

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **VI.1. Kesimpulan**

Setelah aplikasi CRS ini selesai diuji maka dapat ditarik beberapa kesimpulan, yaitu:

1. Perangkat lunak CRS telah berhasil dikembangkan sebagai sistem reservasi antrian service motor berbasis web dan SMS, sehingga pelanggan tidak perlu mengantri terlalu lama ketika menunggu giliran service di Bengkel Chrisna Salatiga.
2. Perangkat lunak CRS telah berhasil diuji sebagai aplikasi yang dapat membantu pelanggan dalam melakukan pemesanan ataupun pembatalan antrian service, serta mempermudah pelanggan dalam memberikan feedback untuk bengkel tanpa membedakan jenis telepon selular yang mereka miliki dengan memanfaatkan layanan SMS Gateway.
3. Perangkat lunak CRS telah berhasil diuji sebagai aplikasi yang dapat membantu administrator dan operator dalam mengelola data-data yang berkaitan dengan reservasi antrian service.
4. Perangkat Lunak CRS telah berhasil diuji sebagai aplikasi yang dapat membantu pemilik dalam melakukan monitoring laporan serta memberikan balasan pada feedback yang telah diberikan oleh pelanggan dimanapun dan kapanpun dia berada.

## VI.2. Saran

Penulis ingin memberikan beberapa saran untuk pengembangan lebih lanjut perangkat lunak CRS ini:

1. Sistem dapat dikembangkan agar dapat menangani proses transaksi atau pembayaran.
2. Sistem dapat dikembangkan agar dapat digunakan oleh pihak bengkel untuk dapat melakukan pengecekan pulsa dan melihat masa aktif berlaku kartu pada nomor GSM yang digunakan untuk modem SMS Gateway.
3. Sistem dapat dikembangkan agar dapat memberikan pemberitahuan kepada pelanggan secara otomatis, ketika kendaraan yang dimiliki pelanggan sudah waktunya untuk melakukan service.
4. Sistem dapat dikembangkan agar dapat memberikan layanan berbasis lokasi yang dapat dimanfaatkan di dealer resmi, sehingga mempermudah pelanggan yang sedang melakukan touring, untuk dapat mencari dealer resmi terdekat.

**DAFTAR PUSTAKA**

- Andriadi, Juhan, 2007. *Analisis dan Perancangan Sistem Pembangunan Mobile Web Pencarian Alamat Penting di Kota Semarang*. Yogyakarta : STMIK Amikom.
- Darmadi, Raden Frederikus Maria Adikusuma, 2009. *Pembangunan Sistem Informasi Penjualan Suplemen Binaraga Berbasis Mobile Commerce*. Yogyakarta : Universitas Atma Jaya Yogyakarta.
- Hartanto, Yustinus Tri, 2010. *Pembangunan Sistem Informasi Rental Film Pada Perangkat Mobile Berbasis J2ME*. Yogyakarta : Universitas Atma Jaya Yogyakarta
- Hartono, Jogiyanto, 2005. *Analisis & Desain: Sistem Informasi*. Yogyakarta : Penerbit Andi.
- Haryadi, Yudi, 2011. Alur Kerja SMS Gateway. <http://www.scribd.com/doc/54119280/Alur-Kerja-SMS-Gateway>, diakses pada tanggal 24 September 2011.
- Kadir, Abdul, 2003. *Dasar Pemrograman Java 2*. Yogyakarta : Penerbit Andi.
- Laura, Lemay, 2001. *Desain Grafik dan Halaman Web*. Bandung : Informatika.
- Nurhadi, T., 2003. *Pemrograman WML dan WMLS : Hadirkan Diri Anda di Mobile Internet*. Yogyakarta : Penerbit Andi.
- Prasetyo, Herlambang, 2009, *Codeigniter Lightweight PHP Framework*. <http://herlambangprasetyo.net/2009/02/codeigniter-lightweight-php-framework>, diakses pada tanggal 15 Oktober 2010.

- Sidhi, Thomas Adi Purnomo, 2009. *Pembangunan Sistem Informasi Pendaftaran Rawat Jalan Berbasis WAP*. Yogyakarta : Universitas Atma Jaya Yogyakarta.
- Rizky, Dhanta, 2009. *Kamus Istilah Komputer Grafis dan Internet*. Surabaya : Indah.
- Rosidi, Romzi Imron, 2004. *Membuat Sendiri SMS Gateway Berbasis Protokol SMPP*. Yogyakarta : Penerbit Andi.
- Saputra, Agus, 2011. *Step by Step Membangun Aplikasi SMS dengan php dan MySQL*. Jakarta : PT Elex Media Komputindo.
- Suhendar, A., 2003. *Teknologi Pemrograman Mobile Commerce*. Bandung : Informatika.
- Supono, 2010. *CodeIgniter Framework PHP*. [supono.wordpress.com/2010/04/16/codeigniter-framework-php/](http://supono.wordpress.com/2010/04/16/codeigniter-framework-php/), diakses pada tanggal 10 November 2010.
- Suyadnya, I Made Arsa, 2008. *Pembangunan Perangkat Lunak Penyewaan VCD/DVD Dengan Layanan Booking Film Terbaru Berbasis SMS Gateway*. Yogyakarta : Universitas Atma Jaya Yogyakarta.
- Upton, David, 2007. *CodeIgniter for Rapid PHP Application Development*. Indianapolis : Wiley Publishing, Inc.
- Whitten, Jeffery, 2004. *Metode Desain dan Analisis Sistem*. Yogyakarta : Penerbit Andi.

**SKPL**

**SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK**

**(CRS)**

**Chrisna Reservation System**

**Untuk :**


**Universitas Atma Jaya Yogyakarta**

**Dipersiapkan oleh:**

**Robby Widyahartono / 5217**

**Program Studi Teknik Informatika - Fakultas Teknologi Industri**

**Universitas Atma Jaya Yogyakarta**

|   |   |                      |  |                |
|---|---|----------------------|--|----------------|
|  | Program Studi Teknik Informatika<br>Fakultas Teknologi Industri | <b>Nomor Dokumen</b> |  | <b>Halaman</b> |
|   |   | <b>SKPL-CRS</b>      |  | 1/47           |
|   |   | <b>Revisi</b>        |  |                |

## DAFTAR PERUBAHAN

| Revisi   | Deskripsi |
|----------|-----------|
| <b>A</b> |           |
| <b>B</b> |           |
| <b>C</b> |           |
| <b>D</b> |           |
| <b>E</b> |           |
| <b>F</b> |           |

| INDEX<br>TGL       | - | A | B | C | D | E | F | G |
|--------------------|---|---|---|---|---|---|---|---|
| Ditulis<br>oleh    |   |   |   |   |   |   |   |   |
| Diperik<br>sa oleh |   |   |   |   |   |   |   |   |
| Disetuj<br>ui oleh |   |   |   |   |   |   |   |   |

### Daftar Halaman Perubahan

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

# Daftar Isi

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>PENDAHULUAN .....</b>   | <b>6</b>  |
| 1.1      | TUJUAN.....  | 6         |
| 1.2      | LINGKUP MASALAH.....   | 6         |
| 1.3      | DEFINISI, AKRONIM DAN SINGKATAN.....   | 7         |
| 1.4      | REFERENSI.....   | 9         |
| 1.5      | DESKRIPSI UMUM (OVERVIEW).....   | 9         |
| <b>2</b> | <b>DESKRIPSI KEBUTUHAN.....</b>  | <b>10</b> |
| 2.1      | PERSPEKTIF PRODUK.....   | 10        |
| 2.2      | FUNGSI PRODUK.....   | 12        |
| 2.2.1    | <i>Fungsi Produk Administrator.....</i>  | <i>12</i> |
| 2.2.2    | <i>Fungsi Produk Operator.....</i>   | <i>14</i> |
| 2.2.3    | <i>Fungsi Produk Pelanggan.....</i>  | <i>17</i> |
| 2.2.4    | <i>Fungsi Produk Pemilik.....</i>  | <i>18</i> |
| 2.3      | KARAKTERISTIK PENGGUNA.....  | 18        |
| 2.4      | BATASAN-BATASAN.....   | 19        |
| 2.5      | ASUMSI DAN KETERGANTUNGAN.....   | 19        |
| <b>3</b> | <b>KEBUTUHAN KHUSUS .....</b>  | <b>20</b> |
| 3.1      | KEBUTUHAN ANTARMUKA EKSTERNAL.....   | 20        |
| 3.1.1    | <i>Antarmuka pemakai.....</i>  | <i>20</i> |
| 3.1.2    | <i>Antarmuka perangkat keras.....</i>  | <i>20</i> |
| 3.1.3    | <i>Antarmuka perangkat lunak.....</i>  | <i>21</i> |
| 3.1.4    | <i>Antarmuka komunikasi.....</i>   | <i>22</i> |
| 3.2      | KEBUTUHAN FUNGSIONALITAS PERANGKAT LUNAK.....                                      | 23        |
| <b>4</b> | <b>SPESIFIKASI RINCI KEBUTUHAN.....</b>  | <b>24</b> |
| 4.1      | SPESIFIKASI KEBUTUHAN FUNGSIONALITAS .....   | 24        |
| 4.1.1    | <i>Use case Spesification : Login.....</i>   | <i>24</i> |
| 4.1.2    | <i>Use case Spesification : Mengelola User.....</i>                                | <i>25</i> |
| 4.1.3    | <i>Use case Spesification : Mengelola Teknisi.....</i>                             | <i>27</i> |
| 4.1.4    | <i>Use case Spesification : Mengelola Reservasi.....</i>                           | <i>30</i> |
| 4.1.5    | <i>Use case Spesification : Mengelola Pelanggan.....</i>                           | <i>32</i> |
| 4.1.6    | <i>Use case Spesification : Mengelola Kendaraan.....</i>                           | <i>35</i> |
| 4.1.7    | <i>Use case Spesification : Mengelola Tarif.....</i>                               | <i>37</i> |
| 4.1.8    | <i>Use case Spesification : Mengelola Onderdil.....</i>                            | <i>39</i> |
| 4.1.9    | <i>Use case Spesification : Mengubah Password.....</i>                             | <i>41</i> |
| 4.1.10   | <i>Use case Spesification : Mendaftar Antrian dan<br/>Memberikan Feedback.....</i> | <i>42</i> |
| 4.1.11   | <i>Use case Spesification : Melakukan Monitoring<br/>Laporan.....</i>              | <i>44</i> |
| 4.1.12   | <i>Use case Spesification : Menanggapi Feedback.....</i>                           | <i>46</i> |
| <b>5</b> | <b>ENTITY RELATIONSHIP DIAGRAM (ERD) .....</b>                                     | <b>47</b> |



## Daftar Gambar

|   |    |
|---|----|
| GAMBAR 1. ARSITEKTUR PERANGKAT LUNAK CRS..... | 12 |
| GAMBAR 2. USE CASE DIAGRAM.....               | 23 |
| GAMBAR 3. ENTITY RELATIONSHIP DIAGRAM.....    | 47 |



## 1 Pendahuluan

### 1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak CRS (*Chrisna Reservation System*) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) dan atribut (*fitur-fitur* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-CRS ini juga mendefinisikan batasan perancangan perangkat lunak.

Dokumen SKPL ini menjadi dasar kesepakatan antara pihak pelanggan dengan pihak pengembang mengenai perangkat lunak yang akan dikembangkan.

### 1.2 Lingkup Masalah

Perangkat Lunak CRS dikembangkan dengan tujuan untuk mempermudah pelanggan dalam mendaftar dan membatalkan antrian *service*, serta memberikan *feedback*, mempermudah administrator dan operator dalam mengelola semua data yang berkaitan dengan reservasi antrian *service*, dan juga mempermudah pemilik dalam melakukan monitoring laporan dan menanggapi *feedback* :

1. Mempermudah administrator dalam menangani pengelolaan user.
2. Mempermudah administrator dalam menangani pengelolaan teknisi.
3. Mempermudah operator dalam menangani pengelolaan reservasi.

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 6/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

4. Mempermudah operator dalam menangani pengelolaan pelanggan.
  5. Mempermudah operator dalam menangani pengelolaan kendaraan.
  6. Mempermudah operator dalam menangani pengelolaan tarif.
  7. Mempermudah operator dalam menangani pengelolaan onderdil.
  8. Mempermudah pelanggan dalam mendaftar antrian service.
  9. Mempermudah pelanggan dalam membatalkan antrian service.
  10. Mempermudah pelanggan dalam memberikan feedback.
  11. Mempermudah pemilik dalam melakukan monitoring laporan.
  12. Mempermudah pemilik dalam menanggapi feedback
- Dan berjalan pada lingkungan dengan platform web.

### 1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

| Keyword/Phrase | Definisi   |
|----------------|--|
| SKPL           | Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.   |
| SKPL-CRS-XXX   | Kode yang merepresentasikan kebutuhan pada CRS ( <i>Chrisna Reservation System</i> ) dimana XXX merupakan nomor fungsi produk. |
| CRS            | Sistem Reservasi Pemesanan Antrian Service Berbasis Web dan SMS.   |
| DBMS           | <i>Database Management System</i> merupakan suatu sistem atau perangkat lunak yang dirancang untuk mengelola suatu basis data  |

|                                  |            |      |
|----------------------------------|------------|------|
| Program Studi Teknik Informatika | SKPL – CRS | 7/47 |
|----------------------------------|------------|------|

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

|          |   |
|----------|---|
|          | dan menjalankan operasi terhadap data yang diminta banyak pengguna.   |
| SMS      | <i>Short Message Service</i> merupakan fitur yang digunakan untuk berkirim pesan dalam format teks.   |
| HTTP     | <i>Hypertext Transfer Protocol</i> merupakan sebuah protokol yang meminta/menjawab antara klien dan server.   |
| WAP      | <i>Wireless Application Protocol</i> merupakan sebuah protokol atau sebuah teknik <i>messaging service</i> yang memungkinkan sebuah telepon genggam digital atau terminal mobile yang mempunyai fasilitas WAP, melihat/membaca isi sebuah situs di <i>internet</i> dalam sebuah format teks khusus. |
| Server   | Komputer yang menyediakan sumber daya bagi klien yang terhubung melalui jaringan.   |
| Internet | Internet merupakan istilah umum yang dipakai untuk menunjuk <i>Network</i> global yang terdiri dari komputer dan layanan servis dengan sekitar 30 sampai 50 juta pemakai komputer dan puluhan layanan informasi termasuk e-mail, FTP, dan World Wide Web.   |
| PHP      | <i>PHP Hypertext Preprocessor</i> , merupakan bahasa yang bersifat <i>server side</i> yang didesain khusus untuk aplikasi web.  |
| BTS      | Base Transceiver Station, merupakan sebuah peralatan yang memfasilitasi komunikasi <i>wireless</i> antara peralatan pengguna dan jaringan.  |

|      |  |
|------|--|
| SMSC | <i>SMS Center, merupakan elemen jaringan dalam jaringan telepon selular yang memberikan pesan SMS.</i> |
|------|--|

#### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Bennet Simon, McRobb Steve, Farmer Ray, *Object-Oriented System Analysis and Design Using UML*, McGraw-Hill Companies, 2002.
2. Boggs Wendy, Boggs Michael, *Mastering UML with Rational Rose 2002*, SYBEX Inc, 2002.

#### 1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak CRS yang akan dikembangkan, mencakup perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak CRS tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak CRS yang akan dikembangkan.

## 2 Deskripsi Kebutuhan

### 2.1 Perspektif produk

CRS merupakan perangkat lunak yang dikembangkan untuk melakukan reservasi antrian *service* di Bengkel Chrisna Salatiga melalui *website* dan SMS. Sistem ini memiliki *website* untuk mempermudah administrator dalam menangani pengelolaan user, pengelolaan teknisi, dan mempermudah operator dalam menangani pengelolaan reservasi, pengelolaan pelanggan, pengelolaan kendaraan, pengelolaan tarif, pengelolaan onderdil. Sistem ini juga memiliki layanan SMS untuk mempermudah pelanggan dalam mendaftar ataupun membatalkan antrian *service*, serta memberikan feedback. Selain itu, sistem ini juga memiliki layanan *mobile website* yang digunakan oleh pemilik untuk dapat melakukan monitoring laporan dan menanggapi feedback.

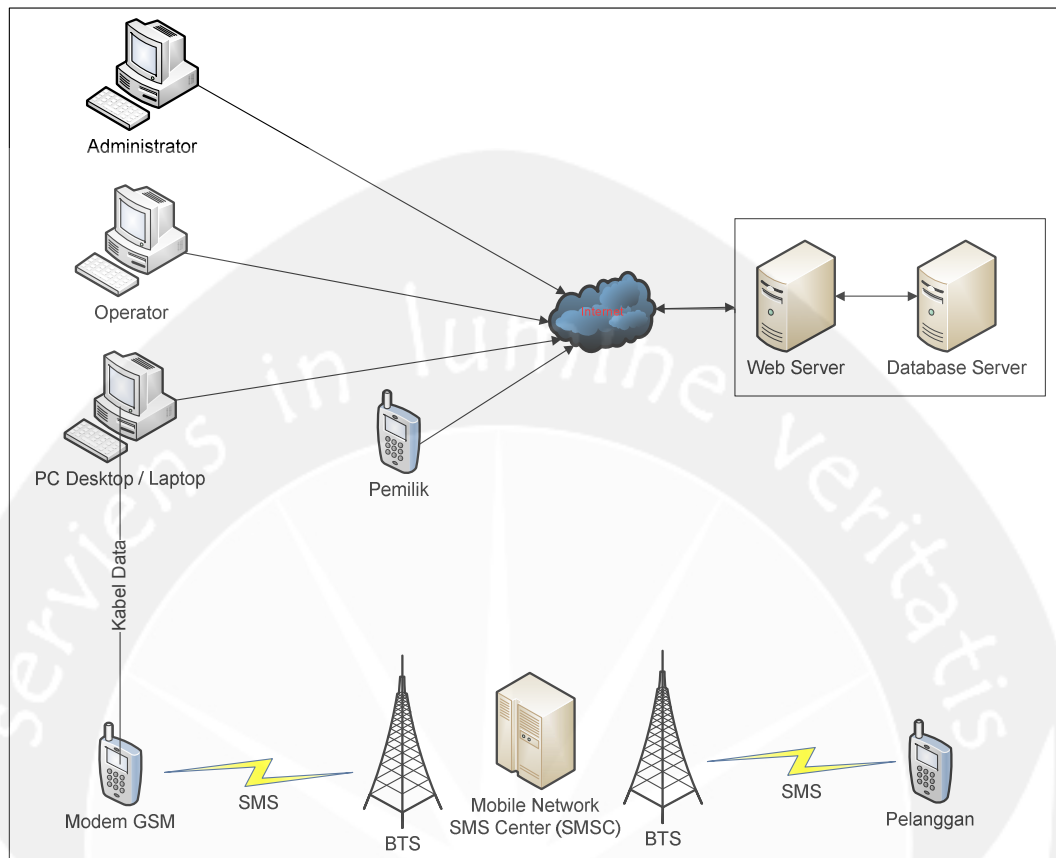
Perangkat lunak CRS ini berjalan pada *platform* Windows dalam bentuk *web based* yang dibuat menggunakan bahasa pemrograman PHP, HTML, XHTML MP, dan JavaScript dengan menggunakan *framework* CodeIgniter 2.0.3 (*framework* PHP) dengan mengacu pada MySQL sebagai DBMS (*Database Management System*). Sedangkan untuk lingkungan pemrogramannya menggunakan Notepad++ 5.9.3.

Administrator, operator, dan pemilik akan berinteraksi dengan sistem melalui antarmuka GUI (*Graphical User Interface*). Pada sistem ini, seperti terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan berupa *client server*, di mana semua data disimpan di *server*. Inputan data yang dimasukkan akan disimpan dalam *database server*, sehingga jika ada pencarian data, maka data yang diinginkan akan dicari

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 10/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

ke *database server* yang selanjutnya dikirimkan ke *client* yang melakukan permintaan melalui *web server*.

Untuk pelanggan akan berinteraksi dengan sistem melalui fitur SMS, setiap mengirimkan pesan melalui SMS berdasar kata kunci tertentu, pesan akan ditangkap oleh BTS dan pesan tadi akan dikirim ke SMS Center terlebih dahulu, sebelum sampai ke modem GSM yang digunakan. Setelah pesan diterima oleh modem GSM, modem GSM akan berkomunikasi dengan PC Desktop / Laptop. Kemudian, melalui aplikasi SMS yang ada pada PC Desktop / Laptop tersebut akan menyimpan data-data yang diperlukan dan pesan balasan ke database server. Setelah itu, melalui database server tersebut pesan balasan dikirimkan ke PC Desktop / Laptop, kemudian diteruskan ke modem GSM untuk mengirim pesan tersebut. Pesan itu akan ditangkap oleh BTS dan akan dikirim ke SMS Center terlebih dahulu, sebelum sampai ke pelanggan. Untuk lebih jelasnya dapat dilihat pada gambar 1.



Gambar 1. Arsitektur Perangkat Lunak CRS

## 2.2 Fungsi Produk

Fungsi produk perangkat lunak CRS adalah sebagai berikut :

### 2.2.1 Fungsi Produk Administrator

#### 1. Fungsi *Login* (SKPL-CRS-001).

Merupakan fungsi yang digunakan oleh pengguna sistem untuk dapat masuk ke dalam sistem yang akan digunakan.

#### 2. Fungsi *Pengelolaan User* (SKPL-CRS-002).

Merupakan fungsi yang digunakan oleh administrator untuk mengelola data *user* yang menggunakan sistem. Fungsi pengelolaan data *user* meliputi :

##### a. Fungsi *Tambah Data User* (SKPL-CRS-002-01)

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 12/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |



Merupakan fungsi yang digunakan oleh administrator untuk menambah data user.

b. Fungsi *Ubah Status User* (**SKPL-CRS-002-02**).

Merupakan fungsi yang digunakan oleh administrator untuk mengubah status user menjadi Aktif atau Blokir.

c. Fungsi *Hapus Data User* (**SKPL-CRS-002-03**).

Merupakan fungsi yang digunakan oleh administrator untuk menghapus data user.

d. Fungsi *Tampil Data User* (**SKPL-CRS-002-04**).

Merupakan fungsi yang digunakan oleh administrator untuk menampilkan data user.

e. Fungsi *Cari Data User* (**SKPL-CRS-002-05**).

Merupakan fungsi yang digunakan oleh administrator untuk mencari data user.

3. Fungsi *Pengelolaan Teknisi* (**SKPL-CRS-003**).

Merupakan fungsi yang digunakan oleh administrator untuk mengelola data teknisi.

Fungsi pengelolaan teknisi meliputi:

a. Fungsi *Tambah Data Teknisi* (**SKPL-CRS-003-01**).

Merupakan fungsi yang digunakan oleh administrator untuk menambah data teknisi.

b. Fungsi *Ubah Data Teknisi* (**SKPL-CRS-003-02**).

Merupakan fungsi yang digunakan oleh administrator untuk mengubah data teknisi.

c. Fungsi *Hapus Data Teknisi* (**SKPL-CRS-003-03**).

Merupakan fungsi yang digunakan oleh administrator untuk menghapus data teknisi.

d. Fungsi *Tampil Data Teknisi* (**SKPL-CRS-003-04**).

Merupakan fungsi yang digunakan oleh administrator untuk menampilkan data teknisi.

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 13/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

e. Fungsi Cari Data Teknisi (**SKPL-CRS-003-05**).

Merupakan fungsi yang digunakan oleh administrator untuk mencari data teknisi.

4. Fungsi *Ubah Password* (**SKPL-CRS-004**).

Merupakan fungsi yang digunakan oleh administrator untuk mengubah *password* yang digunakan untuk masuk ke dalam sistem.

### 2.2.2 Fungsi Produk Operator

1. Fungsi *Login* (**SKPL-CRS-005**).

Merupakan fungsi yang digunakan oleh pengguna sistem untuk dapat masuk ke dalam sistem yang akan digunakan.

2. Fungsi *Pengelolaan Reservasi* (**SKPL-CRS-006**).

Merupakan fungsi yang digunakan oleh operator untuk mengelola data reservasi.

Fungsi pengelolaan reservasi meliputi:

a. Fungsi *Tambah Data Reservasi* (**SKPL-CRS-006-01**).

Merupakan fungsi yang digunakan oleh operator untuk menambah data reservasi.

b. Fungsi *Ubah Data Reservasi* (**SKPL-CRS-006-02**).

Merupakan fungsi yang digunakan oleh operator untuk mengubah data reservasi.

c. Fungsi *Hapus Data Reservasi* (**SKPL-CRS-006-03**).

Merupakan fungsi yang digunakan oleh operator untuk menghapus data reservasi.

d. Fungsi *Tampil Data Reservasi* (**SKPL-CRS-006-04**).

Merupakan fungsi yang digunakan oleh operator untuk menampilkan data reservasi.

e. Fungsi *Cari Data Reservasi* (**SKPL-CRS-006-05**).

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 14/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

Merupakan fungsi yang digunakan oleh operator untuk mencari data reservasi.

3. Fungsi *Pengelolaan Pelanggan* (**SKPL-CRS-007**).

Merupakan fungsi yang digunakan oleh operator untuk mengelola data pelanggan.

Fungsi pengelolaan pelanggan meliputi:

a. Fungsi *Tambah Data Pelanggan* (**SKPL-CRS-007-01**).

Merupakan fungsi yang digunakan oleh operator untuk menambah data pelanggan.

b. Fungsi *Ubah Data Pelanggan* (**SKPL-CRS-007-02**).

Merupakan fungsi yang digunakan oleh operator untuk mengubah data pelanggan.

c. Fungsi *Tampil Data Pelanggan* (**SKPL-CRS-007-03**).

Merupakan fungsi yang digunakan oleh operator untuk menampilkan data pelanggan.

d. Fungsi *Cari Data Pelanggan* (**SKPL-CRS-007-04**).

Merupakan fungsi yang digunakan oleh operator untuk mencari data pelanggan.

4. Fungsi *Pengelolaan Kendaraan* (**SKPL-CBSS-008**).

Merupakan fungsi yang digunakan oleh operator untuk mengelola data kendaraan.

Fungsi pengelolaan kendaraan meliputi:

a. Fungsi *Tambah Data Kendaraan* (**SKPL-CRS-008-01**).

Merupakan fungsi yang digunakan oleh operator untuk menambah data kendaraan.

b. Fungsi *Ubah Data Kendaraan* (**SKPL-CRS-008-02**).

Merupakan fungsi yang digunakan oleh operator untuk mengubah data kendaraan.

c. Fungsi *Tampil Data Kendaraan* (**SKPL-CRS-008-03**).

Merupakan fungsi yang digunakan oleh operator untuk menampilkan data kendaraan.

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 15/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

d. Fungsi *Cari Data Kendaraan* (**SKPL-CRS-008-04**).

Merupakan fungsi yang digunakan oleh operator untuk mencari data kendaraan.

5. Fungsi *Pengelolaan Tarif* (**SKPL-CRS-009**).

Merupakan fungsi yang digunakan oleh operator untuk mengelola data tarif.

Fungsi pengelolaan tarif meliputi:

a. Fungsi *Tambah Data Tarif* (**SKPL-CRS-009-01**).

Merupakan fungsi yang digunakan oleh operator untuk menambah data tarif.

b. Fungsi *Ubah Data Tarif* (**SKPL-CRS-009-02**).

Merupakan fungsi yang digunakan oleh operator untuk mengubah data tarif.

c. Fungsi *Tampil Data Tarif* (**SKPL-CRS-009-03**).

Merupakan fungsi yang digunakan oleh operator untuk menampilkan data tarif.

6. Fungsi *Pengelolaan Onderdil* (**SKPL-CRS-010**).

Merupakan fungsi yang digunakan oleh operator untuk mengelola data onderdil.

Fungsi pengelolaan onderdil meliputi:

a. Fungsi *Tambah Data Onderdil* (**SKPL-CRS-010-01**).

Merupakan fungsi yang digunakan oleh operator untuk menambah data onderdil.

b. Fungsi *Ubah Data Onderdil* (**SKPL-CRS-010-02**).

Merupakan fungsi yang digunakan oleh operator untuk mengubah data onderdil.

c. Fungsi *Tampil Data Onderdil* (**SKPL-CRS-010-03**).

Merupakan fungsi yang digunakan oleh operator untuk menampilkan data onderdil.

d. Fungsi *Cari Data Onderdil* (**SKPL-CRS-010-04**).

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 16/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

Merupakan fungsi yang digunakan oleh operator untuk mencari data onderdil.

7. Fungsi *Ubah Password* (**SKPL-CRS-011**).

Merupakan fungsi yang digunakan oleh operator untuk mengubah *password* yang digunakan untuk masuk ke dalam sistem.

### 2.2.3 Fungsi Produk Pelanggan

1. Fungsi *Login* (**SKPL-CRS-012**).

Merupakan fungsi yang digunakan oleh pengguna sistem untuk dapat masuk ke dalam sistem yang akan digunakan.

2. Fungsi *Daftar Antrian* (**SKPL-CRS-013**).

Merupakan fungsi yang digunakan oleh pelanggan untuk mendaftar antrian service. Pendaftaran antrian service menggunakan kata kunci DAFTAR#[USERNAME]#[PASSWORD]#[NOPLAT].

3. Fungsi *Batal Antrian* (**SKPL-CRS-014**).

Merupakan fungsi yang digunakan oleh pelanggan untuk membatalkan antrian service. Pembatalan antrian service menggunakan kata kunci BATAL#[USERNAME]#[PASSWORD]#[NOPLAT].

4. Fungsi *Tambah Feedback* (**SKPL-CRS-015**).

Merupakan fungsi yang digunakan oleh pelanggan untuk memberikan feedback kepada pemilik bengkel. Untuk dapat memberikan feedback menggunakan kata kunci FEEDBACK#[USERNAME]#[PASSWORD]#[PESAN].

5. Fungsi *Ubah Password* (**SKPL-CRS-016**).

Merupakan fungsi yang digunakan oleh pelanggan untuk mengubah *password* yang digunakan untuk masuk ke dalam sistem. Untuk dapat mengubah password

menggunakan kata kunci  
UBAH#[USERNAME]#[PASSWORD]#[PASSBARU].

#### 2.2.4 Fungsi Produk Pemilik

1. Fungsi *Login* (SKPL-CRS-017).  
Merupakan fungsi yang digunakan oleh pemilik sistem untuk dapat masuk ke dalam sistem yang akan digunakan.
2. Fungsi *Monitoring Laporan* (SKPL-CRS-018).  
Merupakan fungsi yang digunakan oleh pemilik untuk dapat melakukan monitoring laporan.
3. Fungsi *Balas Feedback* (SKPL-CRS-019).  
Merupakan fungsi yang digunakan oleh pemilik untuk dapat membalas feedback dari pelanggan.
4. Fungsi *Ubah Password* (SKPL-CRS-020).  
Merupakan fungsi yang digunakan oleh pemilik untuk mengubah *password* yang digunakan untuk masuk ke dalam sistem.

#### 2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak CRS (Chrisna Reservation System) yaitu :

1. Administrator
  - a. Memahami pengoperasian komputer.
  - b. Memahami perangkat lunak yang digunakan.
  - c. Memahami tentang penggunaan aplikasi berbasis web di komputer.
2. Operator
  - a. Memahami pengoperasian komputer.
  - b. Memahami perangkat lunak yang digunakan.

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 18/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

c. Memahami tentang penggunaan aplikasi berbasis web di komputer.

3. Pelanggan

a. Memahami pengoperasian *handphone*.

4. Pemilik

a. Memahami pengoperasian *handphone*.

b. Memahami perangkat lunak yang digunakan.

c. Memahami tentang penggunaan aplikasi berbasis web di *handphone*.

#### 2.4 Batasan-batasan

Batasan-batasan dalam pengembangan perangkat lunak CRS tersebut adalah :

1. Kebijaksanaan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak CRS yaitu untuk melakukan pengelolaan user, pengelolaan pegawai, pengelolaan reservasi, pengelolaan pelanggan, pengelolaan kendaraan, pengelolaan tarif, pengelolaan onderdil, dan pelaporan.

2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

#### 2.5 Asumsi dan Ketergantungan

Asumsi dan ketergantungan dalam pengembangan perangkat lunak CRS tersebut adalah :

1. Sistem yang digunakan oleh administrator dan operator dapat dijalankan pada perangkat seperti laptop atau PC yang memiliki koneksi internet.

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 19/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

2. Sistem yang digunakan oleh pemilik dapat dijalankan pada perangkat laptop, PC, atau mobile yang memiliki koneksi internet.
3. Sistem yang digunakan oleh pelanggan dapat dijalankan pada perangkat mobile yang memiliki fitur SMS.

### **3 Kebutuhan khusus**

#### **3.1 Kebutuhan antarmuka eksternal**

Kebutuhan antarmuka eksternal pada perangkat lunak CRS meliputi kebutuhan antarmuka pemakai, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

##### **3.1.1 Antarmuka pemakai**

Antarmuka Pemakai dalam pengembangan perangkat lunak CRS tersebut adalah :

1. Administrator dan operator berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form yang dijalankan pada perangkat PC atau laptop.
2. Pemilik berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form yang dijalankan pada perangkat mobile.
3. Pelanggan berinteraksi dengan fitur SMS yang dijalankan pada perangkat mobile.

