BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi *prototype* sistem informasi Pelayanan pada Bengkel Indra Jaya Motor (IJM) Bandung dilakukan menggunakan bahasa dan lingkungan pemrograman JAVA, dengan basis data yang digunakan adalah MySQL *Database*. Implementasi seluruhnya menggunakan media komputer dengan sistem operasi *Windows 7*.

5.1.1 Batasan Implementasi

Dalam mengimplementasikan perangkat lunak pemantau dan pengendalian aplikasi ini ada beberapa hal yang perlu menjadi batasan implementasi yaitu :

1. Basis data yang digunakan dalam Implementasi sistem ini adalah MySQL, baik basis data utama di *server* maupun komputer *client*.
2. Perangkat Lunak ini memfokuskan pada proses pengolahan pelayanan jasa *service* dan suku cadang, *stock* barang dan *stock* minimum barang, pengelolaan data Karyawan dan *Supplier*, dan Transaksi penjualan untuk konsumen dan pembelian suku cadang kepada *supplier*.
3. Dalam Pembuatan sistem informasi pelayanan berbasis *client-server* ini menggunakan bahasa pemograman JAVA.

5.1.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak sistem informasi pelayanan berbasis *client-server* ini digunakan *NetBeans* 7.0.1, *Xamp* 1.7.3, MySQL *Database* dan *Mozilla Firefox*. *NetBeans* 7.0.1 dipilih sebagai perangkat lunak pengembang karena menyediakan fasilitas yang memadai berbasis *client-server* dan memudahkan penggunaan bahasa JAVA, *Xamp* 1.7.3 dipilih sebagai perangkat lunak pengembang karena menyediakan fasilitas yang memadai berbasis *client-server* dan juga mempunyai fungsi sebagai *server* yang berdiri sendir (*localhost)* dan MySQL *Database Server*. Sementara itu MySQL *Database* dan *Mozilla Firefox*.

5.1.3 Implementasi Perangkat Keras

Sistem yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain :

1. Pada sisi *server* perangkat keras yang dibutuhkan antara lain :
    a. Menggunakan *processor* Minimal *Pentium* 4 atau sekelasnya.
    b. Menggunakan minimal RAM 256 MB.
    c. *Hardisk* untuk penampungan data minimal 80 Ghz.
    d. *Mouse, Keyboard* dan *Monitor* sebagai peralatan antar muka.
    e. Sistem Operasi minimal *windows XP*.
2. Pada sisi *client* perangkat keras yang dibutuhkan antara lain :
    a. *Processor* Minimal *Pentium* 4 atau sekelasnya.
    b. Menggunakan minimal RAM 256 MB.
    c. *Hardisk* 80 Gb atau lebih.

d. *Input Device (Mouse, Keyboard)*

e. *Output Device (Monitor, printer)*

f. Sistem Operasi minimal *windows XP*.

