : Peneliti

**Gambar 5.74** *Source Code* Fungsi *Index* Pada *Controller Login*

Pengubahan *source code* menjadi *flowgraph* pada fungsi *index* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.75.



Sumber : Peneliti

**Gambar 5.75** *Flowgraph* Fungsi *Index* Pada *Controller Login*

Berdasarkan *flowgraph* pada fungsi *index* suspect total 9 *statement*, 10 *node* (N), dan 10 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 10 - 10 + 2 = 2 + 2 = 4$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 4. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *index* berdasarkan perolehan CC yaitu ada 4 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-011 = 11, 13, 14
- 2) Jalur P-012 = 11, 15, 16
- 3) Jalur P-013 = 11, 17, 18
- 4) Jalur P-014 = 11, 19, 20, 21

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 4 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *index* dapat dilihat pada gambar 5.76.

```

24
25     public function test_succrssl_response(): void
26     {
27         $response = $this->get('/admin/home');
28
29         $response->assertStatus(200);
30     }

```

Sumber : Peneliti

**Gambar 5.76** *Source Code* Pengujian *Test Fungsi Index* Pada *Controller Login*

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.77.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssl response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.77** Hasil Pengujian *Test Fungsi Index* Pada *Controller Login*

#### b. Proses

Fungsi proses terdapat terdapat 13 *stetment*, dan 3 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.78.

```

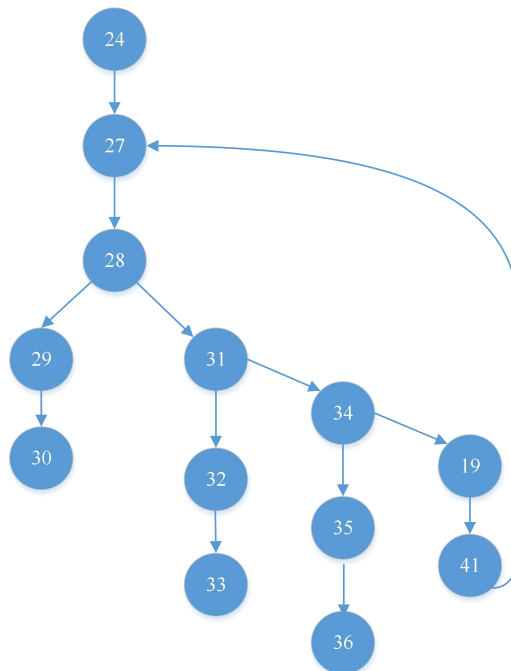
24     public function proses(Request $request)
25     {
26
27         $kredensial = $request->only(['username', 'password']);
28         if (Auth::guard('admin')->attempt($kredensial)) {
29             $request->session()->regenerate();
30             return redirect('/admin/home');
31         } elseif (Auth::guard('penyedia')->attempt($kredensial)) {
32             $request->session()->regenerate();
33             return redirect('/penyedia/home');
34         } elseif (Auth::guard('penyewa')->attempt($kredensial)) {
35             $request->session()->regenerate();
36             return redirect('/penyewa/home');
37         }
38
39         return back()->withErrors([
40             'username' => 'Maaf nama atau password anda salah'
41         ])->onlyInput('username');
42     }
43 }

```

Sumber : Peneliti

**Gambar 5.78** Source Code Fungsi Proses Pada Controller Login

Pengubahan *source code* menjadi *flowgraph* pada fungsi proses untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.79.



Sumber : Peneliti

**Gambar 5.79** Flowgraph Fungsi Proses Pada Controller Login



Berdasarkan *flowgraph* pada fungsi proses suspect total 13 *statement*, 13 *node* (N), dan 13 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 13 - 13 + 2 = 2 + 2 = 4$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 4. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi proses berdasarkan perolehan CC yaitu ada 4 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-021 = 24, 27, 28, 29, 30
- 2) Jalur P-022 = 24, 27, 31, 32, 33
- 3) Jalur P-023 = 24, 27, 34, 35, 36
- 4) Jalur P-024 = 24, 27, 39, 41, 27

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 4 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi proses dapat dilihat pada gambar 5.80.

```

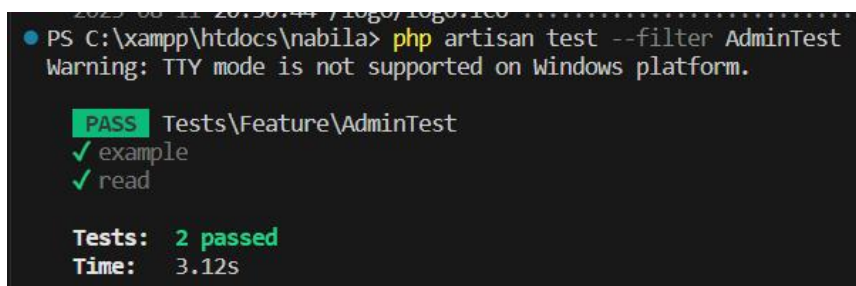
23
24     public function testRead()
25     {
26         $admin = Admin::find(2);
27         $this->assertEquals('Nabila',$admin->nama);
28     }
29 }

```

Sumber : Peneliti

**Gambar 5.80** *Source Code* Pengujian *Test* Fungsi Proses Pada *Controller Login*

Berdasarkan tahapan yang telah dilakukan, diperoleh 2 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.81.



```

2023-08-11 20:30:44 /logs/logs-100 .....
PS C:\xampp\htdocs\nabila> php artisan test --filter AdminTest
Warning: TTY mode is not supported on windows platform.

 PASS Tests\Feature\AdminTest
 ✓ example
 ✓ read

Tests:  2 passed
Time:   3.12s

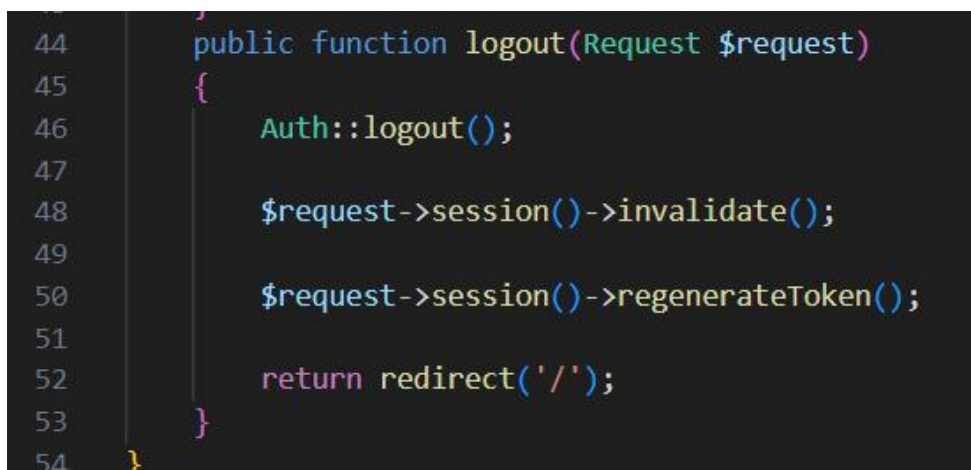
```

Sumber : Peneliti

**Gambar 5.81** Hasil Pengujian *Test* Fungsi Proses Pada *Controller Login*

### c. *Logout*

Fungsi *logout* terdapat terdapat 4 *statement*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.82.



```

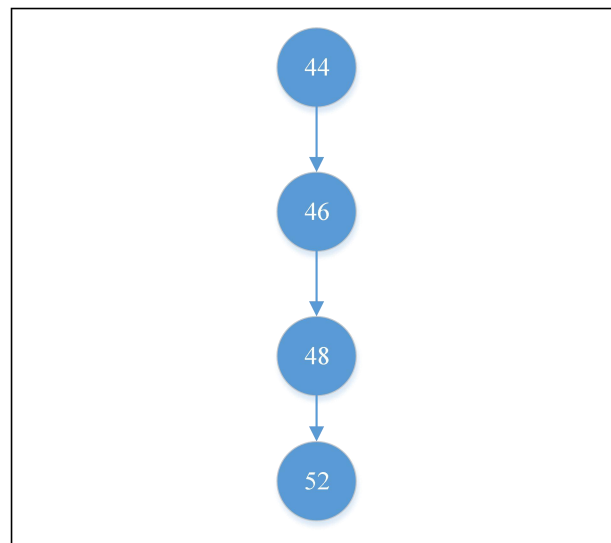
44     public function logout(Request $request)
45     {
46         Auth::logout();
47
48         $request->session()->invalidate();
49
50         $request->session()->regenerateToken();
51
52         return redirect('/');
53     }
54 }

```

Sumber : Peneliti

**Gambar 5.82** *Source Code* Fungsi *Logout* Pada *Controller Login*

Pengubahan *source code* menjadi *flowgraph* pada fungsi *logout* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.83.



Sumber : Peneliti

**Gambar 5.83** *Flowgraph* Fungsi *Logout* Pada *Controller Login*

Berdasarkan *flowgraph* pada fungsi *logout suspect* total 4 *statement*, 4 *node* (N), dan 3 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 3 - 4 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *logout* berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-031 = 44, 46, 48, 52

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *logout* dapat dilihat pada gambar 5.84.

```

25     public function test_succrssl_response(): void
26     {
27         $response = $this->get('/sewa');
28
29         $response->assertStatus(200);
30     }
31 }
32

```

Sumber : Peneliti

**Gambar 5.84** *Source Code* Pengujian *Test* Fungsi *Logout* Pada *Controller Login*

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.85.

```

● PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest
PASS Tests\Feature\AdminTest
✓ succrssl response

Tests: 1 passed (1 assertions)
Duration: 0.44s

```

Sumber : Peneliti

**Gambar 5.85** Hasil Pengujian *Test* Fungsi *Logout* Pada *Controller Login*

### 3. Pengujian UC-03

Pengujian pada UC-03 dengan nama *controller* kendaraan, terdapat fungsi *index*, *edit*, *detail*, cari, dan hapus. Adapun pengujian yang dilakukan adalah sebagai berikut.