##### **3.1.2 Antarmuka perangkat keras**

Antarmuka perangkat keras yang digunakan dalam perangkat lunak CRS adalah :

1. Perangkat mobile (handphone)
2. PC atau laptop
3. Modem GSM

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 20/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |



### 3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak CRS adalah sebagai berikut :

1. Nama : Windows 7  
Sumber : Microsoft  
Sebagai sistem operasi untuk *web based*.
2. Nama : MySQL  
Sumber : MySQL  
Sebagai *Database Management System* (DBMS) yang digunakan untuk menyimpan data di sisi server.
3. Nama : Apache  
Sumber : Apache  
Sebagai web server.
4. Nama : Gammu 1.30.0  
Sumber : Gammu  
Sebagai daemon untuk SMS.
5. Nama : Gingerbread 2.3.4  
Sumber : Android  
Sebagai sistem operasi untuk *mobile web based*.
6. Nama : Mobile Partner  
Sumber : HUAWEI  
Sebagai driver dari modem GSM.
7. Nama : Google Chrome  
Sumber : Google  
Sebagai browser untuk menjalankan aplikasi web.
8. Nama : Dolphin Browser  
Sumber : Dolphin  
Sebagai browser untuk menjalankan aplikasi mobile web.
9. Nama : CodeIgniter 2.0.3

Sumber : CodeIgniter

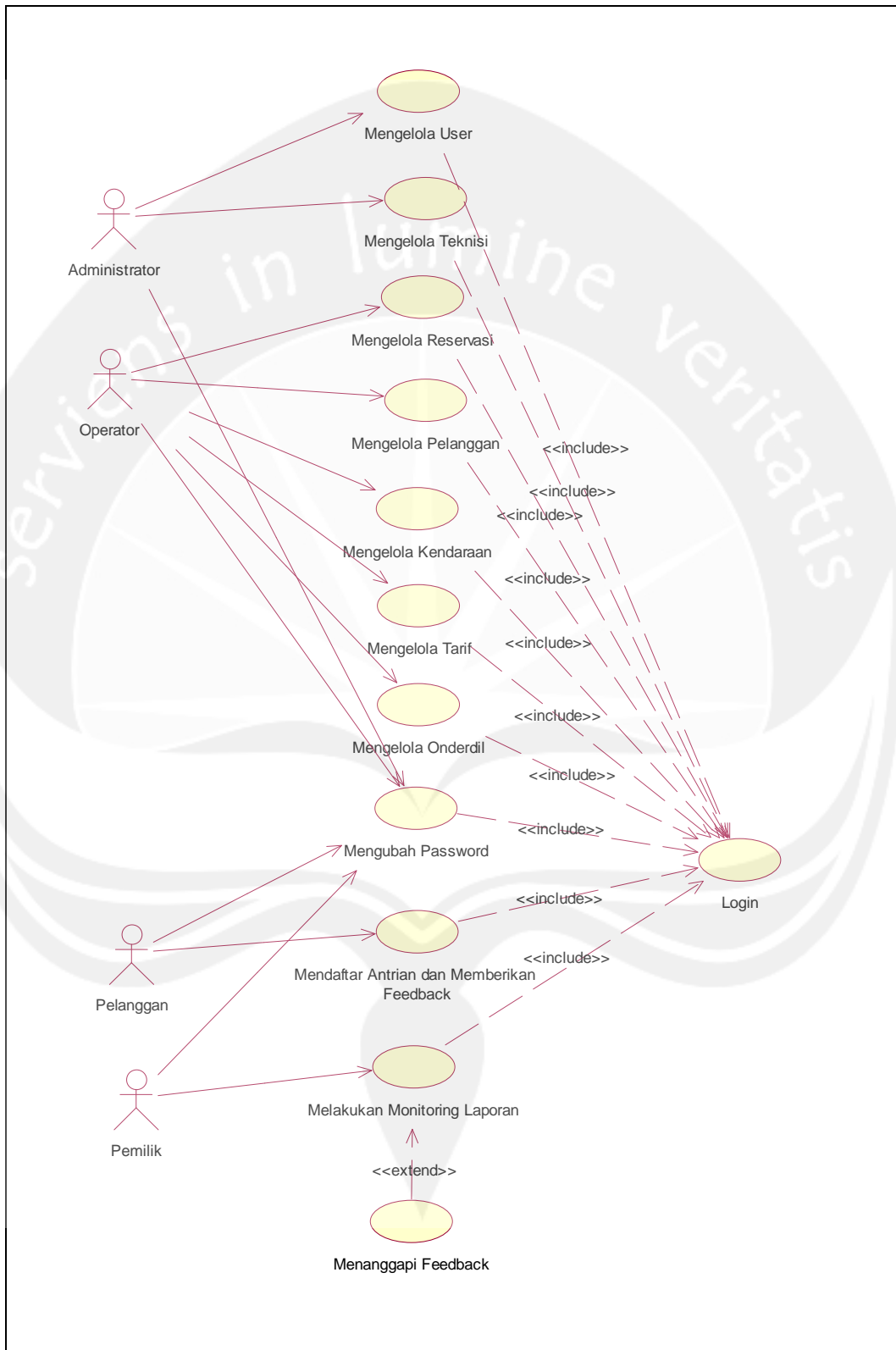
Sebagai framework dalam pembangunan aplikasi berbasis PHP.

#### 3.1.4 Antarmuka komunikasi

Antarmuka komunikasi perangkat lunak CRS menggunakan protokol HTTP, WAP.



### 3.2 Kebutuhan Fungsionalitas Perangkat Lunak



Gambar 2. Use Case Diagram

## **4 Spesifikasi Rinci Kebutuhan**

### **4.1 Spesifikasi Kebutuhan Fungsionalitas**

#### **4.1.1 Use case Spesification : Login**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk masuk ke dalam sistem. Login didasarkan pada sebuah username dan password yang berupa rangkaian karakter.

##### **2. Primary Actor**

1. Administrator
2. Operator
3. Pelanggan
4. Pemilik

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login
2. Sistem menampilkan antarmuka untuk login
3. Aktor memasukkan username dan password
4. Sistem memeriksa username dan password yang diinputkan aktor  
E-1 Password atau username tidak sesuai
5. Sistem memberikan akses ke aktor
6. Use Case selesai

##### **5. Alternative Flow**

none

##### **6. Error Flow**

E-1 Password atau username tidak sesuai

1. Sistem menampilkan peringatan bahwa username atau password tidak sesuai
2. Kembali ke Basic Flow langkah ke 3

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 24/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

## 7. PreConditions

none

## 8. PostConditions

Aktor memasuki sistem dan dapat menggunakan fungsi-fungsi pada sistem.

### 4.1.2 Use case Spesification : Mengelola User

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data user. Aktor dapat melakukan tambah data user, ubah status user, hapus data user, dan cari data user.

#### 2. Primary Actor

Administrator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data user
2. Sistem memberikan pilihan untuk melakukan tambah data user, ubah status user, hapus data user, atau cari data user.
3. Aktor memilih untuk melakukan tambah data user
  - A-1 Aktor memilih untuk melakukan ubah status user
  - A-2 Aktor memilih untuk melakukan hapus data user
  - A-3 Aktor memilih untuk melakukan cari data user
4. Aktor menginputkan data user
5. Aktor meminta sistem untuk menyimpan data user yang telah diinputkan
6. Sistem mengecek data user yang telah diinputkan
  - E-1 Data user yang diinputkan salah

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 25/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

7. Sistem menyimpan data user ke database

8. Use case selesai

## 5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah status user

1. Sistem menampilkan status user
2. Aktor mengubah status user yang ditampilkan
3. Aktor meminta sistem untuk menyimpan status user yang telah diubah
4. Sistem melakukan pengecekan terhadap status user yang telah diubah
5. Sistem menyimpan status user yang telah diubah ke database
6. Berlanjut ke Basic Flow langkah ke 8

A-2 Aktor memilih untuk melakukan hapus data user

1. Sistem menampilkan data user yang sudah tersimpan
2. Aktor memilih data yang akan dihapus
3. Aktor meminta system untuk menghapus data yang sudah dipilih
4. Sistem menghapus data yang sudah dipilih oleh aktor
5. Berlanjut ke Basic Flow langkah ke 8

A-3 Aktor memilih untuk melakukan cari data user

1. Aktor menginputkan kata kunci untuk mencari data user yang diinginkan
2. Aktor meminta system mencari data user sesuai dengan kata kunci yang diinputkan
3. Sistem menampilkan data user sesuai dengan kata kunci yang diinputkan

E-2 Data user yang dicari tidak ditemukan

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 26/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

4. Berlanjut ke Basic Flow langkah ke 8

## 6. Error Flow

E-1 Data user yang diinputkan salah

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow langkah ke 4

E-2 Data user yang dicari tidak ditemukan

1. Sistem memberikan pesan peringatan bahwa data yang dicari tidak ditemukan
2. Kembali ke Alternative Flow A-3 langkah ke 1

## 7. PreConditions

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

## 8. PostConditions

Data user di database telah ter-update

### 4.1.3 Use case Spesification : Mengelola Teknisi

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data teknisi. Aktor dapat melakukan tambah data teknisi, ubah data teknisi, hapus data teknisi, dan cari data teknisi.

#### 2. Primary Actor

Administrator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data teknisi
2. Sistem memberikan pilihan untuk melakukan tambah data teknisi, ubah data teknisi, hapus data teknisi, atau cari data teknisi.

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 27/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

3. Aktor memilih untuk melakukan tambah data teknisi
  - A-1 Aktor memilih untuk melakukan ubah data teknisi
  - A-2 Aktor memilih untuk melakukan hapus data teknisi
  - A-3 Aktor memilih untuk melakukan cari data teknisi
4. Aktor menginputkan data teknisi
5. Aktor meminta sistem untuk menyimpan data teknisi yang telah diinputkan
6. Sistem mengecek data teknisi yang telah diinputkan
  - E-1 Data teknisi yang diinputkan salah
7. Sistem menyimpan data teknisi ke database
8. Use case selesai

#### **5. Alternative Flow**

- A-1 Aktor memilih untuk melakukan ubah data teknisi
  1. Sistem menampilkan data teknisi
  2. Aktor mengubah data pegawai yang ditampilkan
  3. Aktor meminta sistem untuk menyimpan data teknisi yang telah diubah
  4. Sistem melakukan pengecekan terhadap data teknisi yang telah diubah
    - E-2 Data teknisi yang telah diubah salah
  5. Sistem menyimpan data teknisi yang telah diubah ke database
  6. Berlanjut ke Basic Flow langkah ke 8
- A-2 Aktor memilih untuk melakukan hapus data teknisi
  1. Sistem menampilkan data teknisi yang sudah tersimpan
  2. Aktor memilih data yang akan dihapus



3. Aktor meminta sistem untuk menghapus data yang sudah dipilih
  4. Sistem menghapus data yang sudah dipilih oleh aktor
  5. Berlanjut ke Basic Flow langkah ke 8
- A-3 Aktor memilih untuk melakukan cari data teknisi
1. Aktor menginputkan kata kunci untuk mencari data teknisi yang diinginkan
  2. Aktor meminta sistem mencari data teknisi sesuai dengan kata kunci yang diinputkan
  3. Sistem menampilkan data teknisi sesuai dengan kata kunci yang diinputkan
- E-3 Data teknisi yang dicari tidak ditemukan
4. Berlanjut ke Basic Flow langkah ke 8

#### **6. Error Flow**

- E-1 Data teknisi yang diinputkan salah
1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
  2. Kembali ke Basic Flow langkah ke 4
- E-2 Data teknisi yang telah diubah salah
1. Sistem memberikan pesan peringatan bahwa data yang diubah salah
  2. Kembali ke Alternative-Flow A-1 langkah ke 2
- E-2 Data teknisi yang dicari tidak ditemukan
1. Sistem memberikan pesan peringatan bahwa data yang dicari tidak ditemukan
  2. Kembali ke Alternative Flow A-3 langkah ke 1

#### **7. PreConditions**

1. Use case login telah dilakukan
2. Aktor telah memasuki system

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 29/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

## 8. PostConditions

Data teknisi di database telah ter-update

### 4.1.4 Use case Spesification : Mengelola Reservasi

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data reservasi. Aktor dapat melakukan tambah data reservasi, ubah data reservasi, hapus data reservasi, dan cari data reservasi.

#### 2. Primary Actor

Operator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data reservasi
2. Sistem memberikan pilihan untuk melakukan tambah data reservasi, ubah data reservasi, hapus data reservasi, atau cari data reservasi.
3. Aktor memilih untuk melakukan tambah data reservasi
  - A-1 Aktor memilih untuk melakukan ubah data reservasi
  - A-2 Aktor memilih untuk melakukan hapus data reservasi
  - A-3 Aktor memilih untuk melakukan cari data reservasi
4. Aktor menginputkan data reservasi
5. Aktor meminta sistem untuk menyimpan data reservasi yang telah diinputkan
6. Sistem mengecek data reservasi yang telah diinputkan

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 30/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

- E-1 Data reservasi yang diinputkan salah
- 7. Sistem menyimpan data reservasi ke database
- 8. Use case selesai

#### 5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah data reservasi

1. Sistem menampilkan data reservasi
2. Aktor mengubah data reservasi yang ditampilkan
3. Aktor meminta sistem untuk menyimpan data reservasi yang telah diubah
4. Sistem melakukan pengecekan terhadap data reservasi yang telah diubah

E-2 Data reservasi yang telah diubah salah

5. Sistem menyimpan data reservasi yang telah diubah ke database
6. Berlanjut ke Basic Flow langkah ke 8

A-2 Aktor memilih untuk melakukan hapus data reservasi

1. Sistem menampilkan data reservasi yang sudah tersimpan
2. Aktor memilih data yang akan dihapus
3. Aktor meminta sistem untuk menghapus data yang sudah dipilih
4. Sistem menghapus data yang sudah dipilih oleh aktor
5. Berlanjut ke Basic Flow langkah ke 8

A-3 Aktor memilih untuk melakukan cari data reservasi

1. Aktor menginputkan kata kunci untuk mencari data reservasi yang diinginkan

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 31/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

2. Aktor meminta sistem mencari data reservasi sesuai dengan kata kunci yang diinputkan
3. Sistem menampilkan data reservasi sesuai dengan kata kunci yang diinputkan
- E-3 Data reservasi yang dicari tidak ditemukan
4. Berlanjut ke Basic Flow langkah ke 8

#### **6. Error Flow**

E-1 Data reservasi yang diinputkan salah

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow langkah ke 4

E-2 Data reservasi yang telah diubah salah

1. Sistem memberikan pesan peringatan bahwa data yang diubah salah
2. Kembali ke Alternative-Flow A-1 langkah ke 2

E-2 Data reservasi yang dicari tidak ditemukan

1. Sistem memberikan pesan peringatan bahwa data yang dicari tidak ditemukan
2. Kembali ke Alternative Flow A-3 langkah ke 1

#### **7. PreConditions**

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

Data reservasi di database telah ter-update

### **4.1.5 Use case Spesification : Mengelola Pelanggan**

#### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk mengelola data pelanggan. Aktor dapat melakukan tambah data pelanggan, ubah data pelanggan, dan cari data pelanggan.

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 32/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

## 2. Primary Actor

Operator

## 3. Supporting Actor

none

## 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pelanggan
2. Sistem memberikan pilihan untuk melakukan tambah data pelanggan, ubah data pelanggan, atau cari data pelanggan.
3. Aktor memilih untuk melakukan tambah data pelanggan
  - A-1 Aktor memilih untuk melakukan ubah data pelanggan
  - A-2 Aktor memilih untuk melakukan cari data pelanggan
4. Aktor menginputkan data pelanggan
5. Aktor meminta sistem untuk menyimpan data pelanggan yang telah diinputkan
6. Sistem mengecek data pelanggan yang telah diinputkan
  - E-1 Data pelanggan yang diinputkan salah
7. Sistem menyimpan data pelanggan ke database
8. Use case selesai

## 5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah data pelanggan
  1. Sistem menampilkan data pelanggan
  2. Aktor mengubah data pelanggan yang ditampilkan

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 33/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

3. Aktor meminta sistem untuk menyimpan data pelanggan yang telah diubah
  4. Sistem melakukan pengecekan terhadap data pelanggan yang telah diubah
    - E-2 Data pelanggan yang telah diubah salah
  5. Sistem menyimpan data pelanggan yang telah diubah ke database
  6. Berlanjut ke Basic Flow langkah ke 8
- A-2 Aktor memilih untuk melakukan cari data pelanggan
1. Aktor menginputkan kata kunci untuk mencari data pelanggan yang diinginkan
  2. Aktor meminta sistem mencari data pelanggan sesuai dengan kata kunci yang diinputkan
  3. Sistem menampilkan data pelanggan sesuai dengan kata kunci yang diinputkan
    - E-3 Data pelanggan yang dicari tidak ditemukan
  4. Berlanjut ke Basic Flow langkah ke 8

## 6. Error Flow

- E-1 Data pelanggan yang diinputkan salah
1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
  2. Kembali ke Basic Flow langkah ke 4
- E-2 Data pelanggan yang telah diubah salah
1. Sistem memberikan pesan peringatan bahwa data yang diubah salah
  2. Kembali ke Alternative-Flow A-1 langkah ke 2
- E-2 Data pelanggan yang dicari tidak ditemukan
1. Sistem memberikan pesan peringatan bahwa data yang dicari tidak ditemukan

2. Kembali ke Alternative Flow A-3 langkah ke 1

#### **7. PreConditions**

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

Data pelanggan di database telah ter-update

#### **4.1.6 Use case Spesification : Mengelola Kendaraan**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk mengelola data kendaraan. Aktor dapat melakukan tambah data kendaraan, ubah data kendaraan, dan cari data kendaraan.

##### **2. Primary Actor**

Operator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data kendaraan
2. Sistem memberikan pilihan untuk melakukan tambah data kendaraan, ubah data kendaraan, atau cari data kendaraan.
3. Aktor memilih untuk melakukan tambah data kendaraan  
A-1 Aktor memilih untuk melakukan ubah data kendaraan  
A-2 Aktor memilih untuk melakukan cari data kendaraan
4. Aktor menginputkan data kendaraan
5. Aktor meminta sistem untuk menyimpan data kendaraan yang telah diinputkan

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 35/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

6. Sistem mengecek data kendaraan yang telah diinputkan

E-1 Data kendaraan yang diinputkan salah

7. Sistem menyimpan data kendaraan ke database

8. Use case selesai

#### 5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah data kendaraan

1. Sistem menampilkan data kendaraan
2. Aktor mengubah data kendaraan yang ditampilkan
3. Aktor meminta sistem untuk menyimpan data kendaraan yang telah diubah
4. Sistem melakukan pengecekan terhadap data kendaraan yang telah diubah

E-2 Data kendaraan yang telah diubah salah

5. Sistem menyimpan data kendaraan yang telah diubah ke database
6. Berlanjut ke Basic Flow langkah ke 8

A-2 Aktor memilih untuk melakukan cari data kendaraan

1. Aktor menginputkan kata kunci untuk mencari data kendaraan yang diinginkan
2. Aktor meminta sistem mencari data kendaraan sesuai dengan kata kunci yang diinputkan
3. Sistem menampilkan data kendaraan sesuai dengan kata kunci yang diinputkan

E-3 Data kendaraan yang dicari tidak ditemukan

4. Berlanjut ke Basic Flow langkah ke 8



## 6. Error Flow

E-1 Data kendaraan yang diinputkan salah

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow langkah ke 4

E-2 Data kendaraan yang telah diubah salah

1. Sistem memberikan pesan peringatan bahwa data yang diubah salah
2. Kembali ke Alternative-Flow A-1 langkah ke 2

E-2 Data kendaraan yang dicari tidak ditemukan

1. Sistem memberikan pesan peringatan bahwa data yang dicari tidak ditemukan
2. Kembali ke Alternative Flow A-3 langkah ke 1

## 7. PreConditions

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

## 8. PostConditions

Data kendaraan di database telah ter-update

### 4.1.7 Use case Spesification : Mengelola Tarif

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data tarif. Aktor dapat melakukan tambah data tarif, ubah data tarif, dan cari data tarif.

#### 2. Primary Actor

Operator

#### 3. Supporting Actor

none

#### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data tarif

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 37/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

2. Sistem memberikan pilihan untuk melakukan tambah data tarif, ubah data tarif, atau cari data tarif.
3. Aktor memilih untuk melakukan tambah data tarif
  - A-1 Aktor memilih untuk melakukan ubah data tarif
4. Aktor menginputkan data tarif
5. Aktor meminta sistem untuk menyimpan data tarif yang telah diinputkan
6. Sistem mengecek data tarif yang telah diinputkan
  - E-1 Data tarif yang diinputkan salah
7. Sistem menyimpan data tarif ke database
8. Use case selesai

#### **5. Alternative Flow**

- A-1 Aktor memilih untuk melakukan ubah data tarif
  1. Sistem menampilkan data tarif
  2. Aktor mengubah data tarif yang ditampilkan
  3. Aktor meminta sistem untuk menyimpan data tarif yang telah diubah
  4. Sistem melakukan pengecekan terhadap data tarif yang telah diubah
    - E-2 Data tarif yang telah diubah salah
  5. Sistem menyimpan data tarif yang telah diubah ke database
  6. Berlanjut ke Basic Flow langkah ke 8

#### **6. Error Flow**

- E-1 Data tarif yang diinputkan salah
  1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
  2. Kembali ke Basic Flow langkah ke 4
- E-2 Data tarif yang telah diubah salah
  1. Sistem memberikan pesan peringatan bahwa data yang diubah salah

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 38/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

2. Kembali ke Alternative-Flow A-1 langkah ke 2

#### **7. PreConditions**

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

Data tarif di database telah ter-update

#### **4.1.8 Use case Spesification : Mengelola Onderdil**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk mengelola data onderdil. Aktor dapat melakukan tambah data onderdil, ubah data onderdil, dan cari data onderdil.

##### **2. Primary Actor**

Operator

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data onderdil
2. Sistem memberikan pilihan untuk melakukan tambah data onderdil, ubah data onderdil, atau cari data onderdil.
3. Aktor memilih untuk melakukan tambah data onderdil  
A-1 Aktor memilih untuk melakukan ubah data onderdil  
A-2 Aktor memilih untuk melakukan cari data onderdil
4. Aktor menginputkan data onderdil
5. Aktor meminta sistem untuk menyimpan data onderdil yang telah diinputkan

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 39/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

6. Sistem mengecek data onderdil yang telah diinputkan

E-1 Data onderdil yang diinputkan salah

7. Sistem menyimpan data onderdil ke database

8. Use case selesai

#### **5. Alternative Flow**

A-1 Aktor memilih untuk melakukan ubah data onderdil

1. Sistem menampilkan data onderdil
2. Aktor mengubah data onderdil yang ditampilkan
3. Aktor meminta sistem untuk menyimpan data onderdil yang telah diubah
4. Sistem melakukan pengecekan terhadap data onderdil yang telah diubah

E-2 Data onderdil yang telah diubah salah

5. Sistem menyimpan data onderdil yang telah diubah ke database
6. Berlanjut ke Basic Flow langkah ke 8

A-2 Aktor memilih untuk melakukan cari data onderdil

1. Aktor menginputkan kata kunci untuk mencari data onderdil yang diinginkan
2. Aktor meminta sistem mencari data onderdil sesuai dengan kata kunci yang diinputkan
3. Sistem menampilkan data onderdil sesuai dengan kata kunci yang diinputkan

E-3 Data onderdil yang dicari tidak ditemukan

4. Berlanjut ke Basic Flow langkah ke 8

#### **6. Error Flow**

E-1 Data onderdil yang diinputkan salah

1. Sistem memberikan pesan peringatan bahwa data yang diinputkan salah
2. Kembali ke Basic Flow langkah ke 4

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 40/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

E-2 Data onderdil yang telah diubah salah

1. Sistem memberikan pesan peringatan bahwa data yang diubah salah
  2. Kembali ke Alternative-Flow A-1 langkah ke 2
- E-3 Data onderdil yang dicari tidak ditemukan

1. Sistem memberikan pesan peringatan bahwa data yang dicari tidak ditemukan
2. Kembali ke Alternative Flow A-3 langkah ke 1

#### **7. PreConditions**

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

Data onderdil di database telah ter-update

#### **4.1.9 Use case Spesification : Mengubah Password**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk mengubah password yang digunakan untuk masuk ke dalam sistem.

##### **2. Primary Actor**

1. Administrator
2. Operator
3. Pelanggan
4. Pemilik

##### **3. Supporting Actor**

none

##### **4. Basic Flow**

1. Use Case ini dimulai ketika aktor memilih untuk melakukan ubah password
2. Sistem menampilkan antarmuka untuk ubah password
3. Aktor menginputkan password lama, password baru, dan konfirmasi password baru

|  |            |       |
|--|------------|-------|
| Program Studi Teknik Informatika   | SKPL – CRS | 41/47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |       |

4. Aktor meminta sistem untuk mengubah password lama menjadi password baru

5. Sistem memeriksa password lama, password baru, dan konfirmasi password baru yang diinputkan aktor

E-1 Password lama atau konfirmasi password baru tidak sesuai

6. Sistem mengubah password lama menjadi password baru

7. Use Case ini selesai

#### **5. Alternative Flow**

none

#### **6. Error Flow**

E-1 Password lama atau konfirmasi password baru tidak sesuai

1. Sistem menampilkan peringatan bahwa password lama atau konfirmasi password baru tidak sesuai

2. Kembali ke Basic Flow langkah ke 3

#### **7. PreConditions**

1. Use case login telah dilakukan

2. Aktor telah memasuki sistem

#### **8. PostConditions**

Password lama telah diubah menjadi password baru

### **4.1.10 Use case Spesification : Mendaftar Antrian dan Memberikan Feedback**

#### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk mendaftar antrian, membatalkan antrian, dan memberikan feedback.

#### **2. Primary Actor**

Pelanggan

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 42/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

### 3. Supporting Actor

none

### 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk memanfaatkan layanan SMS
2. Sistem memberikan pilihan untuk melakukan daftar antrian, batal antrian, atau memberikan feedback
3. Aktor memilih untuk melakukan daftar antrian  
A-1 Aktor memilih untuk melakukan batal antrian  
A-2 Aktor memilih untuk memberikan feedback
4. Aktor menginputkan kata kunci untuk daftar antrian
5. Aktor meminta sistem untuk menyimpan data antrian
6. Sistem memeriksa kata kunci untuk daftar antrian yang telah diinputkan  
E-1 Kata kunci untuk daftar antrian salah
7. Sistem menyimpan data antrian ke database
8. Use case selesai

### 5. Alternative Flow

- A-1 Aktor memilih untuk melakukan batal antrian
1. Aktor menginputkan kata kunci untuk batal antrian
  2. Aktor meminta sistem untuk membatalkan antrian
  3. Sistem memeriksa data antrian yang sudah tersimpan  
E-2 Kata kunci untuk batal antrian salah
  4. Sistem membatalkan antrian yang diminta
  5. Berlanjut ke Basic Flow langkah ke 8
- A-2 Aktor memilih untuk memberikan feedback
1. Aktor menginputkan kata kunci untuk memberikan feedback

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 43/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

2. Aktor meminta sistem untuk menyimpan data feedback
3. Sistem memeriksa kata kunci yang diberikan  
E-3 Kata kunci untuk memberikan feedback salah
4. Sistem menyimpan data feedback ke database
5. Berlanjut ke Basic Flow langkah ke 8

#### **6. Error Flow**

E-1 Kata kunci untuk daftar antrian salah

1. Sistem memberikan pesan peringatan bahwa kata kunci yang diinputkan salah
2. Kembali ke Basic Flow langkah ke 4

E-2 Kata kunci untuk batal antrian salah

1. Sistem memberikan pesan peringatan bahwa kata kunci yang diinputkan salah
2. Kembali ke Alternative-Flow A-1 langkah ke 1

E-3 Kata kunci untuk memberikan feedback salah

1. Sistem memberikan pesan peringatan bahwa kata kunci yang diinputkan salah
2. Kembali ke Alternative Flow A-2 langkah ke 1

#### **7. PreConditions**

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

#### **8. PostConditions**

Data reservasi atau feedback di database telah ter-update

#### **4.1.11 Use case Spesification : Melakukan Monitoring Laporan**

##### **1. Brief Description**

Use Case ini digunakan oleh aktor untuk melakukan monitoring laporan

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 44/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |



## 2. Primary Actor

Pemilik

## 3. Supporting Actor

none

## 4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan monitoring laporan
2. Sistem memberikan pilihan laporan untuk ditampilkan.
3. Aktor memilih untuk menampilkan laporan total pemasukan
  - A-1 Aktor memilih untuk menampilkan laporan teknisi
  - A-2 Aktor memilih untuk menampilkan laporan feedback
4. Aktor meminta sistem untuk menampilkan laporan total pemasukan
5. Sistem menampilkan laporan total pemasukan
6. Use case selesai

## 5. Alternative Flow

- A-1 Aktor memilih untuk menampilkan laporan teknisi
1. Aktor meminta sistem untuk menampilkan laporan teknisi
  2. Sistem menampilkan laporan teknisi
  3. Berlanjut ke Basic Flow langkah ke 6
- A-2 Aktor memilih untuk menampilkan laporan feedback
1. Aktor meminta sistem untuk menampilkan laporan feedback
  2. Sistem menampilkan laporan feedback
  3. Berlanjut ke Basic Flow langkah ke 6

## 6. Error Flow

none

## 7. PreConditions

1. Use case login telah dilakukan
2. Aktor telah memasuki sistem

## 8. PostConditions

Aktor telah melakukan monitoring laporan

### 4.1.12 Use case Spesification : Menanggapi Feedback

#### 1. Brief Description

Use Case ini digunakan oleh aktor untuk menanggapi feedback

#### 2. Primary Actor

Pemilik

#### 3. Supporting Actor

Pelanggan

#### 4. Basic Flow

1. Use Case ini dimulai ketika pemilik memilih untuk menanggapi feedback
2. Sistem menampilkan laporan feedback
3. Pemilik menginputkan tanggapan untuk feedback
4. Sistem mengirimkan tanggapan feedback untuk pelanggan yang bersangkutan
5. Use Case selesai

#### 5. Alternative Flow

none

#### 6. Error Flow

none

#### 7. PreConditions

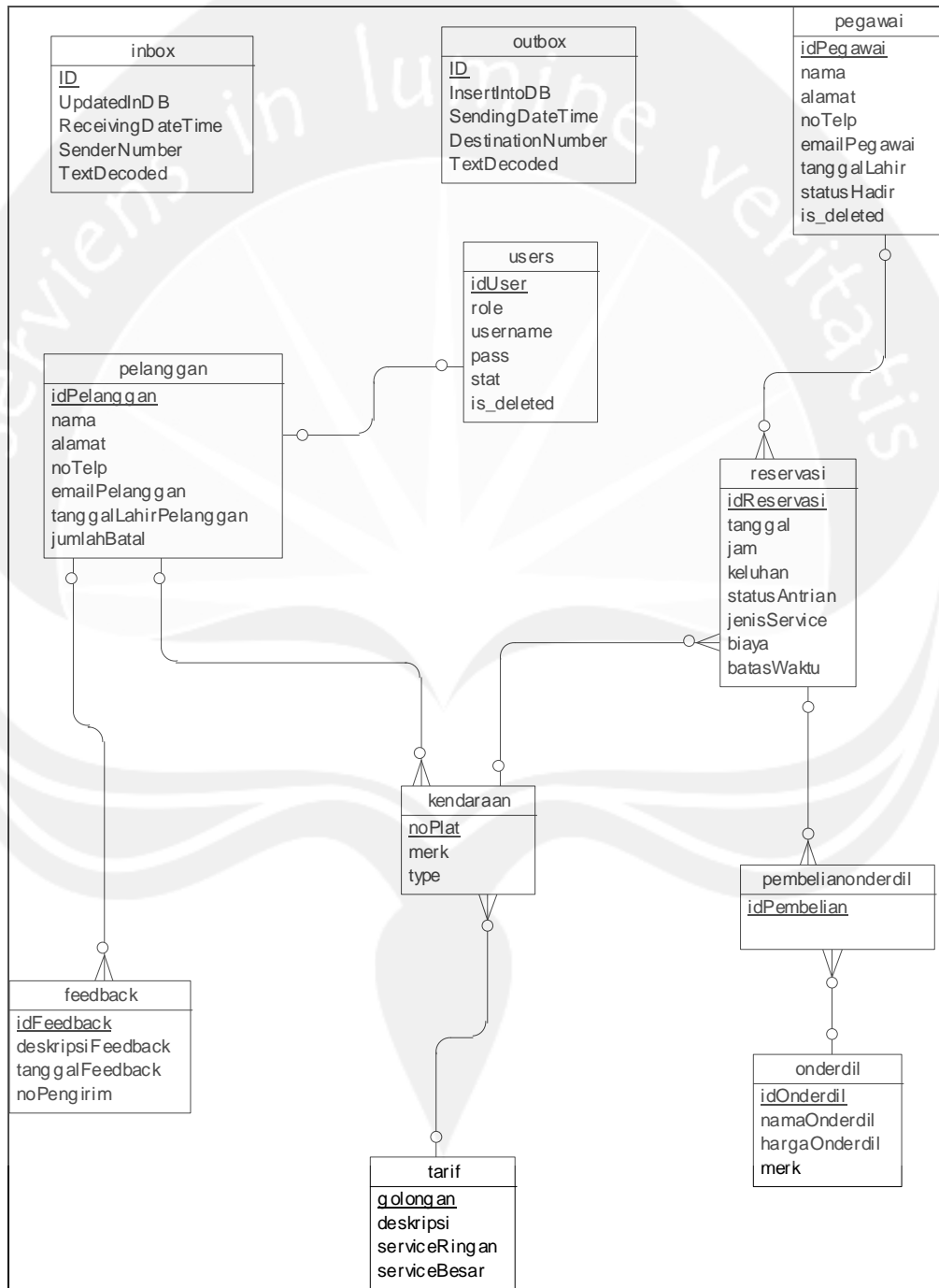
1. Use case login telah dilakukan
2. Aktor telah memasuki sistem
3. Aktor telah melihat laporan feedback

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | SKPL – CRS | 46/ 47 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

## 8. PostConditions

Pemilik telah menanggapi feedback yang dikirimkan oleh pelanggan yang bersangkutan

## 5 Entity Relationship Diagram (ERD)



Gambar 3. Entity Relationship Diagram

# DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

CRS

(Chrisna Reservation System)

Untuk :


Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Robby Widyahartono / 5217

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

|   |                                  |                 |  |         |
|---|----------------------------------|-----------------|--|---------|
|  | Program Studi Teknik Informatika | Nomor Dokumen   |  | Halaman |
|   | Fakultas Teknologi Industri      | <b>DPPL-CRS</b> |  | 1/92    |
|   |                                  | Revisi          |  |         |

## DAFTAR PERUBAHAN

| Revisi   | Deskripsi |
|----------|-----------|
| <b>A</b> |           |
| <b>B</b> |           |
| <b>C</b> |           |
| <b>D</b> |           |
| <b>E</b> |           |
| <b>F</b> |           |

| INDEX<br>TGL      | - | A | B | C | D | E | F | G |
|-------------------|---|---|---|---|---|---|---|---|
| Ditulis<br>oleh   |   |   |   |   |   |   |   |   |
| Diperiksa<br>oleh |   |   |   |   |   |   |   |   |
| Disetujui<br>oleh |   |   |   |   |   |   |   |   |