3. Perangkat Jaringan:

a. Kabel UTP

b. Switch

5.1.4 Implementasi Basis Data (Sintak SQL)

Implementasi basis data dilakukan dengan menggunakan dahasa SQL, , implementasi basis datanya dalam bahas SQL adalah sebagai berikut :

```
-- Database: `bengkel`

-- Struktur dari tabel `barangjasa`

CREATE TABLE IF NOT EXISTS `barangjasa` ( `kode_bj` varchar(12) NOT NULL, `nama_bj` text NOT NULL, `merk_bj` text NOT NULL, `stock_bj` int(10) NOT NULL, `hargabeli_bj` int(10) NOT NULL, `hargajual_bj` int(10) NOT NULL, `ket_bj` text NOT NULL, PRIMARYKEY (`kode_bj`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Table structure for table `gaji`

CREATE TABLE IF NOT EXISTS `gaji` ( `kode_gaji` int(10) NOT NULL, `kode_kry` int(10) NOT NULL, `waktu_gaji` text NOT NULL, `hr_krj` int(10) NOT NULL,`total_gaji` int(10) NOT NULL, PRIMARY KEY (`kode_gaji`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

-- Struktur dari tabel `karyawan`

CREATE TABLE IF NOT EXISTS `karyawan` (`kode_kry` int(10) NOT NULL AUTO_INCREMENT `nama_kry` text NOT NULL, `jbt_kry` text NOT NULL, `gapok_kry` int(10) NOT NULL, `hari_kerja` int(1) NOT NULL, `alamat_kry` text NOT NULL, `no_tlp_kry` text NOT NULL, PRIMARY KEY (`kode_kry`)) ENGINE=InnoDB DEFAULT CHARSET=hp8 AUTO_INCREMENT=8 ;

-- Struktur dari tabel `preorder`

CREATE TABLE IF NOT EXISTS `preorder` (`kode_po` int(11) NOT NULL AUTO_INCREMENT, `nama_bj` text NOT NULL, `merk_bj` text NOT NULL, `stok_bj` int(11) NOT NULL, `tgl_beli` text NOT NULL, `jml_bj` int(11) NOT NULL, `hrg_beli` int(11) NOT NULL, `total_bayar` int(11) NOT NULL, `nama_sup` text NOT NULL, PRIMARY KEY (`kode_po`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;

-- Struktur dari tabel `restock`

CREATE TABLE IF NOT EXISTS `restock` (`kode_res` int(10) NOT NULL AUTO_INCREMENT, `nama_bj` text NOT NULL, `merk_bj` text NOT NULL,`stock_bj` int(10) NOT NULL, `tgl_beli` text NOT NULL, `jml_bj` int(10) NOT NULL, `hargabeli_bj` int(10) NOT NULL, `total_bayar` int(10) NOT NULL, `kode_suplier` text NOT NULL, `stat` int(1) NOT NULL, PRIMARY KEY (`kode_res`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

-- Table structure for table `restock_detail`

CREATE TABLE IF NOT EXISTS `restock_detail` (`id_res_detail` int(11) NOT NULL AUTO_INCREMENT, `kode_res` int(11) NOT NULL, `kode_bj` int(11) NOT NULL, `nama_bj` text NOT NULL, `merk_bj` text NOT NULL, `stock_bj` int(11) NOT NULL, `hb_bj` int(11) NOT NULL, `jumlah_beli` int(11) NOT NULL, `suplier` text NOT NULL, `stat` int(1) NOT NULL, PRIMARY KEY (`id_res_detail`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;

-- Struktur dari tabel `suplier`

CREATE TABLE IF NOT EXISTS `suplier` (`kode_sup` int(10) NOT NULL AUTO_INCREMENT, `nama_sup` text NOT NULL, `alamat_sup` text NOT NULL, `no_tlp_sup` text NOT NULL, PRIMARY KEY (`kode_sup`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

-- Struktur dari tabel `transaksi`