#### a. *Index*

Fungsi *index* terdapat terdapat 13 *stetment*, dan 3 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.86.

```

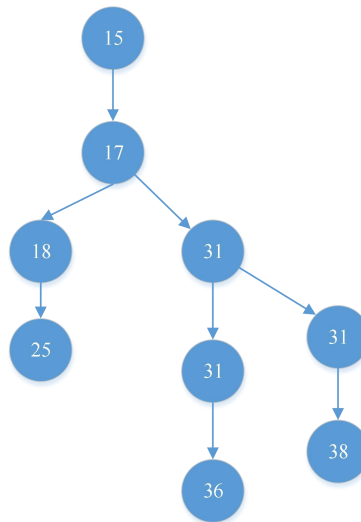
15 public function index(Kendaraan $kendaraan)
16 {
17     if (Auth::guard('penyedia')->check()) {
18         $slider = DB::select('SELECT * from sliders');
19         $no = 1;
20         $id = Auth::guard('penyedia')->user()->id;
21         $data = DB::select(
22             "select *, kendaraan.tanpa_driver as harga_tanpa_driver, kendaraan.dengan_driver as harga_dengan_driver,
23             kendaraan.id as id_kendaraan from kendaraan inner join tersedia on kendaraan.id_tersedia = tersedia.id
24             inner join penyedia on kendaraan.id_penyedia = penyedia.id where kendaraan.id_penyedia = $id "
25         );
26         $tersedia = DB::select("select DISTINCT tersedia.merk, merk from tersedia where tersedia.id not in
27         (select id_tersedia from kendaraan where id_penyedia = " . $id . ")");
28         return view('penyedia.kendaraan', [
29             'data' => $data,
30             'tersedia' => $tersedia,
31             'no' => $no,
32             'slider' => $slider
33         ]);
34     } elseif (Auth::guard('admin')->check()) {
35         $no = 1;
36         $data = DB::select(
37             "select * from tersedia"
38         );
39         return view('admin.kendaraan', ['data' => $data, 'no' => $no]);
40     } else {
41         return redirect('/login');
42     }
43 }

```

Sumber : Peneliti

**Gambar 5.86** Source Code Fungsi Index Pada Controller Kendaraan

Pengubahan *source code* menjadi *flowgraph* pada fungsi *index* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.87.



Sumber : Peneliti

**Gambar 5.87** Flowgraph Fungsi Index Pada Controller Kendaraan

Berdasarkan *flowgraph* pada fungsi *index* suspect total 13 *statement*, 9 *node* (N), dan 8 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 8 - 9 + 2 = -1 + 2 = 3$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 3. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *index* berdasarkan perolehan CC yaitu ada 3 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-011 = 15, 17, 18, 25
- 2) Jalur P-012 = 15, 31, 32, 36
- 3) Jalur P-013 = 15, 37, 38

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 3 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test -filter*. Adapun gambar *source code* pengujian pada fungsi *index* dapat dilihat pada gambar 5.88.

```

24 /
25     public function test_succrssl_response(): void
26     {
27         $response = $this->get('/admin/home');
28
29         $response->assertStatus(200);
30     }

```

Sumber : Peneliti

**Gambar 5.88** *Source Code* Pengujian *Test* Fungsi *Index* Pada *Controller* Kendaraan

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.89.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.89** Hasil Pengujian *Test* Fungsi *Index* Pada *Controller* Kendaraan

b. *Edit*

Fungsi *edit* terdapat terdapat 8 *stetment*, dan 3 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.90.

```

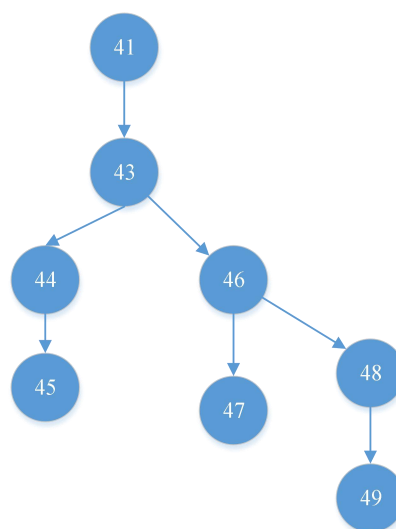
44 public function edit(Kendaraan $kendaraan, Tersedia $tersedia)
45 {
46     if (Auth::guard('penyedia')->check()) {
47         $slider = DB::select('SELECT * from sliders');
48         return view('penyedia.kendaraan_edit', ['kendaraan' => $kendaraan, 'tersedia' => $tersedia, 'slider' => $slider]);
49     } elseif (Auth::guard('admin')->check()) {
50         return view('admin.kendaraan_edit', ['tersedia' => $tersedia]);
51     } else {
52         return redirect('/login');
53     }
54 }

```

Sumber : Peneliti

**Gambar 5.90** *Source Code* Fungsi *Edit* Pada *Controller* Kendaraan

Pengubahan *source code* menjadi *flowgraph* pada fungsi *edit* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.91.



Sumber : Peneliti

**Gambar 5.91** *Flowgraph* Fungsi *Edit* Pada *Controller* Kendaraan

Berdasarkan *flowgraph* pada fungsi *edit suspect* total 8 *statement*, 8 *node* (N), dan 7 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 7 - 8 + 2 = -1 + 2 = 3$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 3. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *edit* berdasarkan perolehan CC yaitu ada 3 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-021 = 42, 43, 44, 45
- 2) Jalur P-022 = 41, 46, 47
- 3) Jalur P-023 = 41, 48, 49

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 3 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test -filter*. Adapun gambar *source code* pengujian pada fungsi *edit* dapat dilihat pada gambar 5.92.

```

32     public function testUpdate()
33     {
34         $kendaraan=Kendaraan::find(1);
35         $kendaraan->id_terdedia='1';
36         $kendaraan->id_penyedia='1';
37         $kendaraan->qty='3';
38         $this->assertTime($kendaraan->Update());
39     }

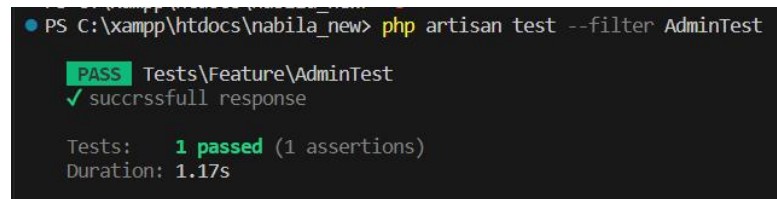
```

Sumber : Peneliti

**Gambar 5.92** *Source Code* Pengujian *Test Fungsi Edit* Pada *Controller* Kendaraan



Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.93.



```
PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s
```

Sumber : Peneliti

**Gambar 5.93** Hasil Pengujian *Test Fungsi Edit* Pada *Controller* Kendaraan

### c. *Detail*

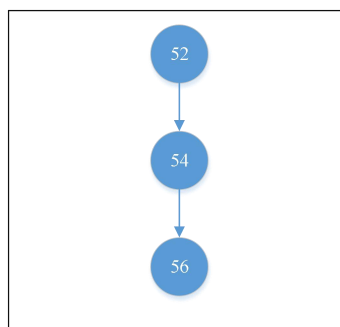
Fungsi *detail* terdapat terdapat 3 *stetment*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.94.

```
55 public function detail(Tersedia $tersedia)
56 {
57     $no = 1;
58     $data = DB::select('SELECT *, kendaaraans.tanpa_driver as harga_tanpa_driver, kendaaraans.dengan_driver
59 as harga_dengan_driver from kendaaraans inner join tersedias on kendaaraans.id_tersedia = tersedias.id
60 inner join penyedia on kendaaraans.id_penyedia = penyedia.id where kendaaraans.id_tersedia = ' . $tersedia->id);
61     return view('admin.detail', ['data' => $data, 'no' => $no]);
62 }
```

Sumber : Peneliti

**Gambar 5.94** *Source Code* Fungsi *Detail* Pada *Controller* Kendaraan

Pengubahan *source code* menjadi *flowgraph* pada fungsi *detail* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.95.