## Daftar Halaman Perubahan

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

# Daftar Isi

|   |           |
|---|-----------|
| <b>1. PENDAHULUAN.....</b>                                  | <b>7</b>  |
| 1.1 TUJUAN.....   | 7         |
| 1.2 RUANG LINGKUP.....                                      | 7         |
| 1.3 DEFINISI DAN AKRONIM.....                               | 8         |
| 1.4 REFERENSI.....  | 9         |
| <b>2. PERANCANGAN SISTEM.....</b>                           | <b>10</b> |
| 2.1 PERANCANGAN ARSITEKTUR.....                             | 10        |
| 2.2 PERANCANGAN RINCI.....                                  | 11        |
| 2.2.1 <i>Sequence Diagram</i> .....                         | 11        |
| 2.2.2 <i>Class Diagram</i> .....                            | 33        |
| 2.2.3 <i>Class Diagram Specific Descriptions</i> .....      | 34        |
| <b>3. PERANCANGAN DATA.....</b>                             | <b>56</b> |
| 3.1 DEKOMPOSISI DATA.....                                   | 56        |
| 3.1.1 <i>Deskripsi Entitas Data user</i> .....              | 56        |
| 3.1.2 <i>Deskripsi Entitas Data pegawai</i> .....           | 56        |
| 3.1.3 <i>Deskripsi Entitas Data pelanggan</i> .....         | 57        |
| 3.1.4 <i>Deskripsi Entitas Data kendaraan</i> .....         | 57        |
| 3.1.5 <i>Deskripsi Entitas Data reservasi</i> .....         | 57        |
| 3.1.6 <i>Deskripsi Entitas Data tarif</i> .....             | 58        |
| 3.1.7 <i>Deskripsi Entitas Data onderdil</i> .....          | 58        |
| 3.1.8 <i>Deskripsi Entitas Data pembelianonderdil</i> ..... | 58        |
| 3.1.9 <i>Deskripsi Entitas Data feedback</i> .....          | 59        |
| 3.1.10 <i>Deskripsi Entitas Data inbox</i> .....            | 59        |
| 3.1.11 <i>Deskripsi Entitas Data outbox</i> .....           | 59        |
| 3.2 PHYSICAL DATA MODEL.....                                | 60        |
| <b>4. PERANCANGAN ANTARMUKA.....</b>                        | <b>61</b> |
| 4.1 LOGIN USER.....   | 61        |
| 4.2 HALAMAN UTAMA.....                                      | 62        |
| 4.3 PENGELOLAAN USER.....                                   | 62        |
| 4.4 PENGELOLAAN TEKNISI.....                                | 66        |
| 4.5 PENGELOLAAN PELANGGAN.....                              | 69        |
| 4.6 PENGELOLAAN KENDARAAN.....                              | 72        |
| 4.7 PENGELOLAAN RESERVASI.....                              | 75        |
| 4.8 PENGELOLAAN ONDERDIL.....                               | 78        |
| 4.9 PENGELOLAAN TARIF.....                                  | 81        |
| 4.10 UBAH PASSWORD.....                                     | 84        |
| 4.11 LOGIN MOBILE WEB.....                                  | 85        |
| 4.12 HALAMAN UTAMA MOBILE WEB.....                          | 86        |
| 4.13 LAPORAN MOBILE WEB.....                                | 86        |
| 4.14 BALAS FEEDBACK MOBILE WEB.....                         | 89        |
| 4.15 UBAH PASSWORD MOBILE WEB.....                          | 90        |
| <b>5. PERANCANGAN FORMAT PENULISAN SMS.....</b>             | <b>91</b> |
| 5.1 DAFTAR.....   | 91        |
| 5.2 BATAL.....  | 91        |
| 5.3 FEEDBACK.....   | 92        |
| 5.4 UBAH PASSWORD.....                                      | 92        |

## Daftar Gambar

|   |    |
|---|----|
| Gambar 2.1 Rancangan Arsitektur CRS.....                        | 10 |
| Gambar 2.2 Sequence Diagram : Login User.....                   | 11 |
| Gambar 2.3 Sequence Diagram : Tambah Data User.....             | 11 |
| Gambar 2.4 Sequence Diagram : Ubah Status User.....             | 12 |
| Gambar 2.5 Sequence Diagram : Hapus Data User.....              | 12 |
| Gambar 2.6 Sequence Diagram : Tampil Data User.....             | 13 |
| Gambar 2.7 Sequence Diagram : Cari Data User.....               | 13 |
| Gambar 2.8 Sequence Diagram : Tambah Data Teknisi.....          | 14 |
| Gambar 2.9 Sequence Diagram : Ubah Data Teknisi.....            | 14 |
| Gambar 2.10 Sequence Diagram : Hapus Data Teknisi.....          | 15 |
| Gambar 2.11 Sequence Diagram : Tampil Data Teknisi.....         | 15 |
| Gambar 2.12 Sequence Diagram : Cari Data Teknisi.....           | 16 |
| Gambar 2.13 Sequence Diagram : Tambah Data Pelanggan.....       | 16 |
| Gambar 2.14 Sequence Diagram : Ubah Data Pelanggan.....         | 17 |
| Gambar 2.15 Sequence Diagram : Tampil Data Pelanggan.....       | 17 |
| Gambar 2.16 Sequence Diagram : Cari Data Pelanggan.....         | 18 |
| Gambar 2.17 Sequence Diagram : Tambah data Kendaraan.....       | 18 |
| Gambar 2.18 Sequence Diagram : Ubah Data Kendaraan.....         | 19 |
| Gambar 2.19 Sequence Diagram : Tampil Data Kendaraan.....       | 19 |
| Gambar 2.20 Sequence Diagram : Cari Data Kendaraan.....         | 20 |
| Gambar 2.21 Sequence Diagram : Tambah Data Reservasi.....       | 20 |
| Gambar 2.22 Sequence Diagram : Ubah Data Reservasi.....         | 21 |
| Gambar 2.23 Sequence Diagram : Hapus Data Reservasi.....        | 21 |
| Gambar 2.24 Sequence Diagram : Tampil Data Reservasi.....       | 22 |
| Gambar 2.25 Sequence Diagram : Cari Data Reservasi.....         | 22 |
| Gambar 2.26 Sequence Diagram : Tambah Data Onderdil.....        | 23 |
| Gambar 2.27 Sequence Diagram : Ubah Data Onderdil.....          | 23 |
| Gambar 2.28 Sequence Diagram : Tampil Data Onderdil.....        | 24 |
| Gambar 2.29 Sequence Diagram : Cari Data Onderdil.....          | 24 |
| Gambar 2.30 Sequence Diagram : Tambah Data Tarif.....           | 25 |
| Gambar 2.31 Sequence Diagram : Ubah Data Tarif.....             | 25 |
| Gambar 2.32 Sequence Diagram : Tampil Data Tarif.....           | 26 |
| Gambar 2.33 Sequence Diagram : Ubah Password.....               | 26 |
| Gambar 2.34 Sequence Diagram : Laporan Total Pemasukan Harian.. | 27 |
| Gambar 2.35 Sequence Diagram : Laporan Total Pemasukan Bulanan. | 27 |
| Gambar 2.36 Sequence Diagram : Laporan Teknisi Harian.....      | 28 |
| Gambar 2.37 Sequence Diagram : Laporan Teknisi Bulanan.....     | 28 |
| Gambar 2.38 Sequence Diagram : Laporan Feedback Harian.....     | 29 |
| Gambar 2.39 Sequence Diagram : Laporan Feedback Bulanan.....    | 29 |
| Gambar 2.40 Sequence Diagram : Ubah Password Mobile Web.....    | 30 |
| Gambar 2.41 Sequence Diagram : Balas Feedback.....              | 30 |
| Gambar 2.42 Sequence Diagram : Daftar.....                      | 31 |
| Gambar 2.43 Sequence Diagram : Batal.....                       | 31 |
| Gambar 2.44 Sequence Diagram : Feedback.....                    | 32 |
| Gambar 2.45 Sequence Diagram : Ubah Password.....               | 32 |
| Gambar 2.46 Class Diagram.....                                  | 33 |
| Gambar 3.1 Physical Data Model.....                             | 60 |
| Gambar 4.1 Rancangan Antarmuka Login User.....                  | 61 |
| Gambar 4.2 Rancangan Halaman Utama.....                         | 62 |
| Gambar 4.3 Rancangan Antarmuka Pengelolaan User.....            | 62 |



|             |  |    |
|-------------|--|----|
| Gambar 4.4  | Rancangan Antarmuka Tambah Data User.....            | 64 |
| Gambar 4.5  | Rancangan Antarmuka Ubah Status User.....            | 65 |
| Gambar 4.6  | Rancangan Antarmuka Pengelolaan Teknisi.....         | 66 |
| Gambar 4.7  | Rancangan Antarmuka Tambah Data Teknisi.....         | 67 |
| Gambar 4.8  | Rancangan Antarmuka Ubah Data Teknisi.....           | 68 |
| Gambar 4.9  | Rancangan Antarmuka Pengelolaan Pelanggan.....       | 69 |
| Gambar 4.10 | Rancangan Antarmuka Tambah Data Pelanggan.....       | 70 |
| Gambar 4.11 | Rancangan Antarmuka Ubah Data Pelanggan.....         | 71 |
| Gambar 4.12 | Rancangan Antarmuka Pengelolaan Kendaraan.....       | 72 |
| Gambar 4.13 | Rancangan Antarmuka Tambah Data Kendaraan.....       | 73 |
| Gambar 4.14 | Rancangan Antarmuka Ubah Data Kendaraan.....         | 74 |
| Gambar 4.15 | Rancangan Antarmuka Pengelolaan Reservasi.....       | 75 |
| Gambar 4.16 | Rancangan Antarmuka Tambah Data Reservasi.....       | 76 |
| Gambar 4.17 | Rancangan Antarmuka Ubah Data Reservasi.....         | 77 |
| Gambar 4.18 | Rancangan Antarmuka Pengelolaan Onderdil.....        | 78 |
| Gambar 4.19 | Rancangan Antarmuka Tambah Data Onderdil.....        | 79 |
| Gambar 4.20 | Rancangan Antarmuka Ubah Data Onderdil.....          | 80 |
| Gambar 4.21 | Rancangan Antarmuka Pengelolaan Tarif.....           | 81 |
| Gambar 4.22 | Rancangan Antarmuka Tambah Data Tarif.....           | 82 |
| Gambar 4.23 | Rancangan Antarmuka Ubah Data Tarif.....             | 83 |
| Gambar 4.24 | Rancangan Antarmuka Ubah Password.....               | 84 |
| Gambar 4.25 | Rancangan Antarmuka Login Mobile Web.....            | 85 |
| Gambar 4.26 | Rancangan Antarmuka Halaman Utama Mobile Web.....    | 86 |
| Gambar 4.27 | Rancangan Antarmuka Laporan Mobile Web.....          | 86 |
| Gambar 4.28 | Rancangan Antarmuka Laporan Total Pemasukan.....     | 87 |
| Gambar 4.29 | Rancangan Antarmuka Laporan Teknisi Mobile Web.....  | 88 |
| Gambar 4.30 | Rancangan Antarmuka Laporan Feedback Mobile Web..... | 88 |
| Gambar 4.31 | Rancangan Antarmuka Balas Feedback Mobile Web.....   | 89 |
| Gambar 4.32 | Rancangan Antarmuka Ubah Password Mobile Web.....    | 90 |
| Gambar 5.1  | Rancangan Format Penulisan SMS Daftar.....           | 91 |
| Gambar 5.2  | Rancangan Format Penulisan SMS Batal.....            | 91 |
| Gambar 5.3  | Rancangan Format Penulisan SMS Feedback.....         | 92 |
| Gambar 5.4  | Rancangan Format Penulisan SMS Ubah Password.....    | 92 |

## 1. Pendahuluan

### 1.1 Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

### 1.2 Ruang Lingkup

Perangkat Lunak CRS dikembangkan dengan tujuan untuk mempermudah pelanggan dalam mendaftar dan membatalkan antrian *service*, serta memberikan *feedback*, mempermudah administrator dan operator dalam mengelola semua data yang berkaitan dengan reservasi antrian *service*, dan juga mempermudah pemilik dalam melakukan monitoring laporan dan menanggapi *feedback* :

1. Mempermudah administrator dalam menangani pengelolaan user.
2. Mempermudah administrator dalam menangani pengelolaan teknisi.
3. Mempermudah operator dalam menangani pengelolaan reservasi.
4. Mempermudah operator dalam menangani pengelolaan pelanggan.
5. Mempermudah operator dalam menangani pengelolaan kendaraan.
6. Mempermudah operator dalam menangani pengelolaan tarif.
7. Mempermudah operator dalam menangani pengelolaan onderdil.

|  |            |      |
|--|------------|------|
| Program Studi Teknik Informatika   | DPPL – CRS | 7/92 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |      |

8. Mempermudah pelanggan dalam mendaftar antrian service.
9. Mempermudah pelanggan dalam membatalkan antrian service.
10. Mempermudah pelanggan dalam memberikan feedback.
11. Mempermudah pemilik dalam melakukan monitoring laporan.
12. Mempermudah pemilik dalam menanggapi feedback Dan berjalan pada lingkungan dengan platform web.

### 1.3 Definisi dan Akronim

Daftar definisi akronim dan singkatan :

| Keyword/Phrase | Definisi   |
|----------------|--|
| DPPL           | Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD) merupakan deskripsi dari perancangan produk/perangkat lunak yang akan dikembangkan. |
| CRS            | Sistem Reservasi Pemesanan Antrian Service Berbasis Web dan SMS.   |
| SMS            | <i>Short Message Service</i> merupakan fitur yang digunakan untuk berkirim pesan dalam format teks.  |
| Database       | Kumpulan data terkait yang diorganisasikan dalam struktur tertentu dan dapat diakses dengan cepat.   |
| GUI            | <i>Graphical User Interface</i> , tampilan antarmuka program bagi pengguna.  |

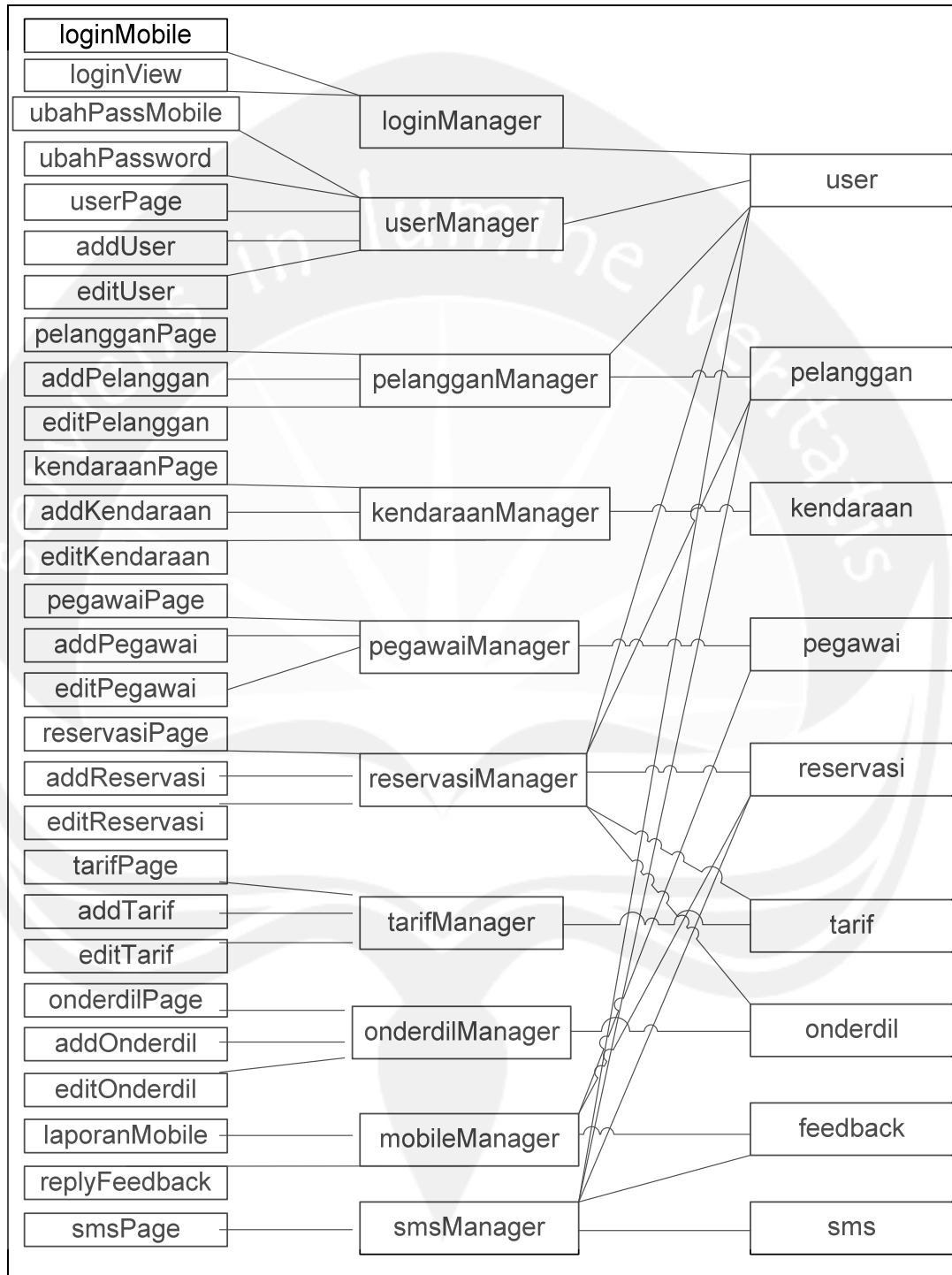
#### 1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Bennet Simon, McRobb Steve, Farmer Ray, *Object-Oriented System Analysis and Design Using UML*, McGraw-Hill Companies, 2002.
2. Boggs Wendy, Boggs Michael, *Mastering UML with Rational Rose 2002*, SYBEX Inc, 2002.
3. Widyahartono, Robby, *Spesifikasi Kebutuhan Perangkat Lunak CRS*, Universitas Atma Jaya Yogyakarta, 2010.

## 2. Perancangan Sistem

### 2.1 Perancangan Arsitektur

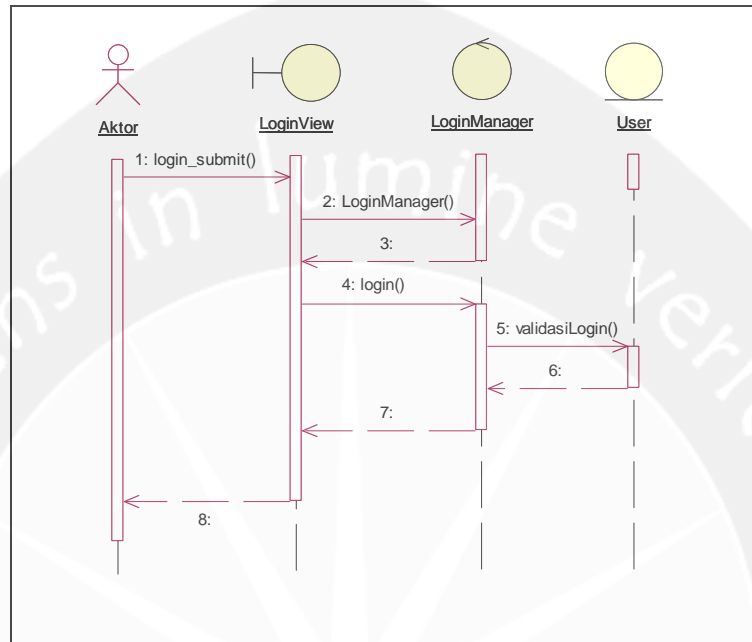


Gambar 2.1 Rancangan Arsitektur CRS

## 2.2 Perancangan Rinci

### 2.2.1 Sequence Diagram

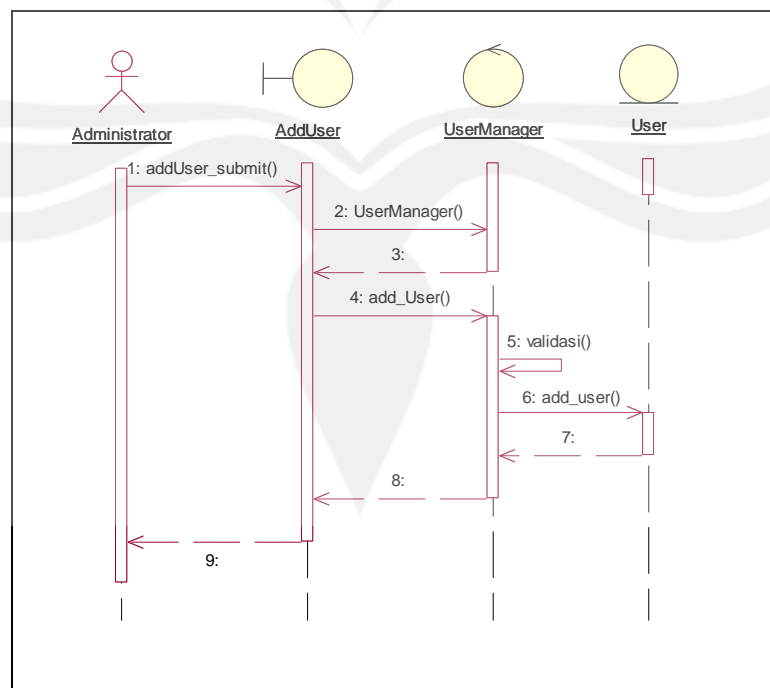
#### 2.2.1.1 Login User



Gambar 2.2 Sequence Diagram : Login User

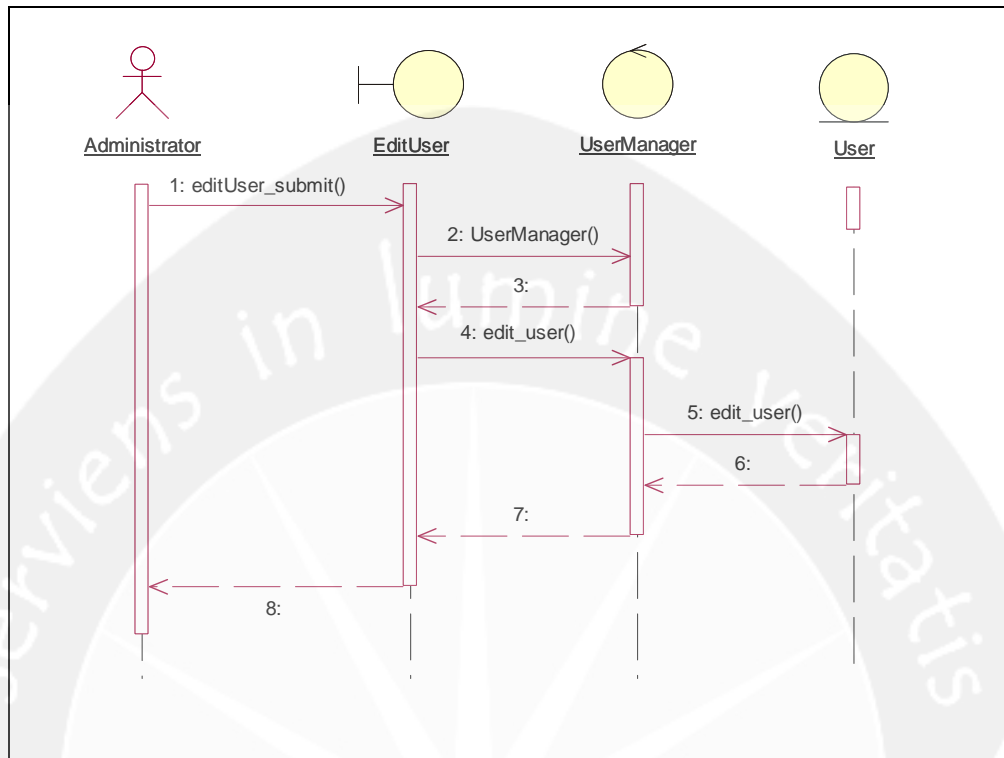
#### 2.2.1.2 Pengelolaan Data User

##### 2.2.1.2.1 Tambah Data User



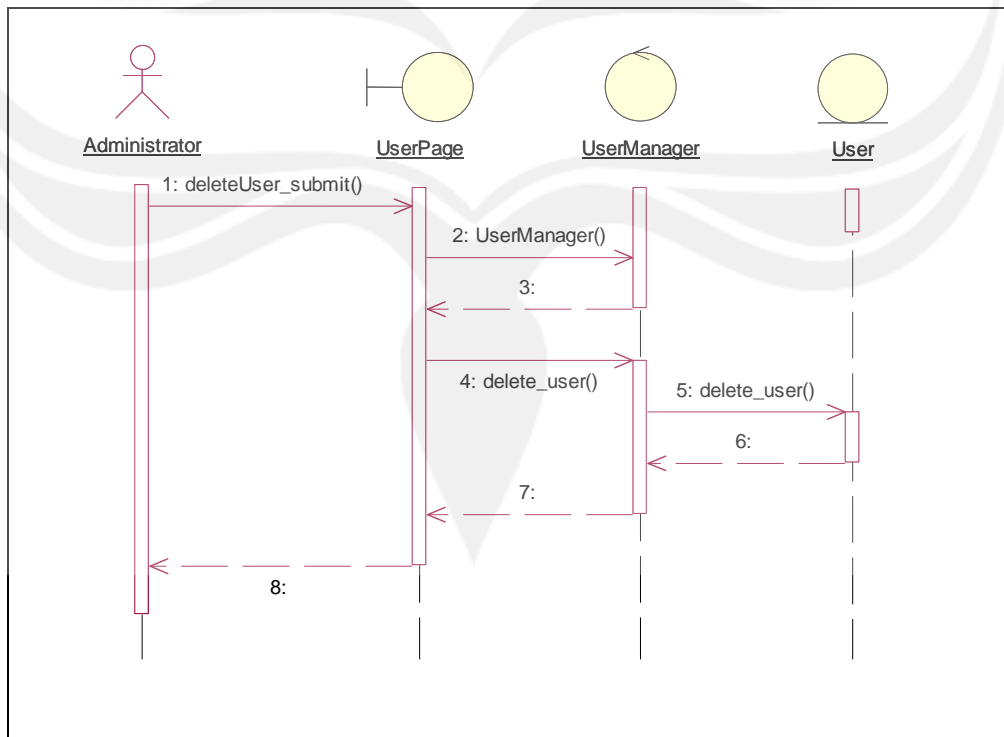
Gambar 2.3 Sequence Diagram : Tambah Data User

### 2.2.1.2.2 Ubah Status User



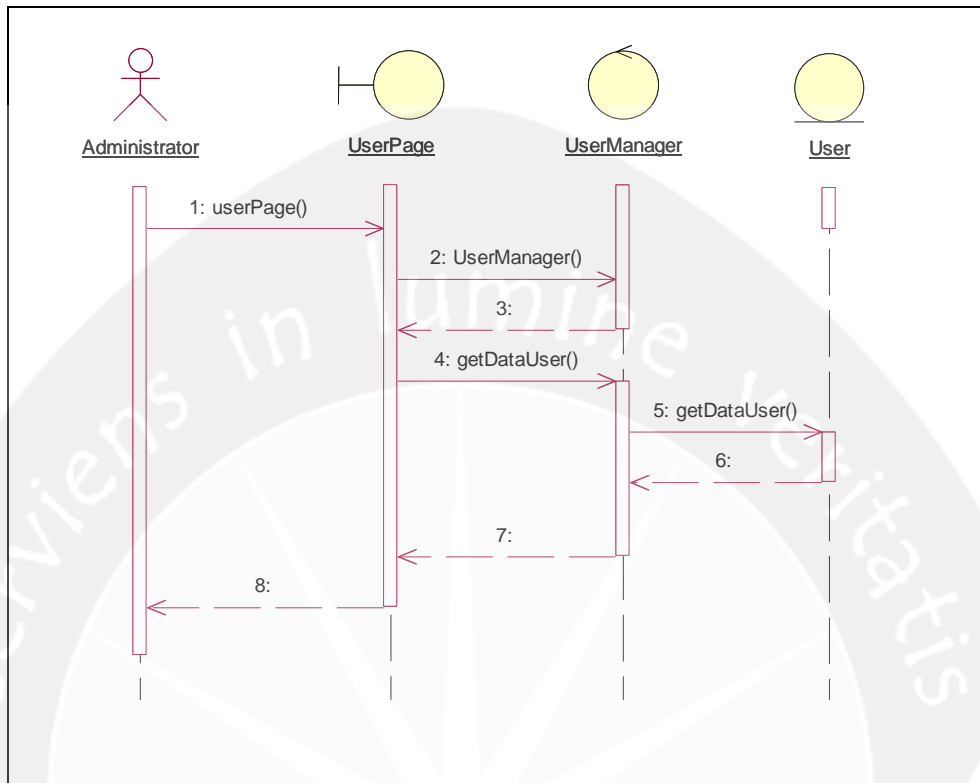
Gambar 2.4 Sequence Diagram : Ubah Status User

### 2.2.1.2.3 Hapus Data User



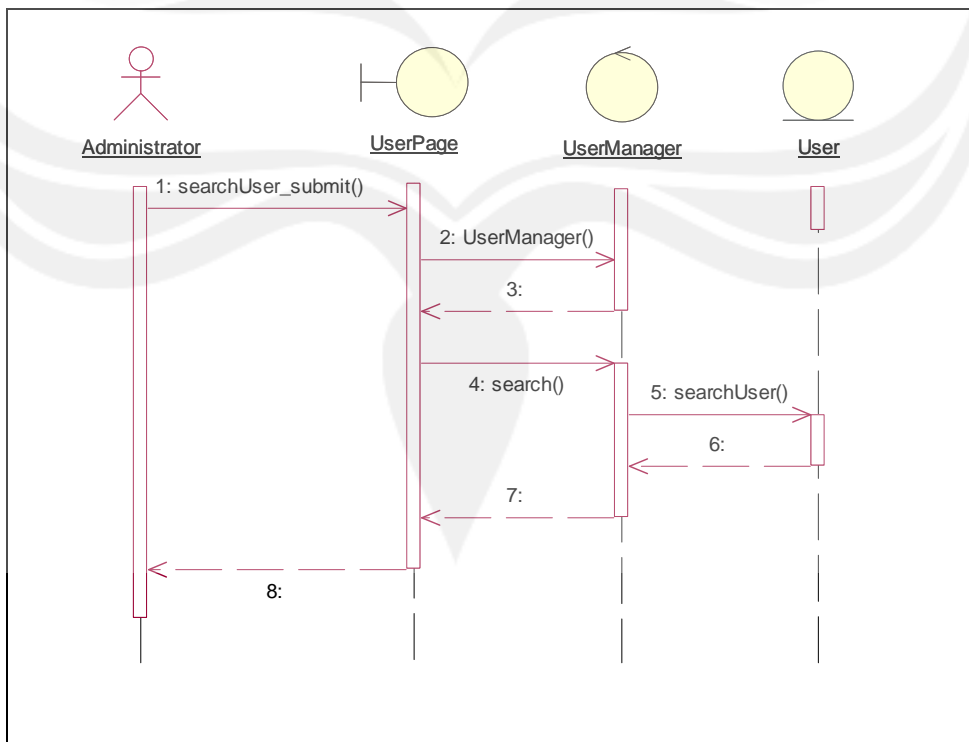
Gambar 2.5 Sequence Diagram : Hapus Data User

#### 2.2.1.2.4 Tampil Data User



Gambar 2.6 Sequence Diagram : Tampil Data User

#### 2.2.1.2.5 Cari Data User

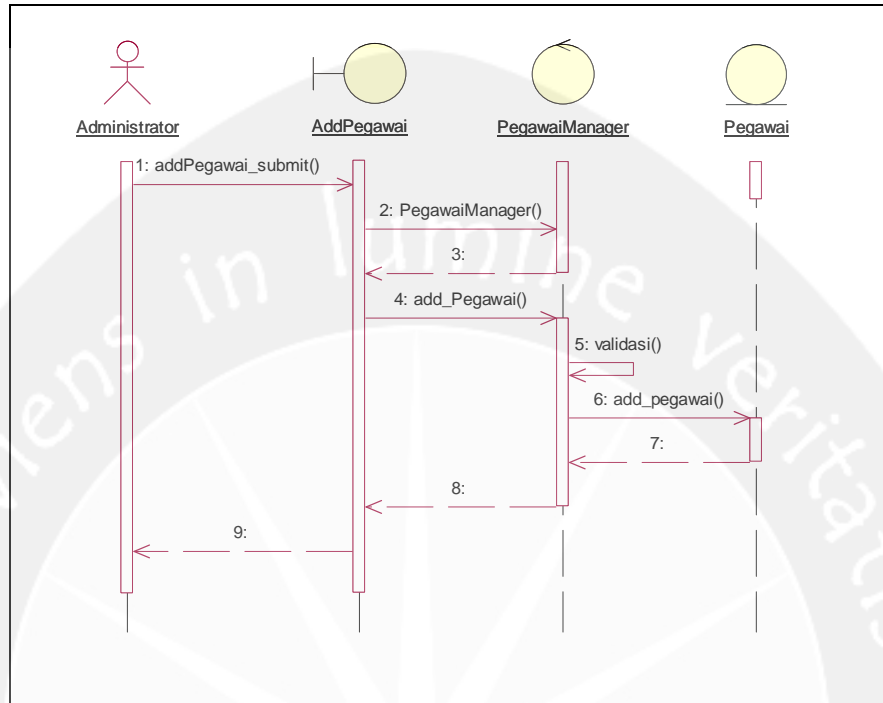


Gambar 2.7 Sequence Diagram : Cari Data User



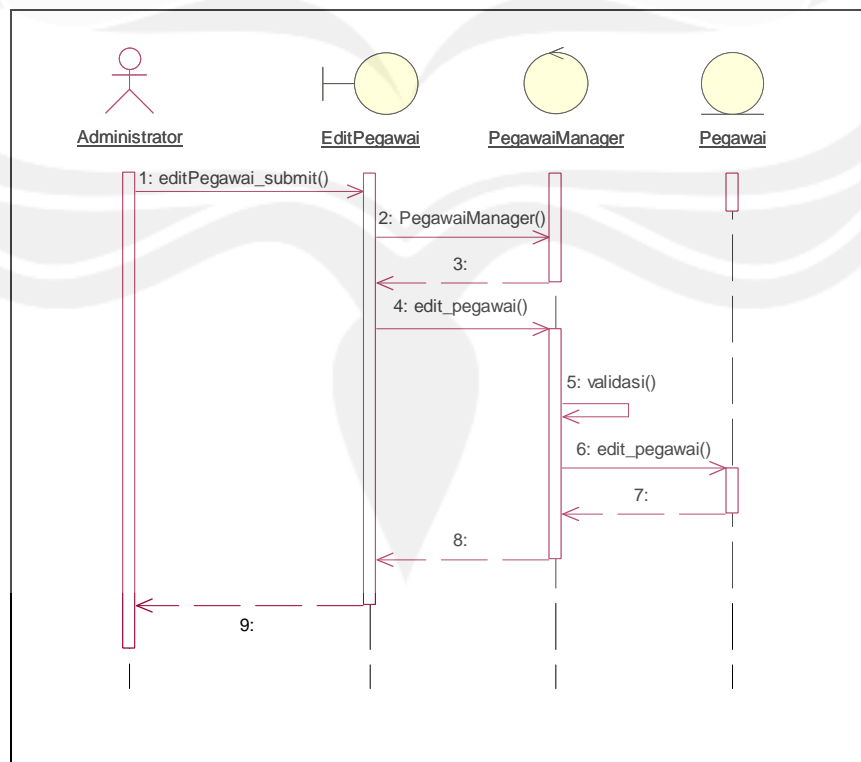
### 2.2.1.3 Pengelolaan Data Teknisi

#### 2.2.1.3.1 Tambah Data Teknisi



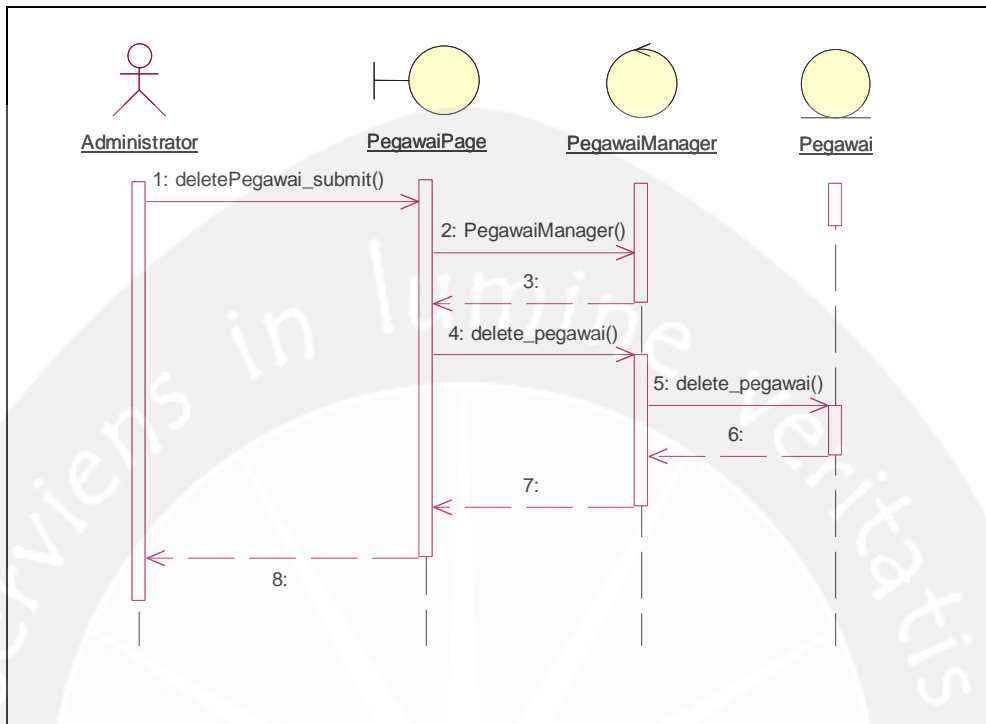
Gambar 2.8 Sequence Diagram : Tambah Data Teknisi