CREATE TABLE IF NOT EXISTS `transaksi` (`kode_trans` int(10) NOT NULL AUTO_INCREMENT, `tgl_trans` text NOT NULL, `nm_pbeli` text NOT NULL,`alamat` text NOT NULL,`total` int(11) NOT NULL, PRIMARY KEY (`kode_trans`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

-- Struktur dari tabel `transaksi_detail`

CREATE TABLE IF NOT EXISTS `transaksi_detail` (`id_detail_transaksi` int(11) NOT NULL AUTO_INCREMENT, `id_transaksi` int(11) NOT NULL,`kode_bj` varchar(12) NOT NULL `nama_bj` text NOT NULL,`harga` int(11) NOT NULL,`jumlah` int(11) NOT NULL,`total_peritem` int(11) NOT NULL,PRIMARY KEY (`id_detail_transaksi`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;

-- Struktur dari tabel `transaksi_temp`

CREATE TABLE IF NOT EXISTS `transaksi_temp` (`id_temp` int(11) NOT NULL AUTO_INCREMENT,`id_item` varchar(12) NOT NULL, `nama_item` text NOT NULL,`harga_item` int(11) NOT NULL, `qty_item` int(11) NOT NULL, `total_item` int(11) NOT NULL, PRIMARY KEY (`id_temp`), UNIQUE KEY `id_item` (`id_item`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

-- Struktur dari tabel `user`

CREATE TABLE IF NOT EXISTS `user` (`id_user` int(2) NOT NULL AUTO_INCREMENT, `username` text NOT NULL, `password` text NOT NULL, `akses` text NOT NULL, PRIMARY KEY (`id_user`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;

5.1.5 Implementasi Antra Muka

Pada implementasi antarmuka ini akan dijelaskan mengenai tampilan antarmuka, petunjuk pengguna dan hasil uji pada perancangan Sistem Informasi Pelayanan pada Bengkel Indra Jaya Motor Bandung.

Berikut ini adalah implemntasi dari setiap halaman yang dibuat dan dibedakan berdasarkan kebutuhan pengguna / user.

A .Implementasi Halaman Login User

Pada halaman *login* terdapat sub-sub menu pilihan. Seperti tabel 5.1 berikut :

Tabel 5.1 Implementasi Halaman *Login User*

Menu	Sub Menu	Deskripsi	Nama File
Login User	*Login*	*File* program untuk masuk program	auth.java
	Close	*File* program untuk batal masuk	

B. Implementasi Halaman Kasir

Pada halaman kasir terdapat sub-sub menu pilihan. Seperti tabel 5.2 berikut :

Tabel 5.2 Implementasi Halaman Kasir

Sub Menu	Sub Menu	Deskripsi	Nama File
Transaksi Penjualan	Penjualan	File program untuk melakukan transaksi penjualan barang	transaksi.java
Logout		File Program untuk keluar dan kembali ke menu login	bgkl.java

C. Implementasi Halaman Gudang

Pada halaman gudang terdapat sub-sub menu pilihan. Seperti tabel 5.3 berikut :

Tabel 5.3 Implementasi Halaman gudang

Sub Menu	Sub Menu	Deskripsi	Nama File
Pembelian		*File* Program untuk Menampilkan menu tentang pembelian berisikan data-data pembelian	bgkl.java
	Buat *Purchase Order*	File Program untuk membuat *Purchase Order* pemesanan barang	preOrder.java
	Verifikasi Purchase Order	File Program untuk mengkonfrimasi PO atau *restock* barang.	restock.java
Barang		*File* Program untuk membuka data dan tampil barang	bgkl.java
	Tambah Barang	*File* Program untuk mendambahkan data barang.	barangjasa.java
	Edit Barang	*File* Program untuk mengubah barang	barangjasaU.java
Supplier		*File* Program untuk membuka	bgkl.java

		data dan tampil *supplier*	
