BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi prototype aplikasi pembelian, penjualan, persedian *spare part* dan *sevis* motor dilakukan menggunakan bahasa pemrograman Java dengan penggunaan *Java Development Kit* 6.9 (JDK6.9) dan menggunakan *tools* pembangun *Netbeans IDE 7.1.2*. Basis data yang digunakan adalah MySql, dengan versi 4.0. Implementasi dan pengujian sepenuhnya hanya dilakukan di perangkat keras PC (*personal computer*) dengan sistem operasi *Microsoft Windows 7*.

5.1.1 Batasan Implementasi

Dalam implementasi perangkat lunak, pemantauan dan pengendalian aplikasi ini ada beberapa hal yang menjadi batasan, yaitu :

1. Basis data yang digunakan dalam Implementasi sistem ini adalah MySql, dengan versi 4.0. sebagai basisdata utama, maupun ekstraksi basisdata yang disebarkan ke *client*. Pengaksesan basisdata di *server* dilakukan melalui aplikasi *XAMPP*.
2. Dalam pembuatan aplikasi ini menggunakan bahasa pemrograman java dengan penggunaan *Java Development Kit* 6.9 (JDK6.9) dan menggunakan *tools Netbeans IDE 7.1.2* sebagai editor.

3. System yang dibuat bertujuan untuk menangani proses yang berjalan pada Dealer Rajamandala Motor dalam keperluan transaksi pembeliaan, penjualan, persediaan *spare part* serta jasa servis motor

5.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan program aplikasi penjualan ini adalah sebagai berikut :

1. *XAMPP* dipilih sebagai perangkat lunak untuk *web server* dan *database* yang digunakan adalah *MySQL*..
2. Perangkat lunak yang digunakan sebagai *tools* pembangun applikasi adalah *Netbeans IDE 7.1.2.*
3. Bahasa pemrograman yang digunakan untuk membangun applikasi ini adalah bahasa pemrograman *java* dengan *Java Development Kit* 6.9 (JDK6.9).
4. Sistem operasi yang digunakan dalam pembangunan dan pengujian applikasi ini adalah *Microsoft Windows 7.*

5.1.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus dipenuhi adalah sebagai berikut :

1. Client
    a. Processor Intel P4 2.4 atau diatasnya.
    b. Memory DDR RAM 512 MB.
    c. Harddisk 40 GB.
    d. Monitor, mouse dan keyboard.

5.1.4 Implementasi Basis Data

Pembahasan pada implementasi dengan menggunakan bahasa SQL, dimana aplikasi pemrograman yang dipakai adalah MySQL, Implementasi basis data dalam MySQL adalah sebagai berikut :

```
-- phpMyAdmin SQL Dump
-- version 3.1.1
-- http://www.phpmyadmin.net
--
-- Host: localhost
-- Generation Time: Jun 08, 2013 at 06:07 AM
-- Server version: 5.1.30
-- PHP Version: 5.2.8
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
--
-- Database: `db_rajamandala`
--


-- --------------------------------------------------------


--
-- Table structure for table `t_belibarang`
--


CREATE TABLE IF NOT EXISTS `t_belibarang` (
```

```
  `noFaktur` varchar(25) NOT NULL,
  `sukuCadangId` varchar(25) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `hargaBeli` bigint(20) NOT NULL,
  `totalBeli` bigint(20) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `t_estimasi`
--

CREATE TABLE IF NOT EXISTS `t_estimasi` (
  `noFaktur` varchar(25) NOT NULL,
  `tanggalTransaksi` date NOT NULL,
  `totalHarga` bigint(20) NOT NULL,
  `IdPelanggan` varchar(25) NOT NULL,
  `idKendaraan` varchar(25) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
```

```
-- Table structure for table `t_jualbarang`
--

CREATE TABLE IF NOT EXISTS `t_jualbarang` (
  `noFaktur` varchar(20) NOT NULL,
  `sukuCadangId` varchar(25) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `totalJual` bigint(20) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--
-- Table structure for table `t_jualservice`
--

CREATE TABLE IF NOT EXISTS `t_jualservice` (
  `noFaktur` varchar(25) NOT NULL,
  `serviceId` varchar(25) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `totalJual` bigint(20) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------
```

```
--

-- Table structure for table `t_kendaraan`

--


CREATE TABLE IF NOT EXISTS `t_kendaraan` (
  `noRangka` varchar(50) NOT NULL,
  `noPolisi` varchar(10) NOT NULL,
  `tipe` varchar(30) NOT NULL,
  `keterangan` text NOT NULL,
  `pelangganId` varchar(25) NOT NULL,
  `tahun` int(11) NOT NULL,
  `noMesin` varchar(25) NOT NULL DEFAULT '',
  PRIMARY KEY (`noRangka`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;


-- --------------------------------------------------------


--

-- Table structure for table `t_pegawai`

--


--

-- Table structure for table `t_pelanggan`
```

```
--

CREATE TABLE IF NOT EXISTS `t_pelanggan` (
  `IdPelanggan` varchar(25) NOT NULL,
  `pelangganName` varchar(50) NOT NULL,
  `pelangganAddress` text NOT NULL,
  `pelangganContact` varchar(20) NOT NULL,
  PRIMARY KEY (`IdPelanggan`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------
--
-- Table structure for table `t_pembelian`
--

CREATE TABLE IF NOT EXISTS `t_pembelian` (
  `noNota` varchar(25) NOT NULL,
  `supplierId` varchar(25) NOT NULL,
  `tanggalTransaksi` date NOT NULL,
  `totalHarga` bigint(20) NOT NULL,
  `status` varchar(10) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------
```

5.1.5 Implementasi Antar Muka

Dalam *Netbeans IDE* 7.1.2, implementasi antarmuka dilakukan dengan sebuah *form* yang berekstensi *form*. Setiap halaman dalam perangkat lunak dibuka *file program* yang ditulis dengan berektensi *form.*

Berikut ini adalah implementasi dari setiap halaman yang dibuat :

5.1.5.1 Implementasi Halaman Utama

Form halaman utama merupakan sentral dari semua halaman form. Form ini berisi menu-menu utama yang dapat mengakses dan menjadi penghubung antara pengguna dengan sistem. Berikut ini implementasi struktur menu pada halaman utama Sistem Bengkel:

Tabel 5. 1 Keterangan Halaman Utama (Menu)

Menu	**Deskripsi**	**Nama File**
Data User	Menu untuk mencari, mengubah dan menambahkan data *user* yang baru pada *database*	UserUI
Data *Customer*	Menu untuk mencari, mengubah dan menambahkan data *customer* pada *database*	PelangganUI
Data Kendaraan	Menu untuk mencari, mengubah dan menambahkan data kendaraan pada *database*	KendaraanUI
Data Jasa	Menu untuk mencari, mengubah dan menambahkan data servis pada *database*	ServiceUI