Sumber : Peneliti

**Gambar 5.95** *Flowgraph* Fungsi *Detail* Pada *Controller* Kendaraan

Berdasarkan *flowgraph* pada fungsi *detail suspect* total 3 *statement*, 3 *node* (N), dan 2 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = -1 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *detail* berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-031 = 52, 54, 56

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *detail* dapat dilihat pada gambar 5.96.

```

31
32     public function testRead()
33     {
34         $kendaraan=Kendaraan::find(1);
35
36         $this->assertEquals('1',$kendaraan->id_tersedia);
37     }
38

```

Sumber : Peneliti

**Gambar 5.96** *Source Code* Pengujian *Test* Fungsi *Detail* Pada *Controller* Kendaraan

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat,

maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.97.

```

● PS C:\xampp\htdocs\nabila_new> php artisan test --filter NabilaTest

PASS Tests\Feature\WabilaTest
✓ read

Tests: 1 passed (1 assertions)
Duration: 0.68s

```

Sumber : Peneliti

**Gambar 5.97** Hasil Pengujian *Test Fungsi Detail* Pada *Controller* Kendaraan

d. Cari

Fungsi cari terdapat terdapat 2 *statement*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.98.

```

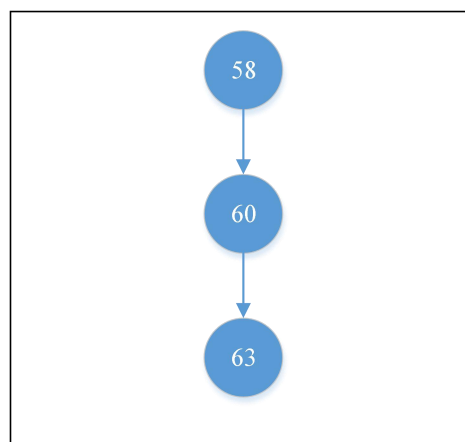
63 public function cari(Request $request)
64 {
65     $merk = str($request->merk);
66     $slider = DB::select('SELECT * from sliders');
67     $data = DB::select(["SELECT * from tersedias where tersedias.id not in (select id_tersedia from kendaraan
68     where id_penyedia = " . Auth::guard('penyedia')->user()->id . ") and merk LIKE '%$merk%'"]);
69     return view('penyedia.cari', ['data' => $data, 'slider' => $slider]);
70 }

```

Sumber : Peneliti

**Gambar 5.98** *Source Code* Fungsi Cari Pada *Controller* Kendaraan

Pengubahan *source code* menjadi *flowgraph* pada fungsi cari untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.99.



Sumber : Peneliti

**Gambar 5.99** *Flowgraph* Fungsi Cari Pada *Controller* Kendaraan

Berdasarkan *flowgraph* pada fungsi cari suspect total 3 *statement*, 3 *node* (N), dan 2 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = -1 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi cari berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-041 = 52, 54, 56

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi cari dapat dilihat pada gambar 5.100.

```

31
32     public function testRead()
33     {
34         $kendaraan=Kendaraan::find(1);
35
36         $this->assertEquals('1',$kendaraan->id_tersedia);
37     }
38

```

Sumber : Peneliti

**Gambar 5.100** *Source Code* Pengujian *Test* Fungsi Cari Pada *Controller* Kendaraan

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat,

maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.101.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter NabilaTest

PASS Tests\Feature\NabilaTest
✓ read

Tests: 1 passed (1 assertions)
Duration: 0.68s
  
```

Sumber : Peneliti

**Gambar 5.101** Hasil Pengujian Fungsi Cari Pada *Controller* Kendaraan

#### e. Hapus

Fungsi hapus terdapat terdapat 10 *statement*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.102.

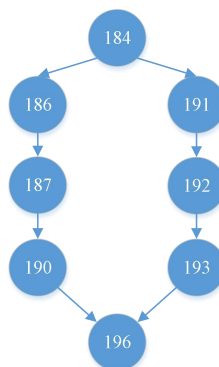
```

184 public function hapus(Kendaraan $kendaraan, Tersedia $tersedia)
185 {
186     if (Auth::guard('penyedia')->check()) {
187         $tersedia->quantity = $tersedia->quantity - $kendaraan->qty;
188         $tersedia->update();
189         $kendaraan->delete();
190         return redirect(Route('kendaraan.index'))->with('pesan', 'Kendaraan berhasil di hapus');
191     } elseif (Auth::guard('admin')) {
192         DB::delete("DELETE from kendaaraans where id_tersedia = ' . $tersedia->id);
193         File::delete("uploads/foto_mobil/" . $tersedia->foto_mobil);
194         $tersedia->delete();
195         return redirect(Route('kendaraan.admin'))->with('pesan', 'Kendaraan berhasil di hapus');
196     }
197 }
  
```

Sumber : Peneliti

**Gambar 5.102** *Source Code* Fungsi Hapus Pada *Controller* Kendaraan

Pengubahan *source code* menjadi *flowgraph* pada fungsi hapus untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.103.



Sumber : Peneliti

**Gambar 5.103** *Flowgraph* Fungsi Hapus Pada *Controller* Kendaraan

Berdasarkan *flowgraph* pada fungsi cari suspect total 10 *statement*, 8 *node* (N), dan 8 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 8 - 8 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi cari berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-051 = 184, 186, 187, 190, 196
- 2) Jalur P-052 = 184, 191, 192, 193, 196

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi hapus dapat dilihat pada gambar 5.104.

```

32     public function testDelete()
33     {
34         $kendaraan=Kendaraan::find(42);
35
36         $this->assertEquals('10000',$kendaraan->qty);
37     }
38

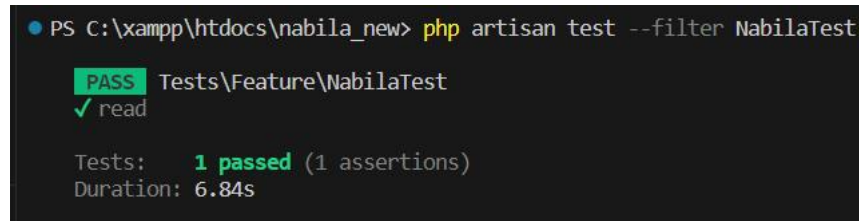
```

Sumber : Peneliti

**Gambar 5.104** *Source Code* Pengujian *Test* Fungsi Hapus Pada *Controller* Kendaraan

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat,

maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.105.



```
PS C:\xampp\htdocs\nabila_new> php artisan test --filter NabilaTest
PASS Tests\Feature\NabilaTest
✓ read

Tests: 1 passed (1 assertions)
Duration: 6.84s
```

Sumber : Peneliti

**Gambar 5.105** Hasil Pengujian Fungsi Hapus Pada *Controller* Kendaraan

#### 4. Pengujian UC-04

Pengujian pada UC-04 dengan nama *controller* pembayaran, terdapat fungsi *index*, *view*, *detail*, *pilih*, dan *lunas*. Adapun pengujian yang dilakukan adalah sebagai berikut.

##### a. *Index*

Fungsi *index* terdapat terdapat 5 *statement*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.106.

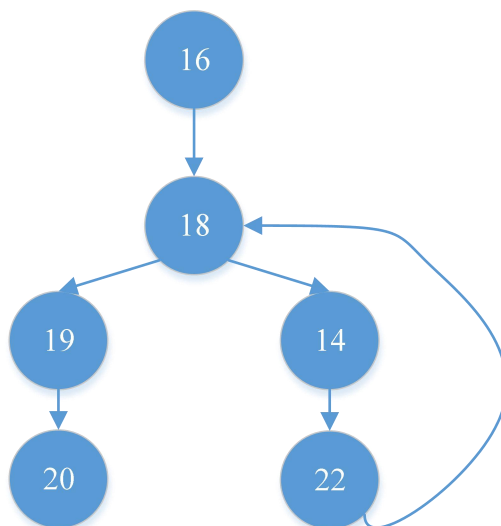


```
15 public function index(Kendaraan $kendaraan)
16 {
17     if (Auth::guard('penyedia')->check()) {
18         $slider = DB::select('SELECT * from sliders');
19         $no = 1;
20         $id = Auth::guard('penyedia')->user()->id;
21         $data = DB::select(
22             "select *, kendaaraans.tanpa_driver as harga_tanpa_driver, kendaaraans.dengan_driver as harga_dengan_driver,
23             kendaaraans.id as id_kendaraan from kendaaraans inner join tersedias on kendaaraans.id_tersedia = tersedias.id
24             inner join penyedias on kendaaraans.id_penyedia = penyedias.id where kendaaraans.id_penyedia = $id "
25         );
26         $tersedia = DB::select("select DISTINCT tersedias.merk, merk from tersedias where tersedias.id not in
27         (select id_tersedia from kendaaraans where id_penyedia = " . $id . ")");
28         return view('penyedia.kendaraan', [
29             'data' => $data,
30             'tersedia' => $tersedia,
31             'no' => $no,
32             'slider' => $slider
33         ]);
34     } elseif (Auth::guard('admin')->check()) {
35         $no = 1;
36         $data = DB::select(
37             "select * from tersedias"
38         );
39         return view('admin.kendaraan', ['data' => $data, 'no' => $no]);
40     } else {
41         return redirect('/login');
42     }
43 }
```

Sumber : Peneliti

**Gambar 5.106** *Source Code* Fungsi *Index* Pada *Controller* Pembayaran

Pengubahan *source code* menjadi *flowgraph* pada fungsi *index* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.107.



Sumber : Peneliti

**Gambar 5.107** *Flowgraph* Fungsi *Index* Pada *Controller* Pembayaran

Berdasarkan *flowgraph* pada fungsi *index* suspect total 5 *statement*, 6 *node* (N), dan 6 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 6 - 6 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *index* berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-011 = 16, 18, 19, 20
- 2) Jalur P-012 = 16, 18, 21, 22

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat



dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *index* dapat dilihat pada gambar 5.108.

```

24 /
25 public function test_succrssl_response(): void
26 {
27     $response = $this->get('/admin/home');
28
29     $response->assertStatus(200);
30 }

```

Sumber : Peneliti

**Gambar 5.108** *Source Code* Pengujian *Test* Fungsi *Index* Pada *Controller* Pembayaran

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.108.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest
PASS Tests\Feature\AdminTest
✓ succrssl response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.108** Hasil Pengujian *Test* Fungsi *Index* Pada *Controller* Pembayaran

#### b. *View*

Fungsi *view* terdapat terdapat 18 *stetment*, dan 4 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.109.