	Tambah *supplier*	*File* Program untuk mendambahkan data *supplier*	suplier. Java
	Edit *supplier*	*File* Program untuk mengubah *supplier*	suplierU. Java

D. Implementasi Halaman Owner

Owner dapat menggunakan semua sistem yang ada di kasir dan gudang, ada beberapa file program yang hanya di gunakan owner. Sebagai tabel 5.4 berikut :

Tabel 5.4 Implementasi Halaman *Owner*

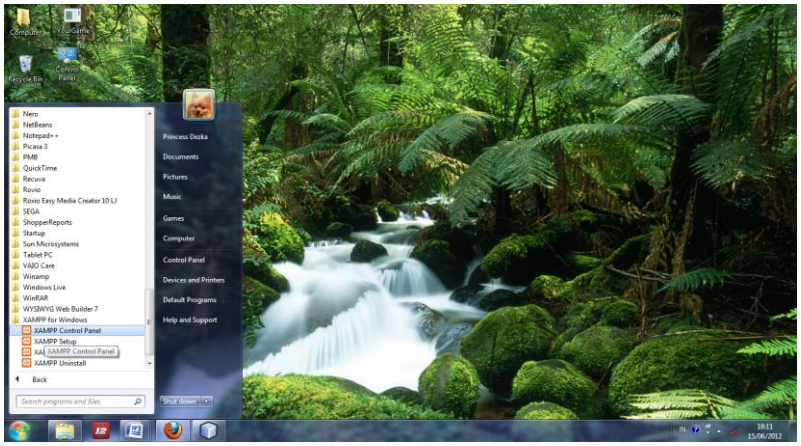
Sub Menu	Sub Menu	Deskripsi	Nama File
Jasa		File Program untuk Menampilkan Jasa *service*	bgkl.java
	Tambah Jasa	File Program untuk membuat Jasa *Service* Baru	barangjasa.java
	Edit Jasa	*File* Program untuk edit data	barangjasaU.java
karyawan		*File* Program untuk membuka data dan tampil karyawan	bgkl.java
	Tambah Data Karyawan	*File* Program untuk mendambahkan data karyawan.	karyawan.java
	Edit Karyawan	*File* Program untuk mengubah data karyawan	karyawanU.java

Kelola *User*		*File* Program untuk membuka data dan tampil *user*	bgkl.java
	Tambah *User*	*File* Program untuk mendambahkan data *user*	userAdd. Java
	Edit User	*File* Program untuk mengubah data *user*	userEdit. Java
Laporan		Untuk menampilkanlaporan	bgkl.java
	Laporan Pembelian	*File* program untuk melihat laporan pembelian	lapRestock.java
	Laporan Penjualan	*File* program untuk melihat laporan penjualan	lapGaji.java
	Laporan *Purchase Order*	File program untuk melihat laporan *Purchase Order*	lapPO.java
	Laporan Pendapatan	File program untuk melihat Laporan Pendapatan perusahaan	lapIncome.java

5.1.6 Implementasi Instalasi Program

Berikut ini adalah implementasi *instalasi* program untuk menggunakan program Sistem Informasi pelayanan Langkah-langkahnya adalah sebagai berikut :

1. Klik menu start kemudian pilih *apachefriends*, xampp, XAMPP Control.



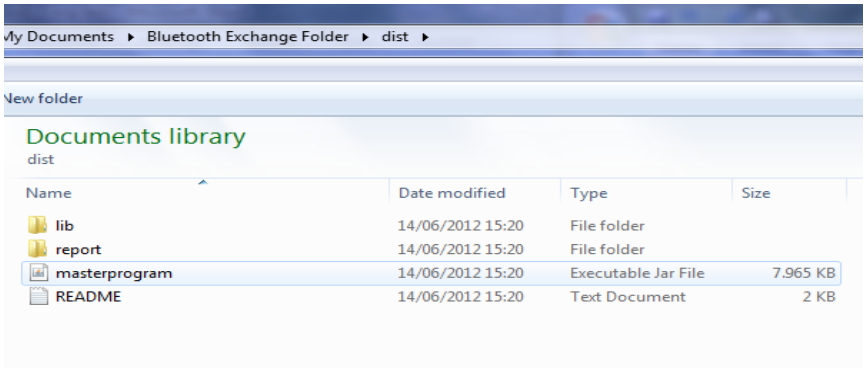
Gambar 5.1 Mengaktifkan XAMPP Control Panel Application.

2. Aktifkan XAMPP *Control Panel Application* seperti pada gambar di bawah.



Gambar 5.2 XAMPP *Control Panel Application.*

3. *Double click* masterprogram dalam menu dist untuk memulai menggunakan program.



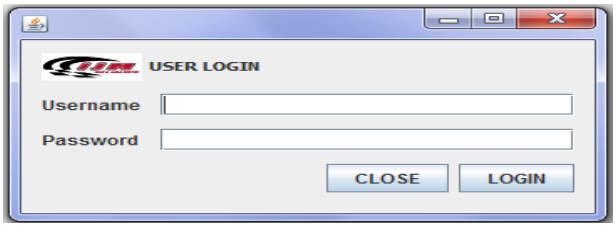
Gambar 5.3 masterprogram

5.1.7 Penggunaan Program

Penggunaan program bertujuan untuk memberikan informasi tentang bagaimana tata cara dalam menjalankan dan menggunakan program Sistem Pelayanan pada Indra Jaya Motor Bandung bagi pengguna atau *user*.

A. *Login User*

Halaman ini merupakan halaman *login user*, sebelum masuk ke sistem terlebih dahulu *user* melakukan *login* dengan memasukan *username* dan *password*.

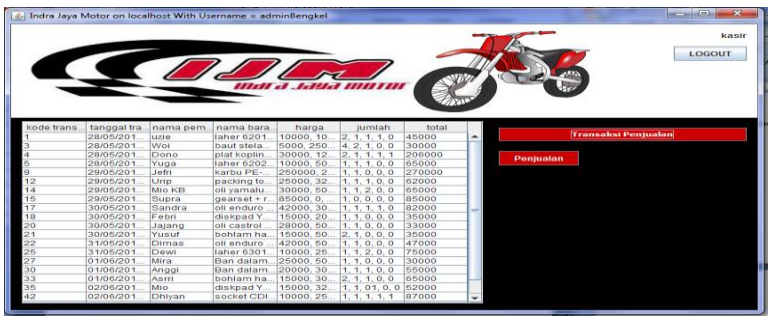


Gambar 5.4 *Login User.*

B. Halaman Kasir

halaman kasir merupakan menu yang dimiliki oleh kasir, setelah kasir *login.*

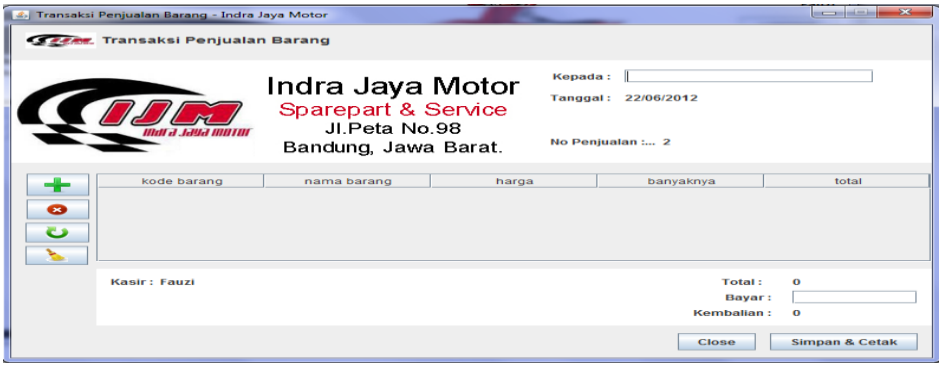
a. Halaman menu Kasir



Gambar 5.5 Halaman Menu Kasir

b. Halaman menu Transaksi Penjualan

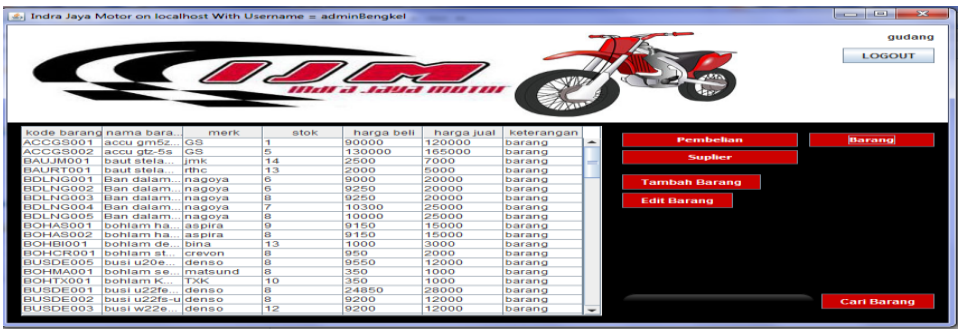
merupakan halaman yang dilakukan kasir untuk transaksi jual – beli.



Gambar 5.6 Halaman Menu Transaksi

C. Halaman Gudang

Merupakan sistem yang dipegang oleh gudang , sebagai berikut :



