BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Sistem

Pada tahapan perancangan sistem ini akan dijelaskan dan digambarkan bagaimana sistem yang akan dibuat, sehingga akan terlihat perbedaan dari sistem awal yang sedang berjalan dengan sistem yang diusulkan.

4.1.1. Tujuan Perancangan

Perancangan sistem secara umum bertujuan untuk memberikan gambaran secara umum kepada pemakai sistem yang baru dan juga merupakan persiapan dari rancangan terperinci dengan mengidentifikasi komponen-komponen sistem informasi, objek yang digunakan, perancangan arsitektur program yang akan dibuat, perancangan tampilan dan perancangan menu. Selain itu, perancangan sistem ini bertujuan untuk mempermudah mengolah data-data penjualan dan pembelian kendaraan di showroom Bersama Motor.

4.1.2. Gambaran Umum Sistem yang Diusulkan

Pada tahap perancangan sistem dilakukan antara lain perancangan untuk format masukan layar *input*, rancangan hasil atau keluaran dari program biasa disebut dengan perancangan *output*, perancangan basis data yang sesuai dengan *file* dan atribut yang ada pada analisis penyimpanan data perancangan menu program yang menjelaskan menu atau dialog yang akan digunakan untuk membangun suatu rancangan struktur program yang berisi mengenai alur proses

program yang dimulai dari masukan sistem menjadi keluaran. Sistem yang diusulkan merupakan tahapan lebih lanjut dari sistem yang sedang berjalan, yang merupakan usulan pemecahan masalah yang dapat membantu dan mempersempit permasalahan yang timbul dari sistem yang dianalisis.

4.1.3. Perancangan Prosedur yang Diusulkan

Berikut prosedur yang diusulkan dalam perancangan Sistem Informasi Pengeloaan Transaksi Jual Beli Kendaraan di Bersama Motor Cirebon.

A. Prosedur Penjualan kendaraan *Cash* di Bersama Motor yang Diusulkan :

1. *Customer* datang ke showroom Bersama Motor kemudian memilih kendaraan yang akan dibeli.
2. Jika *customer* sudah menentukan pilihannya kemudian *customer* menyerahkan data kendaraan kepada administrasi untuk dilakukan pengecekan.
3. Kemudian Administrasi melakukan pengecekan apakah kendaraan tersebut tersedia atau sudah terjual. Jika sudah terjual maka *customer* memilih ulang kendaraan.
4. Jika kendaraan tersedia maka administrasi menginput data penjualan.
5. Kemudian administrasi mencetak *invoice* pembayaran sebanyak dua rangkap. Lembar pertama diserahkan kepada *customer* dan lembar ke dua di arsipkan oleh administrasi.
6. Administrasi mencetak laporan penjualan dan diserahkan kepada pimpinan untuk divalidasi.

7. Pimpinan memvalidasi laporan penjualan dan diserahkan kepada administrasi untuk diarsipkan.

B. Prosedur Penjualan kendaraan Kredit di Bersama Motor yang Diusulkan :

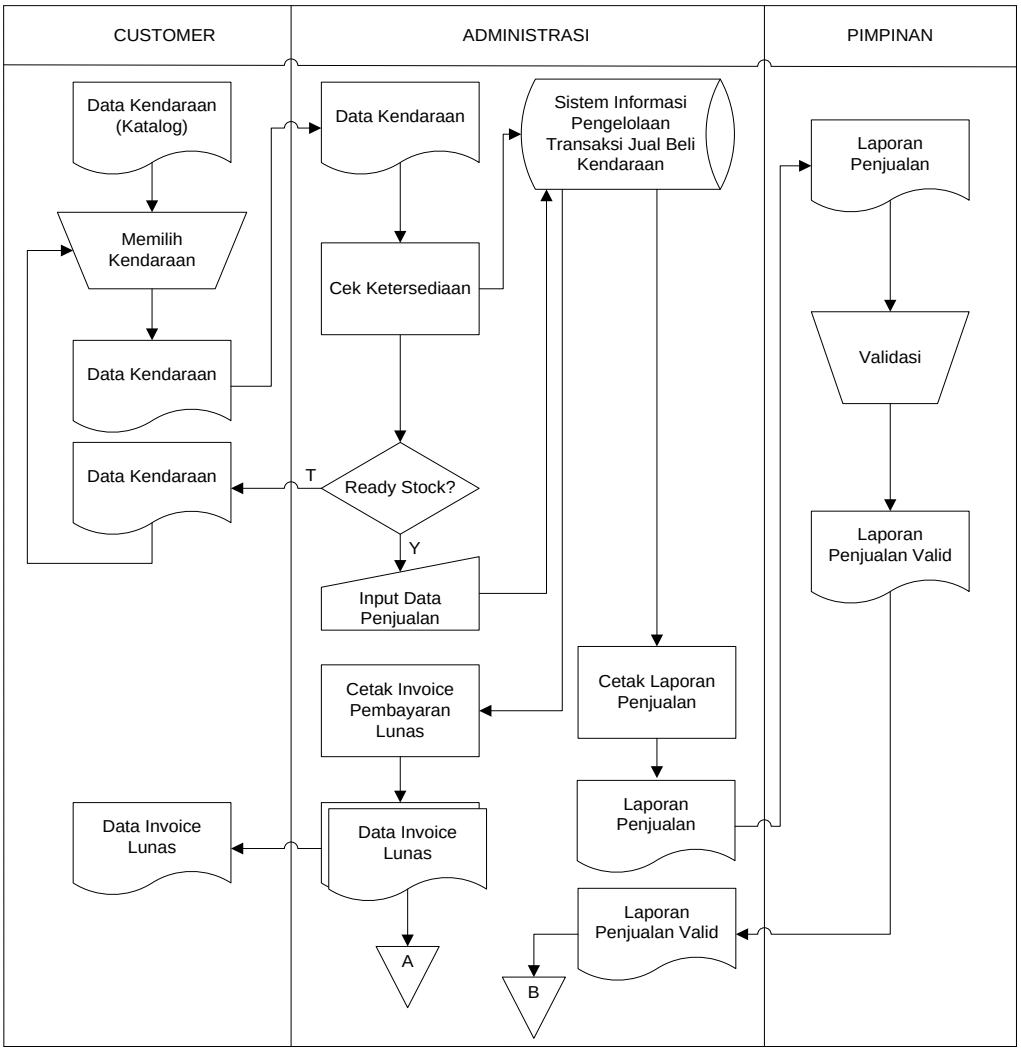
1. *Customer* datang ke showroom kemudian memilih kendaraan yang akan dibeli.
2. Jika *customer* sudah menentukan pilihannya maka *customer* menyerahkan data kendaraan kepada administrasi untuk dilakukan pengecekan.
3. Kemudian Administrasi melakukan pengecekan apakah kendaraan tersebut tersedia atau sudah terjual. Jika sudah terjual maka *customer* memilih ulang kendaraan.
4. Jika kendaraan tersedia maka administrasi akan menginput data penjualan dan pengajuan kredit.
5. Administrasi mencetak *invoice* dan pengajuan kredit sebanyak dua rangkap dan diserahkan kepada *leasing*.
6. Kemudian leasing melakukan *ACC* pengajuan dan data pengajuan kredit *ACC* diserahkan kepada administrasi.
7. Administrasi mencetak *invoice* pembayaran kredit sebanyak dua rangkap, lembar pertama diserahkan kepada *customer* dan lembar ke dua diarsipkan oleh administrasi.
8. Administrasi akan mencetak laporan penjualan dan diserahkan kepada pimpinan untuk divalidasi.
9. Pimpinan memvalidasi laporan penjualan dan diserahkan kepada administrasi untuk diarsipkan.

C. Prosedur Pembelian kendaraan di Bersama Motor yang Diusulkan :

1. *Supplier* menyerahkan daftar stok kendaraan kepada pimpinan.
2. Kemudian pimpinan memilih daftar kendaraan.
3. Daftar kendaraan diserahkan kepada Administrasi.
4. Administrasi menginput *purchese order* dan melihat stok data kendaraan.
5. Kemudian Administrasi mencetak *purchese order* dan diserahkan kepada *supplier.*
6. *Supplier* membuat faktur pembelian dan diserahkan kepada Administrasi.
7. Administrasi menginput data pembelian ke dalam sistem.
8. Kemudian administrasi mencetak laporan pembelian dan diserahkan kepada pimpinan untuk divalidasi.
9. Pimpinan memvalidasi laporan pembelian dan diserahkan kepada administrasi untuk diarsipkan.

4.1.3.1. Flowmap

1. *Flowmap* Penjualan *Cash* yang Diusulkan



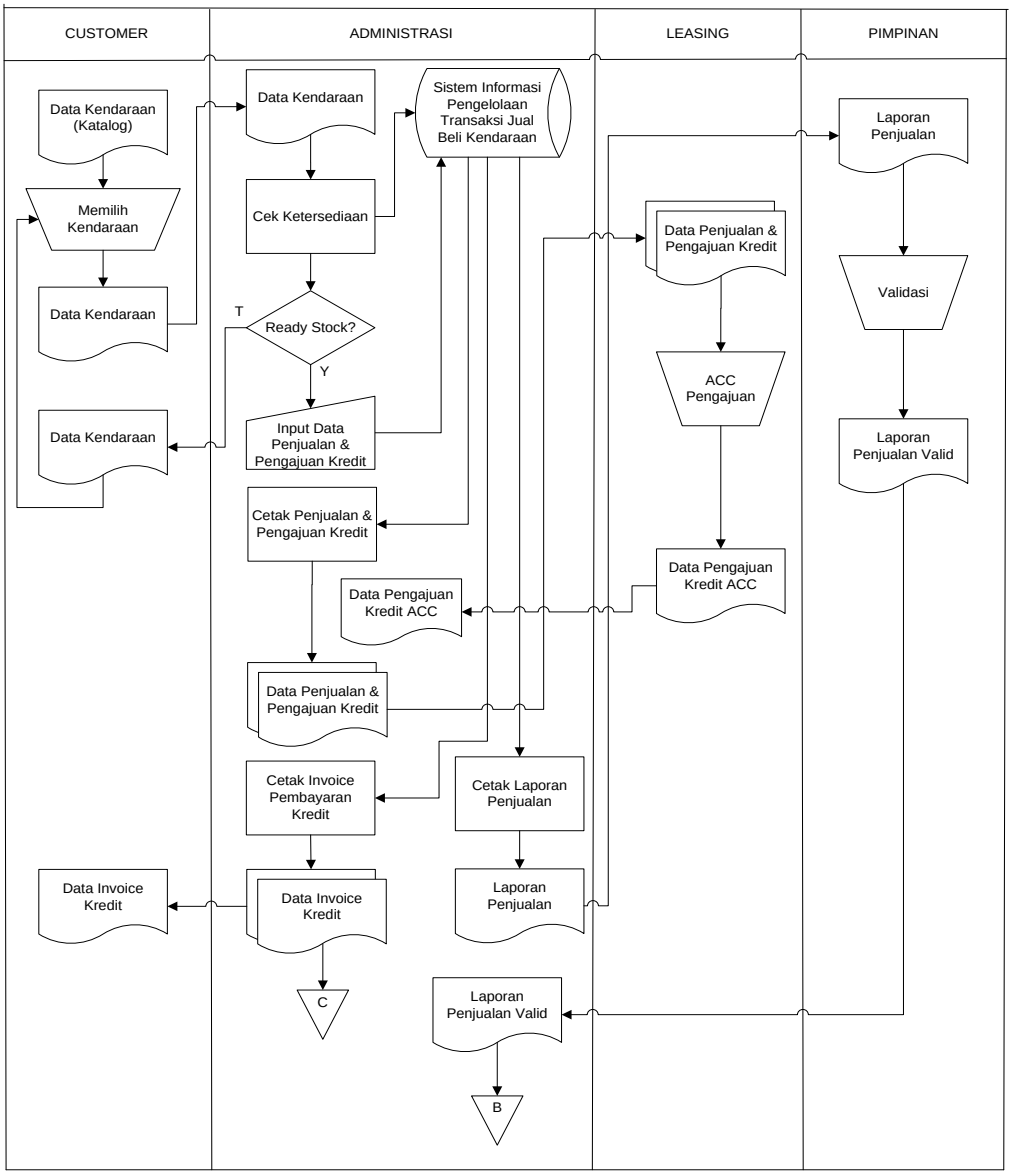
Gambar 4.1 *Flowmap* penjualan *Cash* yang Diusulkan