```

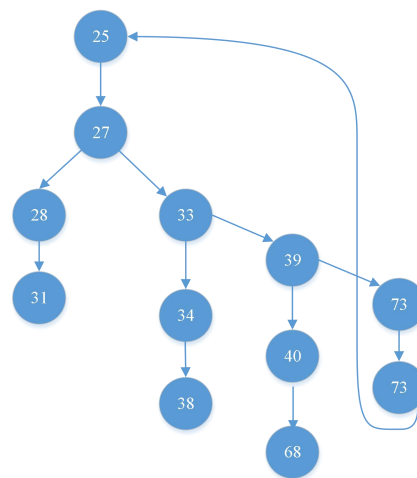
25 public function view()
26 {
27     if (Auth::guard('penyewa')->check()) {
28         $slider = DB::select('SELECT * from sliders');
29         $no = 1;
30         $id = Auth::guard('penyewa')->user()->id;
31         $pembayaran = DB::select('SELECT * from transaksis inner join pembayarans on transaksis.id = pembayarans.id_transaksi
32         where transaksis.id_penyewa = ' . $id . ' order by tanggal_bayar desc limit 25');
33         return view('penyewa.pembayaran_data', ['pembayaran' => $pembayaran, 'no' => $no, 'slider' => $slider]);
34     } elseif (Auth::guard('penyedia')->check()) {
35         $slider = DB::select('SELECT * from sliders');
36         $no = 1;
37         $id = Auth::guard('penyedia')->user()->id;
38         $pembayaran = DB::select('SELECT *,pembayarans.status as status, transaksis.status as status_transaksi from
39         pembayarans inner join transaksis on transaksis.id = pembayarans.id_transaksi inner join kendaraanans on
40         kendaraanans.id = transaksis.id_kendaraan where kendaraanans.id_penyedia = ' . $id . ' order by pembayarans.status
41         asc limit 25');
42         return view('penyedia.pembayaran_data', ['pembayaran' => $pembayaran, 'no' => $no, 'slider' => $slider]);
43     } else if (Auth::guard('admin')->check()) {
44         $no = 1;
45         $pembayaran = DB::select('SELECT
46         *,
47         transaksis.status AS status_transaksi, pembayarans.id as id_pembayaran,
48         pembayarans.status as status,
49         (
50             SELECT penyedias.nama_penyedia
51             FROM penyedias
52             JOIN kendaraanans ON penyedias.id = kendaraanans.id_penyedia
53             WHERE kendaraanans.id = transaksis.id_kendaraan
54             LIMIT 1
55         ) AS nama_penyedia,
56         (
57             SELECT penyedias.id
58             FROM penyedias
59             JOIN kendaraanans ON penyedias.id = kendaraanans.id_penyedia
60             WHERE kendaraanans.id = transaksis.id_kendaraan
61             LIMIT 1
62         ) AS id_penyedia
63     FROM
64         pembayarans
65     INNER JOIN
66         transaksis ON pembayarans.id_transaksi = transaksis.id
67     ORDER BY
68         pembayarans.status ASC
69     LIMIT 25
70     ');
71     };
72     return view('admin.pembayaran', ['pembayaran' => $pembayaran, 'no' => $no]);
73
74
75
76 } else {
77     return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
78 }

```

Sumber : Peneliti

**Gambar 5.109** Source Code Fungsi View Pada Controller Pembayaran

Pengubahan *source code* menjadi *flowgraph* pada fungsi view untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.110.



Sumber : Peneliti

**Gambar 5.110** Flowgraph Fungsi View Pada Controller Pembayaran

Berdasarkan *flowgraph* pada fungsi *view suspect* total 18 *statement*, 12 *node* (N), dan 12 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 12 - 12 + 2 = 2 + 2 = 4$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 4. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *index* berdasarkan perolehan CC yaitu ada 4 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-021 = 25, 27, 28, 32
- 2) Jalur P-022 = 25, 33, 34, 38
- 3) Jalur P-023 = 25, 39, 40, 68
- 4) Jalur P-024 = 25, 71, 73, 25

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *index* dapat dilihat pada gambar 5.111.

```

24
25     public function test_succrssfull_response(): void
26     {
27         $response = $this->get('/admin/home');
28
29         $response->assertStatus(200);
30     }

```

Sumber : Peneliti

**Gambar 5.111** *Source Code* Pengujian *Test* Fungsi *View* Pada *Controller* Pembayaran

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.112.

```
PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ successfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s
```

Sumber : Peneliti

**Gambar 5.112** Hasil Pengujian *Test* Fungsi *View* Pada *Controller* Pembayaran

### c. Pilih

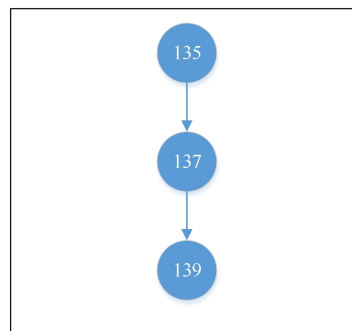
Fungsi pilih terdapat terdapat 3 *stetment*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.113.

```
63 public function cari(Request $request)
64 {
65     $merk = str($request->merk);
66     $slider = DB::select('SELECT * from sliders');
67     $data = DB::select(["SELECT * from tersedia where tersedia.id not in (select id tersedia from kendaraanans
68     |where id_penyedia = " . Auth::guard('penyedia')->user()->id . " ) and merk LIKE '%$merk%'"]);
69     return view('penyedia.cari', ['data' => $data, 'slider' => $slider]);
70 }
```

Sumber : Peneliti

**Gambar 5.113** *Source Code* Fungsi pilih Pada *Controller* Pembayaran

Pengubahan *source code* menjadi *flowgraph* pada fungsi pilih untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.114.



Sumber : Peneliti

**Gambar 5.114** *Flowgraph* Fungsi pilih Pada *Controller* Pembayaran

Berdasarkan *flowgraph* pada fungsi pilih suspect total 3 *statement*, 3 *node* (N), dan 2 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = -1 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi pilih berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-031 = 135, 137, 139

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi pilih dapat dilihat pada gambar 5.115.

```

32     public function testSelect()
33     {
34         $pembayaran=Pembayaran::find(1);
35
36         $this->assertEquals('2023-08-17',$pembayaran->tanggal_bayar);
37     }

```

Sumber : Peneliti

**Gambar 5.115** *Source Code* Pengujian *Test* Fungsi Pilih Pada *Controller* Pembayaran

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.116.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter NabilaTest

PASS Tests\Feature\NabilaTest
✓ select

Tests: 1 passed (1 assertions)
Duration: 0.62s

```

Sumber : Peneliti

**Gambar 5.116** Hasil Pengujian Fungsi Pilih Pada *Controller* Pembayaran

d. Lunas

Fungsi lunas terdapat terdapat 7 *statement*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.117.

```

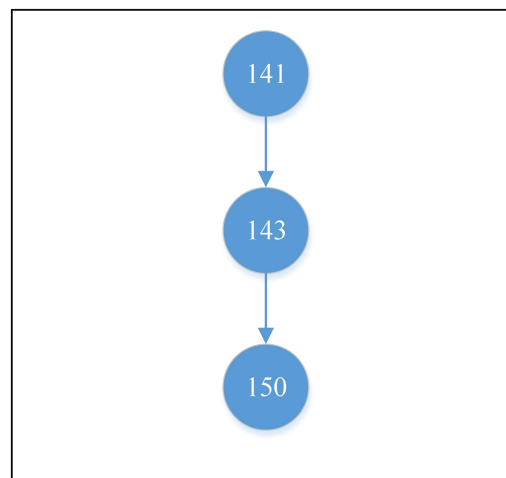
145     public function lunas(Kendaraan $kendaraan, Transaksi $transaksi, Pembayaran $pembayaran)
146     {
147         $kendaraan->qty = $kendaraan->qty - $transaksi->quantity_sewa;
148         $kendaraan->update();
149         $transaksi->id_kendaraan = $kendaraan->id;
150         $transaksi->update();
151         $pembayaran->status = 'Lunas';
152         $pembayaran->update();
153
154         return redirect(Route('pembayaran.admin'))->with('pesan', 'Pembayaran berhasil di lunaskan');
155     }
156

```

Sumber : Peneliti

**Gambar 5.117** *Source Code* Fungsi Lunas Pada *Controller* Pembayaran

Pengubahan *source code* menjadi *flowgraph* pada fungsi lunas untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.118.



Sumber : Peneliti

**Gambar 5.118** *Flowgraph* Fungsi Lunas Pada *Controller* Pembayaran

Berdasarkan *flowgraph* pada fungsi lunas suspect total 7 *statement*, 3 *node* (N), dan 2 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = -1 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi lunas berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-041 = 141, 143, 150

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi Lunas dapat dilihat pada gambar 5.119.

```

32     public function testSelect()
33     {
34         $pembayaran=Pembayaran::find(1);
35
36         $this->assertEquals('2023-08-17',$pembayaran->tanggal_bayar);
37     }

```

Sumber : Peneliti

**Gambar 5.119** *Source Code* Pengujian *Test* Fungsi Lunas Pada *Controller* Pembayaran

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.120.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter NabilaTest

PASS Tests\Feature\NabilaTest
✓ select

Tests: 1 passed (1 assertions)
Duration: 0.62s

```

Sumber : Peneliti

**Gambar 5.120** Hasil Pengujian Fungsi Lunas Pada *Controller* Pembayaran

## 5. Pengujian UC-05

Pengujian pada UC-05 dengan nama *controller* pencairan, terdapat fungsi *index*, dan *print*. Adapun pengujian yang dilakukan adalah sebagai berikut.

### a. Index

Fungsi *index* terdapat terdapat 11 *statement*, dan 3 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.121.