#### 2.2.1.3.2 Ubah Data Teknisi



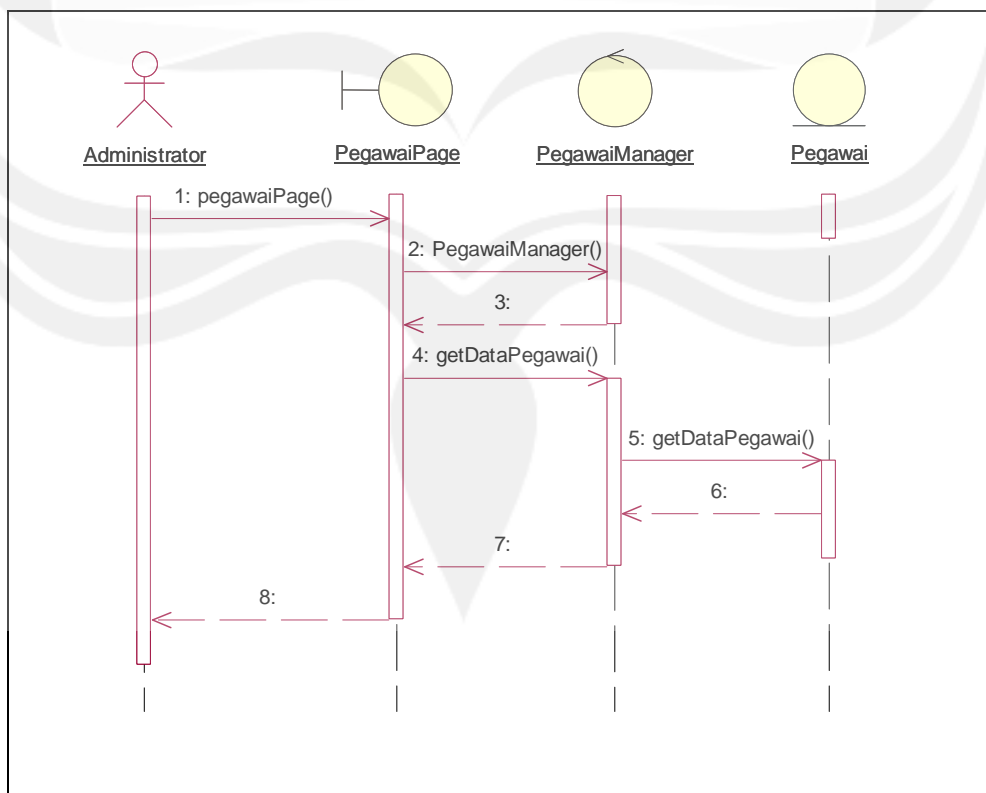
Gambar 2.9 Sequence Diagram : Ubah Data Teknisi

### 2.2.1.3.3 Hapus Data Teknisi



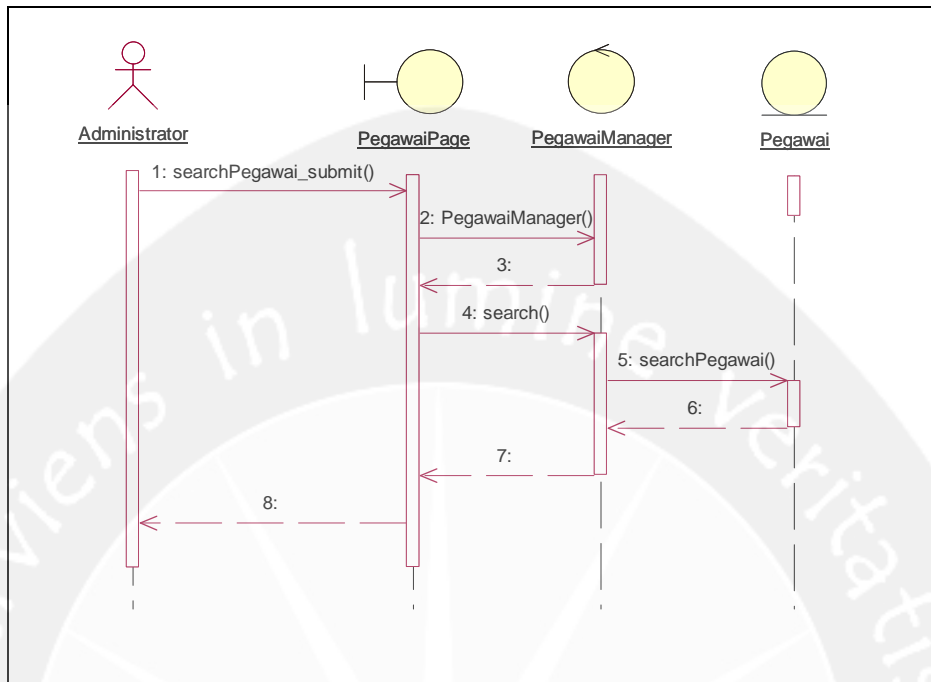
Gambar 2.10 Sequence Diagram : Hapus Data Teknisi

### 2.2.1.3.4 Tampil Data Teknisi



Gambar 2.11 Sequence Diagram : Tampil Data Teknisi

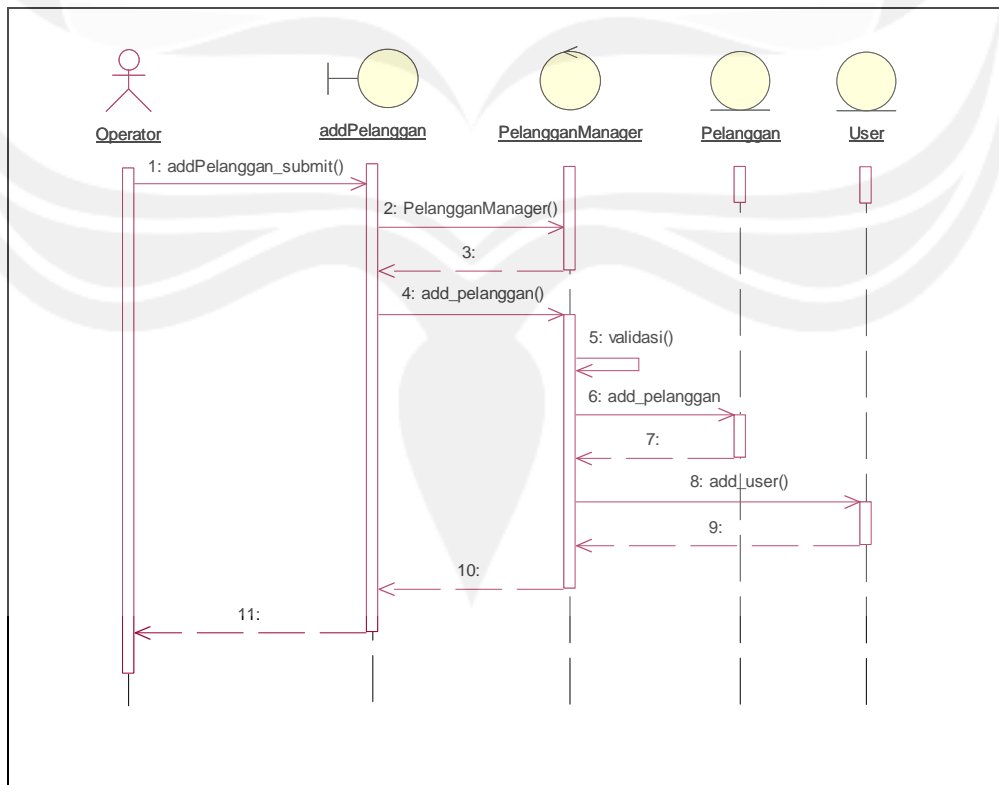
### 2.2.1.3.5 Cari Data Teknisi



Gambar 2.12 Sequence Diagram : Cari Data Teknisi

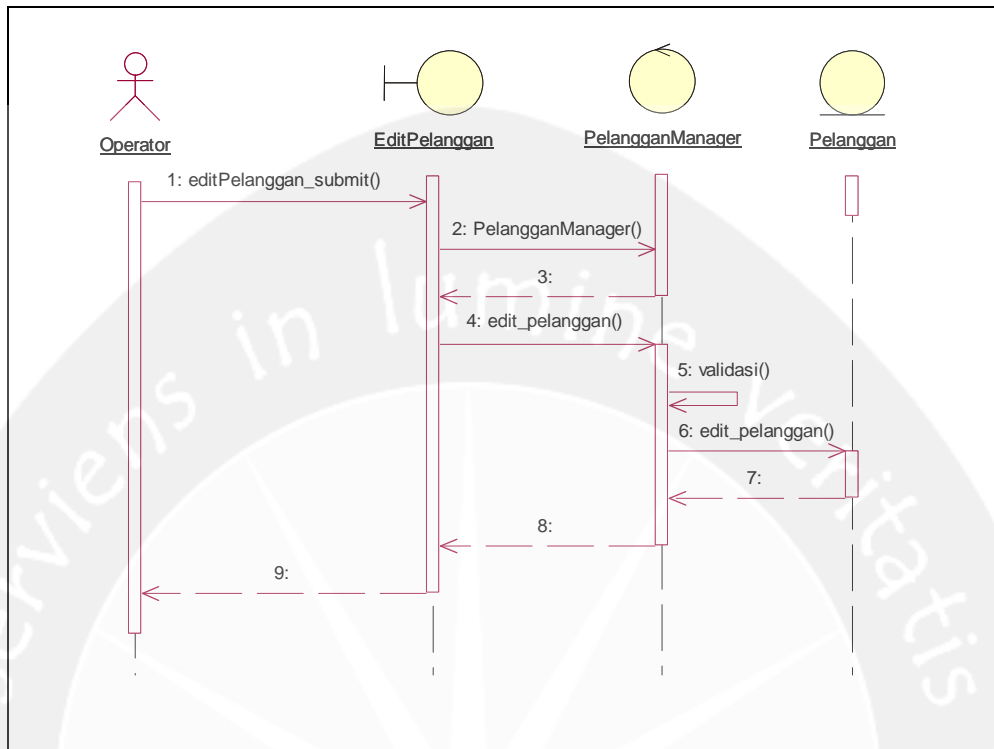
### 2.2.1.4 Pengelolaan Data Pelanggan

#### 2.2.1.4.1 Tambah Data Pelanggan



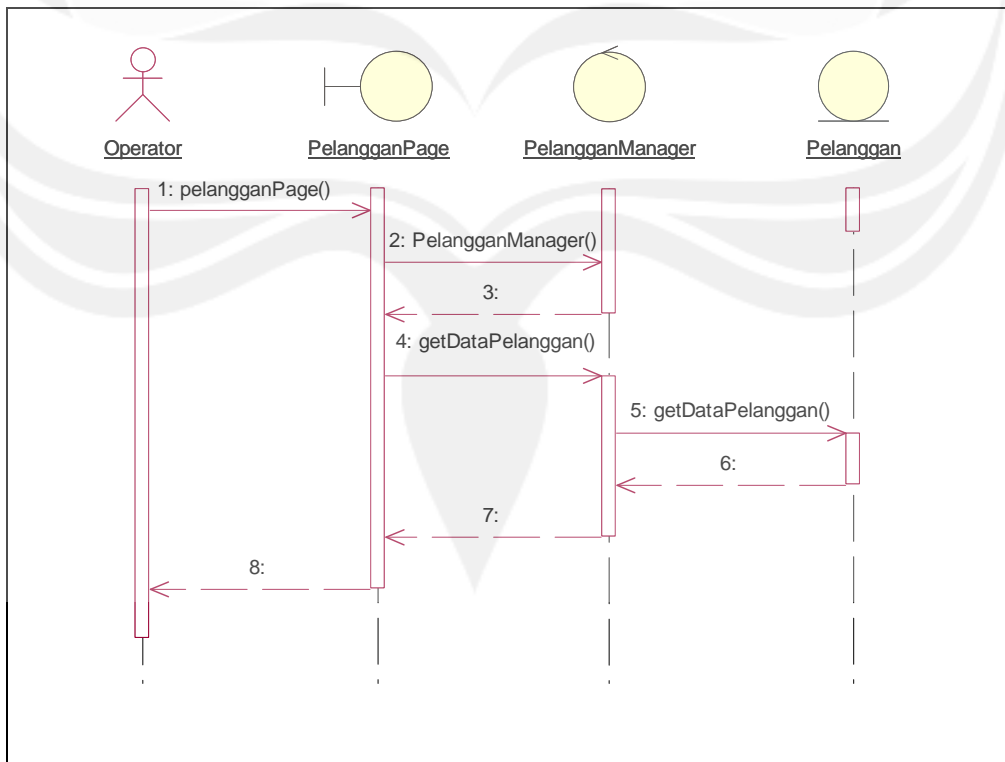
Gambar 2.13 Sequence Diagram : Tambah Data Pelanggan

### 2.2.1.4.2 Ubah Data Pelanggan



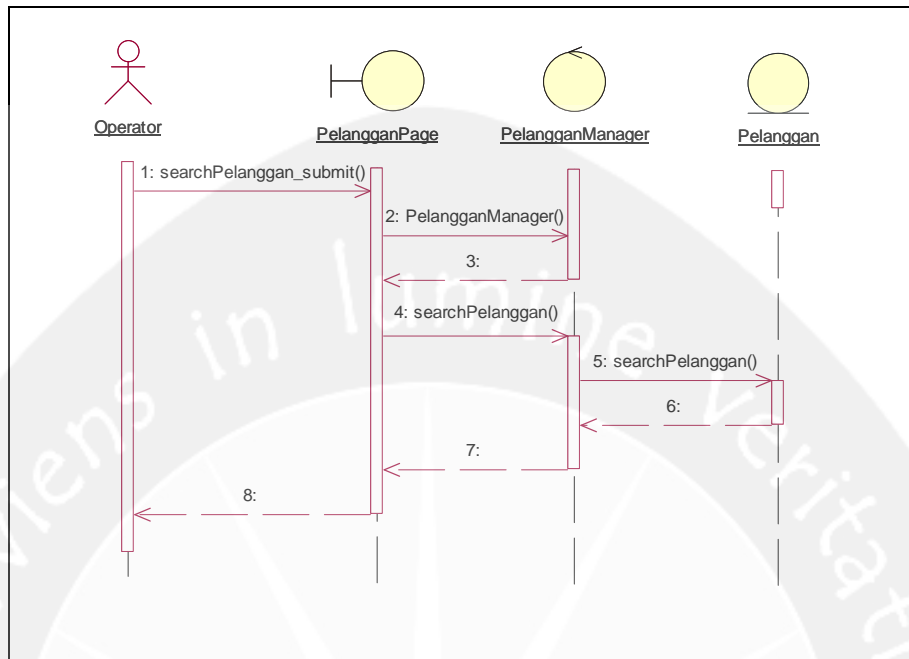
Gambar 2.14 Sequence Diagram : Ubah Data Pelanggan

### 2.2.1.4.3 Tampil Data Pelanggan



Gambar 2.15 Sequence Diagram : Tampil Data Pelanggan

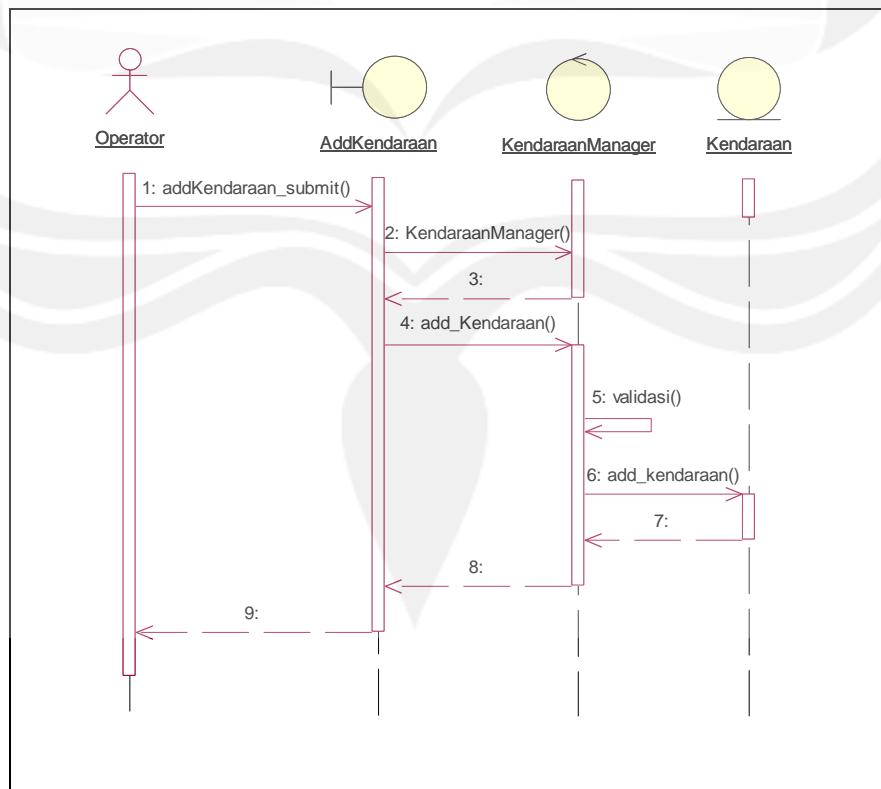
#### 2.2.1.4.4 Cari Data Pelanggan



Gambar 2.16 Sequence Diagram : Cari Data Pelanggan

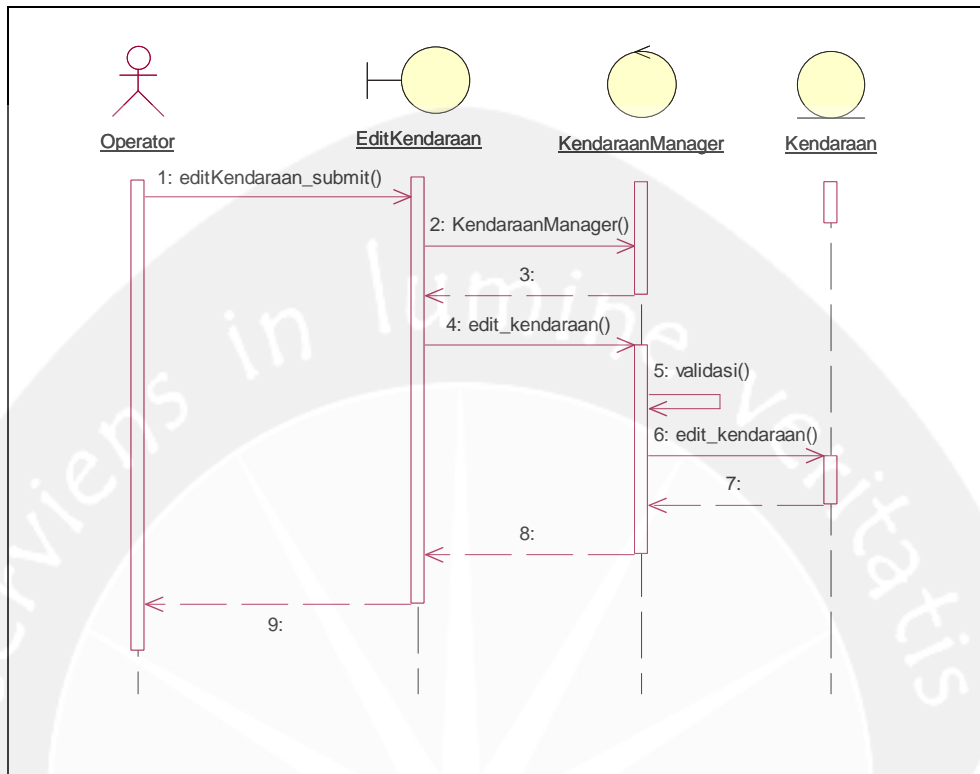
#### 2.2.1.5 Pengelolaan Data Kendaraan

##### 2.2.1.5.1 Tambah Data Kendaraan



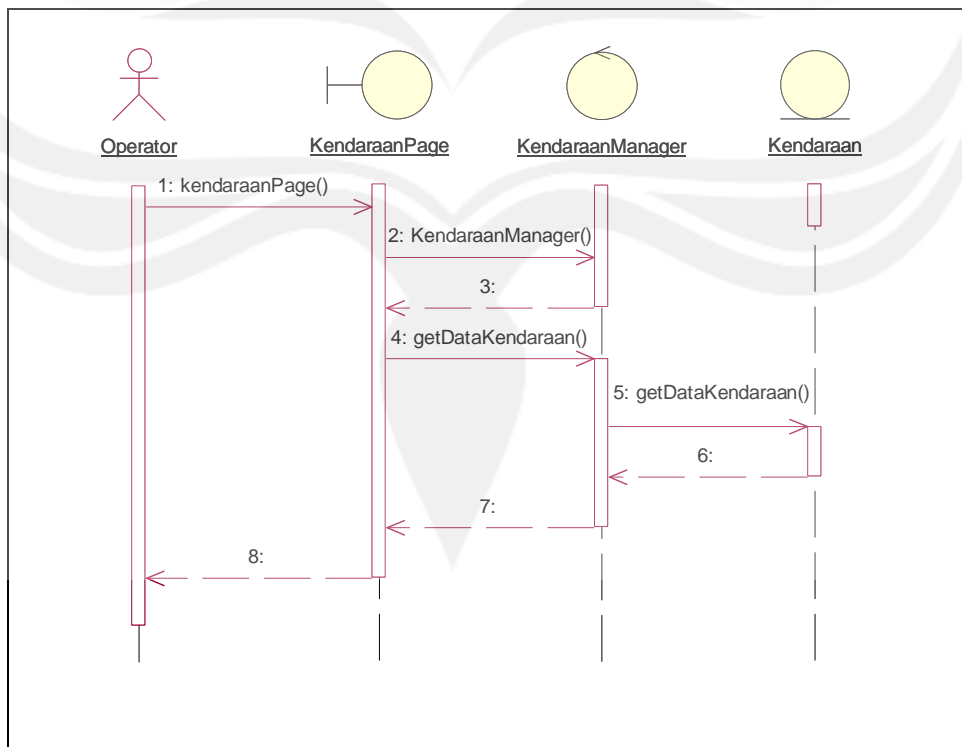
Gambar 2.17 Sequence Diagram : Tambah data Kendaraan

### 2.2.1.5.2 Ubah Data Kendaraan



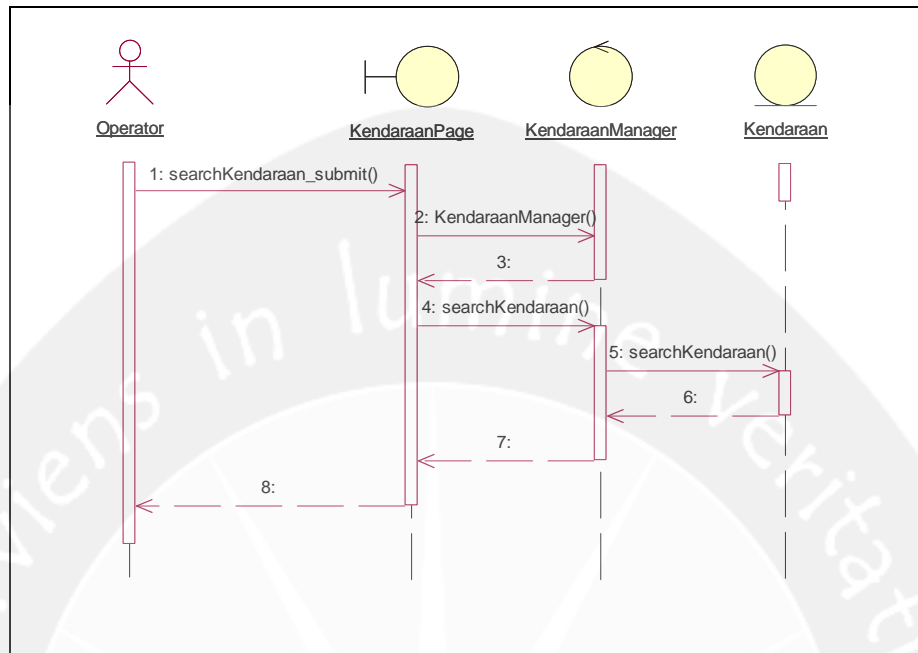
Gambar 2.18 Sequence Diagram : Ubah Data Kendaraan

### 2.2.1.5.3 Tampil Data Kendaraan



Gambar 2.19 Sequence Diagram : Tampil Data Kendaraan

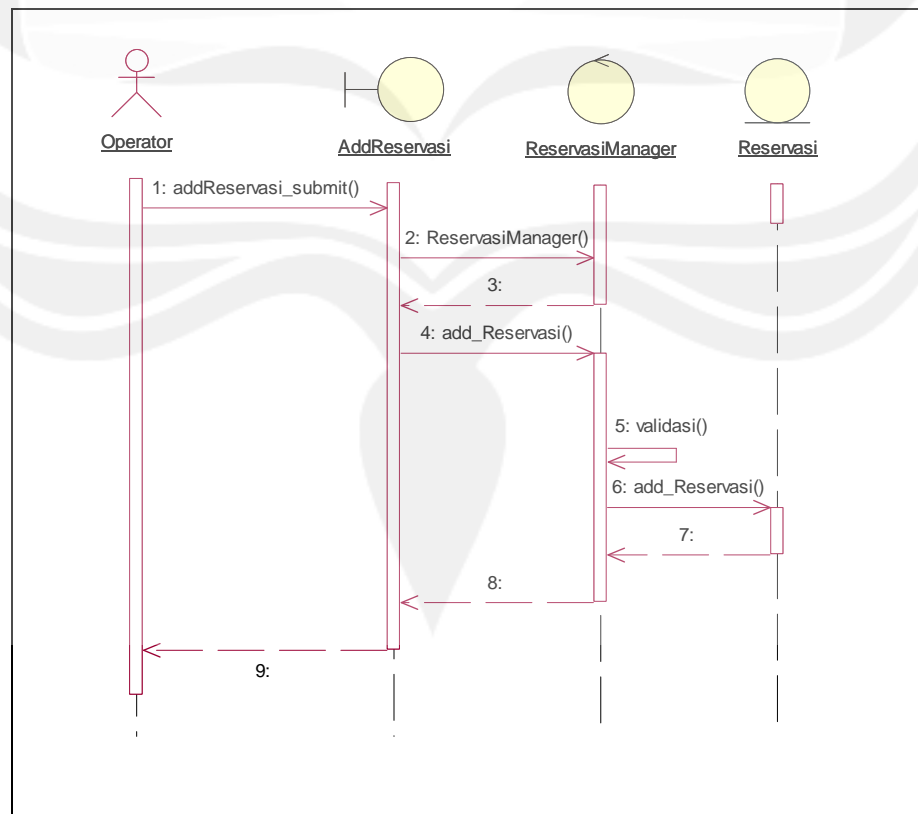
#### 2.2.1.5.4 Cari Data Kendaraan



Gambar 2.20 Sequence Diagram : Cari Data Kendaraan

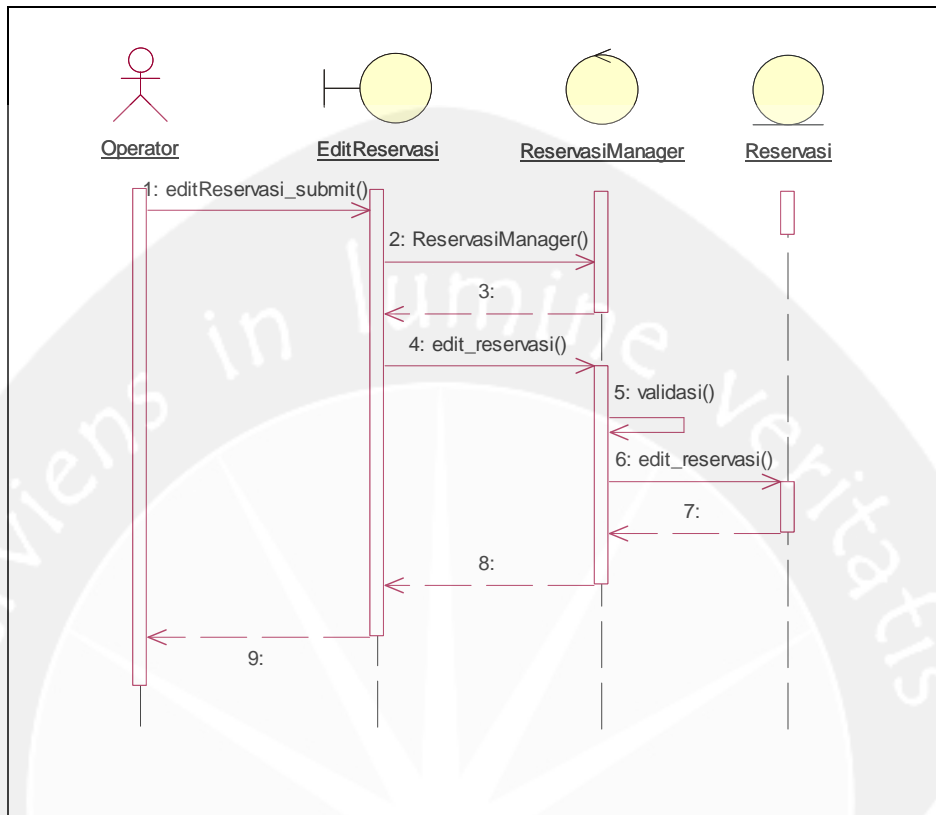
#### 2.2.1.6 Pengelolaan Data Reservasi

##### 2.2.1.6.1 Tambah Data Reservasi



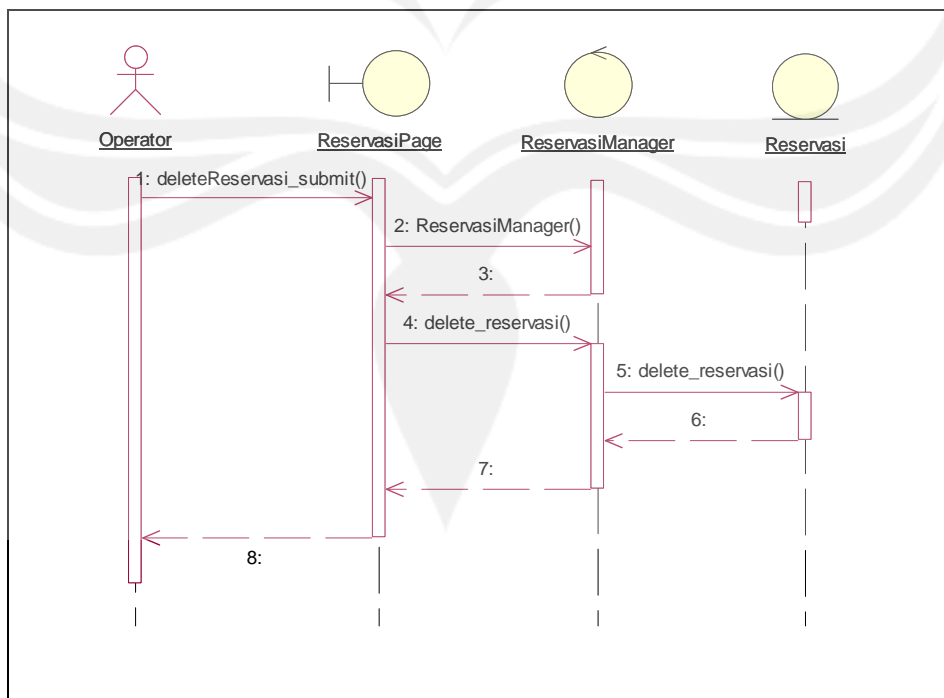
Gambar 2.21 Sequence Diagram : Tambah Data Reservasi