Gambar 5.7 Halaman Menu Gudang

a. *Stock* Minimum, merupakan peringatan data barang yang akan habis.

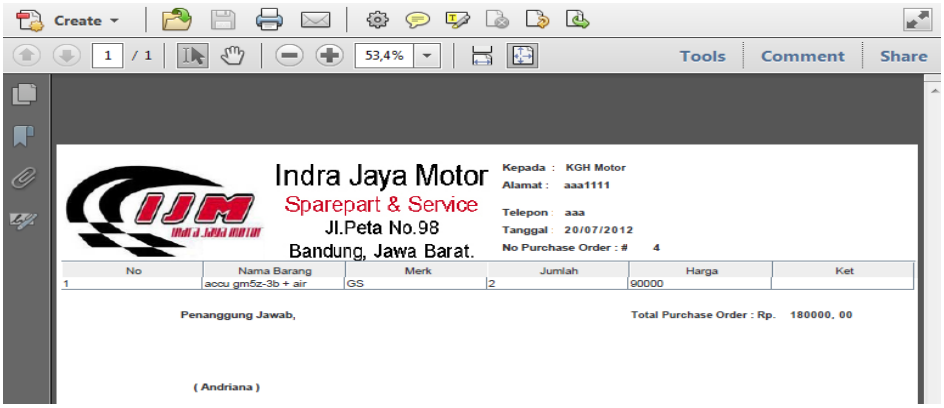


Gambar 5.8 Halaman stock minimum

b. Buat *purchase Order,* merupakan form yang harus diisi ketika *stock* barang muncul hanya *preview* maka data *stock* barang akan tampil kemudian langsung tampil form *purchase Order* maka akan tampil dalam bentuk pdf.

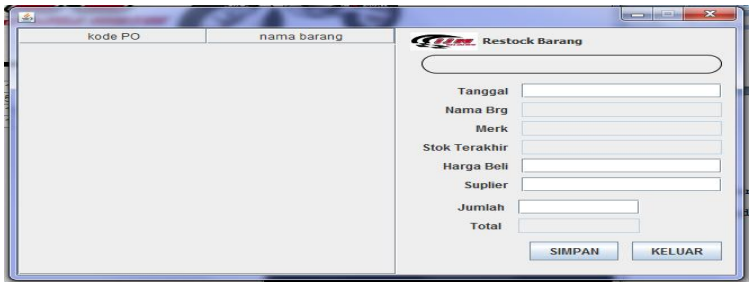


Gambar 5.9 Halaman Buat *purchase Order*



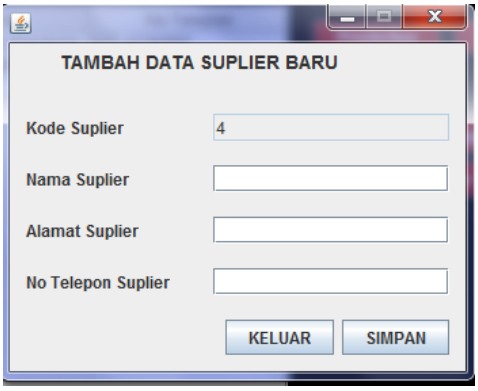
Gambar 5.10 Tampil Print *purchase Order*

c. *Verifikasi purchase Order*, merupakan proses yang dilakukan setelah barang dari *supplier* datang dan data yang diambil berdasarkan data *purchase Order*



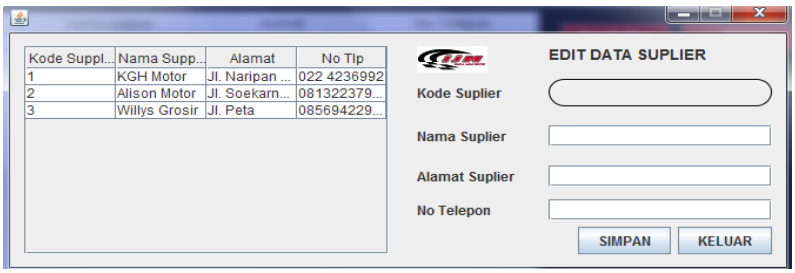
Gambar 5.11 Tampil *Verifikasi purchase Order*

d. Tambah *Supplier*, merupakan halaman tambah data *supplier*



Gambar 5.12 Tambah *supplier*

e. Edit *Supplier*, Merupakan halaman untuk mengedit data *supplier*.



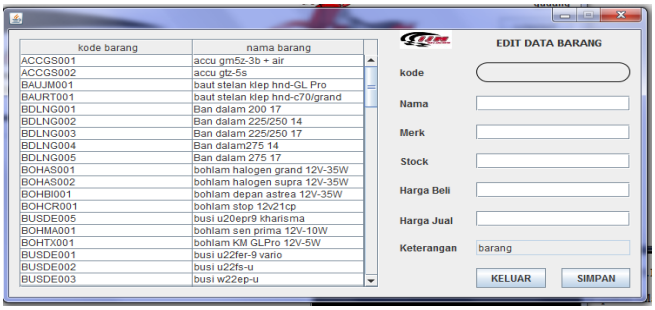
Gambar 5.13 Tampil Edit *Supplier*

f. Tambah Barang, merupakan halaman untuk menambahkan data barang, suku cadang baru yang masuk bengkel dapat dimasukan ke gudang melewati tambah barang..



Gambar 5.14 Halaman Tambah Barang

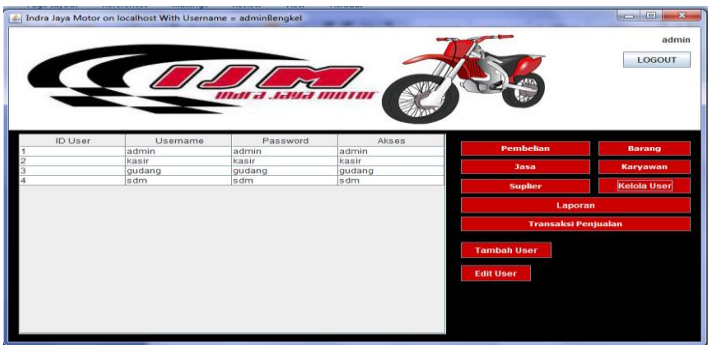
g. Edit Barang, merupakan halaman untuk mengubah data barang.



Gambar 5.15 Halaman Edit Barang

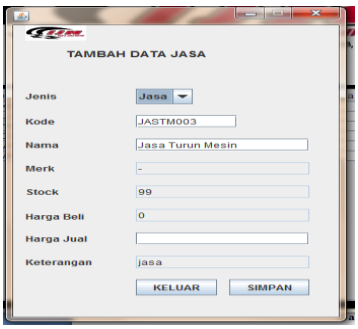
D. Halaman *Owner*

Owner merupakan sebuah *admin*, *owner* dapat melakukan semua proses Selain proses yang dilakukan kasir dan gudang *owner* dapat melakukan sebagai berikut :



Gambar 5.16 Halaman *Owner/admin*

a. Tambah Jasa, merupakan halaman untuk menambahkan jasa *sercvice.*



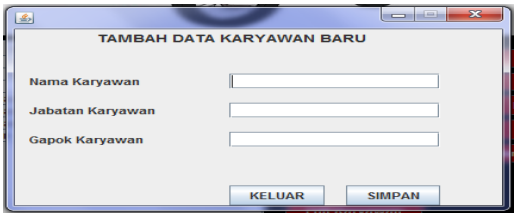
Gambar 5.17 Halaman Tambah Jasa

b. Edit Jasa, merupakan halaman untuk mengubah data Jasa.



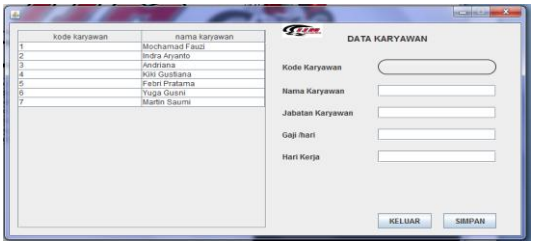
Gambar 5.18 Halaman Edit Jasa

c. Tambah Data Karyawan, merupakan halaman menambahkan data karyawan.



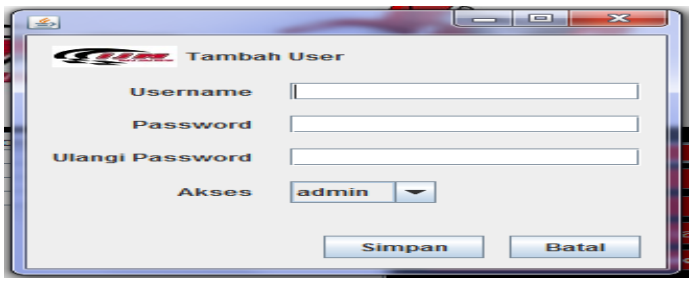
Gambar 5.19 Halaman Tambah Data Karyawan

d. Edit Karyawan, merupakan proses pengubahan data karyawan.



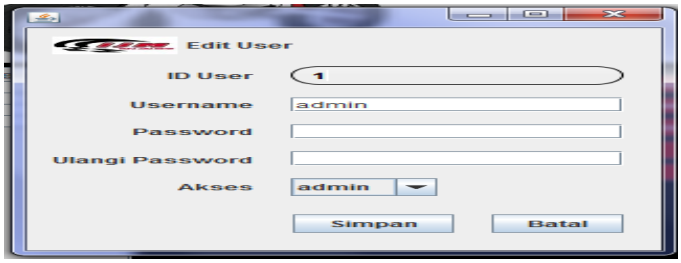
Gambar 5.20 Halaman Edit Karyawan

e. Tambah *User*, merupakan Proses Penambahan *User* baru dalam sistem.



Gambar 5.21 Halaman Tambah *User*

f. Edit *User*, merupakan proses mengubah data *user password* and *username*

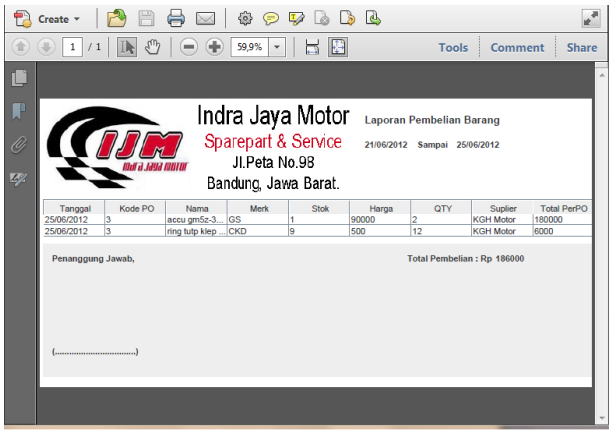


Gambar 5.22 Halaman *Edit User*

g. Laporan Pembelian, merupakan laporan pembelian ke *supplier* , dengan *input* tanggal sesuai keinginan *owner* kemudian akan tampil laporanya.

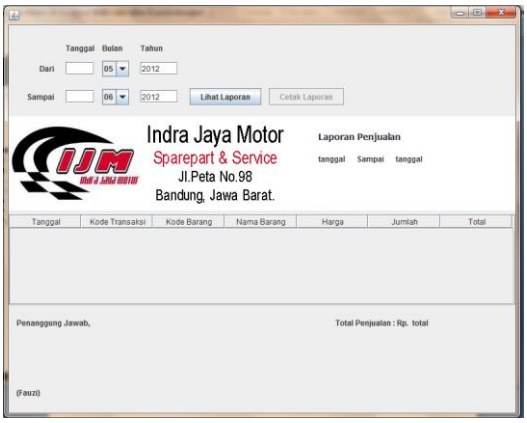


Gambar 5.23 Halaman *Input* Periode Laporan Pembelian

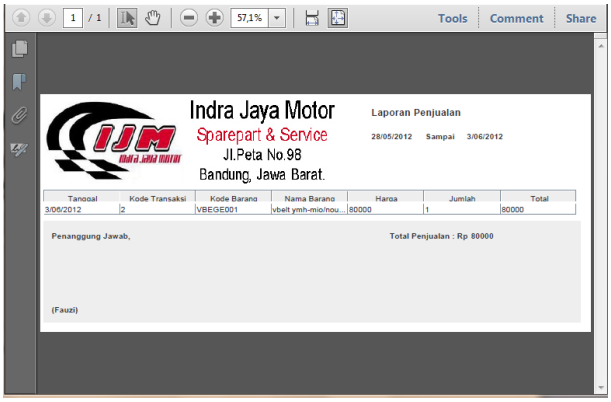


Gambar 5.24 Hasil Cetak Laporan Pembelian

h. Laporan Penjualan, merupakan laporan transaksi penjualan perperiode hanya dengan *input* tanggal dan akan tampil laporannya.

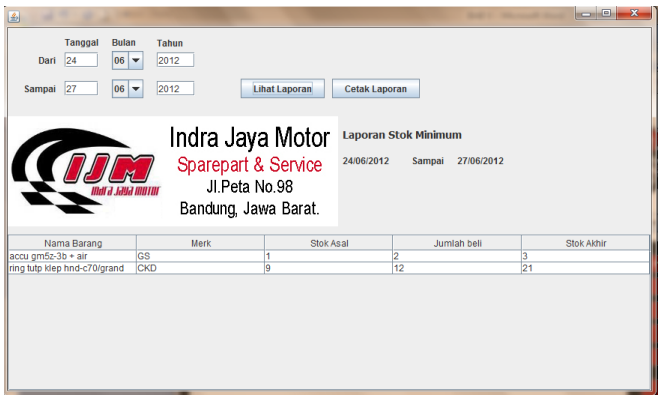