```

19 public function index()
20 {
21     if (Auth::guard('admin')->check()) {
22         $no = 1;
23         $data = DB::select("select * from penyewas inner join transaksis on penyewas.id = transaksis.id_penyewa
24             inner join kendaraans on transaksis.id_kendaraan = kendaraans.id inner join pembayarans on transaksis.id =
25             pembayarans.id_transaksi inner join pencairans on pembayarans.id = pencairans.id_pembayaran inner join
26             penyedias on kendaraans.id_penyedia = penyedias.id order by pembayarans.status asc limit 25 ");
27         return view('admin.pencairan', ['data' => $data, 'no' => $no]);
28     }
29     elseif (Auth::guard('penyedia')->check()) {
30         $slider = DB::select('SELECT * from sliders');
31         $no = 1;
32         $data = DB::select("select *, pencairans.id as id_pencairan from penyewas inner join transaksis on penyewas.id =
33             transaksis.id_penyewa inner join kendaraans on transaksis.id_kendaraan = kendaraans.id inner join pembayarans on
34             transaksis.id = pembayarans.id_transaksi inner join pencairans on pembayarans.id = pencairans.id_pembayaran
35             inner join penyedias on kendaraans.id_penyedia = penyedias.id where penyedias.id = " . Auth::guard('penyedia')
36             |>user()->id . " order by transaksis.tanggal_transaksi desc limit 25");
37         return view('penyedia.pencairan', ['data' => $data, 'no' => $no, 'slider' => $slider]);
38     }
39     else {
40         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
41     }
42 }
43 }

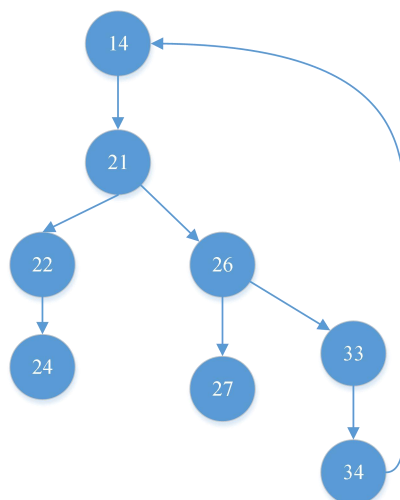
```

Sumber : Peneliti

**Gambar 5.121** Source Code Fungsi *Index* Pada *Controller* Pencairan

Pengubahan *source code* menjadi *flowgraph* pada fungsi *index* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.122.





Sumber : Peneliti

**Gambar 5.122** *Flowgraph* Fungsi *Index* Pada *Controller* Pencairan

Berdasarkan *flowgraph* pada fungsi *index* suspect total 11 *statement*, 9 *node* (N), dan 8 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 8 - 9 + 2 = 1 + 2 = 3$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 3. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *index* berdasarkan perolehan CC yaitu ada 3 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-011 = 19, 21, 22, 24
- 2) Jalur P-012 = 19, 26, 27, 30
- 3) Jalur P-013 = 19, 33, 34, 19

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 3 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *index* dapat dilihat pada gambar 5.123.

```

24 /
25 public function test_succrssfull_response(): void
26 {
27     $response = $this->get('/admin/home');
28
29     $response->assertStatus(200);
30 }

```

Sumber : Peneliti

**Gambar 5.123** *Source Code* Pengujian *Test* Fungsi *Index* Pada *Controller* Pencairan

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.124.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.124** Hasil Pengujian *Test* Fungsi *Index* Pada *Controller* Pencairan

#### b. *Print*

Fungsi *print* terdapat terdapat 4 *stetment*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.125.

```

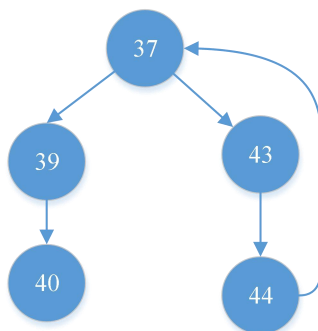
43 }
44 public function print(Pencairan $pencairan, Transaksi $transaksi, Penyewa $penyewa, Penyedia $penyedia, Kendaraan $kendaraan ,
45 {
46     if (Auth::guard('penyedia')->check()) {
47         return view('penyedia.print', ['pencairan' => $pencairan, 'transaksi' => $transaksi, 'penyewa' => $penyewa, 'penyedia'
48
49
50     } else {
51         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
52     }
53 }

```

Sumber : Peneliti

**Gambar 5.125** *Source Code* Fungsi *Print* Pada *Controller* Pencairan

Pengubahan *source code* menjadi *flowgraph* pada fungsi *print* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.126.



Sumber : Peneliti

**Gambar 5.126** *Flowgraph* Fungsi *Print* Pada *Controller* Pencairan

Berdasarkan *flowgraph* pada fungsi *index suspect* total 4 *statement*, 5 *node* (N), dan 5 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 5 - 5 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *print* berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-021 = 37, 39, 40
- 2) Jalur P-022 = 37, 43, 44

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *print* dapat dilihat pada gambar 5.127.

```

31
32 public function testPrint()
33 {
34     $pencairan=Pencairan::find(1);
35
36     $this->assertEquals('2023-08-17',$pencairan->tanggal_pencairan);
37 }
38

```

Sumber : Peneliti

**Gambar 5.127** Source Code Pengujian Test Fungsi Print Pada Controller Pencairan

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.128.

```

● PS C:\xampp\htdocs\nabila_new> php artisan test --filter NabilaTest
PASS Tests\Feature\NabilaTest
✓ print
Tests: 1 passed (1 assertions)
Duration: 0.28s

```

Sumber : Peneliti

**Gambar 5.128** Hasil Pengujian Test Fungsi print Pada Controller Pencairan

## 6. Pengujian UC-06

Pengujian pada UC-06 dengan nama *controller* pengguna, terdapat fungsi penyedia, penyewa, penyedia\_edit, penyewa\_edit, penyedia\_hapus, dan penyewa\_hapus. Adapun pengujian yang dilakukan adalah sebagai berikut.

### a. Penyedia

Fungsi penyedia terdapat terdapat 6 *statement*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.129.

```

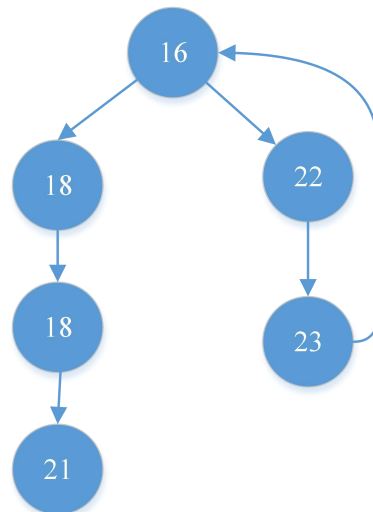
16 public function penyedia()
17 {
18     if (Auth::guard('admin')->check()) {
19         $no = 1;
20         $data = DB::select('select * from penyedias');
21         return view('admin.penyedia', ['data' => $data, 'no' => $no]);
22     } else {
23         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
24     }
25 }

```

Sumber : Peneliti

**Gambar 5.129** Source Code Fungsi Penyedia Pada Controller Pengguna

Pengubahan *source code* menjadi *flowgraph* pada fungsi penyedia untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.130.



Sumber : Peneliti

**Gambar 5.130** Flowgraph Fungsi Penyedia Pada Controller Pengguna

Berdasarkan *flowgraph* pada fungsi penyedia suspect total 6 *statement*, 6 *node* (N), dan 6 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 6 - 6 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan

ditentukan jalur dari fungsi penyedia berdasarkan perolehan CC yaitu ada 2 jalur.

Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-011 = 16, 18, 19, 21
- 2) Jalur P-012 = 16, 22, 23, 16

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi penyedia dapat dilihat pada gambar 5.131.

```

24
25     public function test_succrssl_response(): void
26     {
27         $response = $this->get('/admin/home');
28
29         $response->assertStatus(200);
30     }

```

Sumber : Peneliti

**Gambar 5.131** Source Code Pengujian Test Fungsi Penyedia Pada Controller Pengguna

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.132.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssl response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.132** Hasil Pengujian Test Fungsi Penyedia Pada Controller Pengguna

b. Penyewa

Fungsi penyewa terdapat terdapat 6 *statement*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.133.