### 2.2.1.6.2 Ubah Data Reservasi



Gambar 2.22 Sequence Diagram : Ubah Data Reservasi

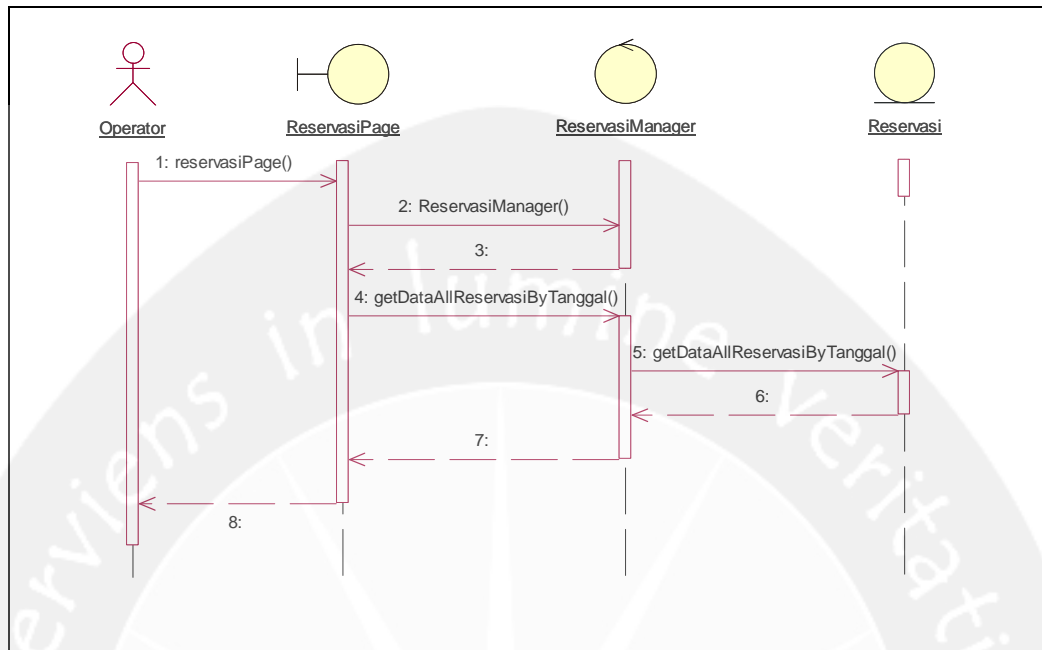
### 2.2.1.6.3 Hapus Data Reservasi



Gambar 2.23 Sequence Diagram : Hapus Data Reservasi

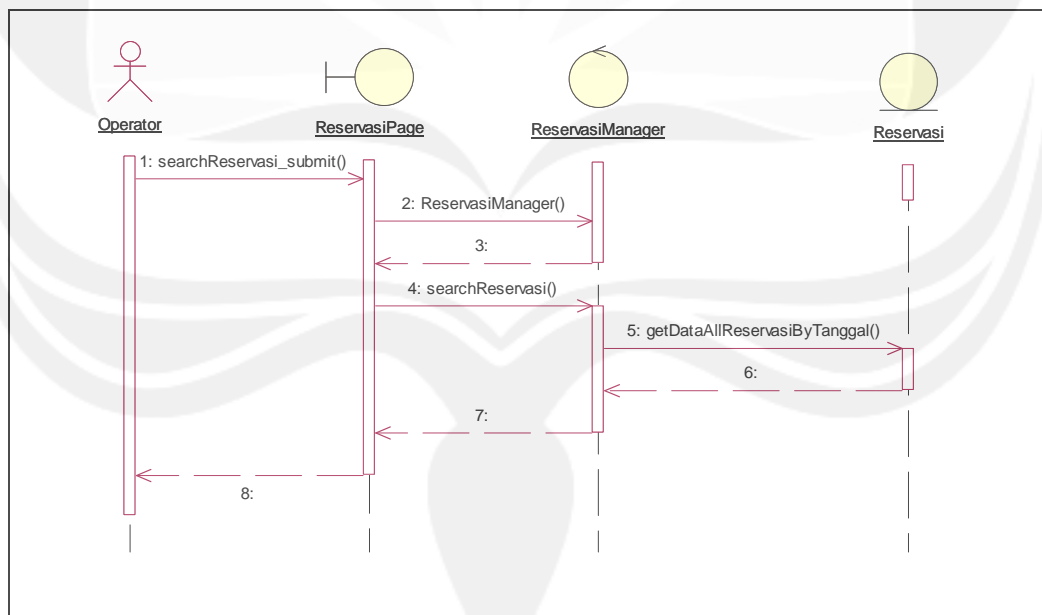


#### 2.2.1.6.4 Tampil Data Reservasi



Gambar 2.24 Sequence Diagram : Tampil Data Reservasi

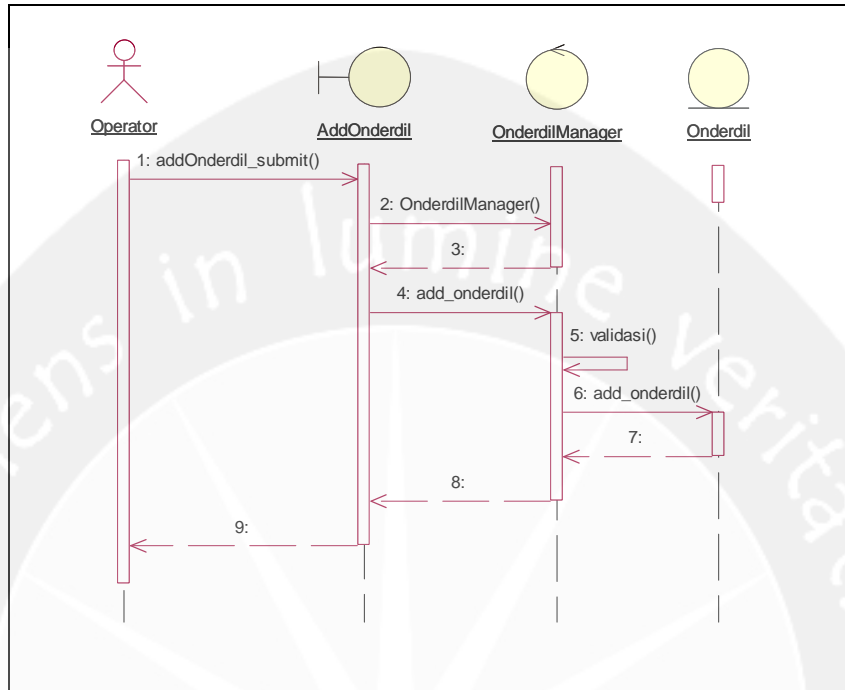
#### 2.2.1.6.5 Cari Data Reservasi



Gambar 2.25 Sequence Diagram : Cari Data Reservasi

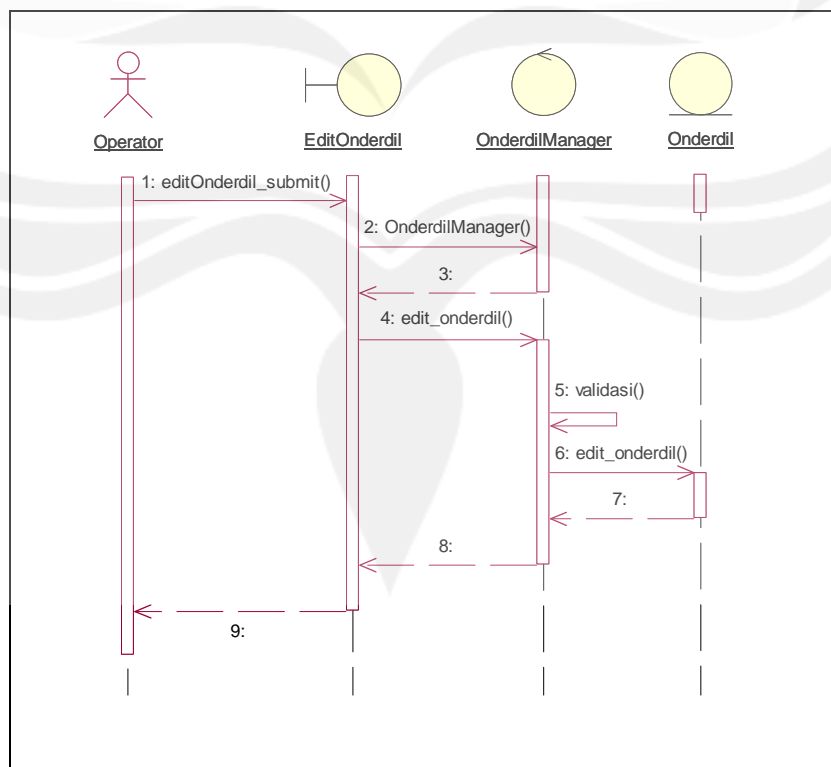
## 2.2.1.7 Pengelolaan Data Onderdil

### 2.2.1.7.1 Tambah Data Onderdil



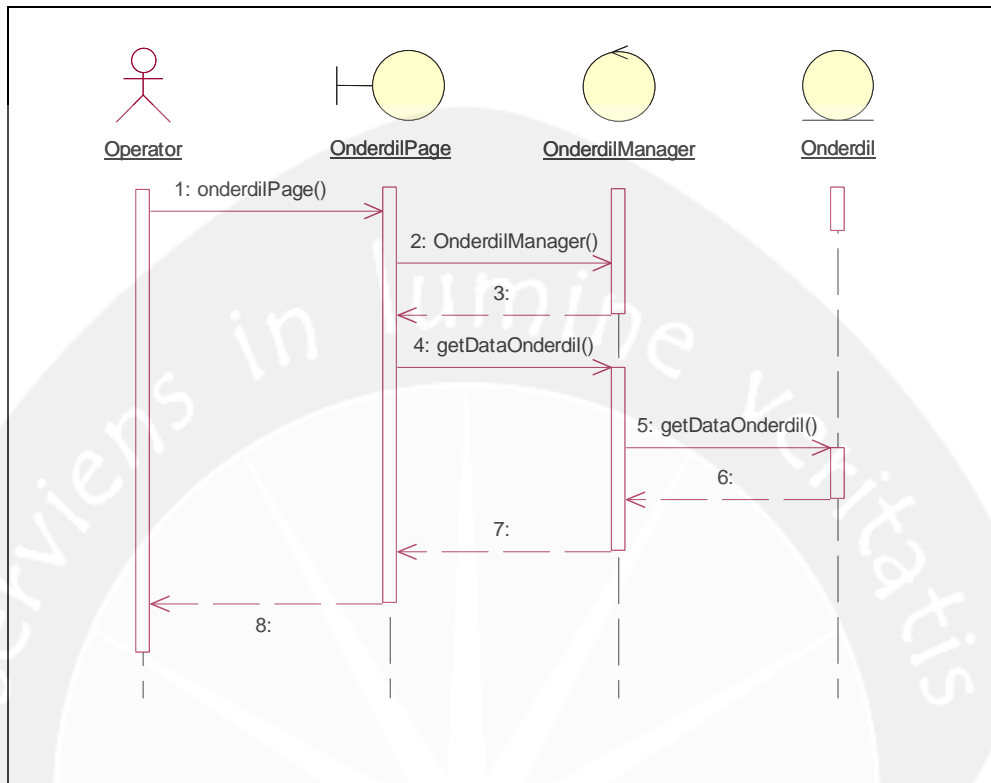
Gambar 2.26 Sequence Diagram : Tambah Data Onderdil

### 2.2.1.7.2 Ubah Data Onderdil



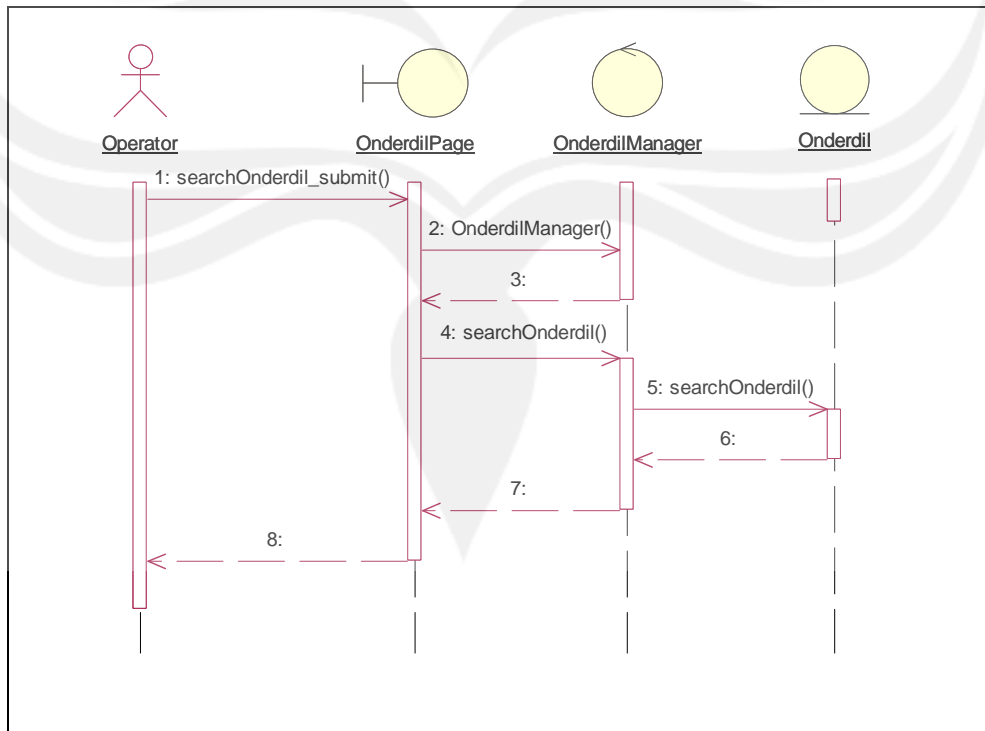
Gambar 2.27 Sequence Diagram : Ubah Data Onderdil

### 2.2.1.7.3 Tampil Data Onderdil



Gambar 2.28 Sequence Diagram : Tampil Data Onderdil

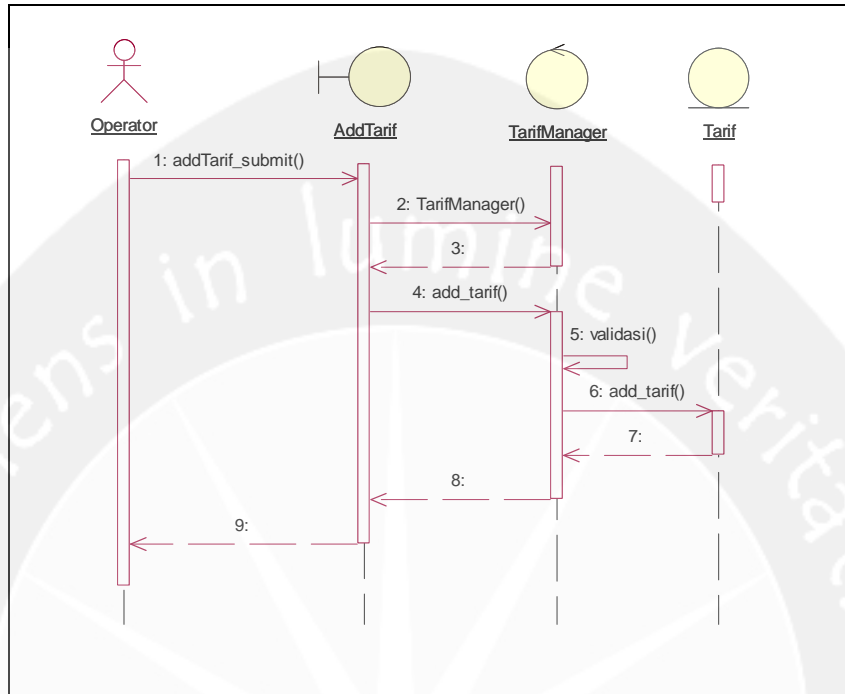
### 2.2.1.7.4 Cari Data Onderdil



Gambar 2.29 Sequence Diagram : Cari Data Onderdil

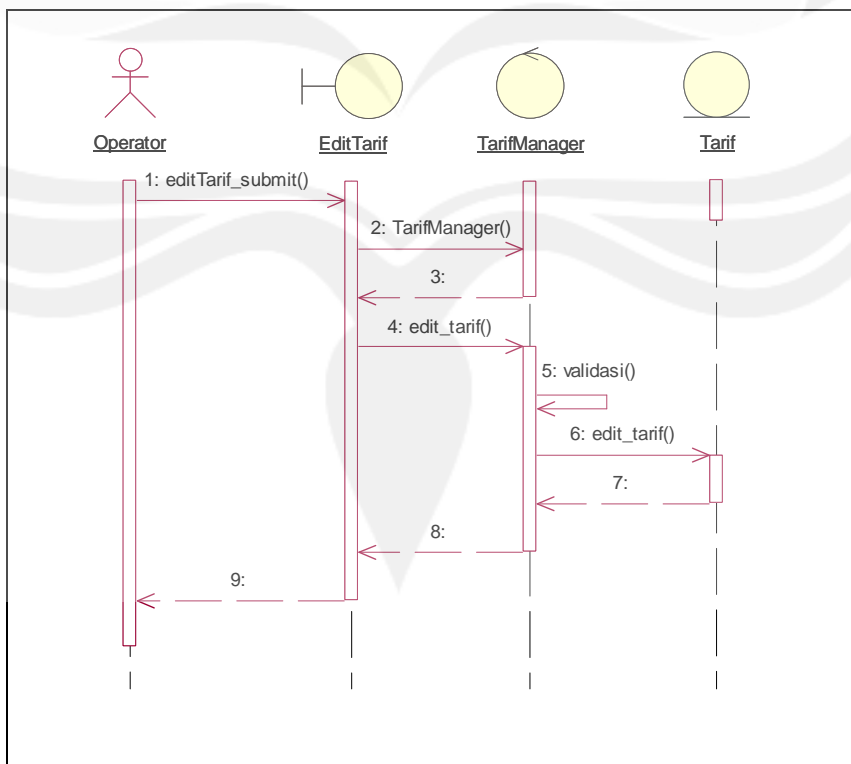
## 2.2.1.8 Pengelolaan Data Tarif

### 2.2.1.8.1 Tambah Data Tarif



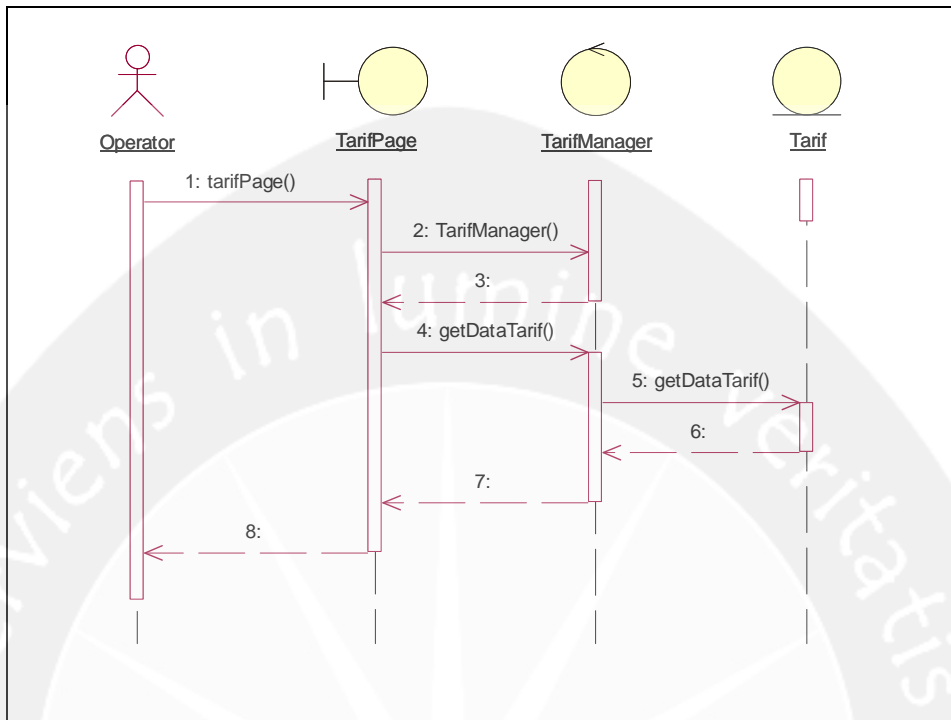
Gambar 2.30 Sequence Diagram : Tambah Data Tarif

### 2.2.1.8.2 Ubah Data Tarif



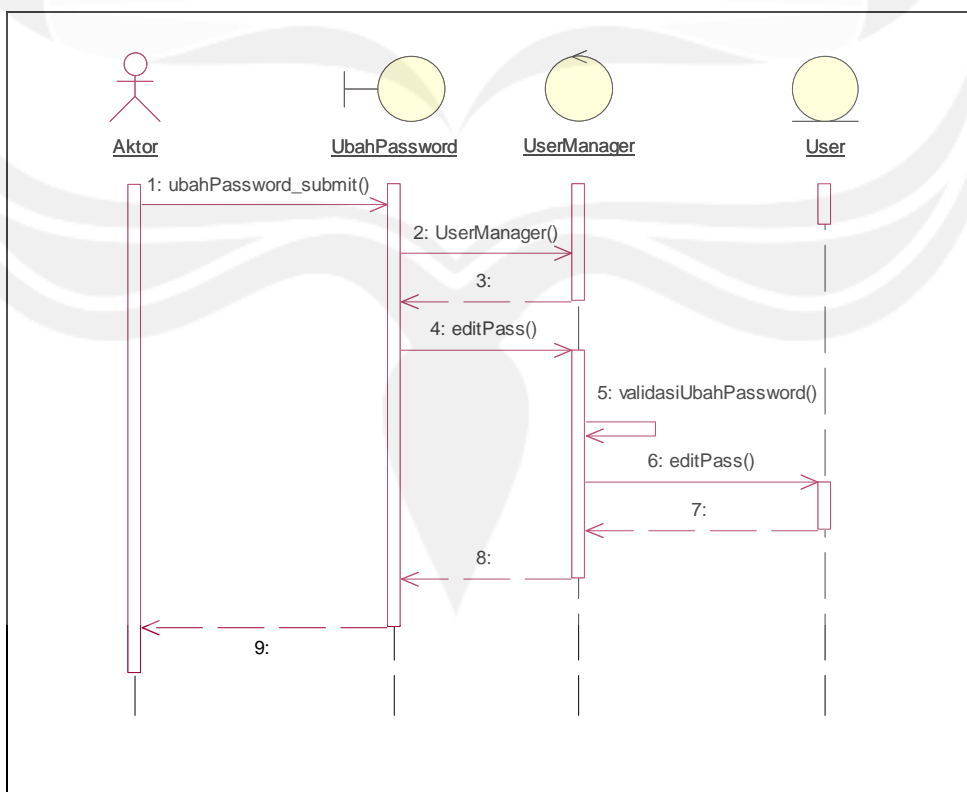
Gambar 2.31 Sequence Diagram : Ubah Data Tarif

### 2.2.1.8.3 Tampil Data Tarif



Gambar 2.32 Sequence Diagram : Tampil Data Tarif

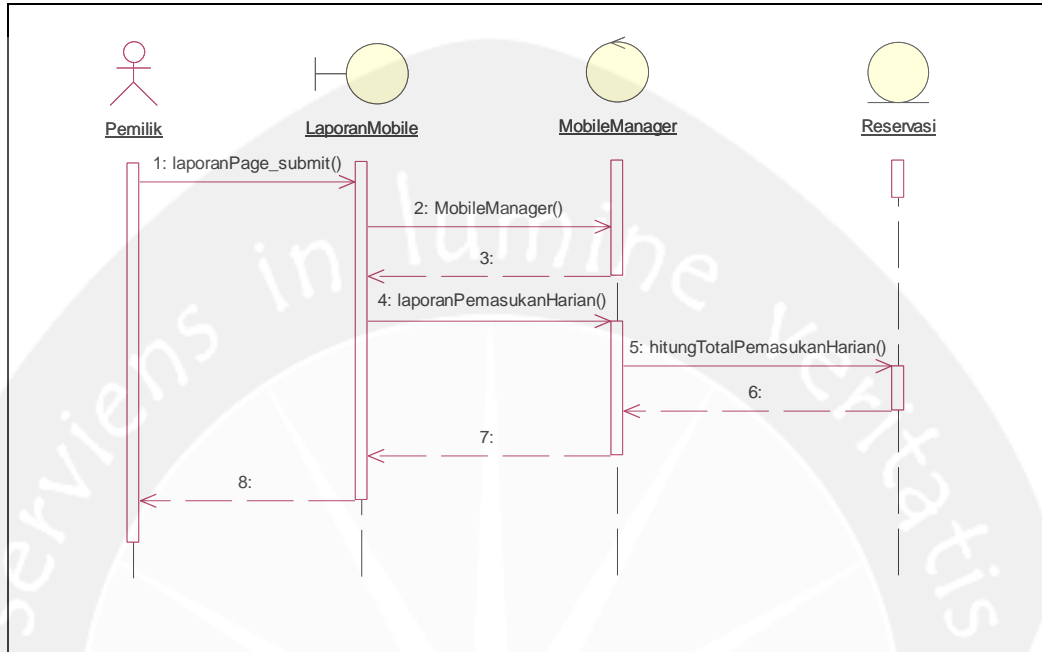
### 2.2.1.9 Ubah Password



Gambar 2.33 Sequence Diagram : Ubah Password

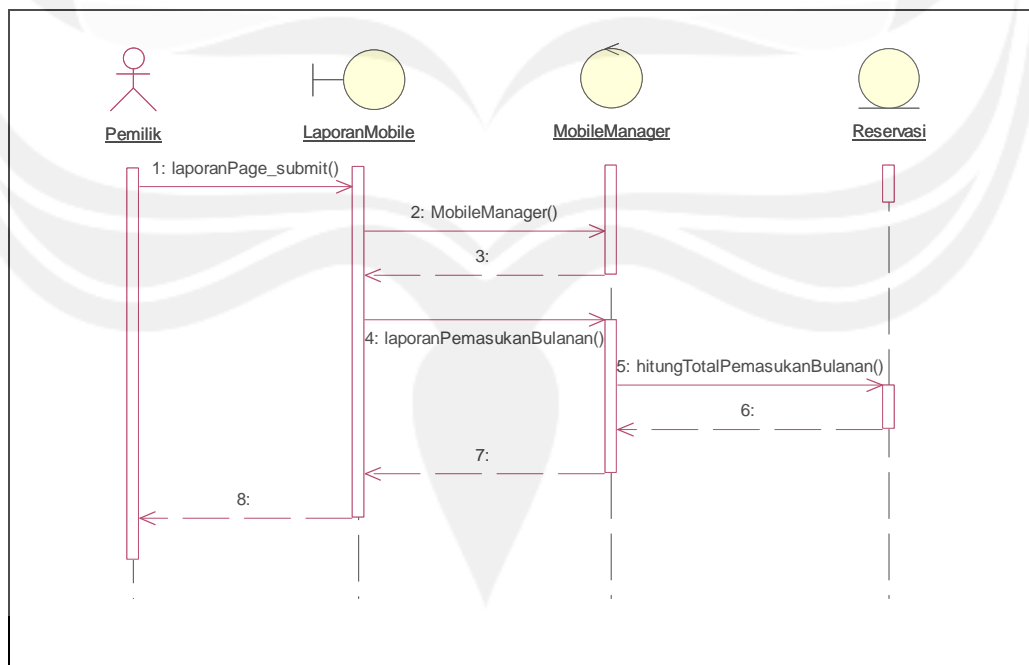
### 2.2.1.10 Monitoring Laporan

#### 2.2.1.10.1 Laporan Total Pemasukan Harian



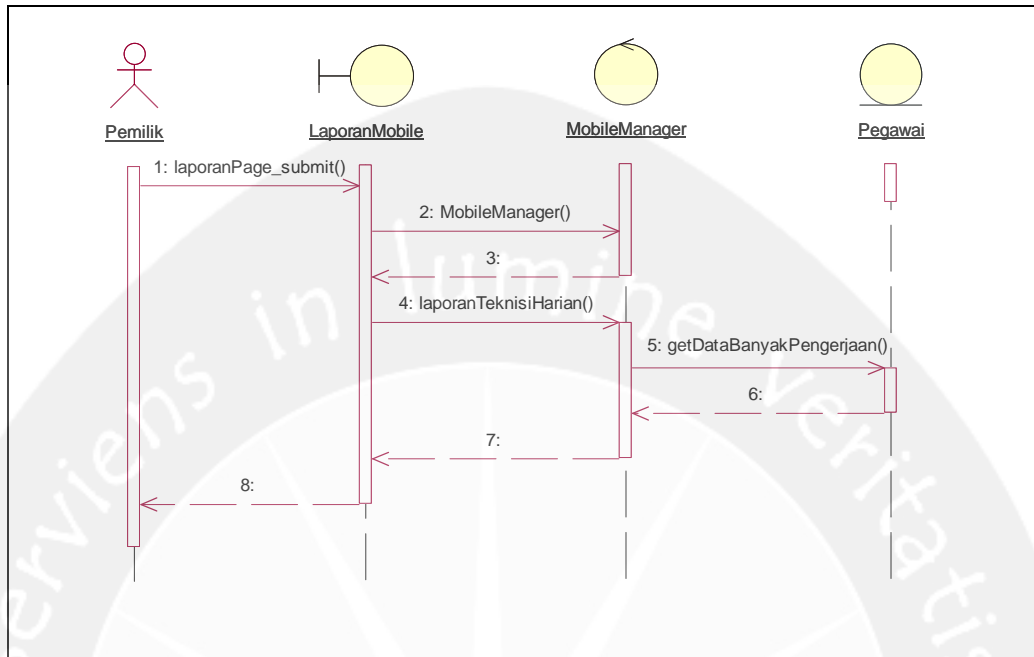
Gambar 2.34 Sequence Diagram : Laporan Total Pemasukan Harian

#### 2.2.1.10.2 Laporan Total Pemasukan Bulanan



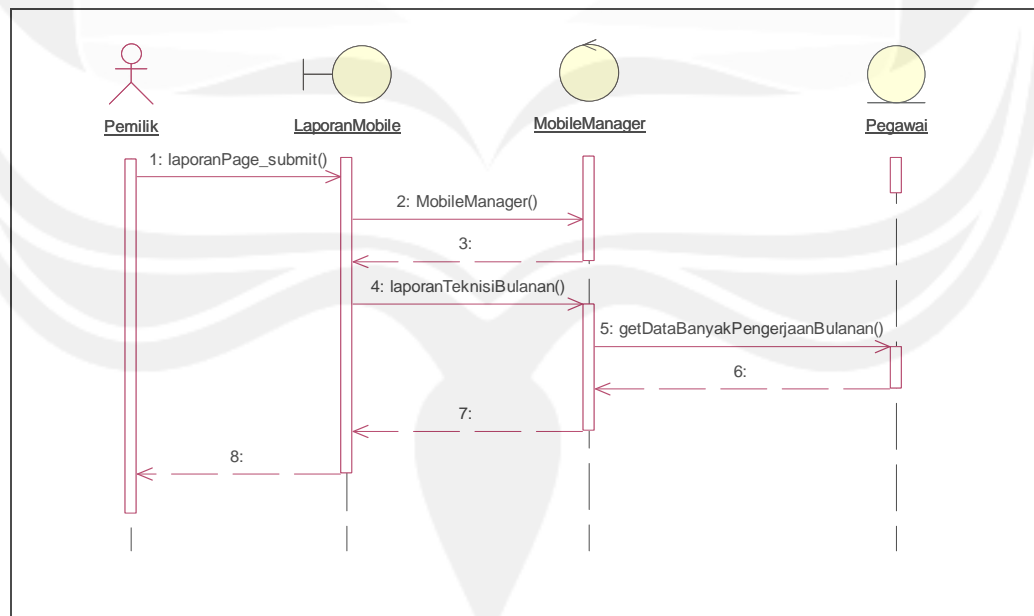
Gambar 2.35 Sequence Diagram : Laporan Total Pemasukan Bulanan

### 2.2.1.10.3 Laporan Teknisi Harian



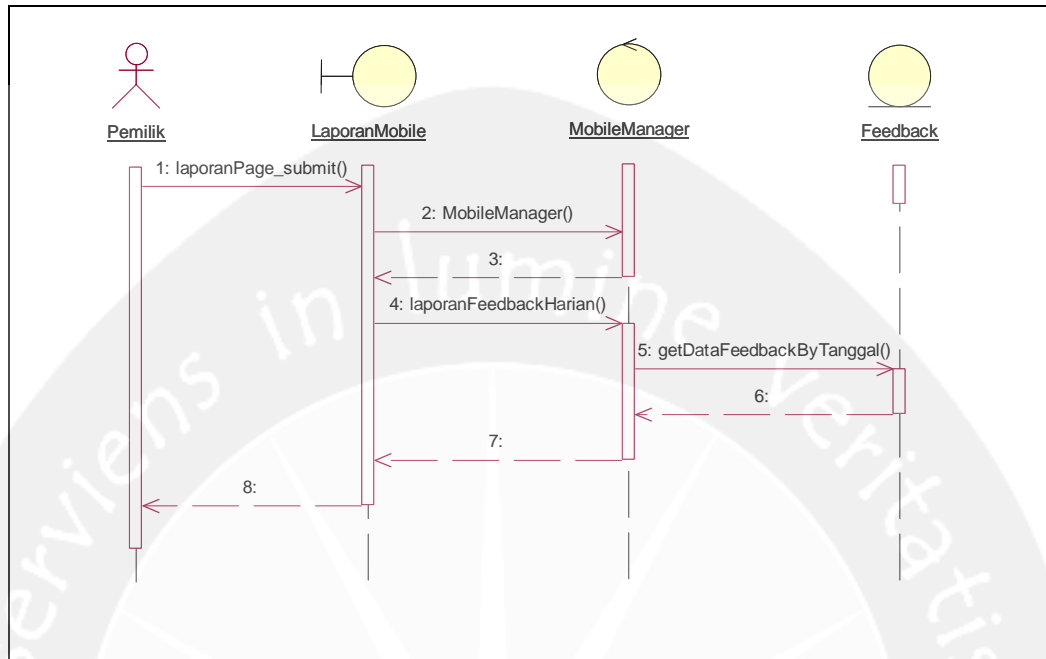
Gambar 2.36 Sequence Diagram : Laporan Teknisi Harian