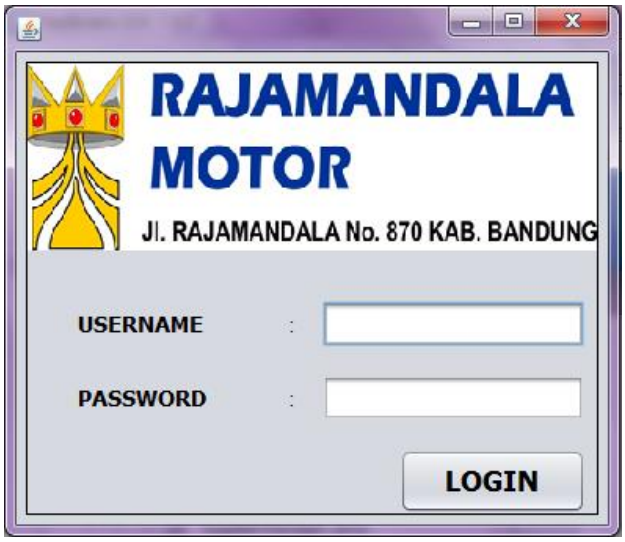
Data Pegawai	Menu untuk mencari, mengubah dan menambahkan data pegawai pada *database*	PegawaiUI
Data *Supplier*	Menu untuk mencari, mengubah dan menambahkan data *supplier* pada *database*	SupplierUI
Data *Spare part*	Menu untuk mencari, mengubah dan menambahkan data *spare part* pada *database*	SukuCadangUI
Work Order	Menu untuk mencari dan menambahkan data *work order* pada *database*	WorkOrderUI
Estimasi Servis	Menu unutk mencari dan menambahkan data *estimasi* harga pada *database*	EstimasiUI
Transaksi Servis	Menu untuk mencari dan menambahkan data transaksi servis yang baru	PerbaikanUI
Transaksi Penjualan	Menu untuk mencari dan menambahkan data transaksi penjualan yang baru	PenjualanUI
Transaksi Pembelian	Menu yang digunakan untuk mencari dan menambahkan data pemesanan barang	PembelianUI
Konfirmasi Pembelian	Menu yang digunakan untuk mencari dan mengkonfirmasi data pemesanan yang ada	KonfirmasiPembelianUI
Laporan	Menu untuk melakukan pencetakan laporan	ReportUI

5.1.6 Penggunaan Program

Kegiatan ini bertujuan untuk menerangkan secara singkat penggunaan program Sistem Informasi Dealer ini. Adapun cara pengoperasiannya adalah sebagai berikut :

1. ***Form Login***

   Ketika pertama kali program aplikasi ini dijalankan maka yang pertama kali muncul adalah form login seperti di bawah ini. Halaman ini merupakan halaman yang berfungsi untuk menentukan hak akses dari pengguna applikasi ini.

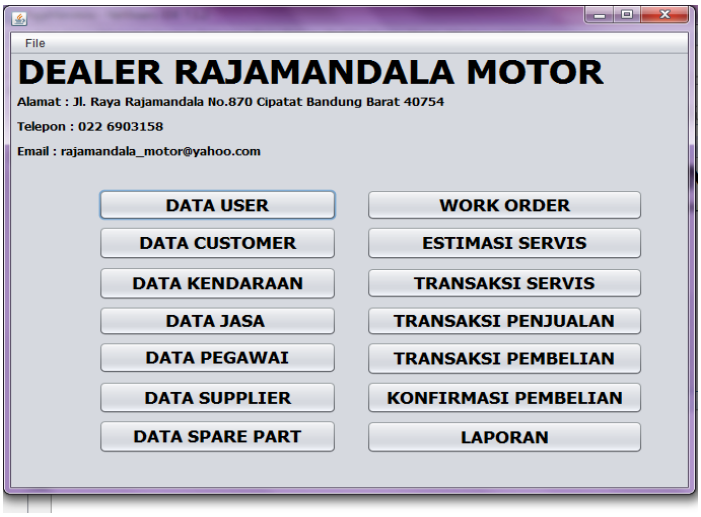


Gambar 5. 1 Tampilan *Form Login*

2. ***Form* Utama**

   Setelah melakukan login dan berhasil masuk ke applikasi, maka akan muncul form utama seperti di bawah ini.

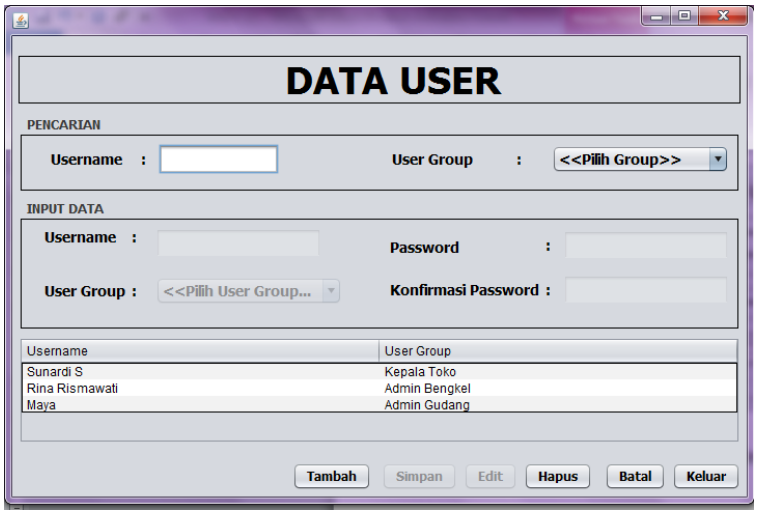
Halaman ini merupakan halaman yang berisikan menu-menu yang dapat dilakukan pada program ini :



Gambar 5. 2 Tampilan Form Utama

3. *Form Master* Data *User*

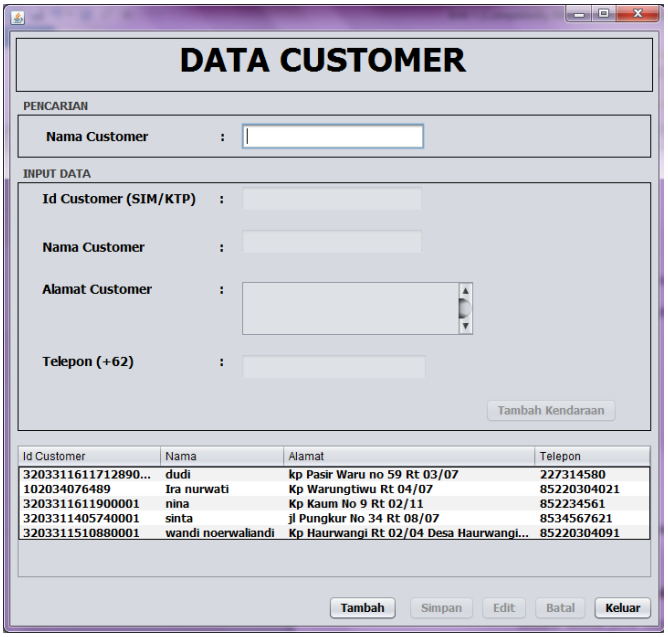
Berikut adalah *form Master* data *user* yang berfungsi untuk menambahkan , mencari, dan mengubah data *user.*