Gambar 5.25 Halaman *Input* Periode Laporan Penjualan

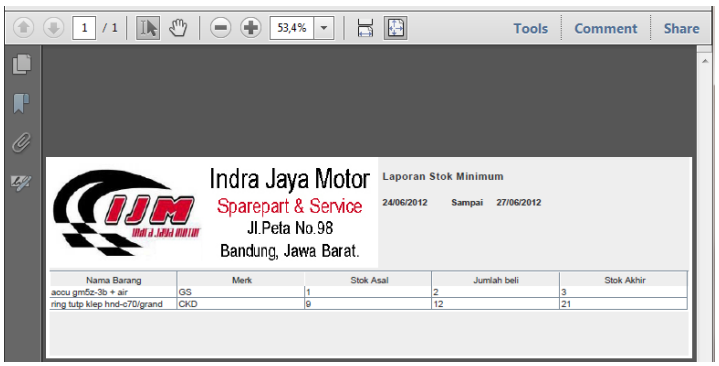


Gambar 5.26 Hasil Cetak Laporan Penjualan

i. Laporan *stock minimum*, merupakan laporan *stock minimu*m yang pernah ada

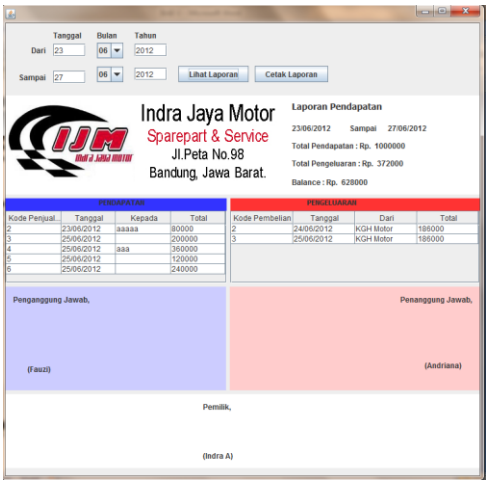


Gambar 5.27 Halaman *Input* stock minimum

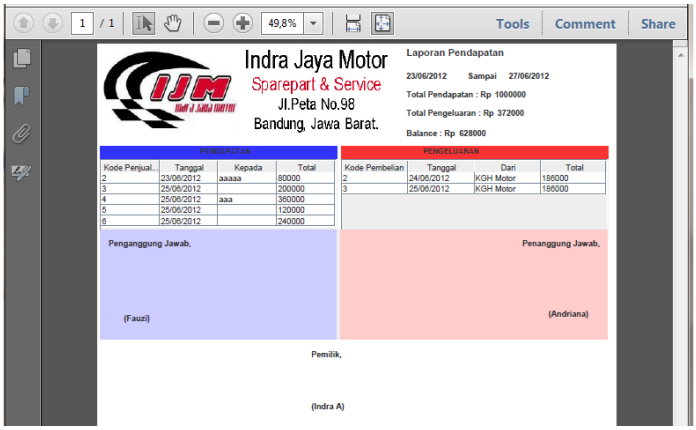


Gambar 5.28 Hasil Cetak Laporan *stock minimum*

j. Laporan Pendapatan, laporan yang diinputkan untuk mengetahui laba rugi atau pendapatan perusahaan.



Gambar 5.29 Input Periode Laporan Pendapatan



Gambar 5.30 Input Periode Laporan Pendapatan

5.2 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu mampu mempresentasikan kajian pokok dari spesifikasi, analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri.

5.2.1 Rencana Pengujian

Pengujian perangkat lunak Sistem Informasi pelayanan pada bengkel Indra Jaya Motor ini menggunakan metode *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

Tabel 5.5 Rencana Pengujian Pada Sistem Informasi Pelayanan

Kelas Uji	**Butir Uji**	**Tingkat Pengujian**	**Jenis Pengujian**
Login	Pengecekan saat memasukan user dan *password* oleh pengguna sistem.	Sistem	*Black box*
Transaksi Penjuaan	Pengecekan Ketika *Input* Transaksi	Sistem	*Black box*
Pembelian	Pengecekan Ketika *Input* Buat *PreOrder* dan *Verifikasi PreOrder*	Sistem	*Black box*
Barang	Pengecekan Ketika *Input* Tambah, Edit dan cari Barang	Sistem	*Black box*
Supplier	Pengecekan ketika *input* Tambah dan edit data *supplier*	Sistem	*Black box*

Jasa	Pengecekan ketika *input* Tambah dan edit data Jasa	Sistem	*Black box*
Karyawan	Pengecekan ketika *input* Tambah dan edit data Karyawan	Sistem	*Black box*
Kelola *User*	Pengecekan ketika *input* Tambah dan edit data Kelola *User*	Sistem	*Black box*
Laporan	Pengecekan ketika *input* periode maka laporan tampil	Sistem	*Black box*

5.2.2 Kasus dan Hasil Penguji

Kasus uji dan hasil pengujian dapat disesuaikan dengan rencana pengujian yang telah dijelaskan sebelumnya. Berikut adalah beberapa kasus uji dan hasil pengujian yang dilakukan, yaitu :

1. Pengujian *Akses Login*

Tabel 5.6 Kasus Uji dan Hasil Pengujian *Login*

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
User name : admin *Password* : admin	Dapat *Login* dan masuk ke halaman utama	Dapat *login* dan masuk ke halaman utama petugas	[X] Diterima [ ] Ditolak
Login	Dapat masuk ke halaman utama petugas	Tombol Simpan dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

User name : bebas *Password* : bebas	Tidak dapat *Login*, Muncul peringatan kesalahan dan kembali lagi ke halaman *login*	Petugas tidak dapat login jika salah *Username* dan / atau *Password*, kembali ke halaman *login*	[ ] Diterima [X] Ditolak

2. Pengujian Transaksi Penjualan

Tabel 5.7 Kasus Uji dan Hasil Pengujian Transaksi Penjualan

Kasus Uji dan Hasil Pengujian (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input Data Transaksi secara benar dan lengkap.	*Input* Transaksi di hitung oleh sistem dan disimpan ke *database*	*Input* Transaksi berhasil menghitung seluruh inputan dan tersimpan di *database*	[X] Diterima [ ] Ditolak