Keterangan :

A : Arsip *Invoice* Lunas

B : Arsip Laporan Penjualan Valid

2. *Flowmap* Penjualan Kredit yang Diusulkan



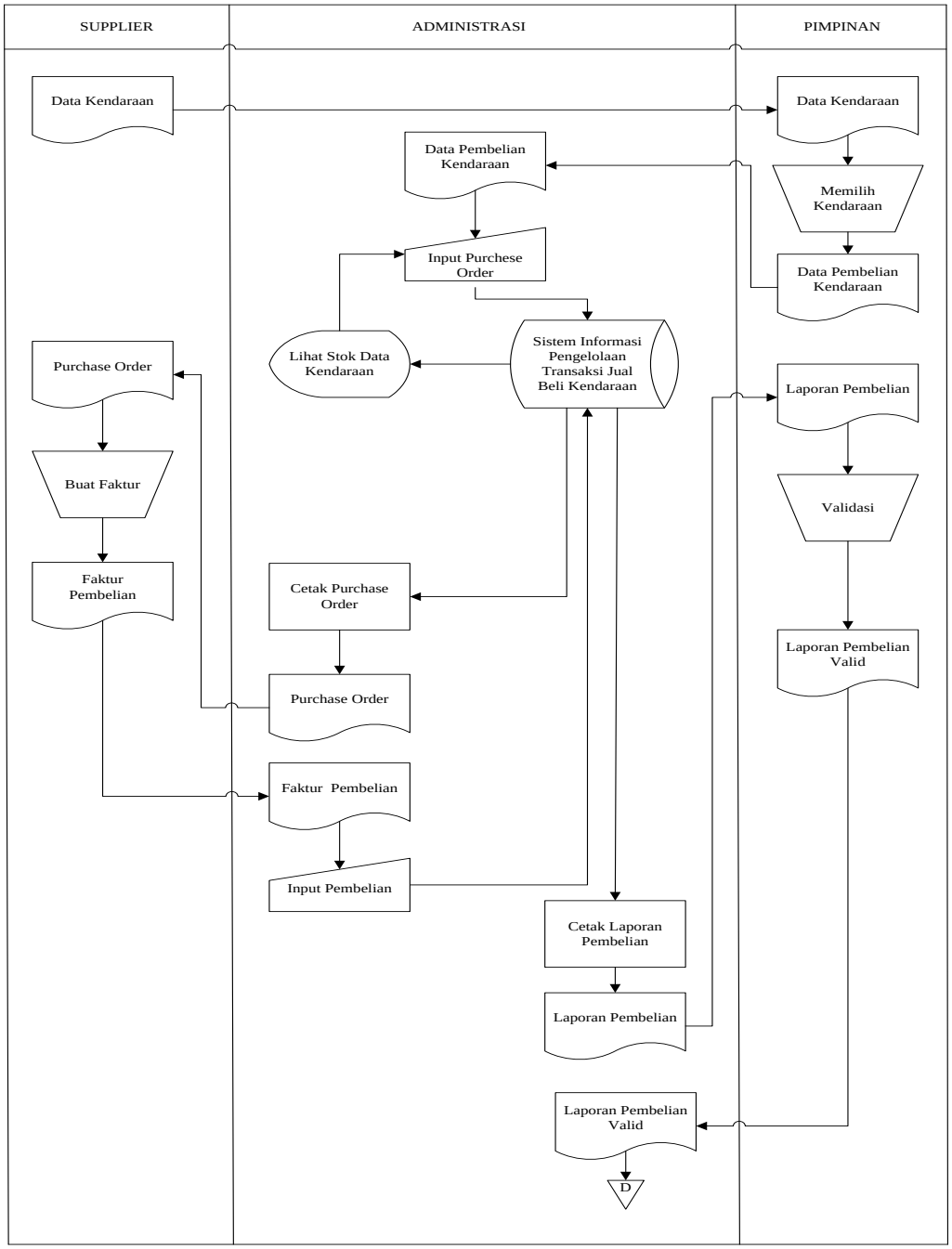
Gambar 4.2 *Flowmap* penjualan Kredit yang Diusulkan

Keterangan :

B : Arsip Laporan Penjualan Valid

C : Arsip *Invoice* Kredit

3. *Flowmap* Pembelian yang Diusulkan



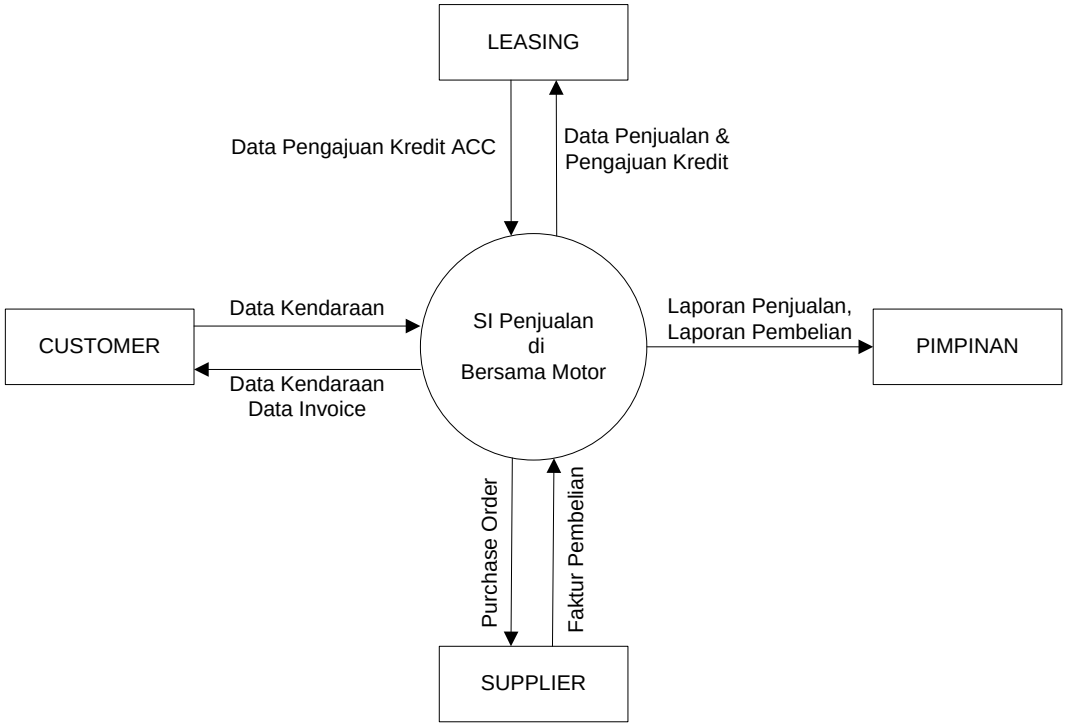
Gambar 4.3 *Flowmap* Pembelian yang diusulkan

Keterangan :

D : Laporan Pembelian Valid

4.1.3.2. Diagram konteks (*context Diagram*)

Diagram konteks (*context Diagram*) merupakan alat bantu untuk struktur analisis. Pendekatan terstruktur ini mencoba untuk menggambarkan sistem secara garis besar atau keseluruhan. Pada diagram konteks ini dianalisis terlebih dahulu. Sistem informasi yang dibuat akan menghasilkan informasi apa saja, sumber informasi yang dibutuhkan dan tujuan dari informasi yang dihasilkan untuk lebih jelasnya mengenai diagram konteks digambarkan sebagai berikut :

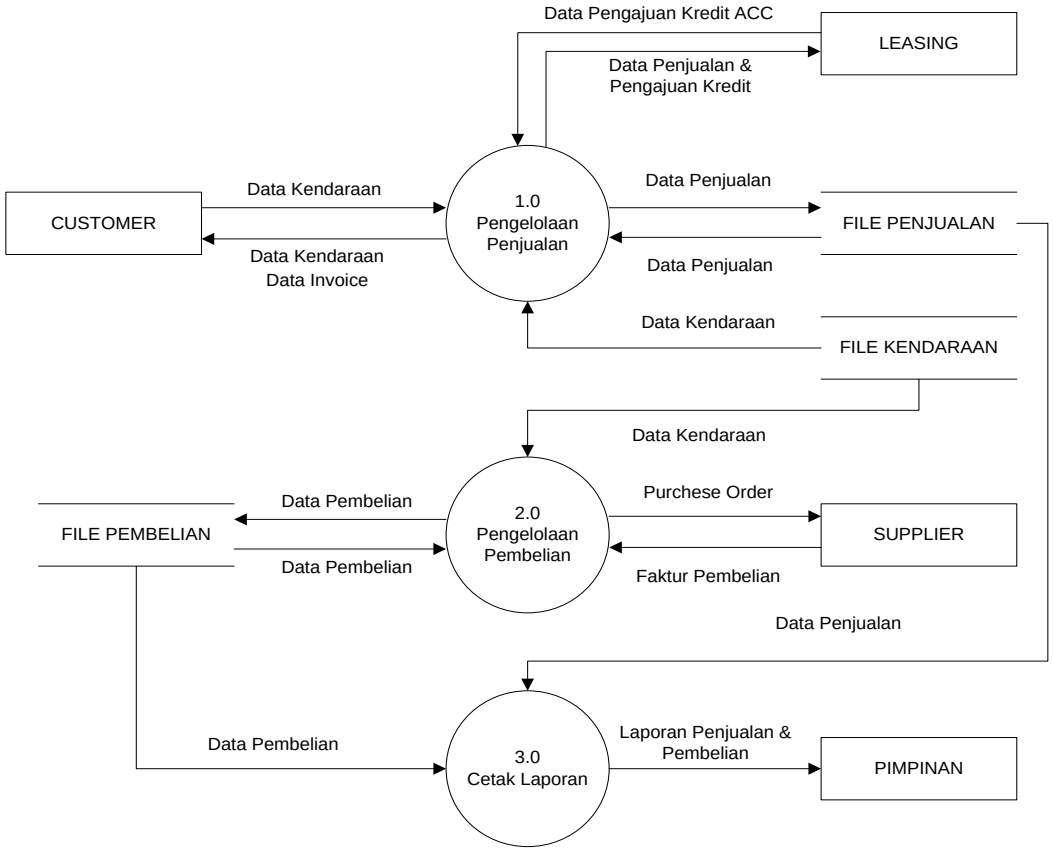


Gambar 4.4. Diagram Konteks yang Diusulkan

4.1.3.3. *Data Flow Diagram* (DFD)