Gambar 5. 3 Tampilan *Form Master* Data *user*

4. *Form Master* Data *Customer*

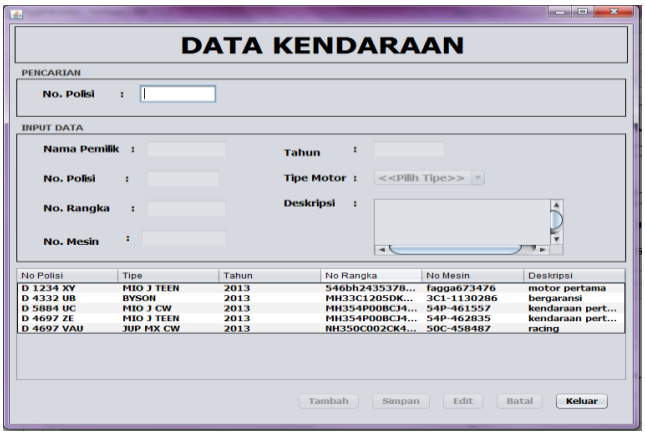
Berikut adalah *form Master* data customer yang berfungsi untuk menambahkan, mencari, dan mengubah data customer.



Gambar 5. 4 Tampilan *Form Master* Data Customer

5. *Form Master* Data Kendaraan

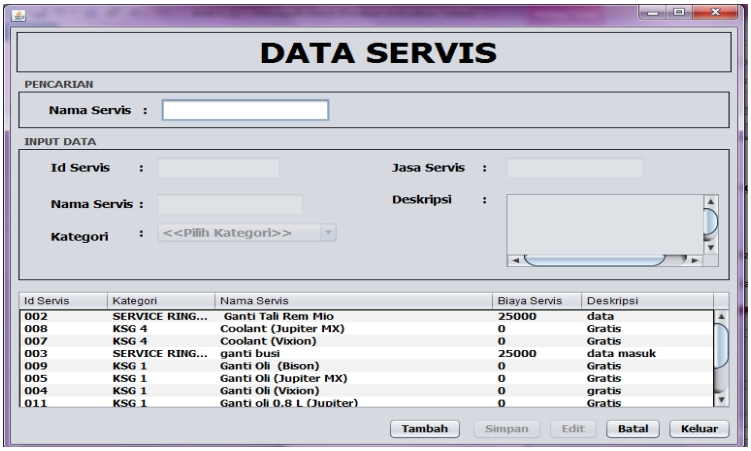
Berikut adalah *form Master* data kendaraan yang berfungsi untuk menambahkan, mencari dan mengubah data kendaraan.



Gambar 5. 5 Tampilan *Form Master* Data kendaraan

6. *Form Master* Data Servis

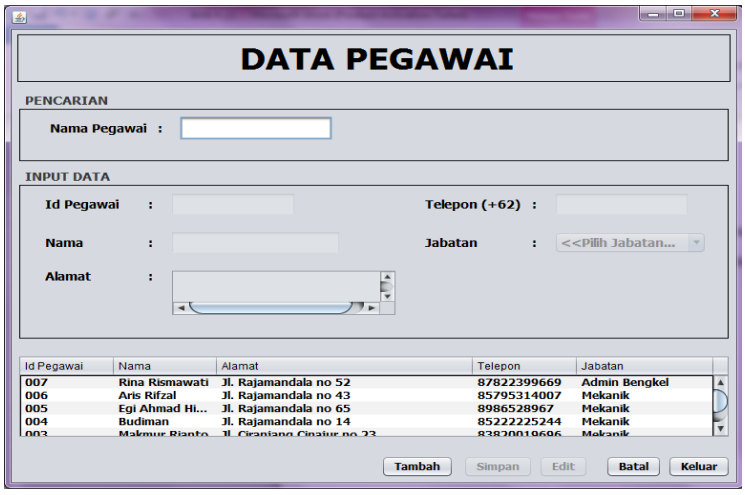
Berikut adalah *form Master* data Servis yang berfungsi untuk menambahkan, mencari, dan mengubah data *service.*



Gambar 5. 6 Tampilan *Form Master* Data servis

7. *Form Master* Data Pegawai

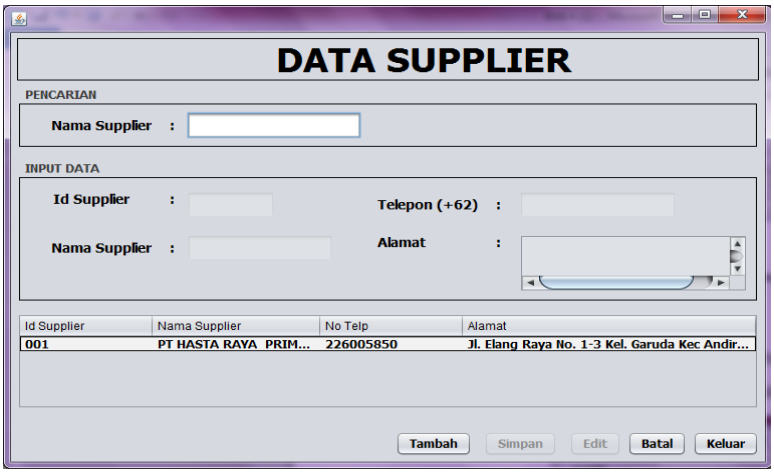
Berikut adalah *form Master* data pegawai yang berfungsi untuk menambahkan, mencari dan mengubah data pegawai.