Kasus Uji dan Hasil Pengujian (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input Transaksi di inputkan secara tidak lengkap dan salah.	*Input* Transaksi tidak akan tersimpan kedalam dalam *database* dan akan muncul pesan box contoh: ("Jumlah	*Input* Transaksi tidak tersimpan ke dalam *database.*	[ ] Diterima [X] Ditolak

	Pembayaran Yang anda masukan salah data tidak tersimpan")		

3. Pengujian Transaksi Pembelian

Tabel 5.8 Kasus Uji dan Hasil Pengujian Pembelian

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
membuat form *purchase order*	Saat tekan tombol *Preview* akan muncul *stock* minimum berdasarkan peringatan yang muncul dan buat form *purchase order* maka akan mencetak Form *purchase order* dan data disimpan ke *database*	Tombol *Preview purchase order* dan Buat Form *purchase order* dapat melakukan fungsinya dengan baik	[X] Diterima [ ] Ditolak
Verifikasi purchase order dengan Input kode PO diatas	Saat i*nput* Kode PO data akan tampil dan dapat melakukan restock kemudian tersimpan di *database*	Fungsi pencarian kode berhasil dan tombol simpan berfungsi simpan ke *database*	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			

Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Membuat Form *purchase order* tetapi data dikosongkan	Saat tekan tombol Buat Form *purchase order* maka tidak dapat disimpan ke database dan tampil peringatan, contoh : "Data Tidak Boleh Kosong"	Sesuai dengan yang diharapkan	[ ] Diterima [X] Ditolak
Inputkan Kode Yang tidak sesuai di *Verifikasi purchase order*	Saat *input* kode salah maka tidak akan tampil apapun dan saat tekan tombol simpan tidak akan berfungsi.	Sesuai dengan yang harapankan	[ ] Diterima [X] Ditolak

4. Pengujian Barang

Tabel 5.9 Kasus Uji dan Hasil Pengujian Barang

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah Barang dan *Inputkan.*	Akan tampil ke halaman tambah barang, dan ketika klik tombol simpan maka akan menambah *stock* barang dan tersimpan	Sesuai yang diharapkan semua fungsi berjalan.	[X] Diterima [ ] Ditolak

	di *database.*		
Klik Edit Barang dan *Input* Editan	*Inputan* Kode Barang dan tombol simpan akan mengubah data yang sudah di edit	Sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kotak inputan tidak diisi seperti yang telah ditentukan.	Menampilkan kotak pesan kesalahan.	Sesuai dengan harapan	[ ] Diterima [X] Ditolak

5. Pengujian *Supplier*

Tabel 5.10 Kasus Uji dan Hasil Pengujian *supplier*

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah *supplier* dan *Inputkan.*	Akan tampil ke halaman tambah *supplier*, dan ketika klik tombol simpan maka tersimpan di *database* dan data *supplier* bertambah	Sesuai yang diharapkan semua fungsi berjalan.	[X] Diterima [ ] Ditolak
Klik Edit *Supplier* dan *Input* Editan data *supplier*	*Inputan* Kode *supplier* dan tombol simpan akan mengubah data yang sudah di edit	Sesuai yang diharapkan	[X] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kotak inputan tidak diisi seperti yang telah ditentukan.	Menampilkan kotak pesan kesalahan.	Sesuai dengan harapan	[ ] Diterima [X] Ditolak