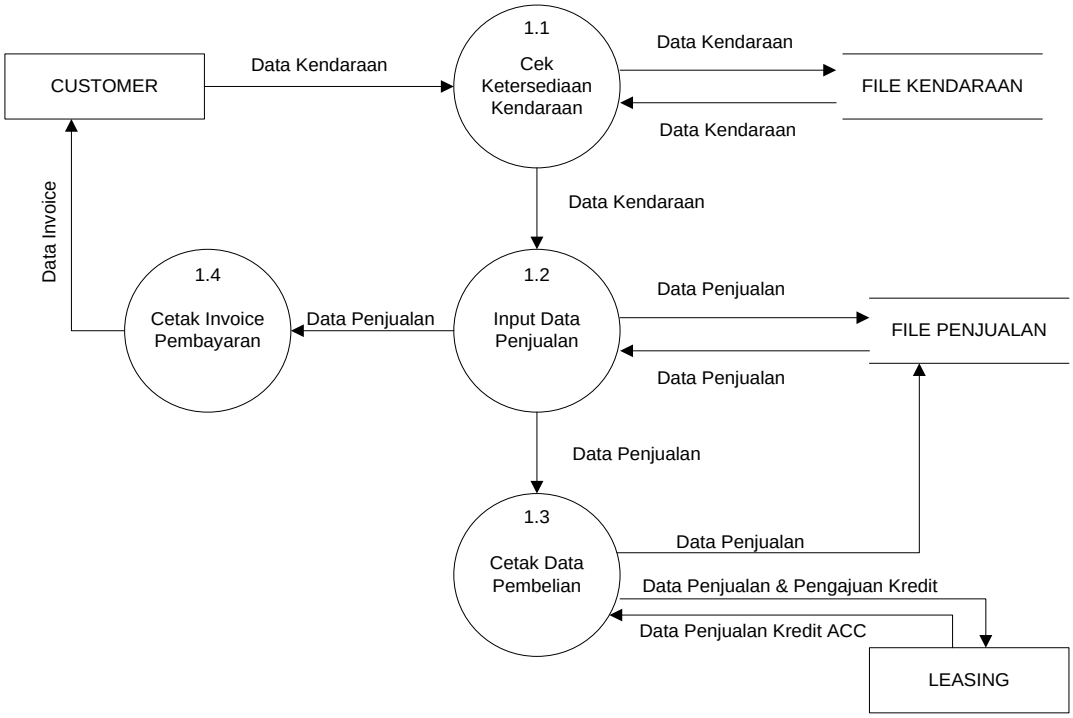
Diagram alir data atau *data flow diagram* ini merupakan alat yang digunakan metodologi pengembangan sistem yang terstruktur. Diagram alir data ini berfungsi untuk menggambarkan arus dalam sistem dengan terstruktur dan jelas. Berikut DFD yang diusulkan di Showroom Bersama Motor.

a. *Data Flow Diagram* Level 1 Sistem Informasi Pengelolaan Transaksi Jual Beli di Showroom Bersama Motor Cirebon



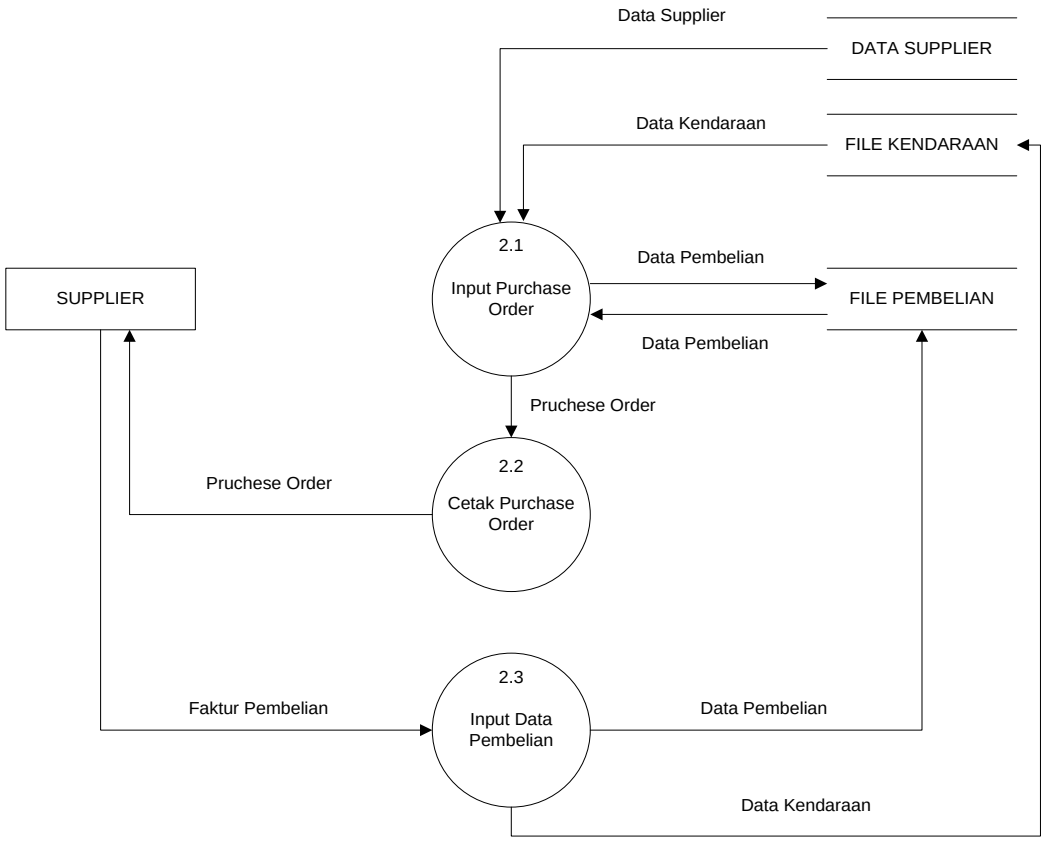
Gambar 4.5. DFD Level 1 yang Diusulkan.

b. *Data Flow Diagram* Level 2 Proses 1 Sistem Informasi Pengelolaan Transaksi Jual Beli di Showroom Bersama Motor Cirebon



Gambar 4.6. DFD Level 2 Proses 1 yang Diusulkan.

c. *Data Flow Diagram* Level 2 Proses 2 Sistem Informasi Pengelolaan Transaksi Jual Beli di Showroom Bersama Motor Cirebon



Gambar 4.7. DFD Level 2 Proses 2 yang Diusulkan.

4.1.3.4. Kamus Data

Kamus data merupakan alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir pada sistem yaitu tentang data yang masuk kedalam sistem dan tentang informasi yang dibutuhkan oleh *user*. Adapun kamus data perancangan sistem yang di usulkan adalah sebagai berikut :

1	Nama Arus Data	:	Data Kendaraan
	Alias	:	-
	Arus Data	:	*Customer* - P.1, P.1 - *Customer*, File Kendaraan - P.1, File Kendaraan - P.2, *Customer* - P.1.1, P.1.1 - File Kendaraan, File Kendaraan - P.1.1, P.1.1 - P.1.2, File Kendaraan - P.2.1, P.2.3 - File Kendaraan
	Struktur Data	:	nomor_rangka, nomor_mesin, merk, type, jenis, warna, isi_silinder, thn_pembuatan, foto, foto_dua, foto_tiga, kondisi, harga_jual, harga_beli, status, no_polisi, bahan_bakar, nama_pemilik, alamat_pemilik, harga_penawaran, no_bpkb.
2	Nama Arus Data	:	Data Penjualan
	Alias	:	*Invoice* Penjualan
	Arus Data	:	P.1 – File Penjualan, File Penjualan - P.1, File Penjualan - P.3, P.3 - Pimpinan, P.1.2 - File Penjualan, File Penjualan - P.1.2, P.1.2 – P.1.4, P.1.4 - *Customer*, P.1.2 - P.1.3, P.1.3 - File Penjualan, P.1.3 – Leasing, Leasing – P.1.3.

	Struktur Data	:	no_invoice, tgl_invoice, nama_cust, alamat, telp, no_rangka, harga_jual, dp, sisa_pembayaran, mediator, pembayaran, bank_pembiayaan.
3	Nama Arus Data	:	Data Pembelian
	Alias	:	Faktur Pembelian
	Arus Data	:	File Pembelian - P.2, P.2 - File Pembelian, P.2 - Supplier, Supplier - P.2, File Pembelian - P.3, P.3 - Pimpinan, P.2.1 - File Pembelian, File Pembelian - P.2.1, P.2.1 - P.2.2, P.2.2 - Supplier, Supplier - P.2.3, P.2.3 - File Pembelian
	Struktur Data	:	no_faktur, tgl_faktur, nama_penjual, alamat, telp, no_rangka, no_mesin, jenis, merk, type, thn_pembuatan, no_polisi, warna, harga_beli, kondisi, bahan_bakar, nama_pemilik, isi_silinder.
4	Nama Arus Data	:	*Purchase Order*
	Alias	:	P.O
	Arus Data	:	P.2 – Supplier, P.2.1 – file_pembelian, file_pembelian – P.2.1, P.2.1 – P.2.2, P.2.2 – Supplier.
	Struktur Data	:	no_po, nama_supplier, tgl, merk, type, jenis, warna, jumlah.

5	Nama Arus Data	:	Data Supplier
	Alias	:	-
	Arus Data	:	data_supplier – P.2.1
	Struktur Data	:	nama_supplier, alamat_supplier, telp_supplier

4.1.4. Perancangan Basis Data

Sebagai penunjang sistem pengolahan data, maka harus ditentukan bagaimana bentuk rancangan basis data yang digunakan. Perancangan basis data ini dimaksudkan untuk mengidentifikasi kebutuhan file-file basis data yang diperlukan sistem. Dalam perancangan basis data ini akan dibahas tentang Normalisasi, Relasi Tabel, Entity Relationship Diagram (ERD), Struktur File, Kodefikasi.

4.1.4.1. Normalisasi

Unnormal

Unnormal : { nomor_rangka, nomor_mesin, merk, type, jenis, warna, isi_silinder, thn_pembuatan, foto, foto_dua, foto_tiga, kondisi, harga_jual, harga_beli, status, no_polisi, bahan_bakar, nama_pemilik, alamat_pemilik, harga_penawaran, no_bpkb, no_invoice, tgl_invoice, nama_cust, alamat, telp, no_rangka, harga_jual, dp, sisa_pembayaran, mediator, pembayaran, no_faktur, tgl_faktur, nama_penjual, alamat, telp, no_rangka, no_mesin, jenis, merk, type, thn_pembuatan, no_polisi, warna, harga_beli, kondisi, bahan_bakar, nama_pemilik, isi_silinder, no_po, nama_supplier, tgl, merk, type, jenis, warna, jumlah, nama_supplier, alamat_supplier, telp_supplier, no_rangka, no_invoice, tgl_invoice, harga_jual, no_faktur, tgl_faktur, harga_beli, pendapatan }

Normal 1

Kendaraan : { nomor_rangka, nomor_mesin, merk, type, jenis, warna, isi_silinder, thn_pembuatan, foto, kondisi, harga_jual, harga_beli, status, no_polisi, bahan_bakar, nama_pemilik, alamat_pemilik, harga_penawaran, no_bpkb, no_invoice, tgl_invoice, nama_cust, alamat, telp, dp, sisa_pembayaran, mediator, pembayaran, no_faktur, tgl_faktur, nama_penjual, no_po, nama_supplier, jumlah, nama_supplier, alamat_supplier, telp_supplier, pendapatan }

Normal 2

Kendaraan : { nomor_rangka*, nomor_mesin, merk, type, jenis, warna, isi_silinder, thn_pembuatan, foto, kondisi, harga_jual, harga_beli, status, no_polisi, bahan_bakar, nama_pemilik, harga_penawaran, no_bpkb }

Penjualan **:** { no_invoice*, tgl_invoice, nama_cust, alamat, telp, no_rangka, harga_jual, dp, sisa_pembayaran, mediator, pembayaran, keterangan }

Pembelian **:** { no_faktur*, tgl_faktur, nama_penjual, alamat, telp, nomor_rangka**, no_mesin, jenis, merk, type, thn_pembuatan, no_polisi, warna, harga_beli, kondisi, bahan_bakar, nama_pemilik, isi_silinder, harga_penawaran }

Normal 3

Penjualan : { no_invoice*, tgl_invoice, nama_cust, alamat, telp, nomor_rangka** }