Gambar 5. 7 Tampilan *Form Master* Data pegawai

8. *Form Master* Data *Supplier*

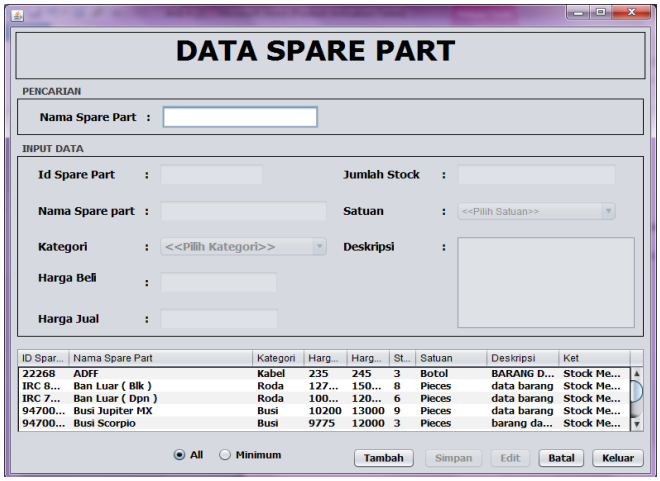
Berikut adalah *form Master* data *supplier* yang berfungsi untuk menambahkan, mencari, dan mengubah data *supplier.*



Gambar 5. 8 Tampilan *Form Master* Data *supplier*

9. *Form Master* Data *Spare part*

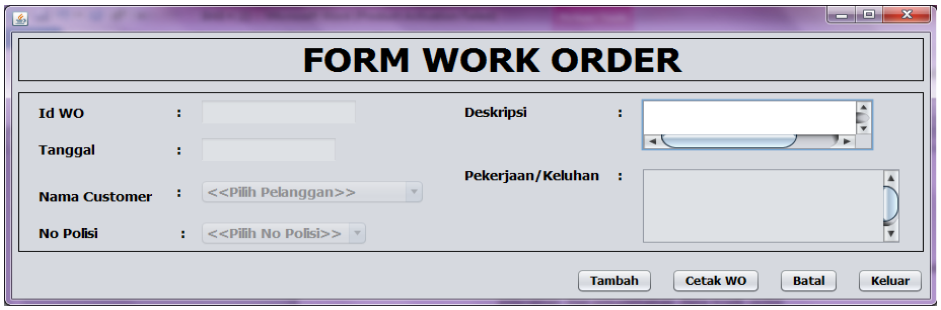
Berikut adalah *form Master* data *spare part* yang berfungsi untuk menambahkan, mencari, dan mengubah data *spare part.*



Gambar 5. 9 Tampilan *Form Master* Data *spare part*

10. *Form Work Order*

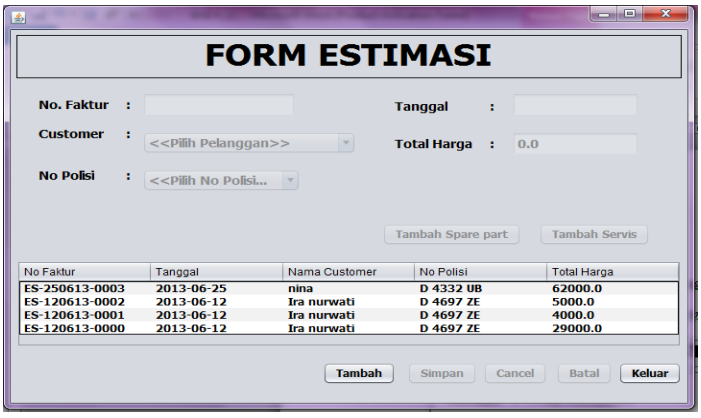
Berikut adalah *form Work order* yang berfungsi untuk menambahkan data *work order* dan mencetak *work order* yang ditambahkan.



Gambar 5. 10 Tampilan *Form Work Order*

11. Form Estimasi

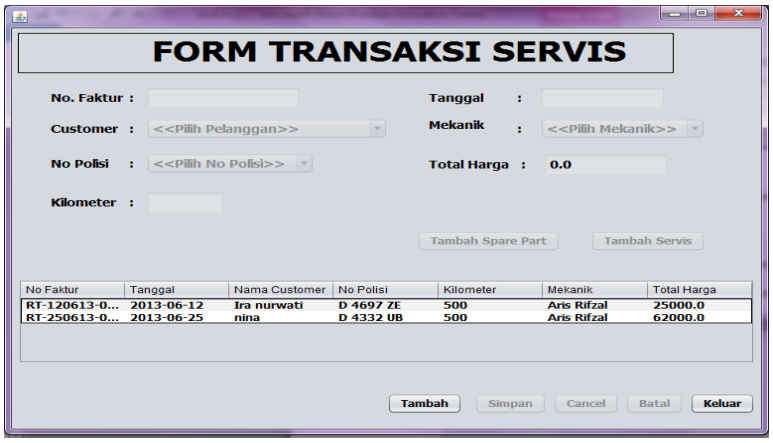
Berikut adalah form estimasi yang berfungsi untuk menambahkan data estimasi yang baru dan mencetak data estimasi yang ditambahkan.



Gambar 5. 11 Tampilan Form Estimasi

12. Form Transaksi Servis

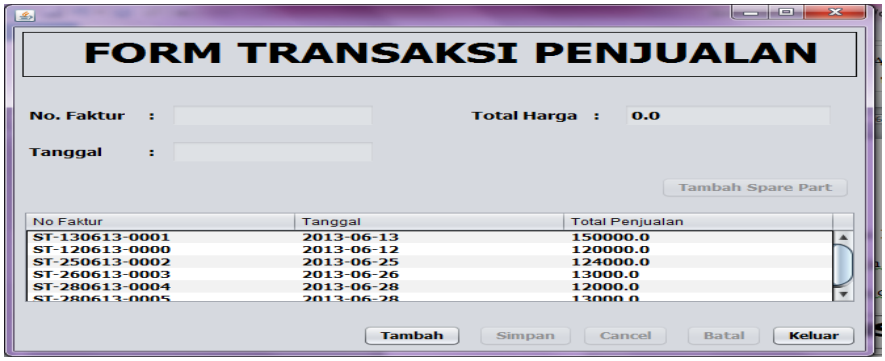
Berikut adalah form Transaksi servis yang berfungsi untuk mencari dan menambahkan data data transaksi s*ervis*, serta mencetak bukti transaksi servis yang baru.



Gambar 5. 12 Tampilan *Form* Transaksi servis

13. Form Transaksi Penjualan

Berikut adalah form Transaksi Penjualan yang berfungsi untuk mencari dan menambahkan data transaksi penjualan serta mencetak transaksi penjualan yang baru.