```

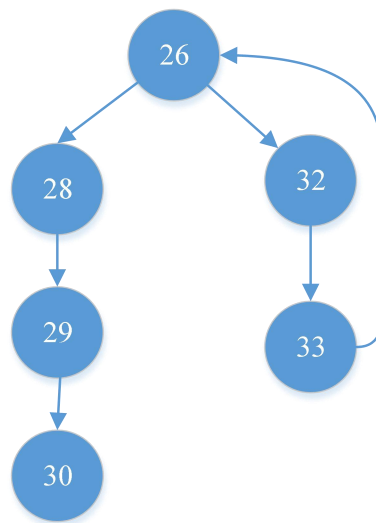
16 public function penyedia()
17 {
18     if (Auth::guard('admin')->check()) {
19         $no = 1;
20         $data = DB::select('select * from penyedia');
21         return view('admin.penyedia', ['data' => $data, 'no' => $no]);
22     } else {
23         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
24     }
25 }

```

Sumber : Peneliti

**Gambar 5.133** *Source Code* Fungsi Penyewa Pada *Controller* Pengguna

Pengubahan *source code* menjadi *flowgraph* pada fungsi penyewa untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.134.



Sumber : Peneliti

**Gambar 5.134** *Flowgraph* Fungsi Penyewa Pada *Controller* Pengguna

Berdasarkan *flowgraph* pada fungsi penyewa suspect total 6 *statement*, 6 *node* (N), dan 6 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 6 - 6 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi penyewa berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-021 = 26, 28, 29, 30
- 2) Jalur P-022 = 26, 32, 33, 26

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test -filter*. Adapun gambar *source code* pengujian pada fungsi penyewa dapat dilihat pada gambar 5.135.

```

24
25     public function test_succrssfull_response(): void
26     {
27         $response = $this->get('/admin/home');
28
29         $response->assertStatus(200);
30     }

```

Sumber : Peneliti

**Gambar 5.135** *Source Code* Pengujian *Test* Fungsi Penyewa Pada *Controller* Pengguna

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.136



```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.136** Hasil Pengujian *Test* Fungsi Penyewa Pada *Controller* Pengguna

c. Penyedia\_Edit

Fungsi penyedia\_edit terdapat terdapat 4 *stetment*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.137.

```

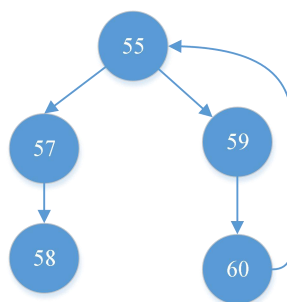
55 public function penyedia_edit(Penyedia $penyedia)
56 {
57     if (Auth::guard('admin')->check()) {
58         return view('admin.ubah_penyedia', ['penyedia' => $penyedia]);
59     } else {
60         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
61     }
62 }

```

Sumber : Peneliti

**Gambar 5.137** *Source Code* Fungsi Penyedia\_Edit Pada *Controller* Pengguna

Pengubahan *source code* menjadi *flowgraph* pada fungsi penyedia\_edit untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.138.



Sumber : Peneliti

**Gambar 5.138** *Flowgraph* Fungsi Penyedia\_Edit Pada *Controller* Pengguna

Berdasarkan *flowgraph* pada fungsi penyedia\_edit suspect total 4 *stetment*, 5 *node* (N), dan 5 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 5 - 5 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *penyedia\_edit* berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-031 = 55, 57, 58
- 2) Jalur P-032 = 55, 59, 60

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *penyedia\_edit* dapat dilihat pada gambar 5.139.

```

39     public function testUpdadte()
40     {
41         $penyedia=Penyedia::find(1);
42         $penyedia->nama='CV. Palembang Berlian Abadi Jaya';
43         $this->assertTrue($penyedia->Update());
44     }
45
46

```

Sumber : Peneliti

**Gambar 5.139** *Source Code* Pengujian *Test* Fungsi *Penyedia\_Edit* Pada *Controller* Pengguna

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.140

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.140** Hasil Pengujian *Test* Fungsi Penyedia\_*Edit* Pada *Controller* Pengguna

d. Penyedia\_Hapus

Fungsi penyedia\_hapus terdapat terdapat 4 *stetment*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.141.

```

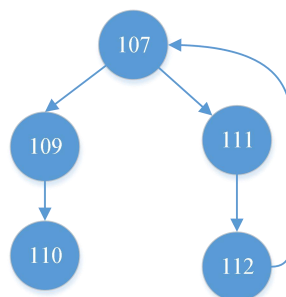
100 public function penyedia_hapus(Penyedia $penyedia)
101 {
102     File::delete("uploads/logo_penyedia/" . $penyedia->logo_penyedia);
103     File::delete("uploads/akte_pendirian/" . $penyedia->akte_pendirian);
104     $penyedia->delete();
105     return redirect(Route('pengguna.penyedia'))->with('pesan', 'Penyedia berhasil di hapus');
106 }

```

Sumber : Peneliti

**Gambar 5.141** *Source Code* Fungsi Penyedia\_*Hapus* Pada *Controller* Pengguna

Pengubahan *source code* menjadi *flowgraph* pada fungsi penyedia\_hapus untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.142.



Sumber : Peneliti

**Gambar 5.142** *Flowgraph* Fungsi Penyedia\_*Hapus* Pada *Controller* Pengguna

Berdasarkan *flowgraph* pada fungsi penyedia\_hapus suspect total 4 *stetment*, 3 *node* (N), dan 2 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = -1 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi penyedia\_hapus berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-041 = 100, 102, 105

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada penyedia\_hapus dapat dilihat pada gambar 5.143.

```

46
47     public function testDelete()
48     {
49         $penyedia=Penyedia::find(1);
50
51         $this->assertDelete($penyedia->delete());
52
53     }

```

Sumber : Peneliti

**Gambar 5.143** *Source Code* Pengujian *Test* Fungsi Penyedia\_Hapus Pada *Controller* Pengguna

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.144

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.144** Hasil Pengujian *Test* Fungsi Penyedia\_Hapus Pada *Controller* Pengguna

e. Penyewa\_Edit

Fungsi *penyewa\_edit* terdapat terdapat 4 *stetment*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.145.

```

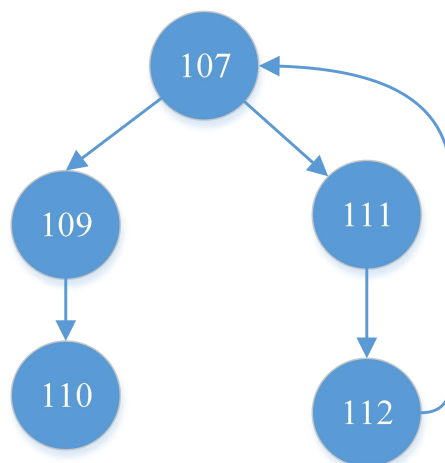
107 public function penyewa_edit(Penyewa $penyewa)
108 {
109     if (Auth::guard('admin')->check()) {
110         return view('admin.ubah_penyewa', ['penyewa' => $penyewa]);
111     } else {
112         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
113     }
114 }

```

Sumber : Peneliti

**Gambar 5.145** *Source Code* Fungsi Penyewa\_Edit Pada *Controller* Pengguna

Pengubahan *source code* menjadi *flowgraph* pada fungsi *penyewa\_edit* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.146.



Sumber : Peneliti

**Gambar 5.146** *Flowgraph* Fungsi Penyewa\_Edit Pada *Controller* Pengguna

Berdasarkan *flowgraph* pada fungsi `penyewa_edit` suspect total 4 *statement*, 5 *node* (N), dan 5 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 5 - 5 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi `penyewa_edit` berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-051 = 107, 109, 110
- 2) Jalur P-052 = 107, 111, 112

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan `php artisan test --filter`. Adapun gambar *source code* pengujian pada fungsi `penyewa_edit` dapat dilihat pada gambar 5.147.

```

54     public function testUpdadte()
55     {
56         $penyewa=Penyewa::find(1);
57         $penyewa->nama_penyewa='Nabila Elfa Rieny';
58         $this->assertTrue($penyewa->Update());
59     }
60
61

```

Sumber : Peneliti

**Gambar 5.147** *Source Code* Pengujian *Test* Fungsi `Penyewa_Edit` Pada *Controller* Pengguna

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat,

maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.148.

```
PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s
```

Sumber : Peneliti

**Gambar 5.148** Hasil Pengujian *Test* Fungsi *Penyewa\_Edit* Pada *Controller* Pengguna

f. *Penyewa\_Hapus*

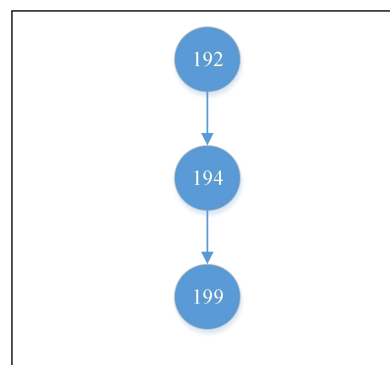
Fungsi *penyewa\_hapus* terdapat terdapat 6 *stetment*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.149.

```
192 public function penyewa_hapus(Penyewa $penyewa)
193 {
194     File::delete("uploads/ktp/" . $penyewa->ktp);
195     File::delete("uploads/kk/" . $penyewa->kk);
196     File::delete("uploads/sim_a/" . $penyewa->sim_a);
197     File::delete("uploads/foto_selfie/" . $penyewa->foto_selfie);
198     $penyewa->delete();
199     return redirect(Route('pengguna.penyewa'))->with('pesan', 'Penyewa berhasil di hapus');
200 }
```

Sumber : Peneliti

**Gambar 5.149** *Source Code* Fungsi *Penyewa\_Hapus* Pada *Controller* Pengguna

Pengubahan *source code* menjadi *flowgraph* pada fungsi *penyewa\_hapus* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.150.