### 2.2.1.10.4 Laporan Teknisi Bulanan



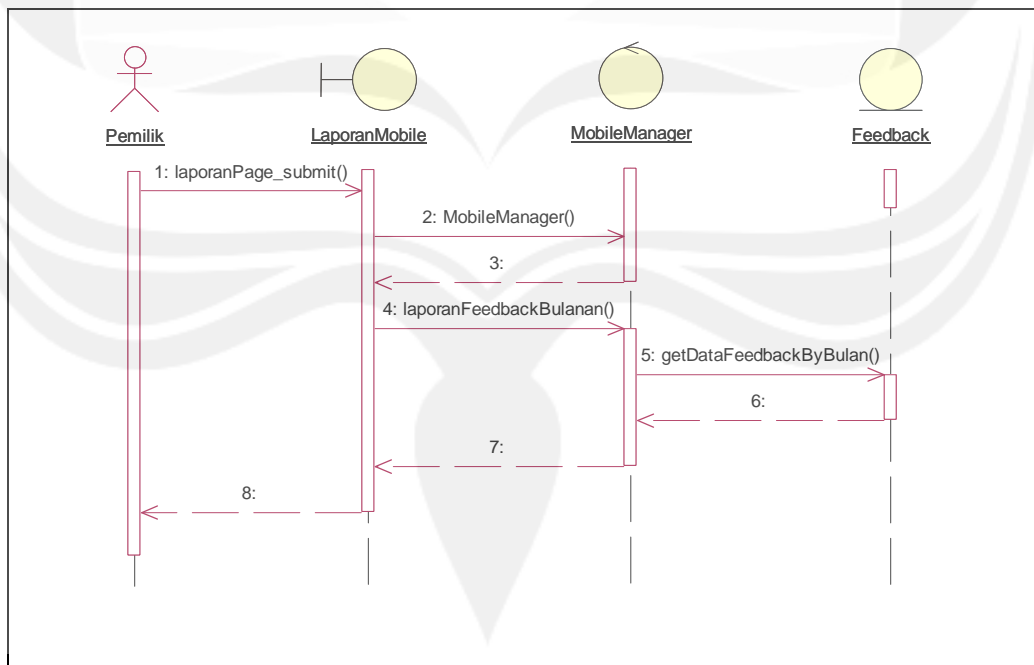
Gambar 2.37 Sequence Diagram : Laporan Teknisi Bulanan

### 2.2.1.10.5 Laporan Feedback Harian



Gambar 2.38 Sequence Diagram : Laporan Feedback Harian

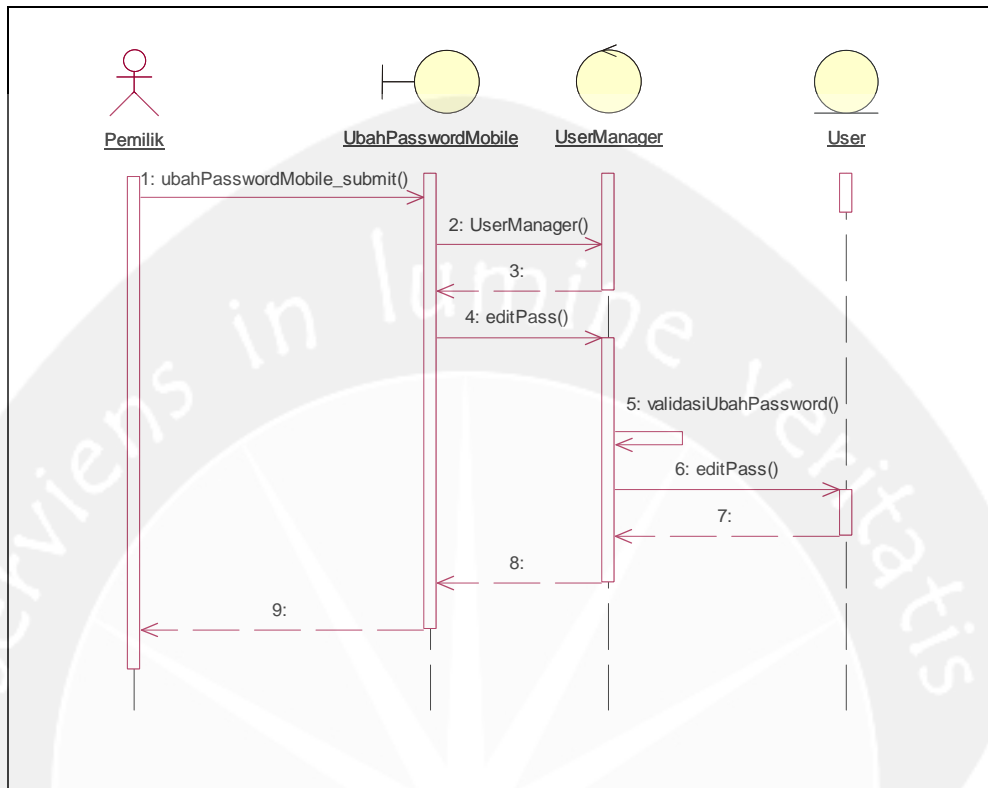
### 2.2.1.10.6 Laporan Feedback Bulanan



Gambar 2.39 Sequence Diagram : Laporan Feedback Bulanan

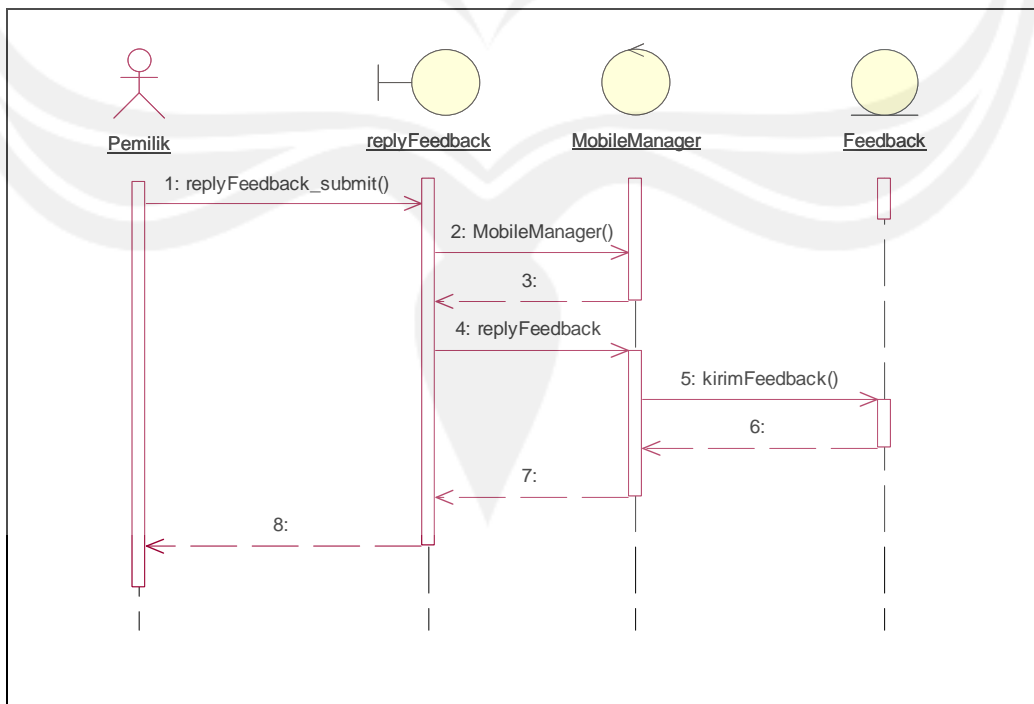


### 2.2.1.11 Ubah Password Mobile Web



Gambar 2.40 Sequence Diagram : Ubah Password Mobile Web

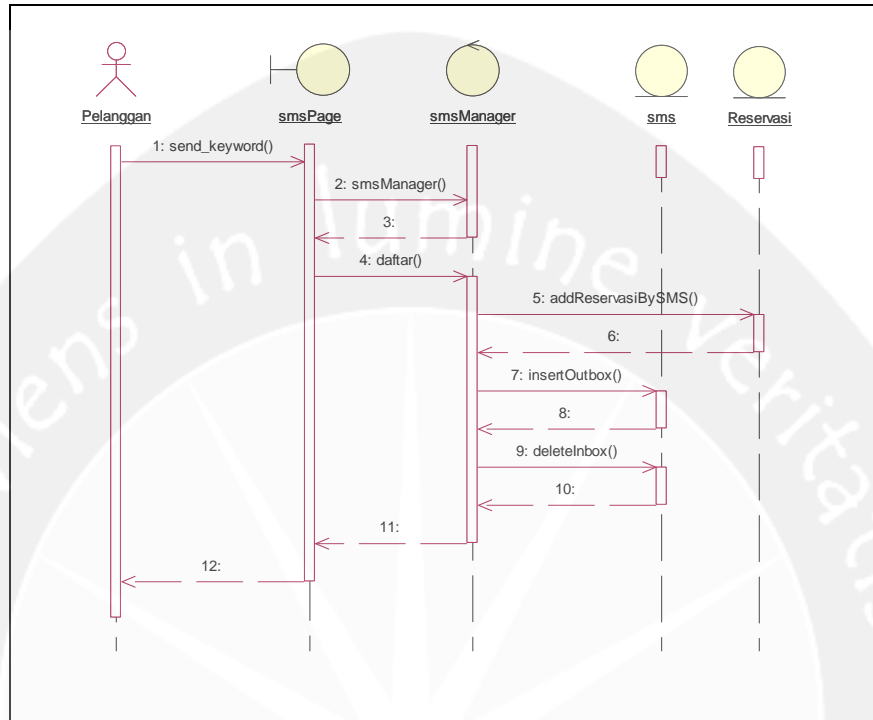
### 2.2.1.12 Balas Feedback



Gambar 2.41 Sequence Diagram : Balas Feedback

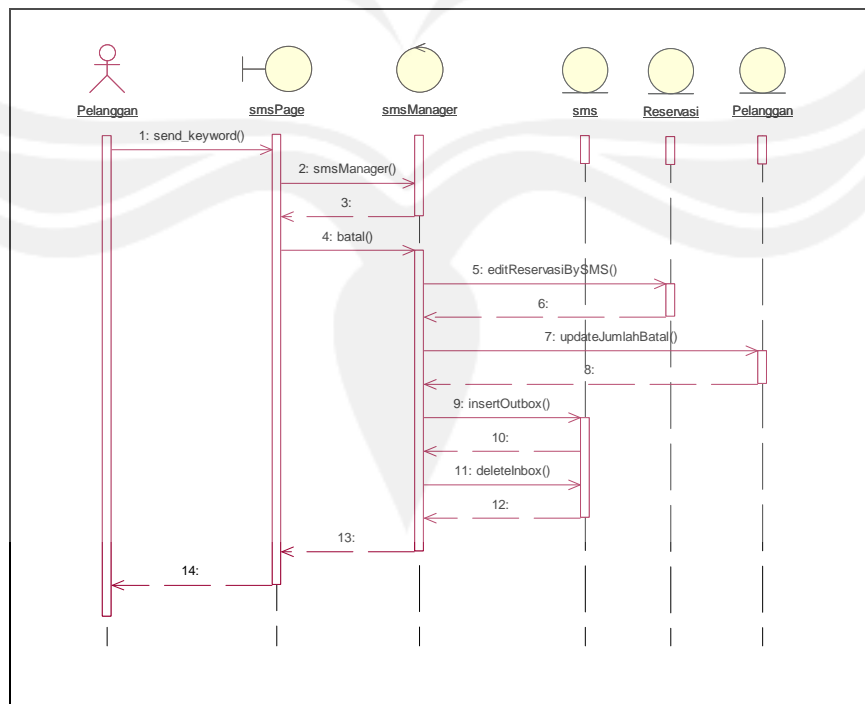
### 2.2.1.13 SMS Gateway

#### 2.2.1.13.1 Daftar



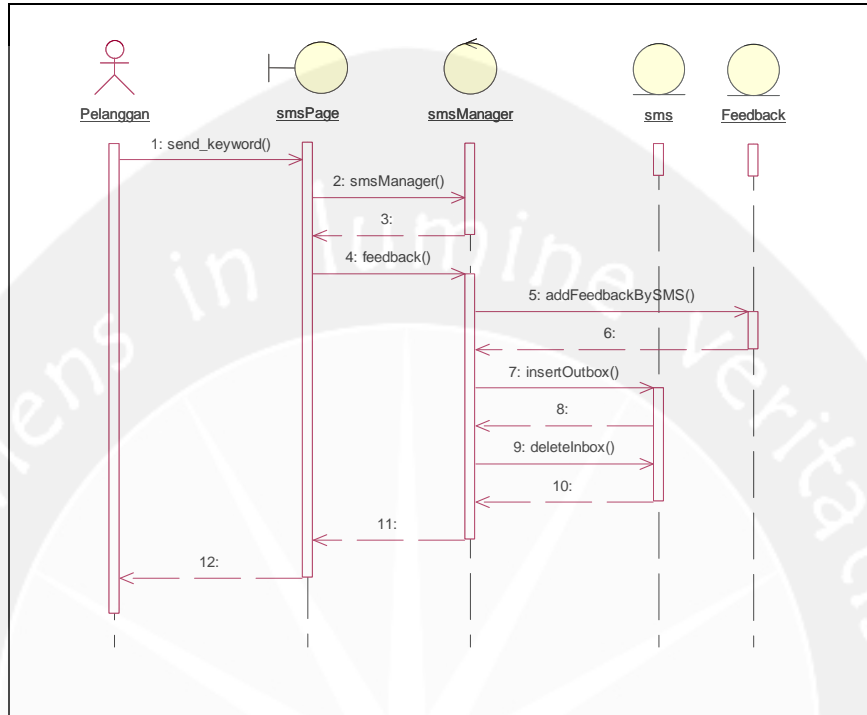
Gambar 2.42 Sequence Diagram : Daftar

#### 2.2.1.13.2 Batal



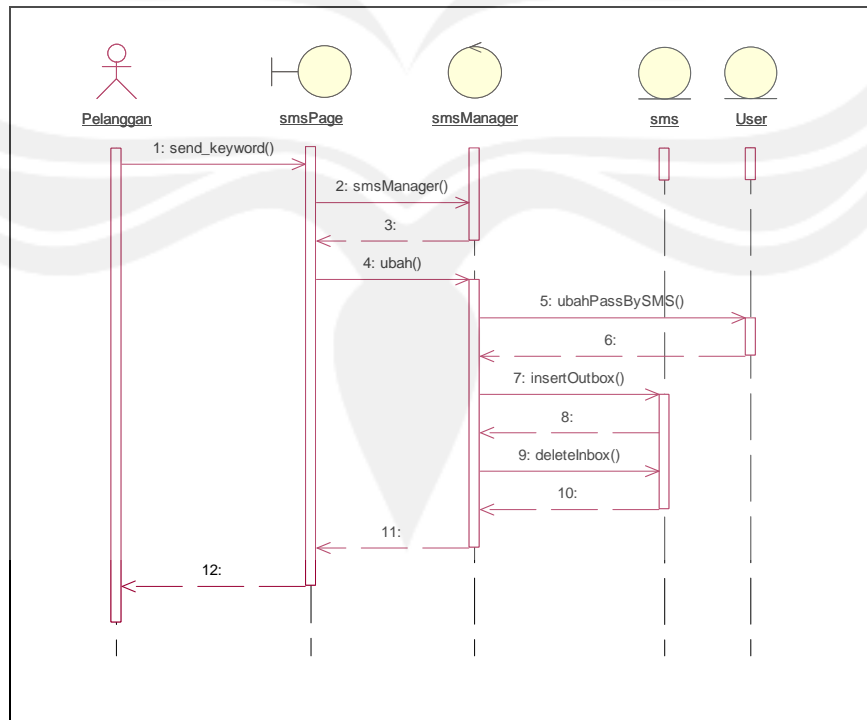
Gambar 2.43 Sequence Diagram : Batal

### 2.2.1.13.3 Feedback



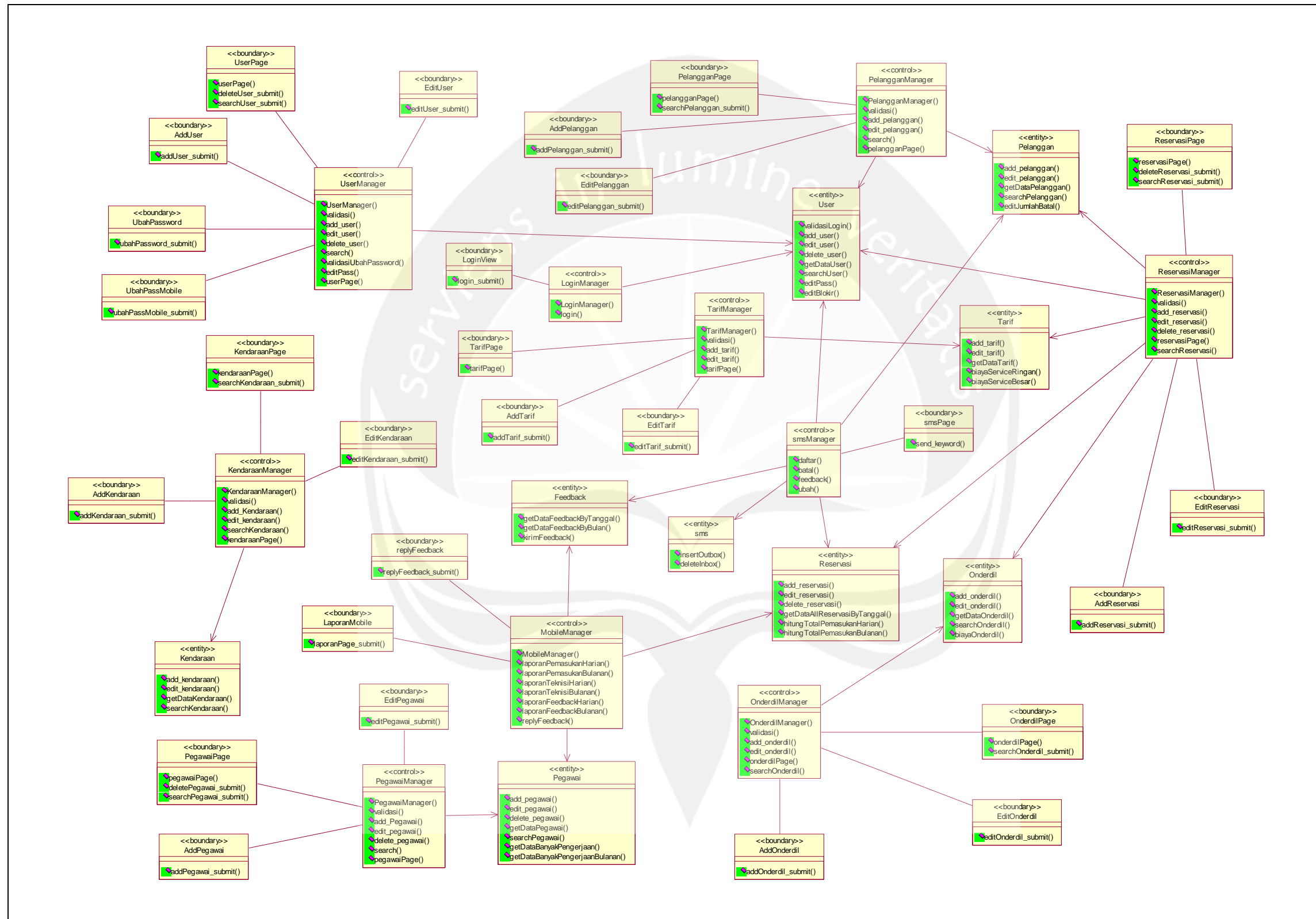
Gambar 2.44 Sequence Diagram : Feedback

### 2.2.1.13.4 Ubah Password



Gambar 2.45 Sequence Diagram : Ubah Password

## 2.2.2 Class Diagram



Gambar 2.46 Class Diagram

## 2.2.3 Class Diagram Specific Descriptions

### 2.2.3.1 Specific Design Class LoginView

|  |                                 |
|--|---------------------------------|
| <b>LoginView</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| +login_submit()<br>Operasi ini digunakan untuk menjalankan fungsi login pada controller. |                                 |

### 2.2.3.2 Specific Design Class UserPage

|   |                                 |
|---|---------------------------------|
| <b>UserPage</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| +userPage()<br>Operasi ini digunakan untuk inisialisasi form UserPage.<br>+deleteUser_submit()<br>Operasi ini digunakan untuk menjalankan fungsi hapus data user yang terdapat pada controller di kelas UserManager.<br>+searchUser_submit()<br>Operasi ini digunakan untuk menjalankan fungsi cari data user yang terdapat pada controller di kelas UserManager. |                                 |

### 2.2.3.3 Specific Design Class AddUser

|  |                                 |
|--|---------------------------------|
| <b>AddUser</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| +addUser_submit()<br>Operasi ini digunakan untuk menjalankan fungsi tambah data user yang terdapat pada controller di kelas UserManager. |                                 |

### 2.2.3.4 Specific Design Class EditUser

|   |                                 |
|---|---------------------------------|
| <b>EditUser</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| +editUser_submit()<br>Operasi ini digunakan untuk menjalankan fungsi ubah data user yang terdapat pada controller di kelas UserManager. |                                 |

### 2.2.3.5 Specific Design Class PegawaiPage

|   |                                 |
|---|---------------------------------|
| <b>PegawaiPage</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
|   |                                 |
| <pre>+pegawaiPage()<br/>Operasi ini digunakan untuk inialisasi form PegawaiPage.<br/>+deletePegawai_submit()<br/>Operasi ini digunakan untuk menjalankan fungsi hapus data<br/>teknisi yang terdapat pada controller di kelas<br/>PegawaiManager.<br/>+searchPegawai_submit()<br/>Operasi ini digunakan untuk menjalankan fungsi cari data<br/>teknisi yang terdapat pada controller di kelas<br/>PegawaiManager.</pre> |                                 |

### 2.2.3.6 Specific Design Class AddPegawai

|   |                                 |
|---|---------------------------------|
| <b>AddPegawai</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
|   |                                 |
| <pre>+addPegawai_submit()<br/>Operasi ini digunakan untuk menjalankan fungsi tambah data<br/>teknisi yang terdapat pada controller di kelas<br/>PegawaiManager.</pre> |                                 |

### 2.2.3.7 Specific Design Class EditPegawai

|  |                                 |
|--|---------------------------------|
| <b>EditPegawai</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
|  |                                 |
| <pre>+editPegawai_submit()<br/>Operasi ini digunakan untuk menjalankan fungsi ubah data<br/>teknisi yang terdapat pada controller di kelas<br/>PegawaiManager.</pre> |                                 |

### 2.2.3.8 Specific Design Class PelangganPage

|   |                                 |
|---|---------------------------------|
| <b>PelangganPage</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
|   |                                 |
| <code>+pelangganPage()</code><br>Operasi ini digunakan untuk inisialisasi form PelangganPage.   |                                 |
| <code>+searchPelanggan_submit()</code><br>Operasi ini digunakan untuk menjalankan fungsi cari data pelanggan yang terdapat pada controller di kelas PelangganManager. |                                 |

### 2.2.3.9 Specific Design Class AddPelanggan

|  |                                 |
|--|---------------------------------|
| <b>AddPelanggan</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
|  |                                 |
| <code>+addPelanggan_submit()</code><br>Operasi ini digunakan untuk menjalankan fungsi tambah data pelanggan yang terdapat pada controller di kelas PelangganManager. |                                 |

### 2.2.3.10 Specific Design Class EditPelanggan

|   |                                 |
|---|---------------------------------|
| <b>EditPelanggan</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
|   |                                 |
| <code>+editPelanggan_submit()</code><br>Operasi ini digunakan untuk menjalankan fungsi ubah data pelanggan yang terdapat pada controller di kelas PelangganManager. |                                 |

### 2.2.3.11 Specific Design Class KendaraanPage

|  |                                 |
|--|---------------------------------|
| <b>KendaraanPage</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
|  |                                 |
| <code>+kendaraanPage()</code><br>Operasi ini digunakan untuk inisialisasi form KendaraanPage.      |                                 |
| <code>+searchKendaraan_submit()</code><br>Operasi ini digunakan untuk menjalankan fungsi cari data |                                 |

kendaraan yang terdapat pada controller di kelas KendaraanManager.

#### 2.2.3.12 Specific Design Class AddKendaraan

|  |                                 |
|--|---------------------------------|
| <b>AddKendaraan</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+addKendaraan_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi tambah data kendaraan yang terdapat pada controller di kelas KendaraanManager.</p> |                                 |

#### 2.2.3.13 Specific Design Class EditKendaraan

|   |                                 |
|---|---------------------------------|
| <b>EditKendaraan</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+editKendaraan_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi ubah data kendaraan yang terdapat pada controller di kelas KendaraanManager.</p> |                                 |

#### 2.2.3.14 Specific Design Class ReservasiPage

|  |                                 |
|--|---------------------------------|
| <b>ReservasiPage</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+reservasiPage()</pre> <p>Operasi ini digunakan untuk inisialisasi form ReservasiPage.</p> <pre>+deleteReservasi_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi hapus data reservasi yang terdapat pada controller di kelas ReservasiManager.</p> <pre>+searchReservasi_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi cari data reservasi berdasarkan tanggal yang terdapat pada controller di kelas ReservasiManager.</p> |                                 |



### 2.2.3.15 Specific Design Class AddReservasi

|   |                                 |
|---|---------------------------------|
| <b>AddReservasi</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+addReservasi_submit()</pre>   |                                 |
| Operasi ini digunakan untuk menjalankan fungsi tambah data reservasi yang terdapat pada controller di kelas ReservasiManager. |                                 |

### 2.2.3.16 Specific Design Class EditReservasi

|   |                                 |
|---|---------------------------------|
| <b>EditReservasi</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+editReservasi_submit()</pre>  |                                 |
| Operasi ini digunakan untuk menjalankan fungsi ubah data reservasi yang terdapat pada controller di kelas ReservasiManager. |                                 |

### 2.2.3.17 Specific Design Class TarifPage

|  |                                 |
|--|---------------------------------|
| <b>TarifPage</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+tarifPage()</pre>                                  |                                 |
| Operasi ini digunakan untuk inisialisasi form TarifPage. |                                 |

### 2.2.3.18 Specific Design Class AddTarif

|   |                                 |
|---|---------------------------------|
| <b>AddTarif</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+addTarif_submit()</pre>   |                                 |
| Operasi ini digunakan untuk menjalankan fungsi tambah data tarif yang terdapat pada controller di kelas TarifManager. |                                 |

### 2.2.3.19 Specific Design Class EditTarif

|   |                                 |
|---|---------------------------------|
| <b>EditTarif</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+editTarif_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi ubah data tarif yang terdapat pada controller di kelas TarifManager.</p> |                                 |

### 2.2.3.20 Specific Design Class OnderdilPage

|  |                                 |
|--|---------------------------------|
| <b>OnderdilPage</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+onderdilPage()</pre> <p>Operasi ini digunakan untuk inisialisasi form OnderdilPage.</p> <pre>+searchOnderdil_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi cari data onderdil yang terdapat pada controller di kelas OnderdilManager.</p> |                                 |

### 2.2.3.21 Specific Design Class AddOnderdil

|   |                                 |
|---|---------------------------------|
| <b>AddOnderdil</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+addOnderdil_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi tambah data onderdil yang terdapat pada controller di kelas OnderdilManager.</p> |                                 |

### 2.2.3.22 Specific Design Class EditOnderdil

|  |                                 |
|--|---------------------------------|
| <b>EditOnderdil</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <pre>+editOnderdil_submit()</pre> <p>Operasi ini digunakan untuk menjalankan fungsi ubah data onderdil yang terdapat pada controller di kelas OnderdilManager.</p> |                                 |

### 2.2.3.23 Specific Design Class UbahPassword

|   |                                 |
|---|---------------------------------|
| <b>UbahPassword</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <b>+ubahPassword_submit()</b><br>Operasi ini digunakan untuk menjalankan fungsi ubah password yang terdapat pada controller di kelas UserManager. |                                 |

### 2.2.3.24 Specific Design Class LaporanMobile

|  |                                 |
|--|---------------------------------|
| <b>LaporanMobile</b>   | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <b>+laporanPage_submit()</b><br>Operasi ini digunakan untuk menjalankan fungsi menampilkan laporan yang terdapat pada controller di kelas MobileManager. |                                 |

### 2.2.3.25 Specific Design Class UbahPasswordMobile

|  |                                 |
|--|---------------------------------|
| <b>UbahPasswordMobile</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <b>+ubahPassMobile_submit()</b><br>Operasi ini digunakan untuk menjalankan fungsi ubah password mobile web yang terdapat pada controller di kelas UserManager. |                                 |

### 2.2.3.26 Specific Design Class ReplyFeedback

|   |                                 |
|---|---------------------------------|
| <b>ReplyFeedback</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <b>+replyFeedback_submit()</b><br>Operasi ini digunakan untuk menjalankan fungsi balas feedback yang terdapat pada controller di kelas MobileManager. |                                 |

### 2.2.3.27 Specific Design Class smsPage

|   |                                 |
|---|---------------------------------|
| <b>smsPage</b>  | <b>&lt;&lt;boundary&gt;&gt;</b> |
| <b>+send_keyword()</b><br>Operasi ini digunakan untuk menjalankan fungsi sms yang terdapat pada controller di kelas smsManager. |                                 |

### 2.2.3.28 Specific Design Class LoginManager

|  |                                |
|--|--------------------------------|
| <b>LoginManager</b>  | <b>&lt;&lt;control&gt;&gt;</b> |
| <b>+LoginManager()</b><br>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.  |                                |
| <b>+login()</b><br>Operasi ini digunakan untuk menjalankan fungsi validasi login yang terdapat pada entity di kelas User serta membuat sesi login. |                                |

### 2.2.3.29 Specific Design Class UserManager

|   |                                |
|---|--------------------------------|
| <b>UserManager</b>  | <b>&lt;&lt;control&gt;&gt;</b> |
| <b>+UserManager()</b><br>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.  |                                |
| <b>+userPage()</b><br>Operasi ini digunakan untuk menampilkan data-data user pada halaman user.   |                                |
| <b>+validasi()</b><br>Operasi ini digunakan untuk melakukan pengecekan ketika menambah data dan mengubah data.  |                                |
| <b>+add_user()</b><br>Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data user, kemudian menjalankan fungsi tambah data user yang |                                |

terdapat pada entity di kelas User, tetapi jika validasi gagal maka akan menampilkan halaman tambah data user.

+edit\_user()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data user, kemudian menjalankan fungsi ubah data user yang terdapat pada entity di kelas User, tetapi jika validasi gagal maka akan menampilkan halaman ubah data user.

+delete\_user()

Operasi ini digunakan untuk menjalankan fungsi hapus data user yang terdapat pada entity di kelas User.

+search()

Operasi ini digunakan untuk menjalankan fungsi cari data user yang terdapat pada entity di kelas User, kemudian dari data yang didapat dicari berdasarkan kategori.

+validasiUbahPassword()

Operasi ini digunakan untuk melakukan pengecekan ketika mengubah password.

+editPass()

Operasi ini digunakan untuk menjalankan fungsi ubah password yang terdapat pada entity di kelas User.

### 2.2.3.30 Specific Design Class PegawaiManager

|  |                                |
|--|--------------------------------|
| <b>PegawaiManager</b>  | <b>&lt;&lt;control&gt;&gt;</b> |
| <p>+PegawaiManager()<br/>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+pegawaiPage()<br/>Operasi ini digunakan untuk menampilkan data-data teknisi pada halaman teknisi.</p> <p>+validasi()<br/>Operasi ini digunakan untuk melakukan pengecekan ketika menambah data dan mengubah data.</p> |                                |

+add\_pegawai()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data teknisi, kemudian menjalankan fungsi tambah data teknisi yang terdapat pada entity di kelas Pegawai, tetapi jika validasi gagal maka akan menampilkan halaman tambah data teknisi.

+edit\_pegawai()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data teknisi, kemudian menjalankan fungsi ubah data teknisi yang terdapat pada entity di kelas Pegawai, tetapi jika validasi gagal maka akan menampilkan halaman ubah data teknisi.

+delete\_pegawai()

Operasi ini digunakan untuk menjalankan fungsi hapus data teknisi yang terdapat pada entity di kelas Pegawai.

+search()

Operasi ini digunakan untuk menjalankan fungsi cari data teknisi yang terdapat pada entity di kelas Pegawai, kemudian dari data yang didapat dicari berdasarkan kategori.

### 2.2.3.31 Specific Design Class PelangganManager

|  |                                |
|--|--------------------------------|
| <b>PelangganManager</b>  | <b>&lt;&lt;control&gt;&gt;</b> |
| <p>+PelangganManager()<br/>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+pelangganPage()<br/>Operasi ini digunakan untuk menampilkan data-data pelanggan pada halaman pelanggan.</p> <p>+validasi()<br/>Operasi ini digunakan untuk melakukan pengecekan ketika menambah data dan mengubah data.</p> <p>+add_pelanggan()</p> |                                |

|                                  |            |        |
|----------------------------------|------------|--------|
| Program Studi Teknik Informatika | DPPL – CRS | 43/ 92 |
|----------------------------------|------------|--------|

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data pelanggan, kemudian menjalankan fungsi tambah data pelanggan yang terdapat pada entity di kelas Pelanggan, tetapi jika validasi gagal maka akan menampilkan halaman tambah data pelanggan.

+edit\_pelanggan()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data pelanggan, kemudian menjalankan fungsi ubah data pelanggan yang terdapat pada entity di kelas Pelanggan, tetapi jika validasi gagal maka akan menampilkan halaman ubah data pelanggan.

+search()

Operasi ini digunakan untuk menjalankan fungsi cari data pelanggan yang terdapat pada entity di kelas Pelanggan, kemudian dari data yang didapat dicari berdasarkan kategori.

### 2.2.3.32 Specific Design Class KendaraanManager

|   |                                |
|---|--------------------------------|
| <b>KendaraanManager</b>   | <b>&lt;&lt;control&gt;&gt;</b> |
| <p>+KendaraanManager()<br/>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+kendaraanPage()<br/>Operasi ini digunakan untuk menampilkan data-data kendaraan pada halaman kendaraan.</p> <p>+validasi()<br/>Operasi ini digunakan untuk melakukan pengecekan ketika menambah data dan mengubah data.</p> <p>+add_kendaraan()<br/>Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data kendaraan, kemudian menjalankan fungsi tambah data kendaraan</p> |                                |

yang terdapat pada entity di kelas Kendaraan, tetapi jika validasi gagal maka akan menampilkan halaman tambah data kendaraan.