Gambar 5. 13 Tampilan Form Transaksi Penjualan

14. Form Transaksi Pemesanan

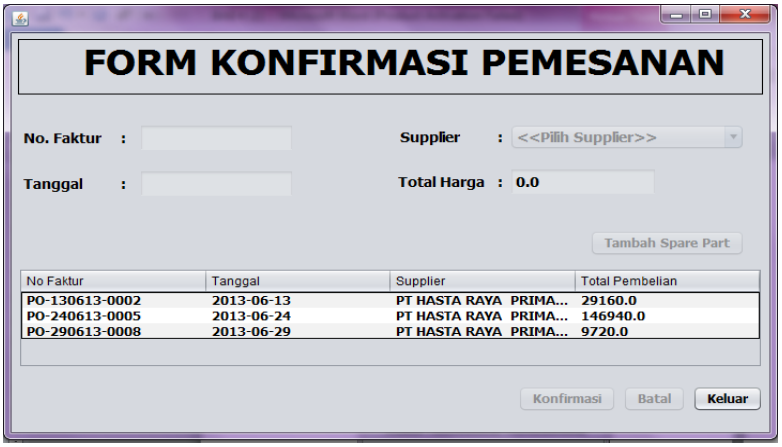
Berikut adalah form transaksi pemesanan yang berfungsi untuk mencari dan menambahkan data transaksi pemesanan serta mencetak bukti pemesanan yang baru.



Gambar 5. 14 Tampilan Form Transaksi Pemesanan

15. Form Konfirmasi Pemesanan

Berikut adalah form konfirmasi pemesanan yang berfungsi untuk mencari dan mengkonfirmasi data transaksi pemesanan serta mencetak bukti konfirmasi yang baru.



Gambar 5. 15 Tampilan Form Konfirmasi Pemesanan

16. Form Pelaporan

Berikut adalah form pelaporan yang berfungsi untuk mencetak data laporan yang diinginkan.



Gambar 5. 16 Tampilan Form Pelaporan

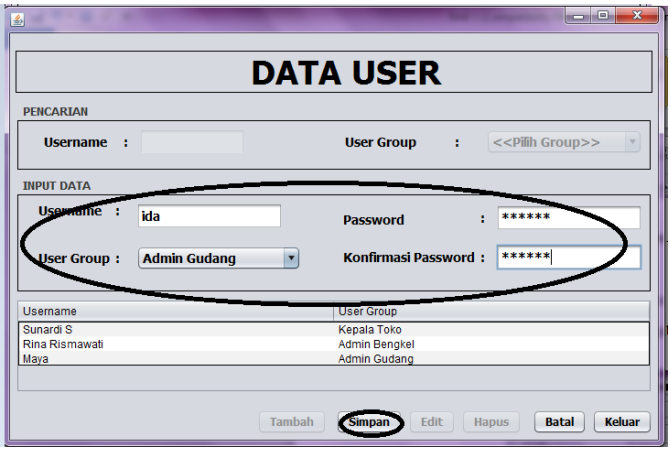
5.2 Pengujian

Pengujian dapat berarti proses untuk memeriksa apakah suatu perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai dengan standar tertentu. Metode pengujian adalah cara atau teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.

5.2.1 Kasus Hasil Pengujian

1. Penambahan data user

   Pengguna dapat menambah data user yang baru dengan mengisi field yang ada dan menentukan tipe user dan kemudian menekan tombol Simpan.



Gambar 5. 17 *Form Master* Data *User* untuk penambahan data *user*

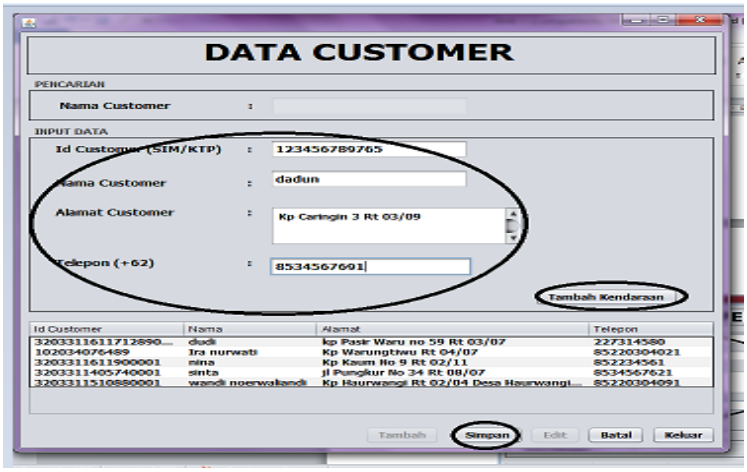
Hasil:

[x] Diterima

[ ] Ditolak

2. Penambahan data Customer

Pengguna dapat menambah data customer yang baru dengan mengisi field yang ada dan menambahkan data kendaraannya dengan menekan tombol tambah kendaraan dan kemudian menekan tombol Simpan setelah selesai penambahan kendaraan.

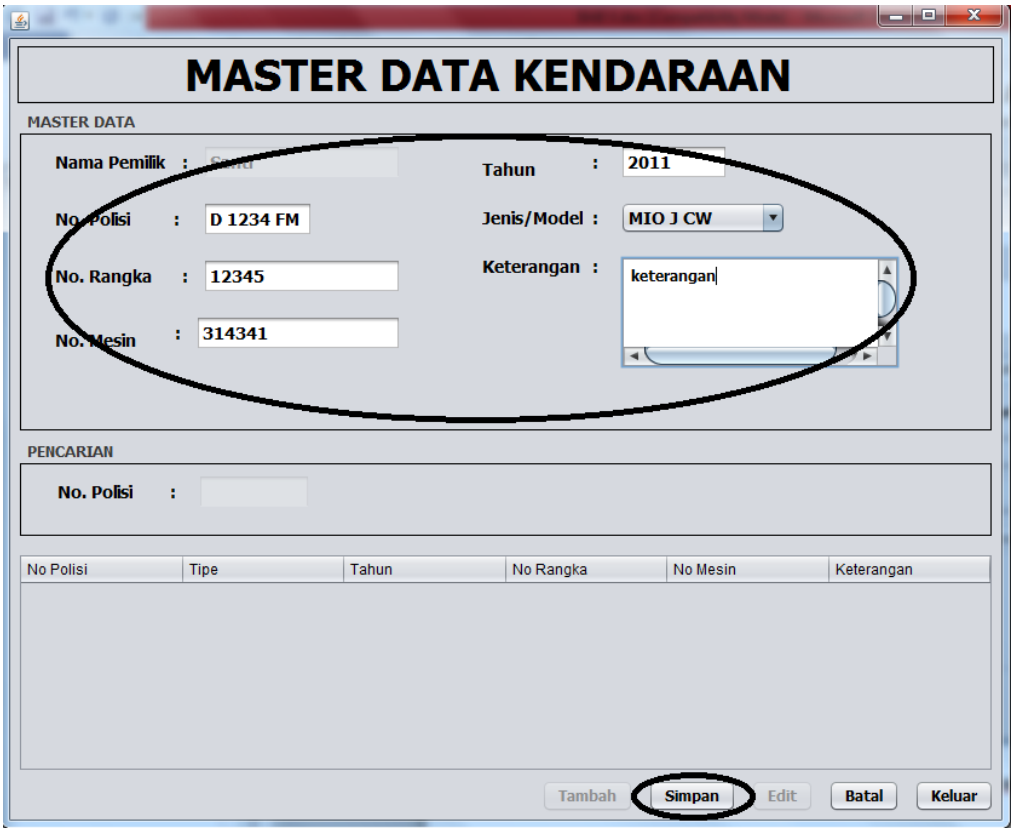


Gambar 5. 18 *Form Master Customer* untuk penambahan data *Customer*

Hasil:

[x] Diterima

[ ] Ditolak



Gambar 5. 19 *Form Master* Kendaraan untuk penambahan data kendaraan milik *customer*

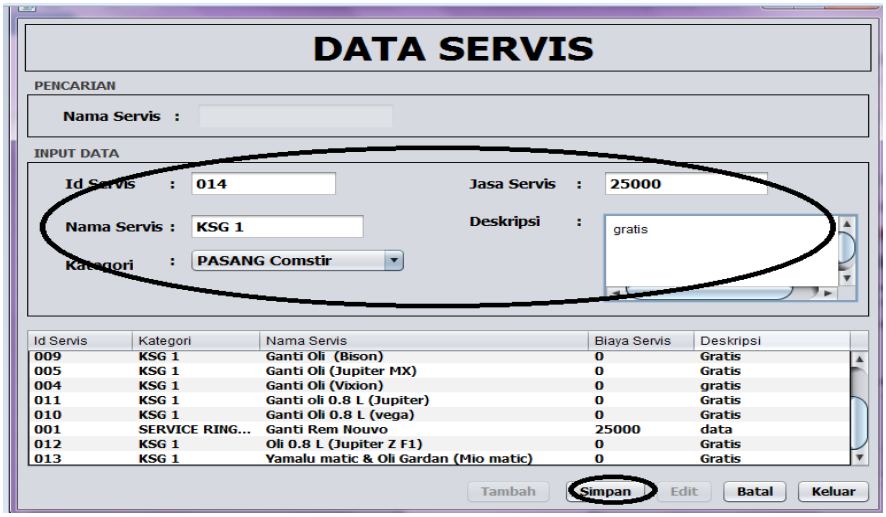
Hasil:

[x] Diterima

[ ] Ditolak

3. Penambahan data Jasa

   Pengguna dapat menambah data servis yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



Gambar 5. 20 *Form Master* Data servis untuk penambahan data servis

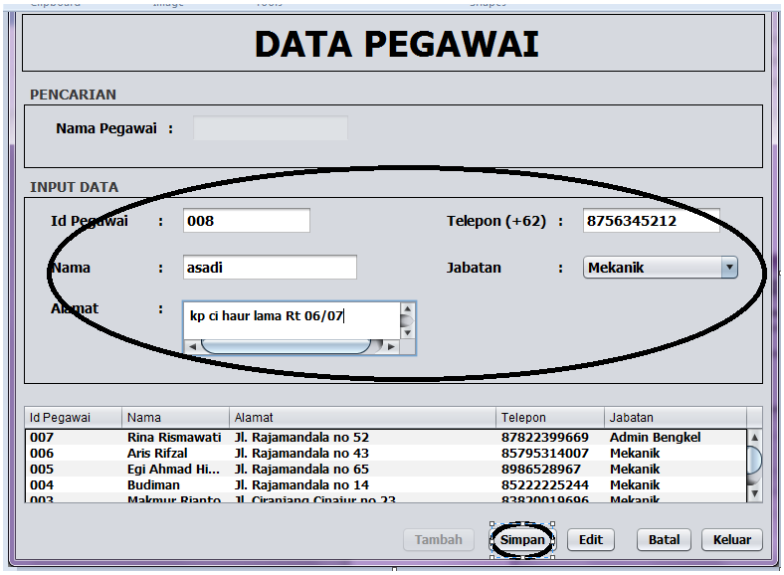
Hasil:

[x] Diterima

[ ] Ditolak

4. Penambahan data Pegawai

   Pengguna dapat menambah data pegawai yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



Gambar 5. 21 Form *Master* Pegawai untuk penambahan data pegawai

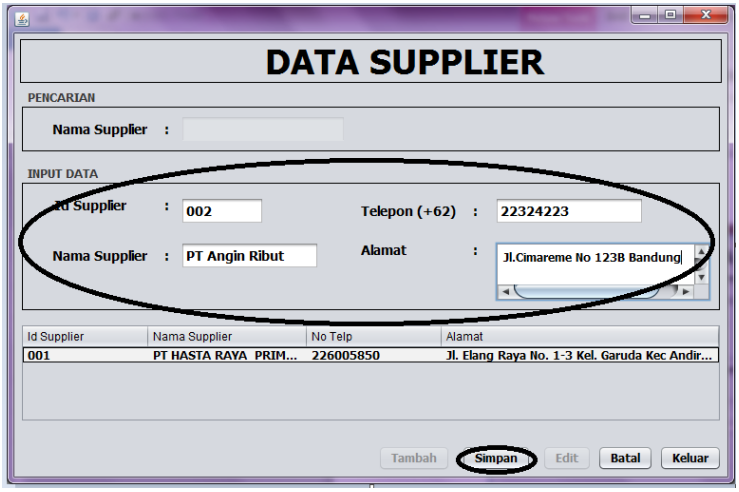
Hasil:

[x] Diterima

[ ] Ditolak

5. Penambahan data *Supplier*

   Pengguna dapat menambah data *supplier* yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