Detail_penjualan **:** { no_invoice**, tgl_invoice, harga_penawaran, harga_jual, dp, sisa_pembayaran, mediator }

Pembelian **:** { no_faktur*, tgl_faktur, nama_supplier, alamat_penjual, telp, nomor_rangka** }

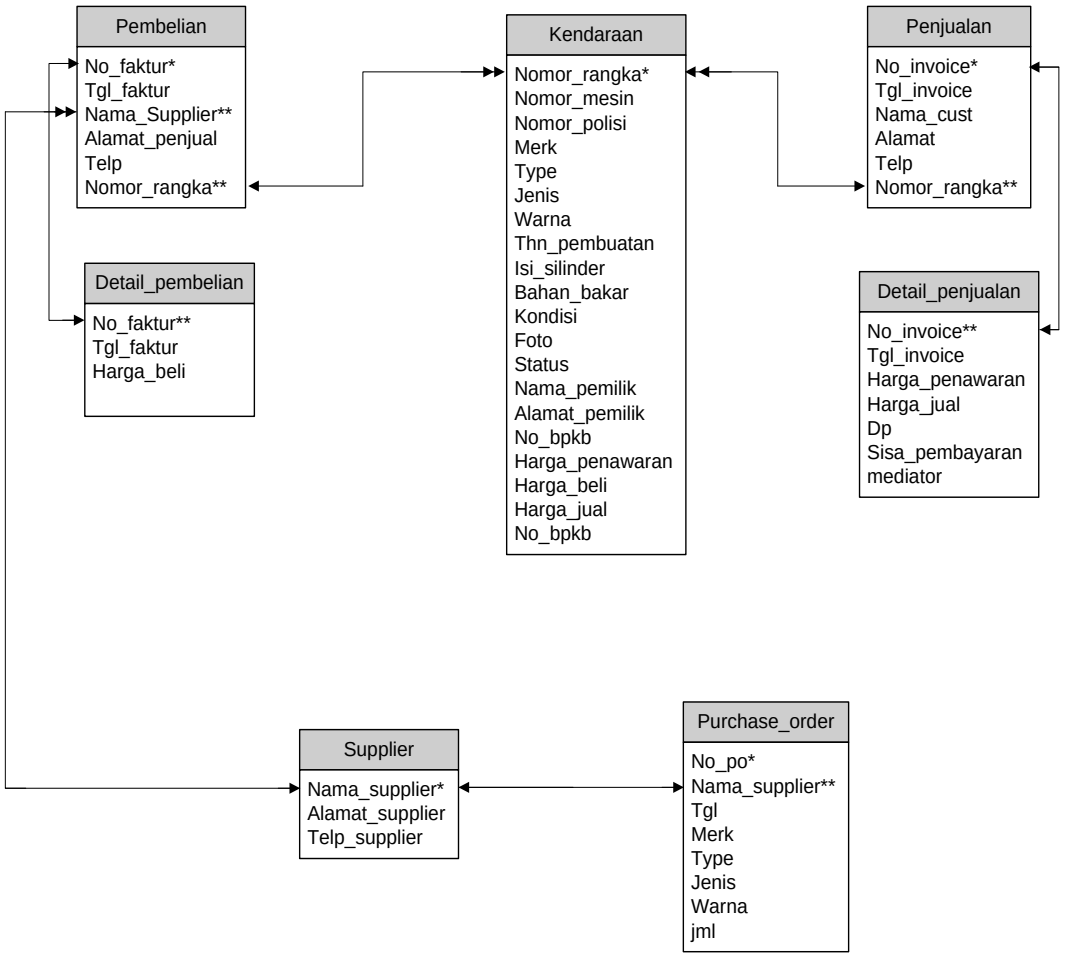
Detail_pembelian : { no_faktur**, tgl_faktur, harga_beli, }

Supplier : { nama_supplier*, alamat_supplier, telp_supplier }

Purchase Order : { no_po*, nama_supplier**, tgl, merk, type, jenis, warna, jml }

4.1.4.2. Relasi Tabel

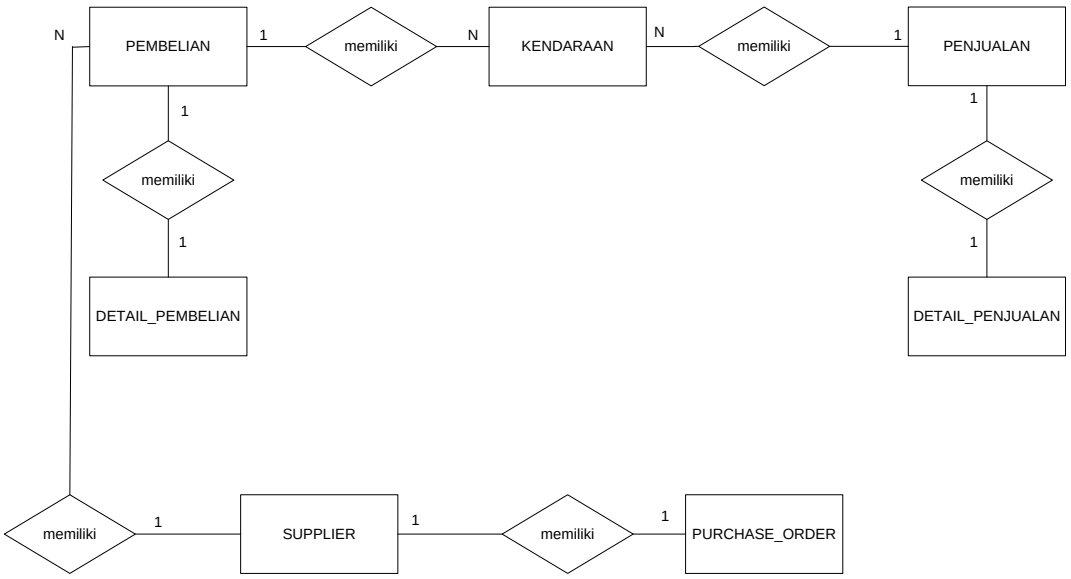
Proses ini merupaka proses hubungan antara file yang satu dengan file yang lain yang saling berhubungan, proses hubungan tersebut antara file yang mempunyai kunci yang sama sehingga file-file tersebut menjadi satu kesatuan yang dihubungkan oleh file kunci tersebut. Adapun gambaran dari bentuk relasinya adalah sebagai berikut :



Gambar 4.8. Relasi Tabel Bersama Motor

4.1.4.3. *Entity Relationship Diagram* (ERD)

Entitas adalah objek yang ada dan dapat dibedakan dari objek yang lain. Sedangkan relasi adalah asosiasi antar entitas, jadi suatu model relasi dan data relasi digambarkan dengan sekumpulan table yang memiliki kolom dengan nama yang unik. Jadi *Entity Relationship Diagram* (ERD) merupakan salah satu cara untuk mengolah *database* sehingga data tersebut dapat diketahui hubungan antar filenya.



Gambar 4.9. ERD Bersama Motor

4.1.4.4. Struktur File

Tujuan dari perancangan strutur file ini yaitu untuk menentukan nama *filed*, *type field*, lebar *field*, dan keterangan dari *field* tersebut yang ada pada setiap file. Struktur file yang akan digunakan dalam perancangan sistem ini akan menentukan struktur fisik *database* yang menunjukkan struktur dari elemen-elemen yang menyatakan panjang data dan tipe datanya. Pengembangan struktur file yang akan diuraikan adalah sebagai berikut :

1. *File User*

   Nama *File* : *File* Data *User*

   Primary Key : *username*

   Foreign Key : -

Tabel 4.1. Atribut Tabel Login

No	Nama Field	Type	Size	Keterangan
1	Nama	Varchar	20	Nama
2	***Username****	Varchar	50	*Username*
3	*Password*	Varchar	50	*Password*

2. *File* Kendaraan

   Nama File : Kendaraan

   Primary Key : nomor_rangka

   Foreign Key : -

Tabel 4.2. Atribut Tabel Kendaraan

No	Nama Field	Type	Size	Keterangan
1	**nomor_rangka***	Varchar	25	Nomor Rangka Kendaraan (primary key)
2	nomor_mesin	Varchar	25	Nomor Mesin Kendaraan
3	nomor_polisi	Varchar	10	Nomor Polisi Kendaraan
4	Merk	Varchar	20	Merk Kendaraan
5	Type	Varchar	20	Type Kendaraan
6	Jenis	Varchar	20	Jenis Kendaraan

7	Warna	Varchar	15	Warna Kendaraan
8	thn_pembuatan	Varchar	4	Tahun Kendaraan
9	Kondisi	Varchar	10	Kondisi Kendaraan
10	isi silinder	Integer	10	Isi Silinder Kendaraan
11	bahan_bakar	Varchar	10	Bahan Bakar Kendaraan
12	Foto	Varchar	500	Foto Kendaraan
13	no_bpkb	Varchar	15	No Bpkb Kendaraan
14	Status	Varchar	15	Status Kendaraan
15	Nama_pemilik	Varchar	30	Nama Pemilik Kendaraan
16	Alamat_pemilik	Varchar	50	Alamat Pemilik Kendaraan
17	Harga_penawaran	Double	10	Harga Penawaran Kendaraan
18	Harga_jual	Double	10	Harga jual Kendaraan
19	Harga_beli	Double	10	Harga Beli Kendaraan

3. *File* Penjualan

   Nama File : Penjualan

   Primary Key : no_invoice

   Foreign Key : nomor_rangka

Tabel 4.3. Atribut Tabel Penjualan

No	Nama Field	Type	Size	Keterangan
1	**no_invoice***	Varchar	15	Nomor Faktur Penjualan (primary

				key)
2	nomor_rangka**	Varchar	25	Nomor Rangka (foreign key)
3	tgl_invoice	Date	10	Tanggal Faktur Penjualan
4	Nama_cust	Varchar	30	Nama
5	Alamat	Varchar	50	Alamat
6	Telp	Varchar	15	Telepon

4. *File* Detail Penjualan

   Nama File : Detail Penjualan

   Primary Key : -

   Foreign Key : no_invoice

Tabel 4.4. Atribut Tabel Detail Penjualan

No	Nama Field	Type	Size	Keterangan
1	**no_invoice****	Varchar	8	Nomor Faktur Penjualan (foreign key)
2	tgl_invoice	date	10	Tgl Invoice Penjualan
3	Harga_penawaran	Double	10	Harga Penawaran
4	Harga_jual	Double	10	Harga Jual Kendaraan
5	Dp	Double	10	Dp
6	Sisa_pembayaran	Double	10	Sisa Pembayaran
7	Mediator	Varchar	30	Mediator

5. File Pembelian

   Nama File : Pembelian

   Primary Key : no_faktur

   Foreign Key : nomor_rangka, nama_supplier

Tabel 4.5. Atribut Tabel Pembelian

No	Nama Field	Type	Size	Keterangan
1	**no_faktur***	Varchar	15	Nomor Faktur Pembelian (primary key)
2	tgl_faktur	Date	10	Tanggal Faktur Pembelian
3	nama_supplier**	Varchar	25	Nama Supplier
4	alamat_penjual	text		Alamat Penjual
5	Telp	Varchar	15	Telpon Penjual
6	nomor_rangka**	Varchar	25	Nomor Rangka (foreign key)

6. File Detail Pembelian

   Nama File : Detail Pembelian

   Primary Key : -

   Foreign Key : no_faktur

Tabel 4.6. Atribut Tabel Detail Pembelian

No	Nama Field	Type	Size	Keterangan
1	**no_faktur****	Varchar	15	Nomor Faktur Pembelian (foreign key)
2	Tgl_faktur	Date	10	Tanggal Faktur Pembelian
3	Harga_beli	Double	10	Harga Beli Kendaraan

7. File Supplier

   Nama File : Supplier

   Primary Key : nama_supplier

   Foreign Key : -

Tabel 4.7. Atribut Tabel Supplier

No	Nama Field	Type	Size	Keterangan
1	**Nama_supplier***	Varchar	25	Nama Supplier (primary key)
2	Alamat_supplier	Varchar	40	Alamat supplier
3	Telp_supplier	Varchar	15	Telp supplier

8. *File Purchese Order*

   Nama File : Purchase Order

   Primary Key : no_po

   Foreign Key : nama_supplier

Tabel 4.8. Atribut Tabel *Purchese Order*

No	Nama Field	Type	Size	Keterangan
1	**no_po***	Varchar	15	No PO (primary key)
2	Nama_supplier**	Varchar	25	Nama supplier (Foreign Key)
3	Tgl	date	10	Tanggal PO
4	Merk	Varchar	15	Merk Kendaraan
5	Type	Varchar	15	Type Kendaraan
6	Jenis	Varchar	15	Jenis Kendaraan
7	Warna	Varchar	15	Warna Kendaraan
8	Jumlah	int	10	Jumlah Pembelian

4.1.4.5 Kodifikasi

Pengkodean dibutuhkan dalam suatu sistem informasi yaitu untuk mempermudah dalam pencarian suatu objek.