+edit\_kendaraan()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data kendaraan, kemudian menjalankan fungsi ubah data kendaraan yang terdapat pada entity di kelas Kendaraan, tetapi jika validasi gagal maka akan menampilkan halaman ubah data kendaraan.

+search()

Operasi ini digunakan untuk menjalankan fungsi cari data kendaraan yang terdapat pada entity di kelas Kendaraan, kemudian dari data yang didapat dicari berdasarkan kategori.

### 2.2.3.33 Specific Design Class ReservasiManager

|   |                                |
|---|--------------------------------|
| <b>ReservasiManager</b>   | <b>&lt;&lt;control&gt;&gt;</b> |
| <p>+ReservasiManager()<br/>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+reservasiPage()<br/>Operasi ini digunakan untuk menampilkan data-data reservasi pada halaman reservasi.</p> <p>+validasi()<br/>Operasi ini digunakan untuk melakukan pengecekan ketika menambah data dan mengubah data.</p> <p>+add_reservasi()<br/>Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data reservasi, kemudian menjalankan fungsi tambah data reservasi yang terdapat pada entity di kelas Reservasi, tetapi jika validasi gagal maka akan menampilkan halaman tambah data reservasi.</p> |                                |
| Program Studi Teknik Informatika  | DPPL – CRS                     |
| 45/ 92  |                                |



+edit\_reservasi()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data reservasi, kemudian menjalankan fungsi ubah data reservasi yang terdapat pada entity di kelas Reservasi, tetapi jika validasi gagal maka akan menampilkan halaman ubah data reservasi.

+delete\_reservasi()

Operasi ini digunakan untuk menjalankan fungsi hapus data reservasi yang terdapat pada entity di kelas Reservasi.

+searchReservasi()

Operasi ini digunakan untuk menjalankan fungsi cari data reservasi berdasarkan tanggal yang terdapat pada entity di kelas Reservasi.

#### 2.2.3.34 Specific Design Class OnderdilManager

|                 |             |
|-----------------|-------------|
| OnderdilManager | <<control>> |
|-----------------|-------------|

+OnderdilManager()

Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.

+onderdilPage()

Operasi ini digunakan untuk menampilkan data-data onderdil yang terdapat pada halaman onderdil.

+validasi()

Operasi ini digunakan untuk melakukan pengecekan ketika menambah data dan mengubah data.

+add\_onderdil()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data onderdil, kemudian menjalankan fungsi tambah data onderdil yang terdapat pada entity di kelas Onderdil, tetapi jika validasi gagal maka akan menampilkan halaman tambah data onderdil.

|                                  |            |        |
|----------------------------------|------------|--------|
| Program Studi Teknik Informatika | DPPL – CRS | 46/ 92 |
|----------------------------------|------------|--------|

+edit\_onderdil ()

Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data onderdil, kemudian menjalankan fungsi ubah data onderdil yang terdapat pada entity di kelas Onderdil, tetapi jika validasi gagal maka akan menampilkan halaman ubah data onderdil.

+searchOnderdil()

Operasi ini digunakan untuk menjalankan fungsi cari data onderdil yang terdapat pada entity di kelas Onderdil, kemudian dari data yang didapat dicari berdasarkan kategori.

#### 2.2.3.35 Specific Design Class TarifManager

|   |                                |
|---|--------------------------------|
| <b>TarifManager</b>   | <b>&lt;&lt;control&gt;&gt;</b> |
| <p>+TarifManager()<br/>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+tarifPage()<br/>Operasi ini digunakan untuk menampilkan data-data tarif pada halaman tarif.</p> <p>+validasi()<br/>Operasi ini digunakan untuk melakukan pengecekan ketika menambah data dan mengubah data.</p> <p>+add_tarif ()<br/>Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data tarif, kemudian menjalankan fungsi tambah data tarif yang terdapat pada entity di kelas Tarif, tetapi jika validasi gagal maka akan menampilkan halaman tambah data tarif.</p> <p>+edit_tarif()<br/>Ketika validasi berhasil operasi ini digunakan untuk menginisialisasi variabel dengan nilai dari inputan data</p> |                                |

tarif, kemudian menjalankan fungsi ubah data tarif yang terdapat pada entity di kelas Tarif, tetapi jika validasi gagal maka akan menampilkan halaman ubah data tarif.

### 2.2.3.36 Specific Design Class MobileManager

| MobileManager   | <<control>> |
|---|-------------|
|   |             |
| <pre> +MobileManager() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +laporanPemasukanHarian() Operasi ini digunakan untuk menjalankan fungsi yang digunakan untuk menghitung total pemasukan harian yang terdapat pada entity di kelas Reservasi. +laporanPemasukanBulanan() Operasi ini digunakan untuk menjalankan fungsi yang digunakan untuk menghitung total pemasukan bulanan yang terdapat pada entity di kelas Reservasi. +laporanTeknisiHarian() Operasi ini digunakan untuk menjalankan fungsi yang digunakan untuk menghitung banyak pengerjaan yang telah dilakukan pegawai pada hari tertentu yang terdapat pada entity di kelas Pegawai. +laporanTeknisiBulanan() Operasi ini digunakan untuk menjalankan fungsi yang digunakan untuk menghitung banyak pengerjaan yang telah dilakukan pegawai pada bulan tertentu yang terdapat pada entity di kelas Pegawai. +laporanFeedbackHarian() Operasi ini digunakan untuk menjalankan fungsi yang digunakan untuk mengambil data-data feedback pada hari tertentu yang terdapat pada entity di kelas Feedback. +laporanFeedbackBulanan() </pre> |             |

Operasi ini digunakan untuk menjalankan fungsi yang digunakan untuk mengambil data-data feedback pada bulan tertentu yang terdapat pada entity di kelas Feedback.

+replyFeedback

Operasi ini digunakan untuk menjalankan fungsi untuk membalas feedback yang terdapat pada entity di kelas Feedback.

### 2.2.3.37 Specific Design Class smsManager

|   |                                |
|---|--------------------------------|
| <b>smsManager</b>   | <b>&lt;&lt;control&gt;&gt;</b> |
| <p>+smsManager()<br/>Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.</p> <p>+daftar()<br/>Operasi ini digunakan ketika pelanggan telah mengirim SMS dengan kata kunci daftar ke nomor tujuan yang sudah ditentukan. Operasi ini akan menjalankan fungsi untuk menyimpan pesan balasan yang terdapat pada entity di kelas sms dan juga menjalankan fungsi untuk menyimpan data reservasi yang terdapat pada entity di kelas Reservasi.</p> <p>+batal()<br/>Operasi ini digunakan ketika pelanggan telah mengirim SMS dengan kata kunci batal ke nomor tujuan yang sudah ditentukan. Operasi ini akan menjalankan fungsi untuk menyimpan pesan balasan yang terdapat pada entity di kelas sms dan juga menjalankan fungsi untuk mengubah status reservasi menjadi batal yang terdapat pada entity di kelas Reservasi dan juga menjalankan fungsi untuk mengubah jumlah batal pelanggan yang terdapat pada entity di kelas Pelanggan.</p> <p>+feedback()<br/>Operasi ini digunakan ketika pelanggan telah mengirim SMS</p> |                                |

dengan kata kunci feedback ke nomor tujuan yang sudah ditentukan. Operasi ini akan menjalankan fungsi untuk menyimpan pesan balasan yang terdapat pada entity di kelas sms dan juga menjalankan fungsi untuk menyimpan data feedback yang terdapat pada entity di kelas Feedback.

+ubah()

Operasi ini digunakan ketika pelanggan telah mengirim SMS dengan kata kunci ubah ke nomor tujuan yang sudah ditentukan. Operasi ini akan menjalankan fungsi untuk menyimpan pesan balasan yang terdapat pada entity di kelas sms dan juga menjalankan fungsi untuk mengubah password yang terdapat pada entity di kelas User.

### 2.2.3.38 Specific Design Class User

|  |            |
|--|------------|
| User   | <<entity>> |
| <p>+validasiLogin()<br/>Operasi ini digunakan untuk melakukan validasi antara nilai inputan dari user dengan nilai di database.</p> <p>+add_user()<br/>Operasi ini digunakan untuk menambahkan data user ke database.</p> <p>+edit_user()<br/>Operasi ini digunakan untuk mengubah data user di database dengan data yang sudah diinputkan.</p> <p>+delete_user()<br/>Operasi ini digunakan untuk menghapus data user di database.</p> <p>+getDataUser()<br/>Operasi ini digunakan untuk mengambil data user dari database.</p> <p>+searchUser()<br/>Operasi ini digunakan untuk mengambil data user dari database, kemudian akan diproses di controller di kelas UserManager.</p> |            |

+editPass()

Operasi ini digunakan untuk mengubah password user di database.

+editBlokir()

Operasi ini digunakan untuk mengubah status user untuk pelanggan yang ada di database menjadi blokir ketika pelanggan telah melakukan pembatalan sebanyak tiga kali.

### 2.2.3.39 Specific Design Class Pegawai

| Pegawai   | <<entity>> |
|---|------------|
| <p>+add_pegawai()<br/>Operasi ini digunakan untuk menambahkan data teknisi ke database.</p> <p>+edit_pegawai ()<br/>Operasi ini digunakan untuk mengubah data teknisi di database dengan data yang sudah diinputkan.</p> <p>+delete_pegawai()<br/>Operasi ini digunakan untuk menghapus data teknisi di database.</p> <p>+getDataPegawai()<br/>Operasi ini digunakan untuk mengambil data teknisi dari database.</p> <p>+getDataBanyakPengerjaan()<br/>Operasi ini digunakan untuk menghitung banyak pengerjaan yang telah dilakukan teknisi pada tanggal tertentu.</p> <p>+getDataBanyakPengerjaanBulanan()<br/>Operasi ini digunakan untuk menghitung banyak pengerjaan yang telah dilakukan teknisi pada bulan tertentu.</p> |            |

#### 2.2.3.40 Specific Design Class Pelanggan

|   |                               |
|---|-------------------------------|
| <b>Pelanggan</b>  | <b>&lt;&lt;entity&gt;&gt;</b> |
| <pre>+add_pelanggan()<br/>Operasi ini digunakan untuk menambahkan data pelanggan ke database.<br/><br/>+edit_pelanggan()<br/>Operasi ini digunakan untuk mengubah data pelanggan di database dengan data yang sudah diinputkan.<br/><br/>+getDataPelanggan()<br/>Operasi ini digunakan untuk mengambil data pelanggan dari database.<br/><br/>+searchPelanggan()<br/>Operasi ini digunakan untuk mengambil data pelanggan dari database, kemudian akan diproses di controller di kelas PelangganManager.<br/><br/>+editJumlahBatal()<br/>Operasi ini digunakan untuk membuat jumlah batal yang terdapat pada database bertambah satu ketika pelanggan melakukan pembatalan.</pre> |                               |

#### 2.2.3.41 Specific Design Class Kendaraan

|  |                               |
|--|-------------------------------|
| <b>Kendaraan</b>   | <b>&lt;&lt;entity&gt;&gt;</b> |
| <pre>+add_kendaraan()<br/>Operasi ini digunakan untuk menambahkan data kendaraan ke database.<br/><br/>+edit_kendaraan()<br/>Operasi ini digunakan untuk mengubah data kendaraan di database dengan data yang sudah diinputkan.<br/><br/>+getDataKendaraan()<br/>Operasi ini digunakan untuk mengambil data kendaraan dari database.</pre> |                               |

+searchKendaraan()

Operasi ini digunakan untuk mengambil data kendaraan dari database, kemudian akan diproses di controller di kelas KendaraanManager.

#### 2.2.3.42 Specific Design Class Reservasi

|   |                               |
|---|-------------------------------|
| <b>Reservasi</b>  | <b>&lt;&lt;entity&gt;&gt;</b> |
| <p>+add_reservasi()<br/>Operasi ini digunakan untuk menambahkan data reservasi ke database.</p> <p>+edit_reservasi()<br/>Operasi ini digunakan untuk mengubah data reservasi di database dengan data yang sudah diinputkan.</p> <p>+delete_reservasi()<br/>Operasi ini digunakan untuk menghapus data reservasi di database.</p> <p>+getDataAllReservasiByTanggal()<br/>Operasi ini digunakan untuk mengambil semua data reservasi di database berdasarkan tanggal.</p> <p>+hitungTotalPemasukanHarian()<br/>Operasi ini digunakan untuk menghitung total pemasukan harian.</p> <p>+hitungTotalPemasukanBulanan()<br/>Operasi ini digunakan untuk menghitung total pemasukan bulanan.</p> |                               |

#### 2.2.3.43 Specific Design Class Tarif

|   |                               |
|---|-------------------------------|
| <b>Tarif</b>  | <b>&lt;&lt;entity&gt;&gt;</b> |
| <p>+add_tarif()<br/>Operasi ini digunakan untuk menambahkan data tarif ke database.</p> |                               |



+edit\_ tarif()

Operasi ini digunakan untuk mengubah data tarif di database dengan data yang sudah diinputkan.

+getDataTarif()

Operasi ini digunakan untuk mengambil data tarif dari database.

+biayaServiceRingan()

Operasi ini digunakan untuk menghitung biaya service ringan kendaraan pada waktu melakukan reservasi.

+biayaServiceBesar()

Operasi ini digunakan untuk menghitung biaya service besar kendaraan pada waktu melakukan reservasi.

#### 2.2.3.44 Specific Design Class Onderdil

|   |            |
|---|------------|
| Onderdil  | <<entity>> |
| <p>+add_ onderdil()<br/>Operasi ini digunakan untuk menambahkan data onderdil ke database.</p> <p>+edit_ onderdil()<br/>Operasi ini digunakan untuk mengubah data onderdil di database dengan data yang sudah diinputkan.</p> <p>+getDataOnderdil()<br/>Operasi ini digunakan untuk mengambil data onderdil dari database.</p> <p>+searchOnderdil()<br/>Operasi ini digunakan untuk mengambil data onderdil dari database, kemudian akan diproses di controller di kelas OnderdilManager.</p> <p>+biayaOnderdil()<br/>Operasi ini digunakan untuk menghitung biaya pembelian onderdil kendaraan pada waktu melakukan reservasi.</p> |            |

#### 2.2.3.45 Specific Design Class Feedback

|   |                               |
|---|-------------------------------|
| <b>Feedback</b>   | <b>&lt;&lt;entity&gt;&gt;</b> |
| <pre>+getDataFeedbackByTanggal()<br/>Operasi ini digunakan mengambil data-data feedback pada tanggal tertentu di database.<br/>+getDataFeedbackByBulan()<br/>Operasi ini digunakan mengambil data-data feedback pada bulan tertentu di database.<br/>+ kirimFeedback()<br/>Operasi ini digunakan unntuk menyimpan balasan feedback ke database.</pre> |                               |

#### 2.2.3.46 Specific Design Class sms

|  |                               |
|--|-------------------------------|
| <b>sms</b>   | <b>&lt;&lt;entity&gt;&gt;</b> |
| <pre>+insertOutboxDaftar()<br/>Operasi ini digunakan untuk menyimpan pesan balasan ke dalam database ketika user mengirim SMS dengan kata kunci daftar.<br/>+insertOutboxBatal()<br/>Operasi ini digunakan untuk menyimpan pesan balasan ke dalam database ketika user mengirim SMS dengan kata kunci batal.<br/>+insertOutboxFeedback()<br/>Operasi ini digunakan untuk menyimpan pesan balasan ke dalam database ketika user mengirim SMS dengan kata kunci feedback.<br/>+insertOutboxUbah()<br/>Operasi ini digunakan untuk menyimpan pesan balasan ke dalam database ketika user mengirim SMS dengan kata kunci ubah.</pre> |                               |

### 3. Perancangan Data

#### 3.1 Dekomposisi Data

##### 3.1.1 Deskripsi Entitas Data user

| Nama        | Tipe      | Panjang | Keterangan   |
|-------------|-----------|---------|--|
| idUser      | Integer   | -       | ID User, Primary Key   |
| idPelanggan | Character | 5       | ID Pelanggan, Foreign Key  |
| username    | Character | 5       | Username dari user   |
| pass        | Varchar   | 10      | Password dari user   |
| role        | Varchar   | 10      | Role dari user   |
| stat        | Varchar   | 10      | Status dari user   |
| is_deleted  | Boolean   | -       | Jika bernilai 1 maka user telah dihapus, sedangkan jika bernilai 0 maka user belum dihapus |

##### 3.1.2 Deskripsi Entitas Data pegawai

| Nama         | Tipe      | Panjang | Keterangan   |
|--------------|-----------|---------|--|
| idPegawai    | Character | 5       | ID Teknisi, Primary Key  |
| nama         | Varchar   | 50      | Nama dari teknisi  |
| alamat       | Varchar   | 100     | Alamat dari teknisi  |
| noTelp       | Varchar   | 15      | Nomor telepon dari teknisi   |
| emailPegawai | Varchar   | 50      | Email dari teknisi   |
| tanggalLahir | Date      | -       | Tanggal lahir dari teknisi   |
| statusHadir  | Character | 5       | Status kehadiran dari teknisi  |
| is_deleted   | Boolean   | -       | Jika bernilai 1 maka teknisi telah dihapus, sedangkan jika bernilai 0 maka teknisi belum dihapus |

### 3.1.3 Deskripsi Entitas Data pelanggan

| Nama                  | Tipe             | Panjang  | Keterangan                       |
|-----------------------|------------------|----------|----------------------------------|
| <b>idPelanggan</b>    | <b>Character</b> | <b>5</b> | <b>ID Pelanggan, Primary Key</b> |
| idUser                | Integer          | -        | ID User, Foreign Key             |
| nama                  | Varchar          | 50       | Nama dari pelanggan              |
| alamat                | Varchar          | 100      | Alamat dari pelanggan            |
| noTelp                | Varchar          | 15       | Nomor telepon dari pelanggan     |
| emailPelanggan        | Varchar          | 50       | Email dari pelanggan             |
| tanggalLahirPelanggan | Date             | -        | Tanggal lahir dari pelanggan     |
| jumlahBatal           | Integer          | -        | Jumlah pembatalan oleh pelanggan |

### 3.1.4 Deskripsi Entitas Data kendaraan

| Nama          | Tipe           | Panjang   | Keterangan                                 |
|---------------|----------------|-----------|--|
| <b>noPlat</b> | <b>Varchar</b> | <b>10</b> | <b>No Plat dari kendaraan, Primary Key</b> |
| idPelanggan   | Character      | 5         | ID pelanggan, Foreign Key                  |
| golongan      | Integer        | -         | Golongan dari tarif, Foreign Key           |
| merk          | Varchar        | 10        | Merk dari kendaraan                        |
| type          | Varchar        | 10        | Tipe dari kendaraan                        |

### 3.1.5 Deskripsi Entitas Data reservasi

| Nama               | Tipe           | Panjang  | Keterangan                       |
|--------------------|----------------|----------|----------------------------------|
| <b>idReservasi</b> | <b>Integer</b> | <b>-</b> | <b>ID Reservasi, Primary Key</b> |
| idPegawai          | Integer        | -        | ID Pegawai, Foreign Key          |

|               |         |    |  |
|---------------|---------|----|--|
| noPlat        | Varchar | 10 | No Plat dari kendaraan,<br>Foreign Key |
| tanggal       | Date    | -  | Tanggal melakukan<br>reservasi         |
| jam           | Time    | -  | Jam melakukan reservasi                |
| keluhan       | Varchar | 50 | Keluhan yang dialami                   |
| statusAntrian | Varchar | 10 | Status antrian dari<br>reservasi       |
| jenisService  | Varchar | 10 | Jenis service pada saat<br>reservasi   |
| biaya         | Double  | -  | Biaya reservasi                        |
| batasWaktu    | Integer | -  | Batas waktu kedatangan                 |

### 3.1.6 Deskripsi Entitas Data tarif

| Nama          | Tipe    | Panjang | Keterangan                        |
|---------------|---------|---------|-----------------------------------|
| golongan      | Integer | -       | Golongan tarif, Primary<br>Key    |
| deskripsi     | Varchar | 50      | Deskripsi dari golongan<br>tariff |
| serviceRingan | Double  | -       | Biaya dari service<br>ringan      |
| serviceBesar  | Double  | -       | Biaya dari service<br>besar       |

### 3.1.7 Deskripsi Entitas Data onderdil

| Nama          | Tipe    | Panjang | Keterangan                  |
|---------------|---------|---------|-----------------------------|
| idOnderdil    | Varchar | 15      | ID Onderdil, Primary<br>Key |
| namaOnderdil  | Varchar | 50      | Nama dari onderdil          |
| hargaOnderdil | Double  | -       | Harga dari onderdil         |
| merk          | Varchar | 10      | Merk dari onderdil          |

### 3.1.8 Deskripsi Entitas Data pembelianonderdil

| Nama        | Tipe    | Panjang | Keterangan                   |
|-------------|---------|---------|------------------------------|
| idPembelian | Integer | -       | ID Pembelian, Primary<br>Key |

|             |         |    |                           |
|-------------|---------|----|---------------------------|
| idReservasi | Integer | -  | ID Reservasi, Foreign Key |
| idOnderdil  | Varchar | 15 | ID Onderdil, Foreign Key  |

### 3.1.9 Deskripsi Entitas Data feedback

| Nama              | Tipe      | Panjang | Keterangan                      |
|-------------------|-----------|---------|---------------------------------|
| idFeedback        | Integer   | -       | ID Feedback, Primary Key        |
| idPelanggan       | Character | 5       | ID Pelanggan, Foreign Key       |
| deskripsiFeedback | Varchar   | 160     | Deskripsi dari feedback         |
| tanggalFeedback   | Date      | -       | Tanggal dari feedback           |
| noPengirim        | Varchar   | 20      | Nomor hp dari pengirim feedback |

### 3.1.10 Deskripsi Entitas Data inbox

| Nama              | Tipe      | Panjang | Keterangan                             |
|-------------------|-----------|---------|--|
| ID                | Integer   | -       | ID inbox, Primary Key                  |
| UpdatedInDB       | Timestamp | -       | Waktu ketika SMS terupdate di database |
| ReceivingDateTime | Timestamp | -       | Waktu ketika SMS diterima              |
| SenderNumber      | Varchar   | 20      | Nomor hp dari pengirim SMS             |
| TextDecoded       | Text      | -       | Pesan yang diterima                    |

### 3.1.11 Deskripsi Entitas Data outbox

| Nama         | Tipe      | Panjang | Keterangan                         |
|--------------|-----------|---------|------------------------------------|
| ID           | Integer   | -       | ID outbox, Primary Key             |
| InsertIntoDB | Timestamp | -       | Waktu ketika SMS masuk ke database |



## 4. Perancangan Antarmuka

### 4.1 Login User

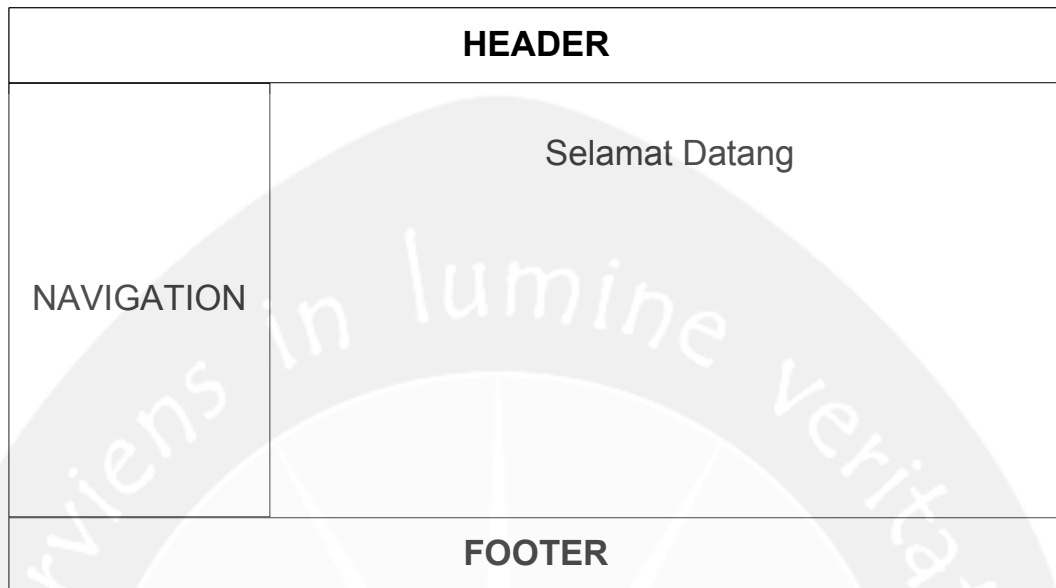
| HEADER   |
|--|
| <div style="text-align: center;"><div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"><p style="text-align: center; margin: 0;"><b>LOGIN</b></p><p>Username : <input style="width: 100%;" type="text"/></p><p>Password : <input style="width: 100%;" type="password"/></p><p style="text-align: center; margin-top: 5px;"><input type="button" value="LOGIN"/></p></div></div> |
| FOOTER   |

Gambar 4.1 Rancangan Antarmuka Login User

Antarmuka pada Gambar 4.1 merupakan antarmuka aplikasi web yang digunakan untuk melakukan login user. Untuk mendapat akses masuk ke dalam sistem, user harus mengisikan username dan password dengan benar pada textbox yang telah disediakan. Pada saat button login diklik, sistem akan mengecek username dan password yang diisi dengan data username dan password yang telah tersimpan di database. Jika data username dan password benar atau cocok maka user akan masuk ke dalam sistem, sebaliknya jika username dan password salah atau tidak cocok maka akan diberikan pesan kesalahan.



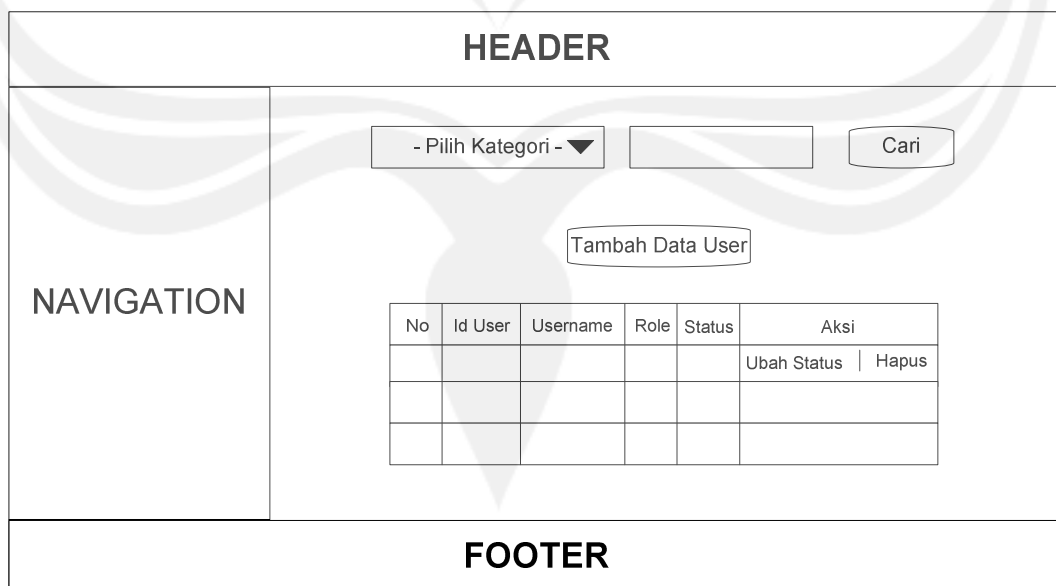
## 4.2 Halaman Utama



Gambar 4.2 Rancangan Halaman Utama

Antarmuka pada Gambar 4.2 merupakan antarmuka aplikasi web yang pertama kali muncul ketika user berhasil melakukan login dan masuk ke dalam sistem.

## 4.3 Pengelolaan User



Gambar 4.3 Rancangan Antarmuka Pengelolaan User

Antarmuka pada Gambar 4.3 merupakan antarmuka aplikasi web yang muncul ketika user masuk sebagai administrator dan memilih menu Pengelolaan User yang ada di navigation. Pada antarmuka ini, administrator dapat melihat tabel yang berisi data-data user, melakukan pencarian data user, melakukan hapus data user, menuju halaman Tambah Data User, dan menuju halaman Ubah Status User.

Untuk melakukan pencarian data user, sebelumnya administrator harus memilih combo box kategori dan mengisikan kata kunci, kemudian klik tombol cari. Jika data yang dicari telah ditemukan, maka sistem akan menampilkan data yang ditemukan ke dalam tabel. Sebaliknya, jika data yang dicari tidak ditemukan, maka sistem akan menampilkan pesan bahwa data tidak ditemukan.

Untuk melakukan hapus data user, sebelumnya administrator harus melakukan klik pada tombol Hapus yang terletak di kolom Aksi pada tabel di Gambar 4.3. Setelah itu, sistem akan menampilkan konfirmasi penghapusan data user, jika administrator memilih Ya, maka data user dihapus, jika dipilih Tidak, maka data user tidak dihapus. Fungsi hapus data user tidak dapat dilakukan untuk data user yang memiliki role sebagai administrator.

Untuk menuju halaman Tambah Data User, sebelumnya administrator harus melakukan klik pada tombol Tambah Data User. Antarmuka Tambah Data User dapat dilihat pada Gambar 4.4.

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | DPPL – CRS | 63/ 92 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

| <b>HEADER</b>     |  |
|-------------------|--|
| <b>NAVIGATION</b> | <p>Id user <input type="text"/></p> <p>Username <input type="text"/></p> <p>Password <input type="text"/></p> <p>Konfirmasi Password <input type="text"/></p> <p style="text-align: right;"> <input type="button" value="Tambah"/> <input type="button" value="Kembali"/> </p> |
| <b>FOOTER</b>     |  |

**Gambar 4.4 Rancangan Antarmuka Tambah Data User**

Antarmuka pada Gambar 4.4 digunakan oleh administrator untuk melakukan tambah data user. Untuk melakukan tambah data user, sebelumnya administrator harus mengisikan semua data yang diperlukan. Setelah itu, administrator melakukan klik pada tombol Tambah. Jika data yang dimasukkan benar, maka sistem akan menambahkan data tersebut ke dalam database dan menampilkan antarmuka Pengelolaan User pada Gambar 4.3, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan User yang ada pada Gambar 4.3, maka administrator harus melakukan klik pada tombol Kembali.

Untuk menuju halaman Ubah Status User, sebelumnya administrator harus melakukan klik pada tombol Ubah Status yang terletak di kolom Aksi pada tabel di Gambar 4.3. Antarmuka Ubah Status User dapat dilihat pada Gambar 4.5.

| HEADER     |  |
|------------|--|
| NAVIGATION | Username <input type="text"/><br>Status <input type="text" value="▼"/><br><input type="button" value="Ubah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

Gambar 4.5 Rancangan Antarmuka Ubah Status User

Antarmuka pada Gambar 4.5 digunakan oleh administrator untuk melakukan ubah status user. Untuk melakukan ubah status user, sebelumnya administrator harus mengisikan semua data yang diperlukan. Setelah itu, administrator melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah data yang ada pada database dengan data yang telah dimasukkan dan menampilkan antarmuka Pengelolaan User pada Gambar 4.3, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Fungsi ubah status user tidak dapat dilakukan untuk data user yang memiliki role sebagai administrator. Untuk kembali ke antarmuka Pengelolaan User yang ada pada Gambar 4.3, maka administrator harus melakukan klik pada tombol Kembali.

#### 4.4 Pengelolaan Teknisi

| HEADER     |                      |      |                      |               |         |       |              |
|------------|----------------------|------|----------------------|---------------|---------|-------|--------------|
| NAVIGATION | - Pilih Kategori - ▼ |      | <input type="text"/> | Cari          |         |       |              |
|            | Tambah Data Teknisi  |      |                      |               |         |       |              |
| No         | Id Teknisi           | Nama | Alamat               | Tanggal Lahir | No Telp | Email | Aksi         |
|            |                      |      |                      |               |         |       | Ubah   Hapus |
|            |                      |      |                      |               |         |       |              |
|            |                      |      |                      |               |         |       |              |

Gambar 4.6 Rancangan Antarmuka Pengelolaan Teknisi

Antarmuka pada Gambar 4.6 merupakan antarmuka aplikasi web yang muncul ketika user masuk sebagai administrator dan memilih menu Pengelolaan Teknisi yang ada di navigation. Pada antarmuka ini, administrator dapat melihat tabel yang berisi data-data teknisi, melakukan pencarian data teknisi, melakukan hapus data teknisi, menuju halaman Tambah Data Teknisi, dan menuju halaman Ubah Data Teknisi.

Untuk melakukan pencarian data teknisi, sebelumnya administrator harus memilih combo box kategori dan mengisikan kata kunci, kemudian klik tombol cari. Jika data yang dicari telah ditemukan, maka sistem akan menampilkan data yang ditemukan ke dalam tabel. Sebaliknya, jika data yang dicari tidak ditemukan, maka sistem akan menampilkan pesan bahwa data tidak ditemukan.

Untuk melakukan hapus data teknisi, sebelumnya administrator harus melakukan klik pada tombol Hapus

yang terletak di kolom Aksi pada tabel di Gambar 4.6. Setelah itu, sistem akan menampilkan konfirmasi penghapusan data teknisi, jika administrator memilih Ya, maka data teknisi dihapus, jika dipilih Tidak, maka data teknisi tidak dihapus.

Untuk menuju halaman Tambah Data Teknisi, sebelumnya administrator harus melakukan klik pada tombol Tambah Data Teknisi. Antarmuka Tambah Data Teknisi dapat dilihat pada Gambar 4.7.

| HEADER     |  |
|------------|--|
| NAVIGATION | Id Teknisi <input type="text"/>  |
|            | Nama <input type="text"/>  |
|            | Alamat <input type="text"/>  |
|            | No Telp <input type="text"/>   |
|            | Email <input type="text"/>   |
|            | Tanggal Lahir <input type="text"/>   |
|            | <input type="button" value="Tambah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

**Gambar 4.7 Rancangan Antarmuka Tambah Data Teknisi**

Antarmuka pada Gambar 4.7 digunakan oleh administrator untuk melakukan tambah data teknisi. Untuk melakukan tambah data teknisi, sebelumnya administrator harus mengisi semua data yang diperlukan. Setelah itu, administrator melakukan klik pada tombol Tambah. Jika data yang dimasukkan benar, maka sistem akan menambahkan data tersebut ke dalam database dan menampilkan antarmuka Pengelolaan Teknisi pada Gambar 4.6, sebaliknya jika data yang dimasukkan

salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Teknisi yang ada pada Gambar 4.6, maka administrator harus melakukan klik pada tombol Kembali.