Gambar 5. 22 Form *Master Supplier* untuk penambahan data *supplier*

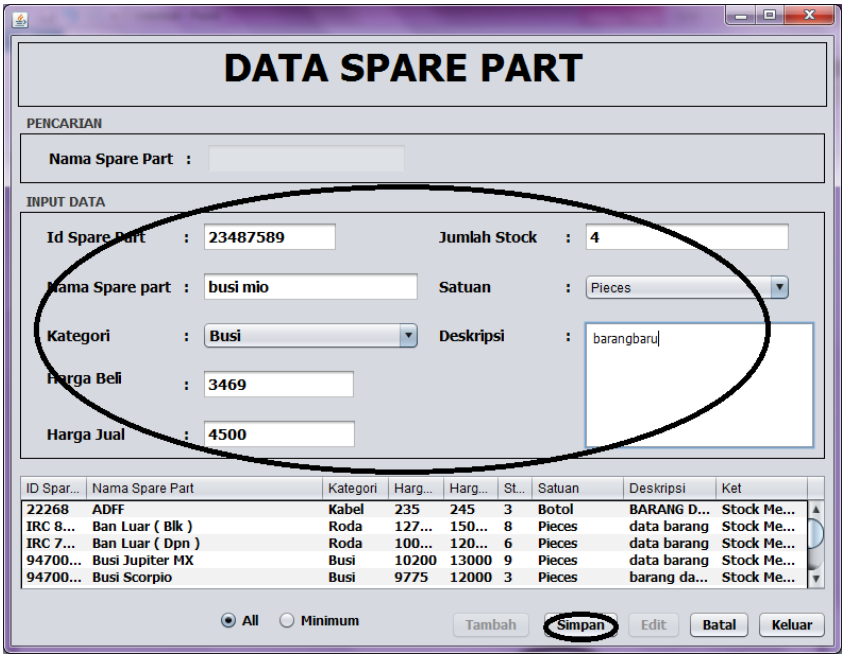
Hasil:

[x] Diterima

[ ] Ditolak

6. Penambahan data *Spare part*

   Pengguna dapat menambah data *Spare part* yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



Gambar 5. 23 *Form Master Spare part* untuk penambahan data *Spare part*

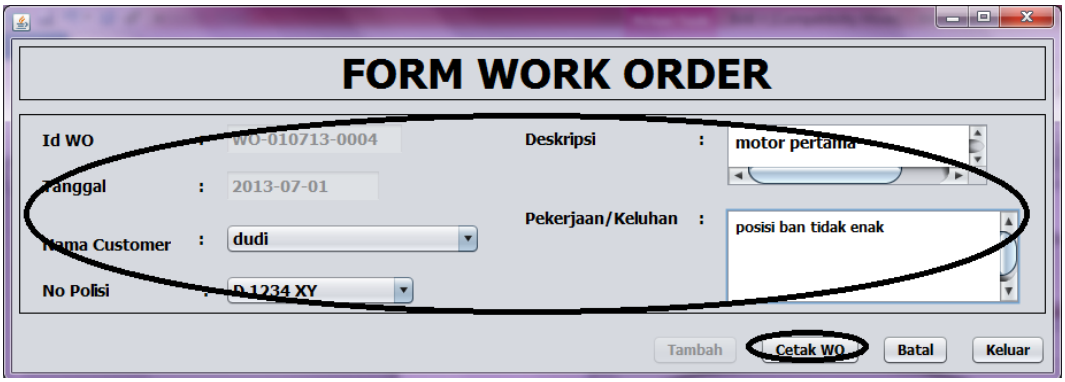
Hasil:

[x] Diterima

[ ] Ditolak

7. Pembuatan *Work Order*

   Pengguna dapat membuat *work order* yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Cetak WO.



Gambar 5. 24 *Form Work Order* untuk pencetakan *Work Order*

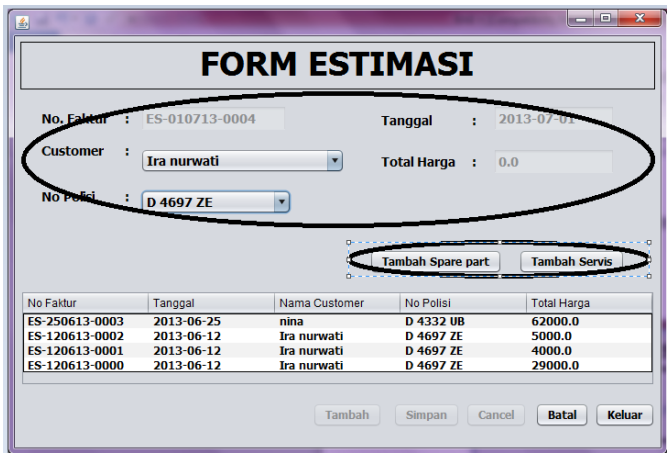
Hasil:

[x] Diterima

[ ] Ditolak

8. Pembuatan Estimasi Harga

   Pengguna dapat membuat Estimasi harga dengan mengisi *field* yang ada dan kemudian menekan tombol penambahan *spare part* atau servis.



Gambar 5. 25 Form Estimasi

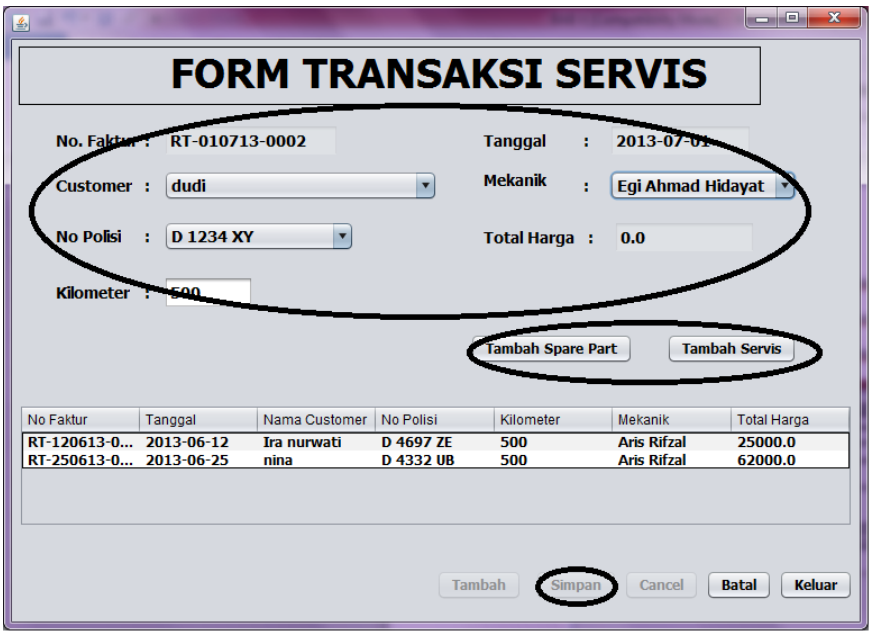
Hasil:

[x] Diterima

[ ] Ditolak

9. Penambahan Transaksi Servis

   Pengguna dapat menambahkan data transaksi Servis dengan mengisi *field* yang ada dan kemudian menekan tombol penambahan *Spare part* atau servis kemudian menyimpannya dengan menekan tombol simpan.