Adapun pengkodean dalam sistem informasi pengelolaan transaksi jual beli yang digunakan adalah sebagai berikut :

1. Faktur Penjualan

   Format : FJL–XXXXXXXX

Keterangan :

FJL merupakan kode untuk *invoice* penjualan.

XX menunjukkan kode jenis kendaraan (01 mempunyai arti kode jenis kendaraan minibus, 02 mempunyai arti kode jenis kendaraan sedan, 03 mempunyai arti kode jenis kendaraan truck, 04 mempunyai arti kode jenis kendaraan dump truck).

XX menunjukkan kode jenis merk kendaraan (01 menpunyai arti kode merk kendaraan mitsubishi, 02 mempunyai arti kode merk kendaraan toyota, 03 mempunyai arti kode merk kendaraan daihatsu, 04 mempunyai arti kode merk kendaraan honda, 05 mempunyai arti kode merk kendaraan hyundai, 06 mempunyai arti kode merk kendaraan mercedes, 07 mempunyai arti kode merk kendaraan bmw).

XXXX menunjukkan nomer urut *invoice* penjualan.

Contoh :

FJL-01020001 kode ini menunjukkan adalah *invoice* penjualan, jenis kendraan minibus, merk kendaraan toyota dengan nomor urut 0001.

2. Faktur Pembelian

Format : FBL–XXXXXXXX

Keterangan :

FBL merupakan kode untuk faktur pembelian.

XX menunjukkan kode jenis kendaraan (01 mempunyai arti kode jenis kendaraan minibus, 02 mempunyai arti kode jenis kendaraan sedan, 03

mempunyai arti kode jenis kendaraan truck, 04 mempunyai arti kode jenis kendaraan dump truck).

XX menunjukkan kode jenis merk kendaraan (01 menpunyai arti kode merk kendaraan mitsubishi, 02 mempunyai arti kode merk kendaraan toyota, 03 mempunyai arti kode merk kendaraan daihatsu, 04 mempunyai arti kode merk kendaraan honda, 05 mempunyai arti kode merk kendaraan hyundai, 06 mempunyai arti kode merk kendaraan mercedes, 07 mempunyai arti kode merk kendaraan bmw).

XXXX menunjukkan nomer urut faktur penjualan.

Contoh :

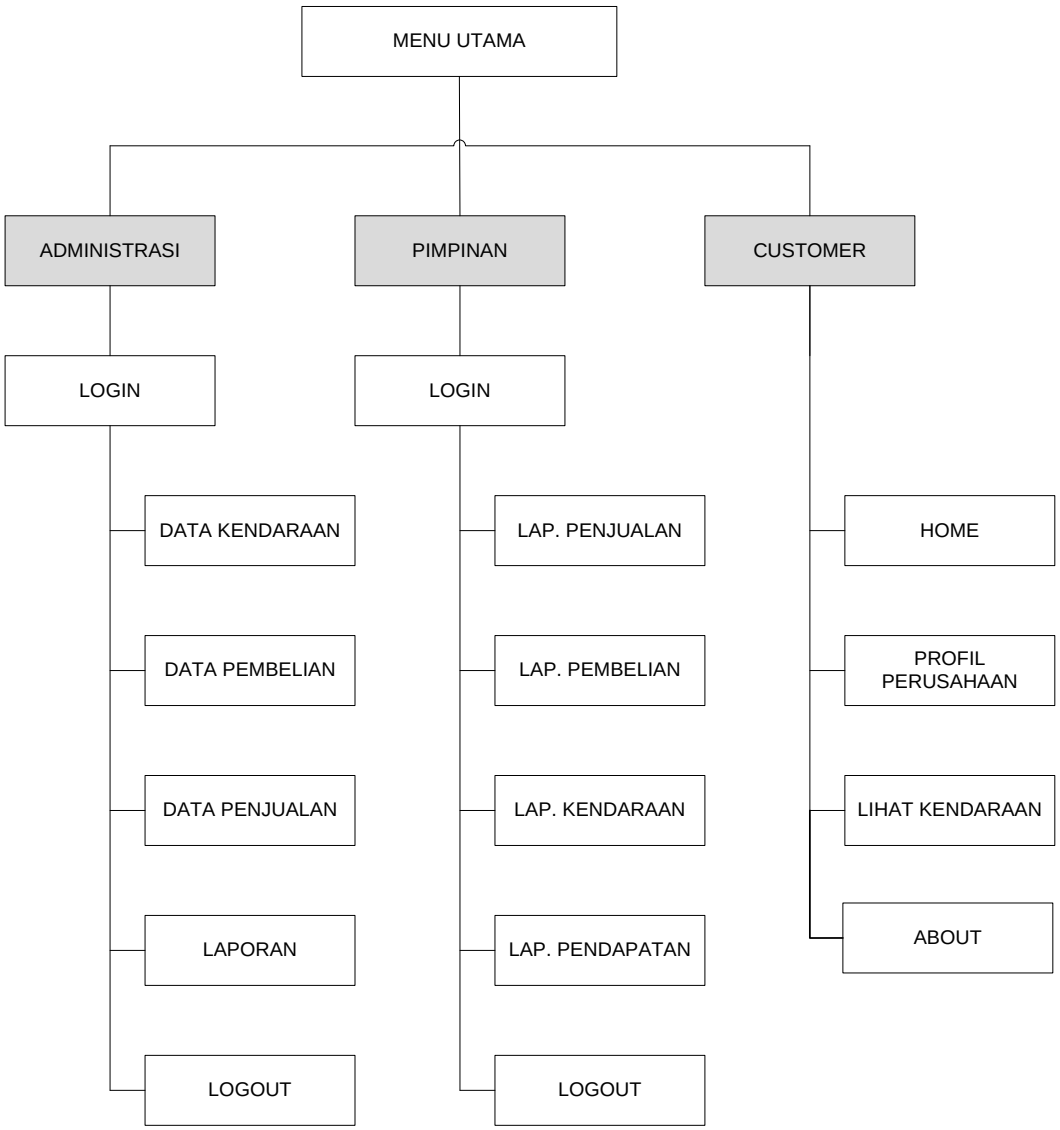
FBL-01020001 kode disamping menunjukkan FBL adalah faktur pembelian, 01 adalah jenis kendraan minibus, 02 adalah merk kendaraan toyota dengan nomor urut 0001.

4.2 Perancangan Antar Muka

Program dapat didefinisikan sebagai kumpulan dari intruksi-intruksi atau perintah-perintah terperinci yang sudah disiapkan oleh komputer sehingga dapat melakukan fungsi sesuai dengan yang telah ditentukan. Tujuan dari pembuatan program ini adalah untuk mempermudah dan mempercepat aktivitas yang berhubungan dengan pengolahan data dan untuk membentuk suatu sistem yang lebih baik.

4.2.1 Struktur Menu

Dalam perancangan program ini menggunakan menu yang mengintegrasikan semua bagian dalam program adapun gambaran menu, seperti yang ditampilkan pada gambar struktur menu berikut ini :

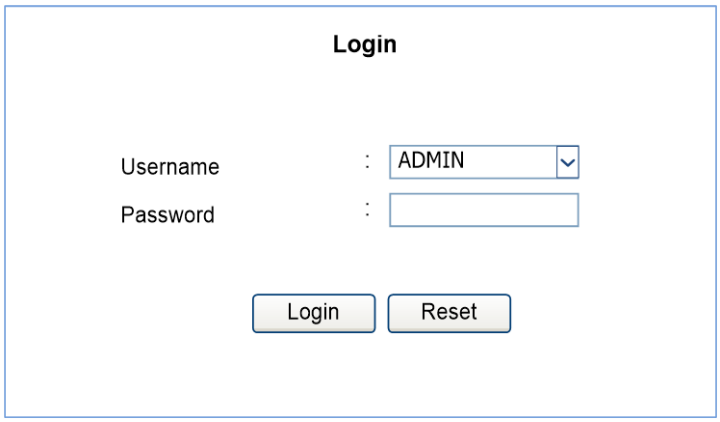


Gambar 4.10. Struktur Menu

4.2.2 Perancangan *Input*

Perancangan *input* dalam proses aplikasi sistem informasi penjualan pada showroom Bersama Motor adalah sebagai berikut :

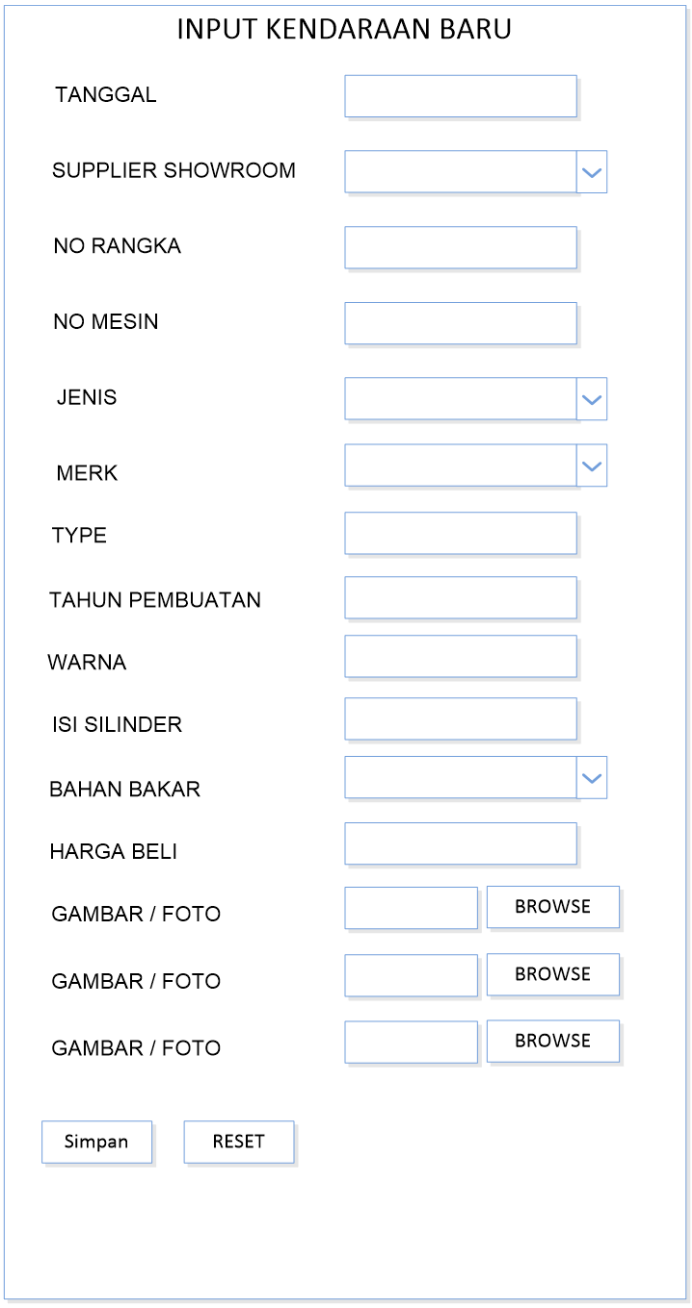
1. *Login*



Gambar 4.11 *Login*

Form *login* ini terdapat pada halaman utama aplikasi dan digunakan oleh *user* dengan hak akses sebagai administrasi, dan pimpinan untuk mengakses halaman sesuai dengan hak aksesnya. *User* harus menginputkan *Username* dan *password*. Apabila *Username* dan *pasword* benar maka akan masuk ke halaman sesuai dengan hak aksesnya, apabila salah maka sistem akan menampilkan peringatan periksa *username* dan *password*.