6. Pengujian Jasa

Tabel 5.11 Kasus Uji dan Hasil Pengujian Jasa

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah Jasa dan Inputkan.	Akan tampil ke halaman tambah jasa, dan ketika klik tombol simpan.	Sesuai yang diharapkan semua fungsi berjalan.	[X] Diterima [ ] Ditolak
Klik Edit Jasa dan Input Editan	Inputan Kode Jasa dan tombol simpan akan mengubah data yang sudah di edit	Sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kotak inputan tidak diisi seperti yang telah ditentukan.	Menampilkan kotak pesan kesalahan.	Sesuai dengan harapan	[ ] Diterima [X] Ditolak

7. Pengujian Karyawan

Tabel 5.12 Kasus Uji dan Hasil Pengujian Karyawan

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah karyawan dan Inputkan.	Akan tampil ke halaman tambah karyawan, dan ketika klik tombol simpan maka akan menambah.	Sesuai yang diharapkan semua fungsi berjalan.	[X] Diterima [ ] Ditolak
Klik Edit Karyawan dan Input Editan	Inputan Kode Karyawan dan tombol simpan.	Sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kotak inputan tidak diisi seperti yang telah ditentukan.	Menampilkan kotak pesan kesalahan.	Sesuai dengan harapan	[ ] Diterima [X] Ditolak

8. Pengujian Kelola *User*

Tabel 5.13 Kasus Uji dan Hasil Pengujian Kelola *User*

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Klik Tambah User dan Inputkan.	Akan tampil ke halaman tambah *user*, dan ketika klik tombol simpan maka akan menambah *user*	Sesuai yang diharapkan semua fungsi berjalan.	[X] Diterima [ ] Ditolak

	baru dan tersimpan di *database.*		
Klik Edit User dan *Input* Editan	Inputan Kode *User* dan tombol simpan akan mengubah data yang sudah di edit	Sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Kotak inputan tidak diisi seperti yang telah ditentukan.	Tombol tidak berfungsi	Sesuai dengan harapan	[ ] Diterima [X] Ditolak

9. Pengujian Laporan

Tabel 5.14 Kasus Uji dan Hasil Pengujian Laporan

Kasus dan Hasil Uji (Data normal)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Input Periode tanggal secara benar. Untuk laporan penjualan, pembelian	Akan tampil semua data berdasarkan tanggal inputan dan ketik Tekan tombol cetak laporan.	Sesuai yang diharapkan semua fungsi berjalan.	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masuk	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Inputkan data yang tidak benar	Tombol tidak berfungsi	Sesuai dengan harapan	[ ] Diterima [X] Ditolak

5.2.3. Kesimpulan Hasil Pengujian

Kesimpulan pengujian yang dapat di ambil dari kasus uji dan hasil pengujian yaitu bahwa suatu perangkat lunak harus di uji terlebih dahulu agar dalam suatu sistem yang dibuat akan mendapatkan suatu komponen sistem yang baik dan fungsional, sehingga jika kemungkinan ada kesalahan pada sistem akan jauh lebih kecil atau sedikit dan terlihat secara detail dimana letak kesalahan pada sistem yang di rancang tersebut.