Gambar 5. 26 Form Transaksi servis

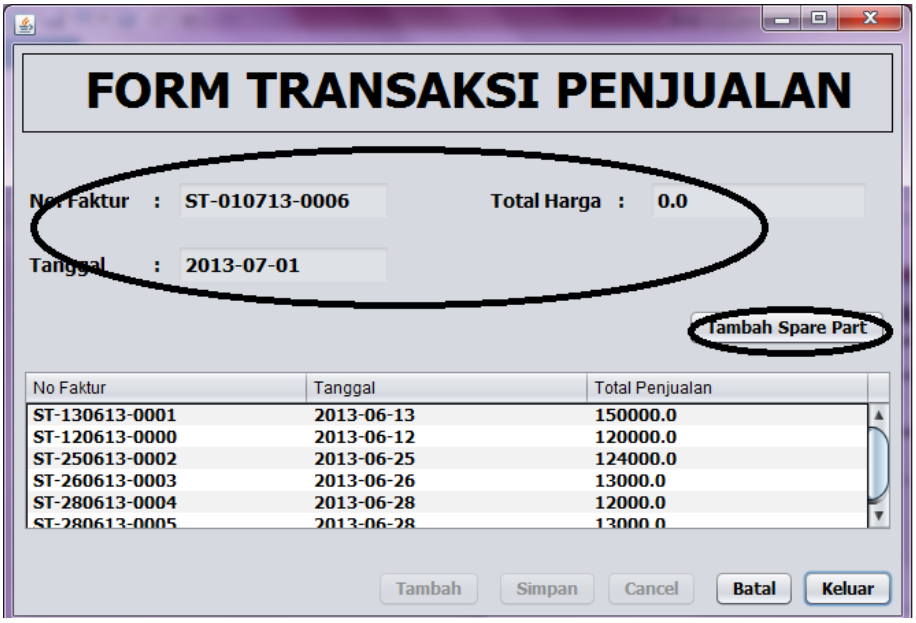
Hasil:

[x] Diterima

[ ] Ditolak

10. Penambahan Transaksi Penjualan

Pengguna dapat menambahkan data transaksi penjualan dengan mengisi *field* yang ada dan kemudian menekan tombol penambahan *Spare part* kemudian menyimpannya dengan menekan tombol simpan.



Gambar 5. 27 Form Transaksi Penjualan

Hasil:

[x] Diterima

[ ] Ditolak

11. Penambahan Transaksi Pemesanan dan Pembelian

    Pengguna dapat menambahkan data transaksi pemesanan dan pembelian dengan mengisi *field* yang ada dan kemudian menekan tombol penambahan *Spare part* kemudian menyimpannya dengan menekan tombol simpan.



Gambar 5. 28 Form Transaksi Pemesanan

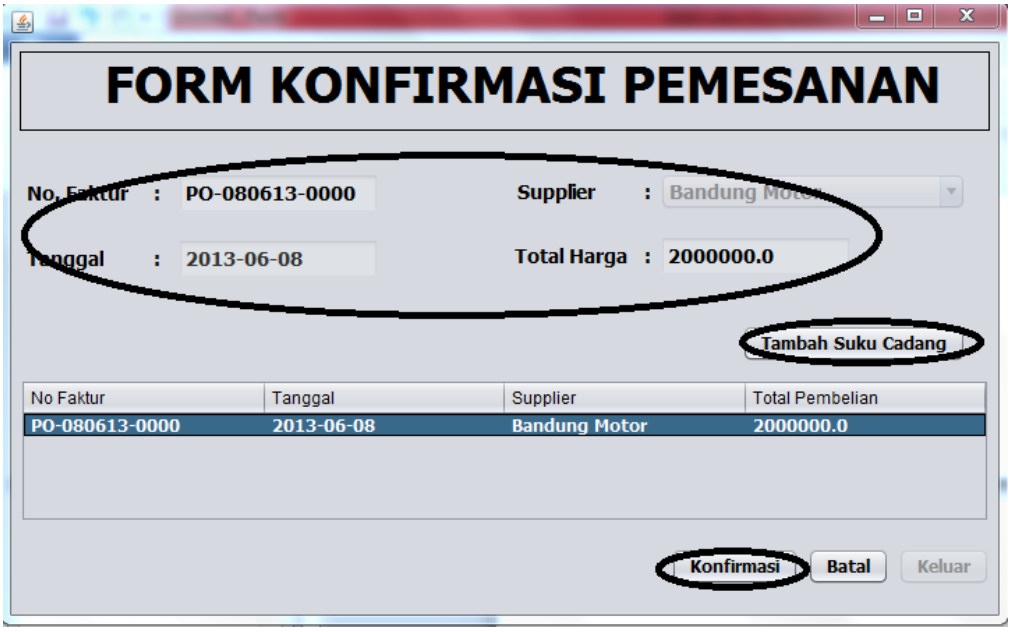
Hasil:

[x] Diterima

[ ] Ditolak

12. Konfirmasi Transaksi Pemesanan

Pengguna dapat mengkonfirmasi data transaksi pemesanan dengan mengisi *field* yang ada dan kemudian menekan tombol penambahan *Spare part* untuk mengkonfirmasi jumlah yang sampai dengan yang dipesan kemudian menyimpannya dengan menekan tombol simpan.



Gambar 5. 29 Form Konfirmasi Transaksi Pemesanan

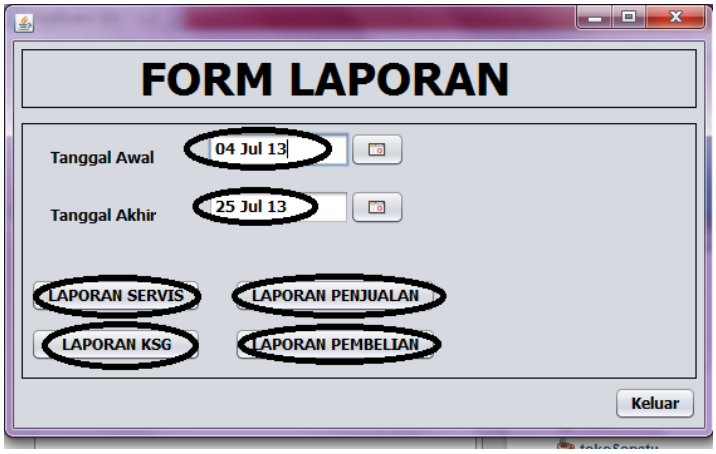
Hasil:

[x] Diterima

[ ] Ditolak

13. Form Pelaporan

Pengguna dapat mencetak laporan dengan mengisi *field* yang ada dan kemudian memilih jenis laporan yang diinginkan.



Gambar 5. 30 Form Pelaporan

Hasil:

[x] Diterima

[ ] Ditolak

5.2.2 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian uji *sample* di atas dapat disimpulan bahwa perangkat lunak Sistem Informasi Dealer Rajamandala Motor dapat berjalan dengan baik dan mengeluarkan hasil sesuai dengan yang diharapkan.