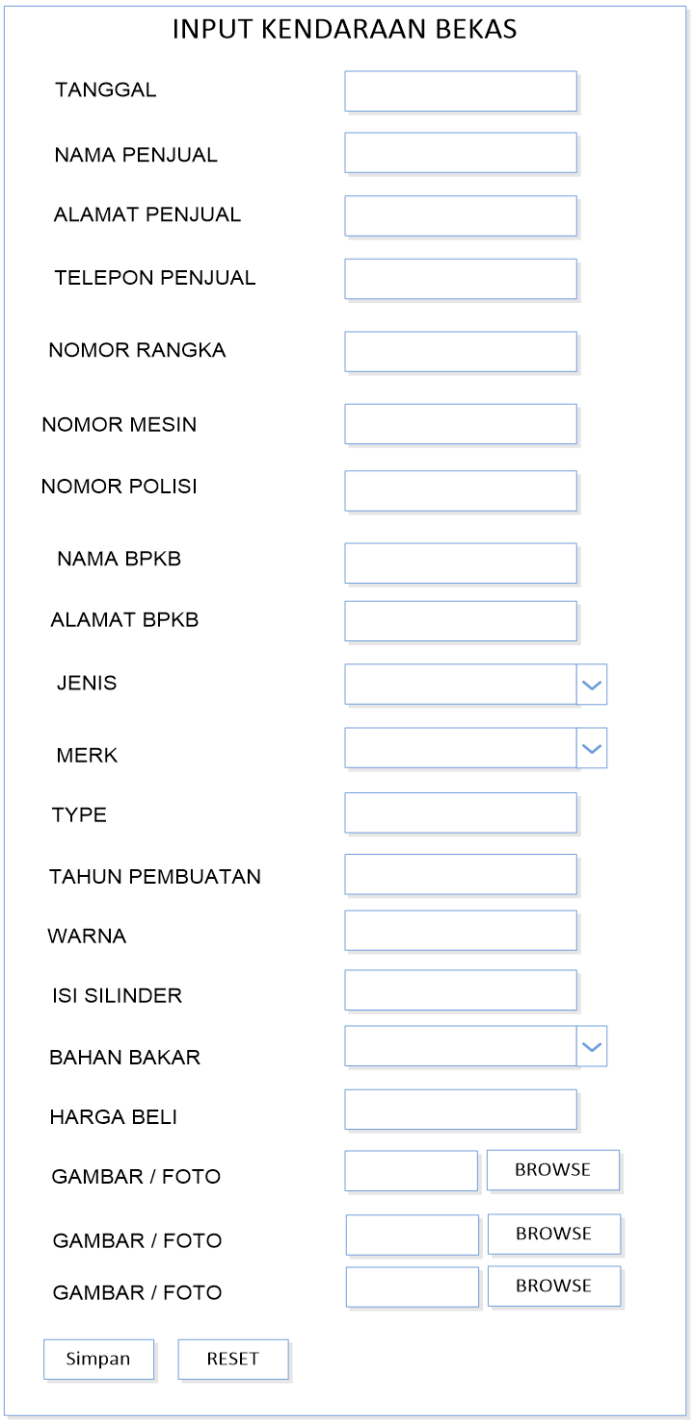
2. *Input* Pembelian Kendaraan Baru



Gambar 4.12 *Input* Pembelian Kendaraan Baru

Form *input* pembelian kendaraan baru adalah untuk menginputkan data pembelian kendaraan baru kedalam sistem.

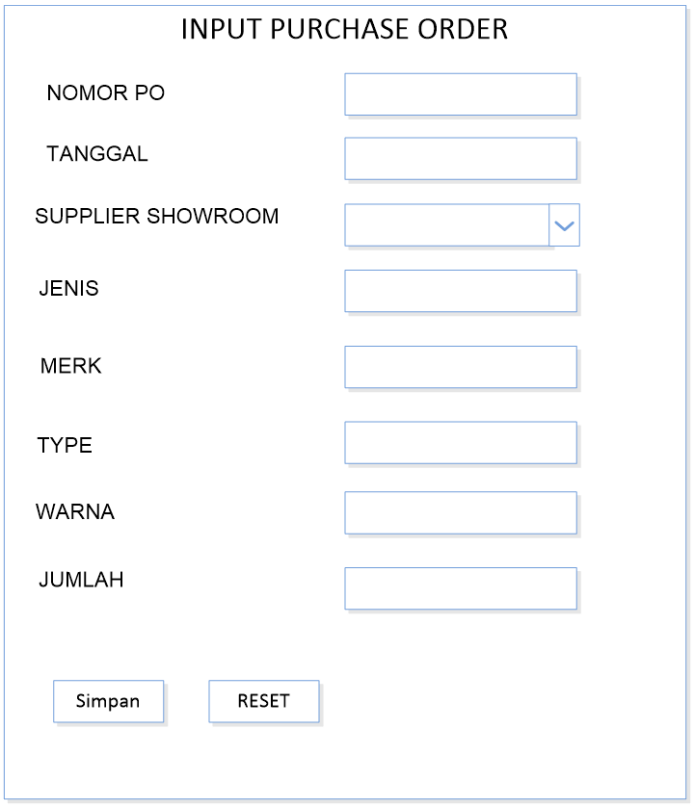
3. *Input* Pembelian Kendaraan Bekas



Gambar 4.13 *Input* Pembelian Kendaraan Bekas

Form *input* pembelian kendaraan bekas adalah untuk menginputkan data pembelian kendaraan bekas kedalam sistem.

4. *Input Purchase Order*



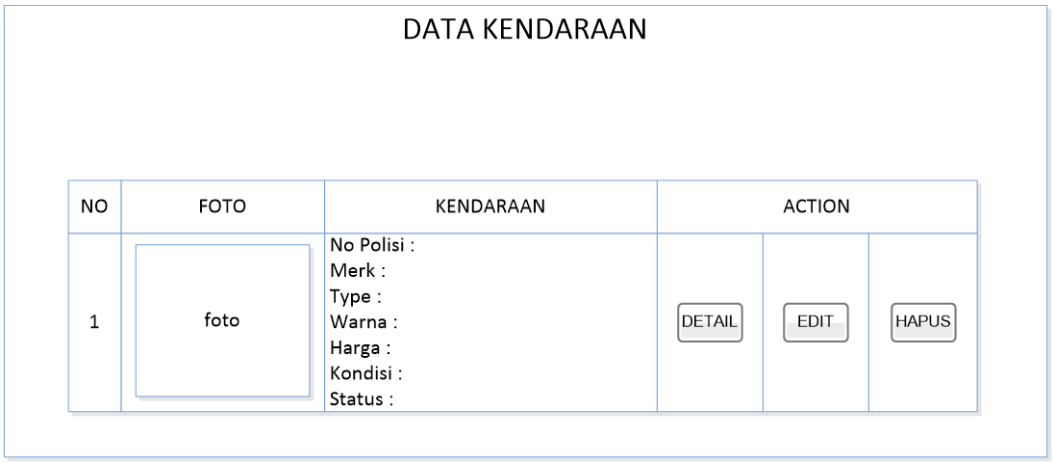
Gambar 4.14 Input *Purchase Order*

Form *input Purchase Order* adalah untuk menginputkan *Purchase Order* ke dalam sistem.

4.2.3 Perancangan *Output*

Perancangan *Output* atau keluaran merupakan hasil dari pengolahan data yang diterima dari proses masukan data berupa informasi yang berguna bagi *user.* Adapun rancangan *output* dari perancangan sistem informasi pengelolaan transaksi jual beli di bersama motor adalah sebagai berikut :

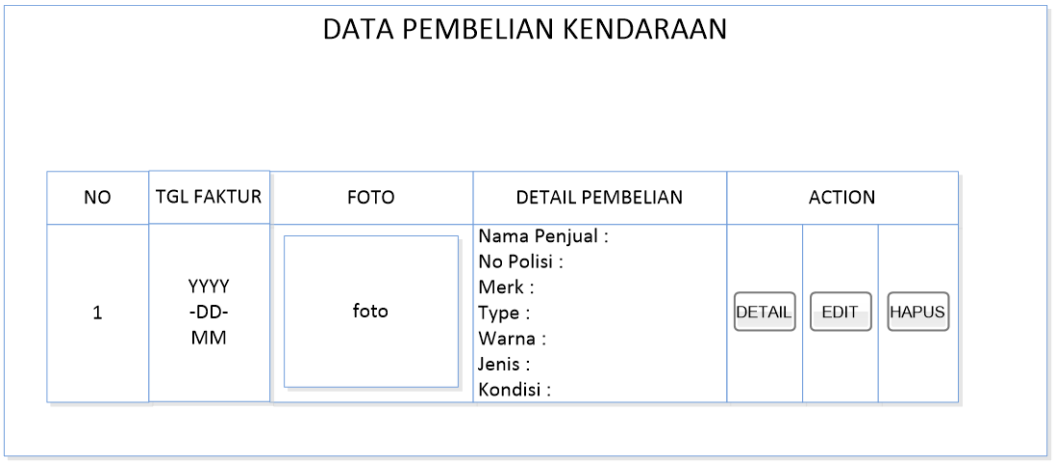
1. *Output* Kendaraan



Gambar 4.15 *Output* Data Kendaraan

Output data kendaraan adalah tampilan untuk menampilkan data kendaraan yang ada di bersama motor.

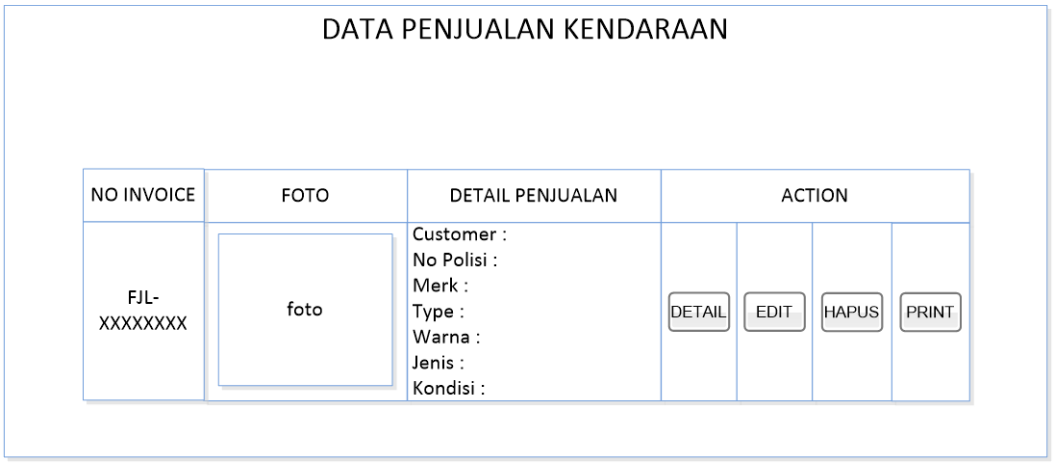
2. *Output* Pembelian Kendaraan



Gambar 4.16 *Output* Data Pembelian Kendaraan

Output data pembelian kendaraan adalah tampilan untuk menampilkan data pembelian kendaraan di Bersama Motor.