Untuk menuju halaman Ubah Data Teknisi, sebelumnya administrator harus melakukan klik pada tombol Ubah yang terletak di kolom Aksi pada tabel di Gambar 4.6. Antarmuka Ubah Data Teknisi dapat dilihat pada Gambar 4.8.

| HEADER     |  |  |
|------------|--|--|
| NAVIGATION | <p>Id Teknisi <input type="text"/></p> <p>Nama <input type="text"/></p> <p>Alamat <input type="text"/></p> <p>No Telp <input type="text"/></p> <p>Email <input type="text"/></p> <p>Tanggal Lahir <input type="text"/></p> <p style="text-align: center;"> <input type="button" value="Ubah"/> <input type="button" value="Kembali"/> </p> |  |
|            | FOOTER   |  |

Gambar 4.8 Rancangan Antarmuka Ubah Data Teknisi

Antarmuka pada Gambar 4.8 digunakan oleh administrator untuk melakukan ubah data teknisi. Untuk melakukan ubah data teknisi, sebelumnya administrator harus mengisikan semua data yang diperlukan. Setelah itu, administrator melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah data yang ada pada database dengan data yang telah dimasukkan dan menampilkan antarmuka Pengelolaan

Teknisi pada Gambar 4.6, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Teknisi yang ada pada Gambar 4.6, maka administrator harus melakukan klik pada tombol Kembali.

#### 4.5 Pengelolaan Pelanggan

The interface is divided into three main sections: HEADER, NAVIGATION, and FOOTER. The NAVIGATION sidebar is on the left. The main content area contains a search form with a dropdown menu for categories, a text input field, and a 'Cari' button. Below the search form is a 'Tambah Data Pelanggan' button. At the bottom of the main content area is a table with the following structure:

| No | Id Teknisi | IdUser | Nama | Alamat | Tanggal Lahir | No Telp | Email | Aksi |
|----|------------|--------|------|--------|---------------|---------|-------|------|
|    |            |        |      |        |               |         |       | Ubah |
|    |            |        |      |        |               |         |       |      |
|    |            |        |      |        |               |         |       |      |

Gambar 4.9 Rancangan Antarmuka Pengelolaan Pelanggan

Antarmuka pada Gambar 4.9 merupakan antarmuka aplikasi web yang muncul ketika user masuk sebagai operator dan memilih menu Pengelolaan Pelanggan yang ada di navigation. Pada antarmuka ini, operator dapat melihat tabel yang berisi data-data pelanggan, melakukan pencarian data pelanggan, menuju halaman Tambah Data Pelanggan, dan menuju halaman Ubah Data Pelanggan.

Untuk melakukan pencarian data pelanggan, sebelumnya operator harus memilih combo box kategori dan mengisikan kata kunci, kemudian klik tombol cari. Jika data yang dicari telah ditemukan, maka sistem akan menampilkan data yang ditemukan ke dalam tabel.



Sebaliknya, jika data yang dicari tidak ditemukan, maka sistem akan menampilkan pesan bahwa data tidak ditemukan.

Untuk menuju halaman Tambah Data Pelanggan, sebelumnya operator harus melakukan klik pada tombol Tambah Data Pelanggan. Antarmuka Tambah Data Pelanggan dapat dilihat pada Gambar 4.10.

| HEADER     |  |
|------------|--|
| NAVIGATION | Id Pelanggan <input type="text"/>  |
|            | Nama <input type="text"/>  |
|            | Alamat <input type="text"/>  |
|            | No Telp <input type="text"/>   |
|            | Email <input type="text"/>   |
|            | Tanggal Lahir <input type="text"/>   |
|            | <input type="button" value="Tambah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

Gambar 4.10 Rancangan Antarmuka Tambah Data Pelanggan

Antarmuka pada Gambar 4.10 digunakan oleh operator untuk melakukan tambah data pelanggan. Untuk melakukan tambah data pelanggan, sebelumnya operator harus mengisi semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Tambah. Jika data yang dimasukkan benar, maka sistem akan menambahkan data tersebut ke dalam database dan menampilkan antarmuka Pengelolaan Pelanggan pada Gambar 4.9, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Pelanggan yang ada pada Gambar

4.9, maka operator harus melakukan klik pada tombol Kembali.

Untuk menuju halaman Ubah Data Pelanggan, sebelumnya operator harus melakukan klik pada tombol Ubah yang terletak di kolom Aksi pada tabel di Gambar 4.9. Antarmuka Ubah Data Pelanggan dapat dilihat pada Gambar 4.11.

| HEADER     |  |
|------------|--|
| NAVIGATION | Id Pelanggan <input type="text"/>  |
|            | Nama <input type="text"/>  |
|            | Alamat <input type="text"/>  |
|            | No Telp <input type="text"/>   |
|            | Email <input type="text"/>   |
|            | Tanggal Lahir <input type="text"/>   |
|            | <input type="button" value="Ubah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

**Gambar 4.11 Rancangan Antarmuka Ubah Data Pelanggan**

Antarmuka pada Gambar 4.11 digunakan oleh operator untuk melakukan ubah data pelanggan. Untuk melakukan ubah data pelanggan, sebelumnya operator harus mengisikan semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah data yang ada pada database dengan data yang telah dimasukkan dan menampilkan antarmuka Pengelolaan Pelanggan pada Gambar 4.9, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan

Pelanggan yang ada pada Gambar 4.9, maka operator harus melakukan klik pada tombol Kembali.

#### 4.6 Pengelolaan Kendaraan

The interface is divided into three main sections: HEADER, NAVIGATION, and FOOTER.

**HEADER:** Contains the title "HEADER".

**NAVIGATION:** Contains a search area with a dropdown menu labeled "- Pilih Kategori -", an input field, and a "Cari" button. Below this is a "Tambah Data Kendaraan" button.

**Table:** A table with 7 columns: No, No Plat, Id Pelanggan, Golongan, Merk, Type, and Aksi. The first row has a "Ubah" button in the Aksi column.

**FOOTER:** Contains the title "FOOTER".

Gambar 4.12 Rancangan Antarmuka Pengelolaan Kendaraan

Antarmuka pada Gambar 4.12 merupakan antarmuka aplikasi web yang muncul ketika user masuk sebagai operator dan memilih menu Pengelolaan Kendaraan yang ada di navigation. Pada antarmuka ini, operator dapat melihat tabel yang berisi data-data kendaraan, melakukan pencarian data kendaraan, menuju halaman Tambah Data Kendaraan, dan menuju halaman Ubah Data Kendaraan.

Untuk melakukan pencarian data kendaraan, sebelumnya operator harus memilih combo box kategori dan mengisikan kata kunci, kemudian klik tombol cari. Jika data yang dicari telah ditemukan, maka sistem akan menampilkan data yang ditemukan ke dalam tabel. Sebaliknya, jika data yang dicari tidak ditemukan, maka

sistem akan menampilkan pesan bahwa data tidak ditemukan.

Untuk menuju halaman Tambah Data Kendaraan, sebelumnya operator harus melakukan klik pada tombol Tambah Data Kendaraan. Antarmuka Tambah Data Kendaraan dapat dilihat pada Gambar 4.13.

| HEADER     |  |
|------------|--|
| NAVIGATION | No Plat <input type="text"/>   |
|            | Id Pelanggan <input type="text"/>  |
|            | Golongan <input type="text"/>  |
|            | Merk <input type="text"/>  |
|            | Type <input type="text"/>  |
|            | <input type="button" value="Tambah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

Gambar 4.13 Rancangan Antarmuka Tambah Data Kendaraan

Antarmuka pada Gambar 4.13 digunakan oleh operator untuk melakukan tambah data kendaraan. Untuk melakukan tambah data kendaraan, sebelumnya operator harus mengisi semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Tambah. Jika data yang dimasukkan benar, maka sistem akan menambahkan data tersebut ke dalam database dan menampilkan antarmuka Pengelolaan Kendaraan pada Gambar 4.12, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Kendaraan yang ada pada Gambar

4.12, maka operator harus melakukan klik pada tombol Kembali.

Untuk menuju halaman Ubah Data Kendaraan, sebelumnya operator harus melakukan klik pada tombol Ubah yang terletak di kolom Aksi pada tabel di Gambar 4.12. Antarmuka Ubah Data Kendaraan dapat dilihat pada Gambar 4.14.

| HEADER     |  |
|------------|--|
| NAVIGATION | No Plat <input type="text"/>   |
|            | Id Pelanggan <input type="text"/>  |
|            | Golongan <input type="text"/>  |
|            | Merk <input type="text"/>  |
|            | Type <input type="text"/>  |
|            | <input type="button" value="Ubah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

Gambar 4.14 Rancangan Antarmuka Ubah Data Kendaraan

Antarmuka pada Gambar 4.14 digunakan oleh operator untuk melakukan ubah data kendaraan. Untuk melakukan ubah data kendaraan, sebelumnya operator harus mengisi semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah data yang ada pada database dengan data yang telah dimasukkan dan menampilkan antarmuka Pengelolaan Kendaraan pada Gambar 4.12, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan

kesalahan. Untuk kembali ke antarmuka Pengelolaan Kendaraan yang ada pada Gambar 4.12, maka operator harus melakukan klik pada tombol Kembali.

#### 4.7 Pengelolaan Reservasi

| HEADER     |  |              |            |         |         |     |         |        |         |      |       |              |            |         |         |     |         |        |         |      |  |  |  |  |  |  |  |  |  |  |      |       |  |  |  |  |  |  |  |  |  |  |  |
|------------|--|--------------|------------|---------|---------|-----|---------|--------|---------|------|-------|--------------|------------|---------|---------|-----|---------|--------|---------|------|--|--|--|--|--|--|--|--|--|--|------|-------|--|--|--|--|--|--|--|--|--|--|--|
| NAVIGATION | <div style="display: flex; justify-content: space-between; align-items: center;"> <span>- Pilih Kategori - ▼</span> <input type="text"/> <input type="button" value="Cari"/> </div> <div style="text-align: center; margin-top: 10px;"> <input type="button" value="Tambah Data Reservasi"/> </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No</th> <th>Id Reservasi</th> <th>Id Teknisi</th> <th>No Plat</th> <th>Tanggal</th> <th>Jam</th> <th>Keluhan</th> <th>Status</th> <th>Service</th> <th colspan="2">Aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Ubah</td> <td>Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> |              |            |         |         |     |         |        |         |      | No    | Id Reservasi | Id Teknisi | No Plat | Tanggal | Jam | Keluhan | Status | Service | Aksi |  |  |  |  |  |  |  |  |  |  | Ubah | Hapus |  |  |  |  |  |  |  |  |  |  |  |
|            | No   | Id Reservasi | Id Teknisi | No Plat | Tanggal | Jam | Keluhan | Status | Service | Aksi |       |              |            |         |         |     |         |        |         |      |  |  |  |  |  |  |  |  |  |  |      |       |  |  |  |  |  |  |  |  |  |  |  |
|            |  |              |            |         |         |     |         |        |         | Ubah | Hapus |              |            |         |         |     |         |        |         |      |  |  |  |  |  |  |  |  |  |  |      |       |  |  |  |  |  |  |  |  |  |  |  |
|            |  |              |            |         |         |     |         |        |         |      |       |              |            |         |         |     |         |        |         |      |  |  |  |  |  |  |  |  |  |  |      |       |  |  |  |  |  |  |  |  |  |  |  |
| FOOTER     |  |              |            |         |         |     |         |        |         |      |       |              |            |         |         |     |         |        |         |      |  |  |  |  |  |  |  |  |  |  |      |       |  |  |  |  |  |  |  |  |  |  |  |

Gambar 4.15 Rancangan Antarmuka Pengelolaan Reservasi

Antarmuka pada Gambar 4.15 merupakan antarmuka aplikasi web yang muncul ketika user masuk sebagai operator dan memilih menu Pengelolaan Reservasi yang ada di navigation. Pada antarmuka ini, operator dapat melihat tabel yang berisi data-data reservasi, melakukan pencarian data reservasi, melakukan hapus data reservasi, menuju halaman Tambah Data Reservasi, dan menuju halaman Ubah Data Reservasi.

Untuk melakukan pencarian data reservasi, sebelumnya operator harus memilih combo box kategori dan mengisikan kata kunci, kemudian klik tombol cari. Jika data yang dicari telah ditemukan, maka sistem akan menampilkan data yang ditemukan ke dalam tabel. Sebaliknya, jika data yang dicari tidak ditemukan, maka sistem akan menampilkan pesan bahwa data tidak ditemukan.

Untuk melakukan hapus data reservasi, sebelumnya operator harus melakukan klik pada tombol Hapus yang terletak di kolom Aksi pada tabel di Gambar 4.15. Setelah itu, sistem akan menampilkan konfirmasi penghapusan data reservasi, jika operator memilih Ya, maka data reservasi dihapus, jika dipilih Tidak, maka data reservasi tidak dihapus.

Untuk menuju halaman Tambah Data Reservasi, sebelumnya operator harus melakukan klik pada tombol Tambah Data Reservasi. Antarmuka Tambah Data Reservasi dapat dilihat pada Gambar 4.16.

**HEADER**

Id Reservasi

Id Teknisi

No Plat

Jam Service

Keluhan

Jenis Service

**NAVIGATION**

| Id Onderdil | Nama Onderdil | Harga | Check                    |
|-------------|---------------|-------|--------------------------|
|             |               |       | <input type="checkbox"/> |
|             |               |       | <input type="checkbox"/> |
|             |               |       | <input type="checkbox"/> |

Tambah Kembali

**FOOTER**

Gambar 4.16 Rancangan Antarmuka Tambah Data Reservasi

Antarmuka pada Gambar 4.16 digunakan oleh operator untuk melakukan tambah data reservasi. Untuk melakukan tambah data reservasi, sebelumnya operator harus mengisi semua data yang diperlukan. Setelah itu,

operator melakukan klik pada tombol Tambah. Jika data yang dimasukkan benar, maka sistem akan menambahkan data tersebut ke dalam database dan menampilkan antarmuka Pengelolaan Reservasi pada Gambar 4.15, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Kendaraan yang ada pada Gambar 4.15, maka operator harus melakukan klik pada tombol Kembali.

Untuk menuju halaman Ubah Data Reservasi, sebelumnya operator harus melakukan klik pada tombol Ubah yang terletak di kolom Aksi pada tabel di Gambar 4.15. Antarmuka Ubah Data Reservasi dapat dilihat pada Gambar 4.17.

HEADER

NAVIGATION

Id Reservasi

Id Teknisi

No Plat

Jam Service

Keluhan

Jenis Service

| Id Onderdil | Nama Onderdil | Harga | Check                    |
|-------------|---------------|-------|--------------------------|
|             |               |       | <input type="checkbox"/> |
|             |               |       | <input type="checkbox"/> |
|             |               |       | <input type="checkbox"/> |

Kembali

FOOTER

**Gambar 4.17 Rancangan Antarmuka Ubah Data Reservasi**



Antarmuka pada Gambar 4.17 digunakan oleh operator untuk melakukan ubah data reservasi. Untuk melakukan ubah data reservasi, sebelumnya operator harus mengisikan semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah data yang ada pada database dengan data yang telah dimasukkan dan menampilkan antarmuka Pengelolaan Reservasi pada Gambar 4.15, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Reservasi yang ada pada Gambar 4.15, maka operator harus melakukan klik pada tombol Kembali.

#### 4.8 Pengelolaan Onderdil

| HEADER     |   |      |       |      |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |
|------------|---|------|-------|------|----|-------------|------|-------|------|--|--|--|--|------|--|--|--|--|--|--|--|--|--|
| NAVIGATION | <input type="text" value="- Pilih Kategori -"/> <input type="text"/> <input type="button" value="Cari"/>  |      |       |      |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |
|            | <input type="button" value="Tambah Data Onderdil"/>   |      |       |      |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |
|            | <table border="1"> <thead> <tr> <th>No</th> <th>Id Onderdil</th> <th>Nama</th> <th>Harga</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td>Ubah</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table> |      |       |      | No | Id Onderdil | Nama | Harga | Aksi |  |  |  |  | Ubah |  |  |  |  |  |  |  |  |  |
| No         | Id Onderdil   | Nama | Harga | Aksi |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |
|            |   |      |       | Ubah |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |
|            |   |      |       |      |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |
|            |   |      |       |      |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |
| FOOTER     |   |      |       |      |    |             |      |       |      |  |  |  |  |      |  |  |  |  |  |  |  |  |  |

Gambar 4.18 Rancangan Antarmuka Pengelolaan Onderdil

Antarmuka pada Gambar 4.18 merupakan antarmuka aplikasi web yang muncul ketika user masuk sebagai operator dan memilih menu Pengelolaan Onderdil yang ada di navigation. Pada antarmuka ini, operator dapat

melihat tabel yang berisi data-data onderdil, melakukan pencarian data onderdil, menuju halaman Tambah Data Onderdil, dan menuju halaman Ubah Data Onderdil.

Untuk melakukan pencarian data onderdil, sebelumnya operator harus memilih combo box kategori dan mengisikan kata kunci, kemudian klik tombol cari. Jika data yang dicari telah ditemukan, maka sistem akan menampilkan data yang ditemukan ke dalam tabel. Sebaliknya, jika data yang dicari tidak ditemukan, maka sistem akan menampilkan pesan bahwa data tidak ditemukan.

Untuk menuju halaman Tambah Data Onderdil, sebelumnya operator harus melakukan klik pada tombol Tambah Data Onderdil. Antarmuka Tambah Data Onderdil dapat dilihat pada Gambar 4.19.

| HEADER                                |                                    |  |
|---------------------------------------|------------------------------------|--|
| NAVIGATION                            | Id Onderdil <input type="text"/>   |  |
|                                       | Nama Onderdil <input type="text"/> |  |
| Harga Onderdil <input type="text"/>   |                                    |  |
| <input type="button" value="Tambah"/> |                                    | <input type="button" value="Kembali"/> |
| FOOTER                                |                                    |  |

Gambar 4.19 Rancangan Antarmuka Tambah Data Onderdil

Antarmuka pada Gambar 4.19 digunakan oleh operator untuk melakukan tambah data. Untuk melakukan tambah data onderdil, sebelumnya operator harus mengisikan

semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Tambah. Jika data yang dimasukkan benar, maka sistem akan menambahkan data tersebut ke dalam database dan menampilkan antarmuka Pengelolaan Onderdil pada Gambar 4.18, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Kendaraan yang ada pada Gambar 4.18, maka operator harus melakukan klik pada tombol Kembali.

Untuk menuju halaman Ubah Data Onderdil, sebelumnya operator harus melakukan klik pada tombol Ubah yang terletak di kolom Aksi pada tabel di Gambar 4.18. Antarmuka Ubah Data Onderdil dapat dilihat pada Gambar 4.20.

| HEADER     |  |
|------------|--|
| NAVIGATION | Id Onderdil <input type="text"/>   |
|            | Nama Onderdil <input type="text"/>   |
|            | Harga Onderdil <input type="text"/>  |
|            | <input type="button" value="Ubah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

Gambar 4.20 Rancangan Antarmuka Ubah Data Onderdil

Antarmuka pada Gambar 4.20 digunakan oleh operator untuk melakukan ubah data onderdil. Untuk melakukan ubah data onderdil, sebelumnya operator harus

mengisikan semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah data yang ada pada database dengan data yang telah dimasukkan dan menampilkan antarmuka Pengelolaan Onderdil pada Gambar 4.18, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Onderdil yang ada pada Gambar 4.18, maka operator harus melakukan klik pada tombol Kembali.

#### 4.9 Pengelolaan Tarif

| HEADER     |  |          |           |                      |                     |      |
|------------|--|----------|-----------|----------------------|---------------------|------|
| NAVIGATION | <input type="button" value="Tambah Data Tarif"/> |          |           |                      |                     |      |
|            | No   | Golongan | Deskripsi | Biaya Service Ringan | Biaya Service Besar | Aksi |
|            |  |          |           |                      |                     | Ubah |
|            |  |          |           |                      |                     |      |
|            |  |          |           |                      |                     |      |
| FOOTER     |  |          |           |                      |                     |      |

Gambar 4.21 Rancangan Antarmuka Pengelolaan Tarif

Antarmuka pada Gambar 4.21 merupakan antarmuka aplikasi web yang muncul ketika user masuk sebagai operator dan memilih menu Pengelolaan Tarif yang ada di navigation. Pada antarmuka ini, operator dapat melihat tabel yang berisi data-data tarif, menuju halaman Tambah Data Tarif, dan menuju halaman Ubah Data Tarif.

Untuk menuju halaman Tambah Data Tarif, sebelumnya operator harus melakukan klik pada tombol Tambah Data

Tarif. Antarmuka Tambah Data Tarif dapat dilihat pada Gambar 4.22.

| HEADER     |  |
|------------|--|
| NAVIGATION | Golongan <input type="text"/>  |
|            | Deskripsi <input type="text"/>   |
|            | Biaya Service Ringan <input type="text"/>                                    |
|            | Biaya Service Besar <input type="text"/>                                     |
|            | <input type="button" value="Tambah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

Gambar 4.22 Rancangan Antarmuka Tambah Data Tarif

Antarmuka pada Gambar 4.22 digunakan oleh operator untuk melakukan tambah data tarif. Untuk melakukan tambah data tarif, sebelumnya operator harus mengisikan semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Tambah. Jika data yang dimasukkan benar, maka sistem akan menambahkan data tersebut ke dalam database dan menampilkan antarmuka Pengelolaan Tarif pada Gambar 4.21, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Tarif yang ada pada Gambar 4.21, maka operator harus melakukan klik pada tombol Kembali.

Untuk menuju halaman Ubah Data Tarif, sebelumnya operator harus melakukan klik pada tombol Ubah yang terletak di kolom Aksi pada tabel di Gambar 4.21.

Antarmuka Ubah Data Tarif dapat dilihat pada Gambar 4.23.

| HEADER     |  |
|------------|--|
| NAVIGATION | Golongan <input type="text"/>  |
|            | Deskripsi <input type="text"/>   |
|            | Biaya Service Ringan <input type="text"/>                                  |
|            | Biaya Service Besar <input type="text"/>                                   |
|            | <input type="button" value="Ubah"/> <input type="button" value="Kembali"/> |
| FOOTER     |  |

Gambar 4.23 Rancangan Antarmuka Ubah Data Tarif

Antarmuka pada Gambar 4.23 digunakan oleh operator untuk melakukan ubah data tarif. Untuk melakukan ubah data tarif, sebelumnya operator harus mengisikan semua data yang diperlukan. Setelah itu, operator melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah data yang ada pada database dengan data yang telah dimasukkan dan menampilkan antarmuka Pengelolaan Tarif pada Gambar 4.21, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan. Untuk kembali ke antarmuka Pengelolaan Tarif yang ada pada Gambar 4.21, maka operator harus melakukan klik pada tombol Kembali.

#### 4.10 Ubah Password

| HEADER     |   |
|------------|---|
| NAVIGATION | <p>Username <input type="text"/></p> <p>Password Lama <input type="text"/></p> <p>Password Baru <input type="text"/></p> <p>Konfirmasi Password <input type="text"/></p> <p><input type="button" value="Ubah"/></p> |
|            | FOOTER  |

Gambar 4.24 Rancangan Antarmuka Ubah Password

Antarmuka pada Gambar 4.24 merupakan antarmuka aplikasi web yang digunakan untuk melakukan ubah password pada username yang bersangkutan. Sebelum melakukan ubah password, user yang bersangkutan diharuskan memasukkan data-data yang diperlukan. Setelah itu, user yang bersangkutan melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah password lama yang ada pada database dengan password baru yang telah dimasukkan, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan.

#### 4.11 Login Mobile Web

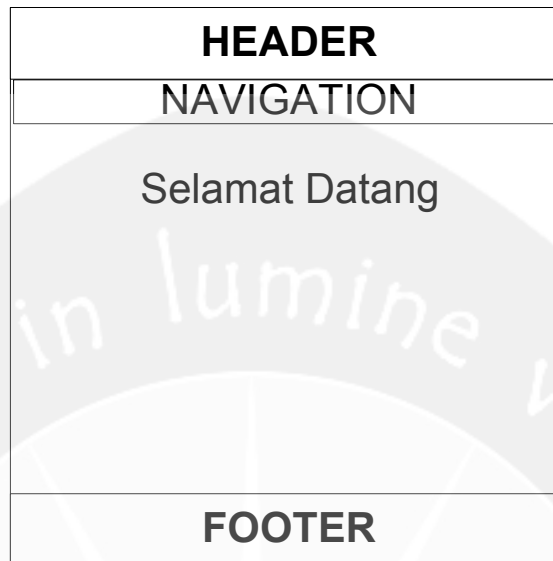
The image shows a wireframe for a mobile web login page. It is divided into three horizontal sections: a top section labeled 'HEADER', a middle section for the login form, and a bottom section labeled 'FOOTER'. The login form contains two text input fields: one for 'Username' and one for 'Password'. Below these fields is a button labeled 'LOGIN'. A large, semi-transparent watermark with the Latin phrase 'serviens in lumine veritatis' is overlaid on the entire wireframe.

Gambar 4.25 Rancangan Antarmuka Login Mobile Web

Antarmuka pada Gambar 4.25 merupakan antarmuka aplikasi mobile web yang digunakan untuk melakukan login user. Untuk mendapat akses masuk ke dalam sistem, user harus mengisikan username dan password dengan benar pada textbox yang telah disediakan. Pada saat button login diklik, sistem akan mengecek username dan password yang diisi dengan data username dan password yang telah tersimpan di database. Jika data username dan password benar atau cocok maka user akan masuk ke dalam sistem, sebaliknya jika username dan password salah atau tidak cocok maka akan diberikan pesan kesalahan.



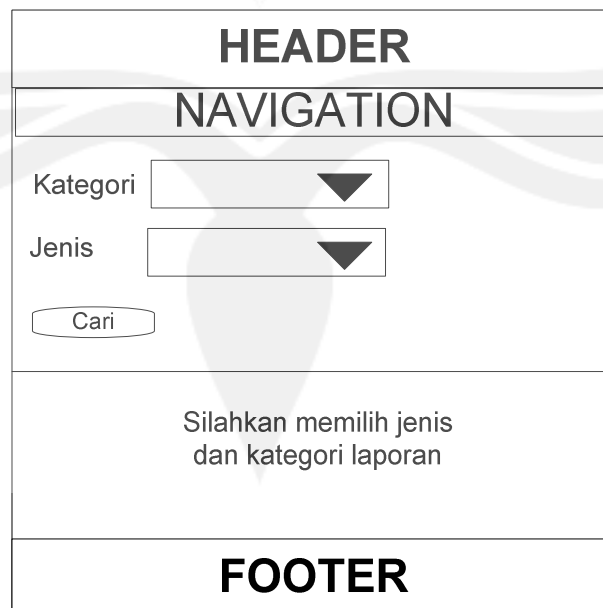
#### 4.12 Halaman Utama Mobile Web



Gambar 4.26 Rancangan Antarmuka Halaman Utama Mobile Web

Antarmuka pada Gambar 4.26 merupakan antarmuka aplikasi mobile web yang pertama kali muncul ketika user berhasil melakukan login dan masuk ke dalam sistem.

#### 4.13 Laporan Mobile Web



Gambar 4.27 Rancangan Antarmuka Laporan Mobile Web

Antarmuka pada Gambar 4.27 merupakan antarmuka aplikasi mobile web yang muncul ketika user memilih untuk melakukan monitoring laporan. User diminta untuk memilih kategori laporan dan jenis laporan. Kategori laporan meliputi pemasukan, teknisi, dan feedback. Jenis laporan meliputi harian dan bulanan.

| <b>HEADER</b>                       |  |
|-------------------------------------|--|
| <b>NAVIGATION</b>                   |  |
| Kategori                            | <input type="text" value="Pemasukan"/> ▼ |
| Jenis                               | <input type="text"/> ▼                   |
| <input type="button" value="Cari"/> |  |
| Laporan Total Pemasukan             |  |
| Total : _                           |  |
| <b>FOOTER</b>                       |  |

**Gambar 4.28 Rancangan Antarmuka Laporan Total Pemasukan Mobile Web**

Antarmuka pada Gambar 4.28 merupakan antarmuka aplikasi mobile web yang muncul ketika user memilih kategori laporan pemasukan dan memilih jenis laporan harian/bulanan dan melakukan klik pada tombol Cari.

| <b>HEADER</b>                       |           |            |
|-------------------------------------|-----------|------------|
| <b>NAVIGATION</b>                   |           |            |
| Kategori                            | Teknisi ▼ |            |
| Jenis                               | ▼         |            |
| <input type="button" value="Cari"/> |           |            |
| Laporan Teknisi                     |           |            |
| Id Teknisi                          | Nama      | Pengerjaan |
|                                     |           |            |
|                                     |           |            |
| <b>FOOTER</b>                       |           |            |

**Gambar 4.29 Rancangan Antarmuka Laporan Teknisi Mobile Web**

Antarmuka pada Gambar 4.29 merupakan antarmuka aplikasi mobile web yang muncul ketika user memilih kategori laporan teknisi dan memilih jenis laporan harian/bulanan dan melakukan klik pada tombol Cari.

| <b>HEADER</b>                       |            |    |       |
|-------------------------------------|------------|----|-------|
| <b>NAVIGATION</b>                   |            |    |       |
| Kategori                            | Feedback ▼ |    |       |
| Jenis                               | ▼          |    |       |
| <input type="button" value="Cari"/> |            |    |       |
| Laporan Feedback                    |            |    |       |
| Id Pelanggan                        | Feedback   | HP | Aksi  |
|                                     |            |    | Balas |
|                                     |            |    | Balas |
|                                     |            |    | Balas |
| <b>FOOTER</b>                       |            |    |       |

**Gambar 4.30 Rancangan Antarmuka Laporan Feedback Mobile Web**

Antarmuka pada Gambar 4.30 merupakan antarmuka aplikasi mobile web yang muncul ketika user memilih kategori laporan feedback dan memilih jenis laporan harian/bulanan dan melakukan klik pada tombol Cari.

#### 4.14 Balas Feedback Mobile Web

|  |  |
|--|--|
| <b>HEADER</b>                              |  |
| <b>NAVIGATION</b>                          |  |
| <b>Balas Feedback</b>                      |  |
| Nomor                                      | <input type="text"/>                   |
| Pesan                                      | <input type="text"/>                   |
|  | 160                                    |
| <input type="button" value="Kirim Pesan"/> | <input type="button" value="Kembali"/> |
| <b>FOOTER</b>                              |  |

Gambar 4.31 Rancangan Antarmuka Balas Feedback Mobile Web

Antarmuka pada Gambar 4.31 merupakan antarmuka aplikasi mobile web yang muncul ketika user melakukan klik tombol Balas pada kolom Aksi yang terdapat pada Gambar 4.30. Antarmuka ini digunakan oleh user untuk memberikan balasan feedback pada data feedback yang terdapat pada Gambar 4.30.

#### 4.15 Ubah Password Mobile Web

The image shows a wireframe for a mobile web password change interface. It is divided into four main sections: a top 'HEADER', a 'NAVIGATION' bar, a central content area, and a bottom 'FOOTER'. The content area contains four input fields with labels: 'Username', 'Pass Lama', 'Pass Baru', and 'Konfirmasi'. Each label is followed by a rectangular input box. Below these fields is a button labeled 'Ubah'.

Gambar 4.32 Rancangan Antarmuka Ubah Password Mobile Web

Antarmuka pada Gambar 4.32 merupakan antarmuka aplikasi mobile web yang digunakan untuk melakukan ubah password pada username yang bersangkutan. Sebelum melakukan ubah password, user yang bersangkutan diharuskan memasukkan data-data yang diperlukan. Setelah itu, user yang bersangkutan melakukan klik pada tombol Ubah. Jika data yang dimasukkan benar, maka sistem akan mengubah password lama yang ada pada database dengan password baru yang telah dimasukkan, sebaliknya jika data yang dimasukkan salah, maka sistem akan menampilkan pesan kesalahan.

## 5. Perancangan Format Penulisan SMS

### 5.1 Daftar

DAFTAR#[USERNAME]#[PASSWORD]#[NOPLAT]

Gambar 5.1 Rancangan Format Penulisan SMS Daftar

Format penulisan SMS pada Gambar 5.1 merupakan kata kunci yang digunakan oleh pelanggan untuk mendaftar antrian service. Kata kunci tersebut harus diketik dengan format yang benar dan dikirimkan melalui SMS menuju nomer SMS Gateway yang telah disediakan. Dengan mengirimkan SMS tersebut, pelanggan akan secara otomatis mendapat pesan balasan dan terdaftar di antrian service. Pelanggan tidak dapat melakukan pendaftaran, ketika kendaraan yang bersangkutan sudah melakukan pendaftaran atau sedang melakukan service.

### 5.2 Batal

BATAL#[USERNAME]#[PASSWORD]#[NOPLAT]

Gambar 5.2 Rancangan Format Penulisan SMS Batal

Format penulisan SMS pada Gambar 5.2 merupakan kata kunci yang digunakan oleh pelanggan untuk membatalkan antrian service. Kata kunci tersebut harus diketik dengan format yang benar dan dikirimkan melalui SMS menuju nomer SMS Gateway yang telah disediakan. Dengan mengirimkan SMS tersebut, pelanggan akan secara otomatis mendapat pesan balasan dan membatalkan antrian service. Untuk membatalkan antrian service, pelanggan harus sudah melakukan pendaftaran terlebih dahulu.

|  |            |        |
|--|------------|--------|
| Program Studi Teknik Informatika   | DPPL – CRS | 91/ 92 |
| Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika |            |        |

### 5.3 Feedback

```
FEEDBACK#[USERNAME]#[PASSWORD]#[PESAN]
```

Gambar 5.3 Rancangan Format Penulisan SMS Feedback

Format penulisan SMS pada Gambar 5.3 merupakan kata kunci yang digunakan oleh pelanggan untuk memberikan feedback. Kata kunci tersebut harus diketik dengan format yang benar dan dikirimkan melalui SMS menuju nomer SMS Gateway yang telah disediakan. Dengan mengirimkan SMS tersebut, pelanggan akan secara otomatis mendapat pesan balasan dan memberikan feedback kepada pemilik. Feedback hanya dapat diberikan satu hari satu kali.

### 5.4 Ubah Password

```
UBAH#[USERNAME]#[PASSWORD]#[PASSBARU]
```

Gambar 5.4 Rancangan Format Penulisan SMS Ubah Password

Format penulisan SMS pada Gambar 5.4 merupakan kata kunci yang digunakan oleh pelanggan untuk mengubah password. Kata kunci tersebut harus diketik dengan format yang benar dan dikirimkan melalui SMS menuju nomer SMS Gateway yang telah disediakan. Dengan mengirimkan SMS tersebut, pelanggan akan secara otomatis mendapat pesan balasan dan mengubah password.