Sumber : Peneliti

**Gambar 5.150** *Flowgraph* Fungsi *Penyewa\_Hapus* Pada *Controller* Pengguna

Berdasarkan *flowgraph* pada fungsi `penyewa_hapus` suspect total 6 *statement*, 3 *node* (N), dan 2 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = -1 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi `penyewa_hapus` berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-61 = 192, 194, 199

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan `php artisan test --filter`. Adapun gambar *source code* pengujian pada `penyewa_hapus` dapat dilihat pada gambar 5.151.

```

62     public function testDelete()
63     {
64         $penyewa=Penyewa::find(1);
65
66         $this->assertDelete($penyewa->delete());
67     }
68

```

Sumber : Peneliti

**Gambar 5.151** *Source Code* Pengujian *Test* Fungsi `Penyewa_Hapus` Pada *Controller* Pengguna

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.152.



```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest
PASS Tests\Feature\AdminTest
✓ successful response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```

Sumber : Peneliti

**Gambar 5.152** Hasil Pengujian *Test* Fungsi Penyewa\_Hapus Pada *Controller* Pengguna

## 7. Pengujian UC-07

Pengujian pada UC-07 dengan nama *controller* slider, terdapat fungsi *index*, *edit* dan *hapus*. Adapun pengujian yang dilakukan adalah sebagai berikut.

### a. *Index*

Fungsi *index* terdapat terdapat 6 *statement*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.153.

```

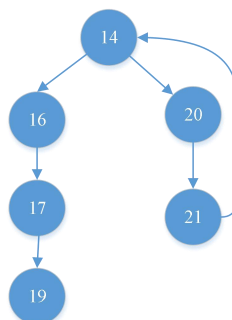
14 public function index()
15 {
16     if (Auth::guard('admin')->check()) {
17         $no = 1;
18         $data = DB::select('SELECT * from sliders');
19         return view('admin.slider', ['data' => $data, 'no' => $no]);
20     } else {
21         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
22     }
23 }

```

Sumber : Peneliti

**Gambar 5.153** *Source Code* Fungsi *Index* Pada *Controller* Slider

Pengubahan *source code* menjadi *flowgraph* pada fungsi *index* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.154.



Sumber : Peneliti

**Gambar 5.154** *Flowgraph* Fungsi *Index* Pada *Controller* Slider

Berdasarkan *flowgraph* pada fungsi *index suspect* total 6 *statement*, 6 *node* (N), dan 6 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 6 - 6 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *index* berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-011 = 14, 16, 17, 19
- 2) Jalur P-012 = 14, 29, 21, 14

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test -filter*. Adapun gambar *source code* pengujian pada fungsi *index* dapat dilihat pada gambar 5.155.

```

54 public function test_sucessful_response():
55     {
56         $response=$this->get('admin/slider');
57
58         $response->assertStatus(200);
59     }

```

Sumber : Peneliti

**Gambar 5.155** *Source Code* Pengujian *Test Fungsi Index* Pada *Controller Slider*

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat,

maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.156

```
PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s
```

Sumber : Peneliti

**Gambar 5.156** Hasil Pengujian *Test* Fungsi *Index* Pada *Controller* Slider

b. *Edit*

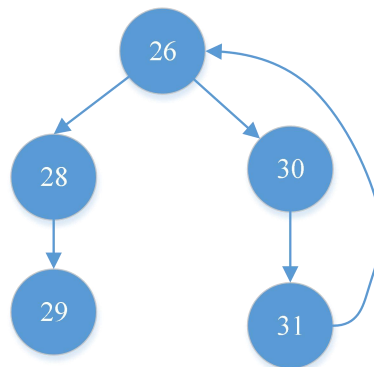
Fungsi *edit* terdapat terdapat 4 *stetment*, dan 2 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.157.

```
26 public function edit(Slider $slider)
27 {
28     if (Auth::guard('admin')->check()) {
29         return view('admin.ubah_slider', ['slider' => $slider]);
30     } else {
31         return redirect('/login')->with('pesan', 'Anda Tidak Memiliki Akses');
32     }
33 }
34
```

Sumber : Peneliti

**Gambar 5.157** *Source Code* Fungsi *Edit* Pada *Controller* Slider

Pengubahan *source code* menjadi *flowgraph* pada fungsi *edit* untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.158.



Sumber : Peneliti

**Gambar 5.158** *Flowgraph* Fungsi *Edit* Pada *Controller* Slider

Berdasarkan *flowgraph* pada fungsi *edit suspect* total 4 *statement*, 5 *node* (N), dan 5 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 5 - 5 + 2 = 0 + 2 = 2$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 2. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi *edit* berdasarkan perolehan CC yaitu ada 2 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-021 = 26, 28, 29
- 2) Jalur P-022 = 26, 30, 31

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 2 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada fungsi *edit* dapat dilihat pada gambar 5.159.

```

39     public function testUpdadte()
40     {
41         $slider=Slider::find(5);
42         $Slider->foto='slider7.png';
43         $this->assertTrue($slider->Update());
44     }
45

```

Sumber : Peneliti

#### **Gambar 5.159** *Source Code* Pengujian *Test Fungsi Edit* Pada *Controller Slider*

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat,

maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.160.

```
PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest

PASS Tests\Feature\AdminTest
✓ succrssfull response

Tests: 1 passed (1 assertions)
Duration: 1.17s
```

Sumber : Peneliti

**Gambar 5.160** Hasil Pengujian *Test Fungsi Edit Pada Controller Slider*

c. Hapus

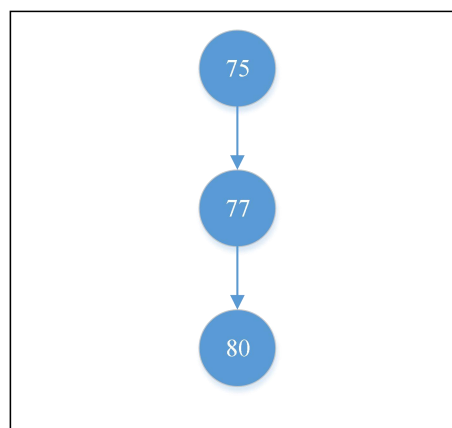
Fungsi hapus terdapat terdapat 3 *stetment*, dan 0 kondisi. Adapun *source code* nya dapat dilihat pada gambar 5.160.

```
75 public function hapus(Slider $slider)
76 {
77     File::delete("uploads/foto/" . $slider->foto);
78     $slider->delete();
79
80     return redirect(Route('slider.index'))->with('pesan', 'slider berhasil di hapus');
81 }
82
```

Sumber : Peneliti

**Gambar 5.160** *Source Code* Fungsi Hapus Pada *Controller Slider*

Pengubahan *source code* menjadi *flowgraph* pada fungsi hapus untuk mencari jalur yang dapat dilewati dapat dilihat pada gambar 5.161.



Sumber : Peneliti

**Gambar 5.161** *Flowgraph* Fungsi Hapus Pada *Controller Slider*

Berdasarkan *flowgraph* pada fungsi hapus suspect total 3 *statement*, 3 *node* (N), dan 2 *edge* (E). Dari *flowgraph* diatas maka dapat dihitung nilai *cyclomatic complexiy* sebagai berikut :

$$V(G) = E - N + 2 = 2 - 3 + 2 = -1 + 2 = 1$$

Jadi, *cyclomatic complexiy* yang diperoleh berdasarkan *flowgraph* pada gambar diatas adalah 1. Berdasarkan perhitungan *cyclomatic complexiy* tersebut, maka akan ditentukan jalur dari fungsi hapus berdasarkan perolehan CC yaitu ada 1 jalur. Adapun identifikasi jalur yang mungkin untuk dilakukan uji coba antara lain :

- 1) Jalur P-031 = 75, 77, 80

Data uji yang dilakukan dalam pengujian berdasarkan jalur yang ditentukan dan diukur menggunakan *cyclomatic complexiy*, dimana terdapat 1 jalur yang dapat dilewati kemudian dapat diberi masukan berupa data input dan keluaran yang diharapkan.

Pengujian ini dilakukan dengan *php artisan test --filter*. Adapun gambar *source code* pengujian pada hapus dapat dilihat pada gambar 5.162.

```

74
75     public function hapus(Slider $slider)
76     {
77         File::delete("uploads/foto/" . $slider->foto);
78         $slider->delete();
79
80         return redirect(Route('slider.index'))->with('pesan', 'Slider berhasil di hapus');
81
82     }
83

```

Sumber : Peneliti

**Gambar 5.162** *Source Code* Pengujian *Test* Fungsi Hapus Pada *Controller* Slider

Berdasarkan tahapan yang telah dilakukan, diperoleh 1 jalur dan pengujian berhasil. Untuk mengetahui tingkatan keberhasilan dari program yang telah dibuat, maka dibuatlah sebuah kasus uji. Adapun hasil pengujian testing dapat dilihat pada gambar 5.163.