3. *Output* Penjualan Kendaraan



Gambar 4.17 *Output* Data Penjualan Kendaraan

Output data penjualan kendaraan adalah tampilan untuk menampilkan data penjualan kendaraan di Bersama Motor.

4.3 Perancangan Arsitektur Jaringan

Adapun Kebutuhan Jaringan untuk mendukung program ini adalah sebagai berikut ;

a. Tipe jaringan yang digunakan adalah LAN (*Local Area Network*), karena area yang relatif kecil, dengan model konfigurasi dimana satu komputer bertindak sebagai *server*, dan yang lainnya sebagai *client* yang mengakses file dalam *server*.

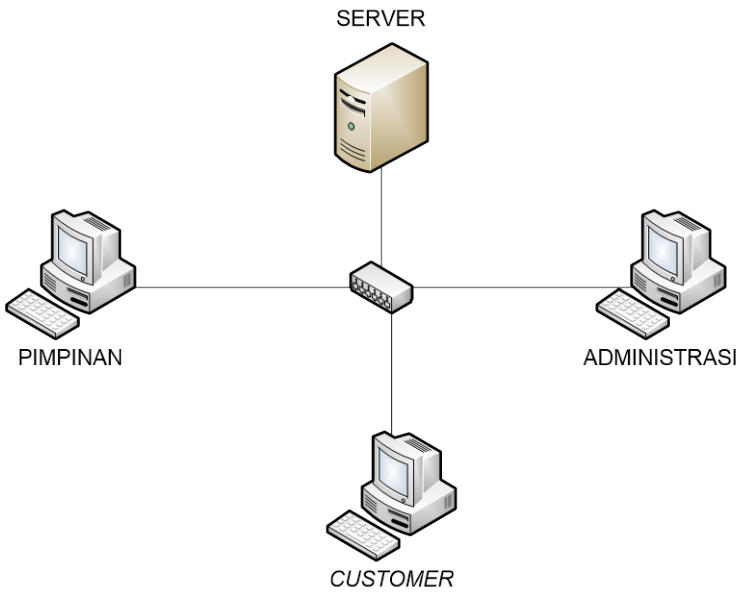
b. Media transmisi menggunakan kabel twisted pair, yang tipe *Unshielded twisted pair* (UTP). Dan dengan konektor RJ 45.

c. Topologi yang digunakan adalah Topologi Star.

   Topologi model ini dirancang, yang mana setiap *nodes* (*file server*, *workstation*, dan perangkat lainnya) terkoneksi ke jaringan melewati

sebuah *concentrator*. Data yang dikirim ke jaringan local akan melewati *concentrator* sebelum melanjutkan ke tempat tujuannya. *Concentrator* akan mengatur dan mengendalikan keseluruh fungsi jaringan, dan juga bertindak sebagai *repeater* (penguat alitan data).

Topologi ini kinerjanya sangat optimal, karena lintas kabel dari terminal ke server yang pendek.



Gambar 4.18 Perancangan Arsitektur Jaringan

d. Unsur yang terkait lainnya yaitu,

   a. File *server*, yang digunakan sebagai pusat dari jaringan,

   b. *WorkStations*, Komputer yang terhubung ke file server dalam sebuah jaringan,

   c. *Network Interface Cards* (Kartu Jaringan), Perangkat keras yang berfungsi sebagai media penghubung antar komputer,

d. *Enthernet Card* (Karu Jaringan Enthernet), Peerangkat keras yang menyediakan port untuk media transmisi (Kabel),

e. *Repeaters*, berfungsi sebagai penguat sinyal pada media transmisi yang digunakan.

f. *Bridges*, Perangkat keras yang mengatur jalur arus data / informasi agar tetap berjalan dengan baik dan teratur.

4.4 Implementasi

Implementasi adalah suatu proses penerapan rancangan program yang telah dibuat ke dalam sebuah aplikasi pemrograman sesuai dengan tujuan yang diharapkan dari program aplikasi tersebut. Kegiatan implementasi dilakukan dengan dasar yang telah direncanakan dalam tahap perancangan.

4.4.1 Batasan Implementasi (optional)

Batasan implementasi ini ditinjau agar pengguna dapat mengakses perangkat lunak dengan baik, batasannya adalah sebagai berikut :

1. Sistem hanya dapat diakses dengan menggunakan intranet atau *localhost*.
2. *Database* yang digunakan dalam mengimplementasikan sistem ini adalah MySql.
3. Tampilan pada program aplikasi ini menggunakan bahasa Indonesia.

4.4.2 Implementasi Perangkat Lunak

Dalam Implementasi Sistem Informasi Pengelolaan Transaksi Jual Beli Kendaraan, harus didukung oleh perangkat lunak yang digunakan agar sistem ini berjalan sebagaimana mestinya. Berikut perangkat lunak yang dibutuhkan :

a. Spesifikasi Untuk *Client* dan Admin Jaringan Lokal (lokal server saat perancangan dan pembangunan)

1. Sistem Operasi. Contoh : Window Xp, Vista, Windows 7, Unix dan lain lain.
2. *Web Browser* adalah *browser* internet sederhana yang dirancang untuk menampilkan halaman PHP, HTML, AJAX, dan lain lain dengan benar. Contoh : Opera, Mozilla Firefox, Internet Explorer dan lain lain
3. Lokal *server* Untuk menjadikan komputer kita agar bisa dijadikan sebagai localhost, kita membutuhkan sebuah program yang harus kita install.
   Contoh : xampp, AppServ dan lain lain

4.4.3 Implementasi Perangkat Keras

Perangkat lunak saja belum cukup untuk mengimplementasikan Sistem Informasi Ini. Agar Sistem Informasi Pengelolaan Transaksi Jual Beli Kendaraan ini dapat digunakan dibutuhkan perangkat keras (*Hardware*) komputer yang berfungsi menjalankan intruksi intruksi dan menampilkan secara visual informasi-informasi yang berguna bagi pengguna yang membutuhkan. Adapun perangkat keras yang digunakan untuk mendukung pembuatan maupun menjalankan aplikasi ini sebagai berikut :

1. Processor dengan *clock speed* 2.0 GHZ.
2. Harddisk 80 GB.
3. RAM 1 GB.
4. Perangkat Jaringan (Kabel UTP, Hub, dan lain-lain) .
5. VGA 256 MB.
6. Mouse, Keyboard dan Monitor LCD.
7. Printer.

4.4.4 Implementasi Basis Data (Sintaks SQL)

Sebuah basis data atau *database*, sangat dibutuhkan dalam sebuah sistem informasi yang terkomputerisasi. Pada sistem pendukung Sistem Informasi Pengelolaan Transaksi Jual Beli Kendaraan di Bersama Motor ini dibangun database demi terciptanya sistem informasi yang terintegrasi. Berikut adalah basis data menggunakan bahasa SQL Sistem Informasi Pengelolaan Transaksi Jual Beli Kendaraan di Bersama Motor Cirebon :

```
-- phpMyAdmin SQL Dump

-- version 2.11.4

-- http://www.phpmyadmin.net

--

-- Host: localhost

-- Generation Time: Jun 24, 2014 at 01:42 AM

-- Server version: 5.0.51
```

```
-- PHP Version: 5.2.5

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

--
-- Database: `bersamamotor`
--

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

--
-- Stand-in structure for view `detail_pembelian`
--
CREATE TABLE IF NOT EXISTS `detail_pembelian` (
`no_faktur` varchar(15)
,`tgl_faktur` date
,`nama_penjual` varchar(25)
```

```
,`alamat` text
,`telp` varchar(15)
,`no_rangka` varchar(25)
,`nomor_mesin` varchar(15)
,`jenis` varchar(15)
,`merk` varchar(15)
,`type` varchar(15)
,`warna` varchar(15)
,`thn_pembuatan` varchar(4)
,`model` varchar(15)
,`isi_silinder` int(10)
,`no_polisi` varchar(15)
,`foto` varchar(500)
,`foto_dua` varchar(500)
,`foto_tiga` varchar(100)
,`kondisi` text
,`harga_jual` double
```

```
,`harga_beli` double
,`nama_pemilik` varchar(30)
,`bahan_bakar` varchar(10)
,`no_bpkb` varchar(100)
);
-- --------------------------------------------------------

--
-- Stand-in structure for view `detail_penjualan`
--
CREATE TABLE IF NOT EXISTS `detail_penjualan` (
`no_invoice` varchar(15)
,`tgl_invoice` date
,`nama_cust` varchar(30)
,`alamat` text
,`telp` varchar(15)
,`no_rangka` varchar(25)
```

,`nomor_mesin` varchar(15)

,`jenis` varchar(15)

,`merk` varchar(15)

,`type` varchar(15)

,`warna` varchar(15)

,`thn_pembuatan` varchar(4)

,`model` varchar(15)

,`isi_silinder` int(10)

,`no_polisi` varchar(15)

,`foto` varchar(500)

,`foto_dua` varchar(500)

,`foto_tiga` varchar(100)

,`kondisi` text

,`harga_beli` double

,`no_bpkb` varchar(100)

,`harga_jual` double

,`dp` double

```
,`sisa_pembayaran` double
,`mediator` varchar(30)
,`nama_pemilik` varchar(30)
,`bahan_bakar` varchar(10)
,`pembayaran` varchar(10)
,`bank_pembiayaan` varchar(20)
);
-- --------------------------------------------------------

--
-- Table structure for table `kendaraan`
--

CREATE TABLE IF NOT EXISTS `kendaraan` (
  `nomor_rangka` varchar(25) collate latin1_general_ci NOT NULL,
  `nomor_mesin` varchar(15) collate latin1_general_ci NOT NULL,
  `merk` varchar(15) collate latin1_general_ci NOT NULL,
```

```
`type` varchar(15) collate latin1_general_ci NOT NULL,
`jenis` varchar(15) collate latin1_general_ci NOT NULL,
`model` varchar(15) collate latin1_general_ci NOT NULL,
`warna` varchar(15) collate latin1_general_ci NOT NULL,
`isi_silinder` int(10) NOT NULL,
`thn_pembuatan` varchar(4) collate latin1_general_ci NOT NULL,
`foto` varchar(500) collate latin1_general_ci NOT NULL,
`foto_dua` varchar(500) collate latin1_general_ci NOT NULL,
`foto_tiga` varchar(100) collate latin1_general_ci NOT NULL,
`kondisi` text collate latin1_general_ci NOT NULL,
`harga_jual` double NOT NULL,
`harga_beli` double NOT NULL,
`status` varchar(15) collate latin1_general_ci NOT NULL,
`no_polisi` varchar(15) collate latin1_general_ci NOT NULL,
`bahan_bakar` varchar(10) collate latin1_general_ci NOT NULL,
`nama_pemilik` varchar(30) collate latin1_general_ci NOT NULL,
`alamat_pemilik` varchar(50) collate latin1_general_ci NOT NULL,
```

```
  `harga_penawaran` double NOT NULL,
  `no_bpkb` varchar(100) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`nomor_rangka`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;

--
-- Dumping data for table `kendaraan`
--

INSERT INTO `kendaraan` (`nomor_rangka`, `nomor_mesin`, `merk`, `type`,
`jenis`, `model`, `warna`, `isi_silinder`, `thn_pembuatan`, `foto`, `foto_dua`,
`foto_tiga`, `kondisi`, `harga_jual`, `harga_beli`, `status`, `no_polisi`,
`bahan_bakar`, `nama_pemilik`, `alamat_pemilik`, `harga_penawaran`,
`no_bpkb`) VALUES
('325345435', '134531453', 'Honda', 'Jazz', 'Mini Bus', '', 'Merah', 2500, '2014',
'index.jpg', '', '', 'Baru', 0, 200000000, 'Tersedia', 'Baru', 'Bensin', '', '', 23, ''),
```