```

PS C:\xampp\htdocs\nabila_new> php artisan test --filter AdminTest
PASS Tests\Feature\AdminTest
✓ successful response

Tests: 1 passed (1 assertions)
Duration: 1.17s

```


Sumber : Peneliti





**Gambar 5.163** Hasil Pengujian *Test* Fungsi Hapus Pada *Controller* Slider

### 5.1.3.5. Pengenalan E-Marketplace Penyewaan Mobil Riemera

Peneliti melakukan pengenalan *e-marketplace* penyewaan mobil Riemera kepada penyedia jasa penyewaan mobil untuk memberikan arahan bagaimana cara penggunaan sistem, memberikan informasi mengenai menu yang tersedia, kelebihan, keuntungan dan perbedaan pada sistem yang saat ini digunakan. Pada tahap ini juga penyedia jasa penyewaan mobil memberikan tanggapan akan sistem yang sudah dibuat. Adapun dokumentasi yang peneliti lakukan dapat dilihat pada tabel 5.11.

**Tabel 5.11** Dokumentasi Pengenalan E-Marketplace Penyewaan Mobil Riemera

No	Nama Penyedia	Dokumentasi
1	PT. Sekayu Serasan Sekate	

No	Nama Penyedia	Dokumentasi
2	CV. Palembang Berlian Abadi Jaya	
3	CV. Sriwijaya XI	
4	CV. Sriwijaya Maju Gemilang	
5	CV. Anugrah Esa Utama	

Sumber : Peneliti



## BAB VI

### KESIMPULAN DAN SARAN

#### 6.1. Kesimpulan

Berdasarkan penelitian yang telah peneliti lakukan, dapat disimpulkan bahwa telah dihasilkan *e-marketplace* penyewaan mobil yang telah peneliti beri nama Rimera dimana *e-marketplace* ini bisa diakses oleh penyedia, penyewa, dan admin. Berdasarkan hasil pengujian dengan metode *white box* menggunakan metode *statement coverage* yang dilakukan oleh penguji *e-marketplace* penyewaan mobil rimera dapat digunakan dengan baik oleh pengguna dan memberikan pengalaman yang memuaskan dalam memilih, mencari, dan menyewa mobil sesuai dengan kebutuhan penyewa.

#### 6.2. Saran

Saran yang diusulkan peneliti untuk proses pengembangan lebih lanjut dari *e-marketplace* penyewaan mobil rimera yang telah dibangun ditambahkan fitur sebagai berikut :

1. Fitur *chat* yang dapat mempermudah komunikasi antara penyedia, penyewa, dan admin.
2. Pembayaran secara *realtime* menggunakan API yang dapat mempermudah proses pembayaran tepat waktu dan kelancaran bisnis.
3. Hendaknya dibuat berbasis android sehingga dapat mempermudah penyewa melakukan transaksi penyewaan mobil melalui *handphone*.
4. Serta dilakukan pemeliharaan agar dapat berfungsi dengan baik.

## DAFTAR PUSTAKA

- Aditama, Roki. 2017. *Web Service Pembayaran Uang Kuliah Online dengan PHP dan SOAP WSDL*. CV Lokomedia: Yogyakarta.
- Al Afghani, Dhian Yusuf. 2022. *Perancangan Sistem Informasi Marketplace dan Reseller pada PT. XYZ dengan Metode RAD (Rapid Application Development)*. ISMETEK 14.2.
- Alfiah, Alfiah, and Damayanti Damayanti. 2020. *Aplikasi E-Marketplace Penjualan Hasil Panen Ikan Lele (Studi Kasus: Kabupaten Pringsewu Kecamatan Pagelaran)*. Jurnal Teknologi Dan Sistem Informasi 1.1 111-117.
- Aswati, S., Ramadhan, M. S., Firmansyah, A., & Anwar, K. 2017. *Studi Analisis Model Rapid Application Development dalam Pengembangan Sistem Informasi*. Matrik: Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer, 16(2), 20-27.
- Christiana, Sandra. 2022 *Sistem Informasi Penyewaan Lapangan Bulu Tangkis Berbasis Web Pada Gor Pandiga Jati Cimahi*. Diss. Universitas Komputer Indonesia.
- Connolly, Thomas dan Carolyn Begg. 2015. *Database Systems : A Practical Approach to Design, implementation and management. Sixth Edition*. Boston : Pearson Education.
- Dewi, K. I., Doni Syofiawan, and Titania Ulfa Grace. 2018. *Pembangunan Dashboard Sebagai Alat Monitoring Dan Evaluasi Pada Toko Permata Batam*. Jurnal Responsive 2.2.

- Enterprise, Jubilee. 2014. *MySQL untuk pemula*. Elex Media Komputindo.
- Ernanda, Dimas. 2017. *Pengaruh Store Atmosphere, Hedonic Motive Dan Service Quality Terhadap Keputusan Pembelian*. Jurnal Ilmu dan Riset Manajemen, Vol.6 No.10. ISSN: 2461-0593.
- Fahrisal, Fahrisal, Sentosa Pohan, and Marnis Nasution. 2018. *Perancangan Sistem Inventory Barang Pada Ud.Minang Dewi Berbasis Website*. Informatika, 6(2) 17-23.
- Fitriansyah, Ahmad, and Ibnu Harris. 2018. *Pengukuran kepuasan pengguna situs web dengan metode End User Computing Satisfaction (EUCS)*. Query : Journal of Information Systems, 2(1).
- Gunadi, David Christian Aditya, and Teddy Marcus Zakaria. 2018. *Perancangan E-Marketplace Sewa-Menyewa sebagai Alternatif Bisnis di Era Digital*. Jurnal Teknik Informatika dan Sistem Informasi 4. 281-294.
- Imah, Milla Tunna, and Budi Purwoko. 2018. *Studi Kepustakaan Penerapan Konseling Neuro Linguistic Programming (NLP) dalam Lingkup Pendidikan*. Diss. State University of Surabaya.
- Lenti, Febri Nova. 2017. *Rekayasa proses bisnis pada e-commerce b2b-b2c menggunakan sistem afiliasi*. JIKO (Jurnal Informatika dan Komputer) 2.1.
- Massie, Juan V., Tommy, Parengkuan., dkk. 2017. *Analisis Finansial Terhadap Nilai Perusahaan (Studi Pada Perusahaan perusahaan Consumer Goods Sub Sektor Makanan Dan Minuman Yang Terdaftar Di Bei Tahun 2011 - 2016)*. Jurnal EMBA, Vol.5 No.3. ISSN: 2303-1174.

- Maulana, Djodi, Rahmat Tullah, and Siti Maisaroh. 2022. *Application Helpdesk User Friendly IT Solution berbasis Android*. JURNAL TOPIK GLOBAL 1.1.
- Melani, Yayuk Ike. 2020. *Pengembangan Sistem Informasi Monitoring Sarana Dan Prasarana Dan Penerapannya Untuk Kegiatan Belajar Mengajar Pada Perguruan Tinggi Swasta*. Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer 11.2 672-680.
- Pradana, Mahir. 2015. *Klasifikasi jenis-jenis bisnis e-commerce di Indonesia*. Neo-Bis 9.2 32-40.
- Priyaungga, Bayu Aji, et al. 2020. *Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions*. Jurnal Teknologi Sistem Informasi dan Aplikasi ISSN 2654 3788.
- Putra, Darma Setiawan, and Ami Fauzijah. 2018. *Perancangan Aplikasi Presensi Dosen Realtime Dengan Metode Rapid Application Development (RAD) Menggunakan Fingerprint Berbasis Web*. Jurnal Informatika : Jurnal Pengembangan IT 3.2 167-171.
- Rachmah, Ucti Nur, and Chandra Kesuma. 2021. *Sistem Informasi E-Marketplace Penyedia Jasa Sewa Dekorasi Berbasis Website*. Indonesian Journal of Networking and Security (IJNS) 10.2.
- Ramadhan, Surya, Sumitro Sarkum, and Iwan Purnama. 2019. *Sistem Informasi Penilaian Kinerja Pegawai Berbasis Web Pada Operasi Perangkat Daerah Kantor Camat Rantau Utara Labuhanbatu*. Jurnal Teknik Komputer 5(1), 93-96.

- Riyanto, Verry. 2017. *Implementasi Metode Rapid Application Development Dalam Membangun E-Commerce Di Bidang UKM*. Jurnal Pilar Nusa Mandiri 13.1 122-127.
- Rosalyn, Rizqi, and Andy Prasetyo. 2019. *Pengertian Flowchart Beserta Fungsi dan Simbol-simbol Flowchart yang Paling Umum Digunakan*.
- S. Supriyanto, I. Fitri, and N. Nurhayati. 2022. *Aplikasi Inventory Peralatan Mekanik Unit BRT UNAS Berbasis Web Menggunakan Metode Black-Box dan White-Box Testing*. Jurnal Teknol. Inf. dan Komunikasi 224–233.
- Wahyuningrum, Tenia, and Dwi Januarita. 2014. *Perancangan web e-commerce dengan metode Rapid Application Development (RAD) untuk produk unggulan desa*. Semantik 4.1.
- Yuliati, Eka, and Dian Ratanasari Yahya. 2018. *Laporan Pengabdian Kepada Masyarakat : Penanaman 1000 Pohon di Taman Purbakala Kerajaan Sriwijaya (TPKS) Palembang*.
- Yunisa, R. 2018. “Perbandingan 2 teknik white box testing (Studi kasus : sistem informasi reporting community TB-HIV ‘Aisyah Tanggamus)’”.
- Yunisa, Rahmaria, and Hanson Prihantoro Putro. 2020. *Perbandingan 2 Teknik White Box Testing: Statement Coverage dan Branch Coverage Testing (Studi Kasus: Sistem Informasi Reporting Community TB-HIV Care)*. Annual Research Seminar (ARS). Vol. 5. No. 1.
- Yunisa, Rahmaria, and Hanson Prihantoro Putro. 2020. *Perbandingan 2 Teknik White Box Testing: Statement Coverage dan Branch Coverage Testing (Studi Kasus: Sistem Informasi Reporting Community TB-HIV Care)*. Annual Research Seminar (ARS). Vol. 5. No. 1.

Yustiani, Rini, and Rio Yunanto. 2017. *Peran Marketplace Sebagai Alternatif  
Bisnis Di Era Teknologi Informasi*. Komputa: Jurnal Ilmiah Komputer Dan  
Informatika 6.2 43-48.