('283741834', '482374837', 'Honda', 'Genio', 'Mini Bus', '', 'Silver', 200, '2000', 'images.jpg', '', '', 'Bekas', 85000000, 80000000, 'Terjual', 'D-2012-PI', 'Bensin', 'Rfiki', 'Cimahi', 0, ''),

('8343GGAG', '7878768955', 'Toyota', 'Honda City', 'Mini Bus', '', 'Biru', 1900, '2000', '1426296355.jpg', '1426296355-5.jpg', '', 'Baru', 0, 180000000, 'Tersedia', '-', 'Bensin', '', '', 0, '-'),

('8343GGAGFCC', '82E2D82H8', 'Honda', 'Honda City', 'Sedan', '', 'Biru', 1900, '2000', 'Detective_Conan_DVD_10.png', 'url.png', 'whatapp.png', 'Bekas', 0, 180000000, 'Tersedia', 'A 8989 DF', 'Pertamax', 'Dedi Homblo', '', 0, '1118978702FFF');

-- --

--

-- Table structure for table `kritiksaran`

--

CREATE TABLE IF NOT EXISTS `kritiksaran` (

  `id` int(11) NOT NULL auto_increment,

```
  `nama` varchar(30) collate latin1_general_ci NOT NULL,
  `subjek` varchar(30) collate latin1_general_ci NOT NULL,
  `pesan` text collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=4 ;

--
-- Dumping data for table `kritiksaran`
--

INSERT INTO `kritiksaran` (`id`, `nama`, `subjek`, `pesan`) VALUES
(1, 'rifki', 'saran', 'saran'),
(2, 'isma', 'saran', 'saran aja deh'),
(3, 'dgt', 'grt', 'grtg');

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

```
--

-- Table structure for table `laporan`

--


CREATE TABLE IF NOT EXISTS `laporan` (
  `no_rangka` varchar(25) collate latin1_general_ci default NULL,
  `no_invoice` varchar(15) collate latin1_general_ci default NULL,
  `tgl_invoice` date default NULL,
  `harga_jual` double default NULL,
  `no_faktur` varchar(15) collate latin1_general_ci default NULL,
  `tgl_faktur` date default NULL,
  `harga_beli` double default NULL,
  `pendapatan` double default NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;


--

-- Dumping data for table `laporan`
```

```
--



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

--

-- Table structure for table `login`

--

CREATE TABLE IF NOT EXISTS `login` (
  `nama` varchar(20) collate latin1_general_ci NOT NULL,
  `username` varchar(50) collate latin1_general_ci NOT NULL,
  `password` varchar(50) collate latin1_general_ci NOT NULL,
  `email` varchar(50) collate latin1_general_ci NOT NULL,
  `status` varchar(10) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`username`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;
```

```
--

-- Dumping data for table `login`

--


INSERT INTO `login` (`nama`, `username`, `password`, `email`, `status`)
VALUES
('Admin', 'admin', 'admin', 'admin@mail.com', 'admin'),
('Manajer', 'manajer', 'manajer', 'manajer@mail.com', 'manajer');


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


--

-- Table structure for table `pembelian`

--


CREATE TABLE IF NOT EXISTS `pembelian` (
  `no_faktur` varchar(15) collate latin1_general_ci NOT NULL,
```

```
`tgl_faktur` date NOT NULL,
`nama_penjual` varchar(25) collate latin1_general_ci NOT NULL,
`alamat` text collate latin1_general_ci NOT NULL,
`telp` varchar(15) collate latin1_general_ci NOT NULL,
`no_rangka` varchar(25) collate latin1_general_ci NOT NULL,
`no_mesin` varchar(30) collate latin1_general_ci NOT NULL,
`jenis` varchar(25) collate latin1_general_ci NOT NULL,
`merk` varchar(30) collate latin1_general_ci NOT NULL,
`type` varchar(25) collate latin1_general_ci NOT NULL,
`thn_pembuatan` varchar(4) collate latin1_general_ci NOT NULL,
`no_polisi` varchar(10) collate latin1_general_ci NOT NULL,
`warna` varchar(15) collate latin1_general_ci NOT NULL,
`harga_beli` double NOT NULL,
`kondisi` text collate latin1_general_ci NOT NULL,
`bahan_bakar` varchar(10) collate latin1_general_ci NOT NULL,
`nama_pemilik` varchar(30) collate latin1_general_ci NOT NULL,
`isi_silinder` int(10) NOT NULL,
```

```
  `harga_penawaran` double NOT NULL,
  PRIMARY KEY  (`no_faktur`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;

--
-- Dumping data for table `pembelian`
--

INSERT INTO `pembelian` (`no_faktur`, `tgl_faktur`, `nama_penjual`, `alamat`,
`telp`, `no_rangka`, `no_mesin`, `jenis`, `merk`, `type`, `thn_pembuatan`,
`no_polisi`, `warna`, `harga_beli`, `kondisi`, `bahan_bakar`, `nama_pemilik`,
`isi_silinder`, `harga_penawaran`) VALUES
('FBL-0401012', '2014-06-03', 'Rifki', 'Cimahi', '48237481', '283741834',
'482374837', 'Mini Bus', 'Honda', 'Genio', '2005', 'D-2012-PI', 'Silver', 80000000,
'Bekas', 'Bensin', 'Rifki Arsyad', 2000, 85000000),
('FBL-04010011', '2014-06-03', 'Honda', '', '', '325345435', '134531453', 'Mini
Bus', 'Honda', 'Jazz', '2013', 'Baru', 'Merah', 200000000, 'Baru', 'Bensin', '', 200, 0),
```

('FBL-0201013', '2014-06-23', 'Honda', '', '', '8343GGAG', '7878768955', 'Mini Bus', 'Toyota', 'Honda City', '2000', '-', 'Biru', 180000000, 'Baru', 'Bensin', '', 1900, 0),

('FBL-0402013', '2014-06-24', 'Dedi', 'Cilegon', '0812345678', '8343GGAGFCC', '82E2D82H8', 'Sedan', 'Honda', 'Honda City', '2000', 'A 8989 DF', 'Biru', 180000000, 'Bekas', 'Pertamax', 'Dedi Homblo', 1900, 0);

-- --

--

-- Table structure for table `penjualan`

--

CREATE TABLE IF NOT EXISTS `penjualan` (

`no_invoice` varchar(15) collate latin1_general_ci NOT NULL,

`tgl_invoice` date NOT NULL,

`nama_cust` varchar(30) collate latin1_general_ci NOT NULL,

```
  `alamat` text collate latin1_general_ci NOT NULL,
  `telp` varchar(15) collate latin1_general_ci NOT NULL,
  `no_rangka` varchar(25) collate latin1_general_ci NOT NULL,
  `harga_jual` double NOT NULL,
  `dp` double NOT NULL,
  `sisa_pembayaran` double NOT NULL,
  `mediator` varchar(30) collate latin1_general_ci NOT NULL,
  `pembayaran` varchar(10) collate latin1_general_ci NOT NULL,
  `bank_pembiayaan` varchar(20) collate latin1_general_ci NOT NULL,
  `keterangan` text collate latin1_general_ci NOT NULL,
  `durasi_kredit` int(10) NOT NULL,
  PRIMARY KEY  (`no_invoice`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;

--
-- Dumping data for table `penjualan`
--
```

```
INSERT INTO `penjualan` (`no_invoice`, `tgl_invoice`, `nama_cust`, `alamat`,
`telp`, `no_rangka`, `harga_jual`, `dp`, `sisa_pembayaran`, `mediator`,
`pembayaran`, `bank_pembiayaan`, `keterangan`, `durasi_kredit`) VALUES
('FJL-04010011', '2014-06-03', 'Isma', 'bandung', '82711831782', '283741834',
85000000, 0, 0, 'dede', 'Cash', '', '', 0);


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


--
-- Table structure for table `po`
--


CREATE TABLE IF NOT EXISTS `po` (
  `no_po` varchar(15) collate latin1_general_ci NOT NULL,
  `nama_supplier` varchar(25) collate latin1_general_ci NOT NULL,
  `tgl` date NOT NULL,
  `merk` varchar(15) collate latin1_general_ci NOT NULL,
```

```
  `type` varchar(15) collate latin1_general_ci NOT NULL,
  `jenis` varchar(15) collate latin1_general_ci NOT NULL,
  `warna` varchar(15) collate latin1_general_ci NOT NULL,
  `jml` int(10) NOT NULL,
  PRIMARY KEY  (`no_po`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;

--
-- Dumping data for table `po`
--

INSERT INTO `po` (`no_po`, `nama_supplier`, `tgl`, `merk`, `type`, `jenis`,
`warna`, `jml`) VALUES
('PO-0011', 'Honda', '2014-06-15', 'Honda', 'Jazz', 'Sedan', 'Merah', 1),
('PO-0012', 'Honda', '2014-06-16', 'Honda', 'JAZZ RS', 'Mini Bus', 'HITAM', 1),
('PO-0013', 'Toyota', '2014-06-16', 'Toyota', 'Avanza', 'Mini Bus', 'Putih', 1);
```

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

--

-- Table structure for table `supplier`

--

CREATE TABLE IF NOT EXISTS `supplier` (
  `nama_supplier` varchar(25) collate latin1_general_ci NOT NULL,
  `alamat_supplier` varchar(40) collate latin1_general_ci NOT NULL,
  `telp_supplier` varchar(15) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`nama_supplier`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;

--

-- Dumping data for table `supplier`

--
```

```
INSERT INTO `supplier` (`nama_supplier`, `alamat_supplier`, `telp_supplier`)
VALUES
('Honda', 'Cirebon', '0924273923'),
('Toyota', 'Cirebon', '0237282828'),
('BMW', 'Cirebon', '213487123412'),
('Mercedes', 'Cirebon', '34218648123'),
('Daihatzu', 'Cirebon', '24151435'),
('Hyundai', 'Cirebon', '4351353151'),
('Mitsubishi', 'Cirebon', '21421275');


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


--
-- Structure for view `detail_pembelian`
--
DROP TABLE IF EXISTS `detail_pembelian`;
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL
SECURITY DEFINER VIEW `bersamamotor`.`detail_pembelian` AS select
`bersamamotor`.`pembelian`.`no_faktur` AS
`no_faktur`,`bersamamotor`.`pembelian`.`tgl_faktur` AS
`tgl_faktur`,`bersamamotor`.`pembelian`.`nama_penjual` AS
`nama_penjual`,`bersamamotor`.`pembelian`.`alamat` AS
`alamat`,`bersamamotor`.`pembelian`.`telp` AS
`telp`,`bersamamotor`.`pembelian`.`no_rangka` AS
`no_rangka`,`bersamamotor`.`kendaraan`.`nomor_mesin` AS
`nomor_mesin`,`bersamamotor`.`kendaraan`.`jenis` AS
`jenis`,`bersamamotor`.`kendaraan`.`merk` AS
`merk`,`bersamamotor`.`kendaraan`.`type` AS
`type`,`bersamamotor`.`kendaraan`.`warna` AS
`warna`,`bersamamotor`.`kendaraan`.`thn_pembuatan` AS
`thn_pembuatan`,`bersamamotor`.`kendaraan`.`model` AS
`model`,`bersamamotor`.`kendaraan`.`isi_silinder` AS
`isi_silinder`,`bersamamotor`.`kendaraan`.`no_polisi` AS
`no_polisi`,`bersamamotor`.`kendaraan`.`foto` AS
`foto`,`bersamamotor`.`kendaraan`.`foto_dua` AS
`foto_dua`,`bersamamotor`.`kendaraan`.`foto_tiga` AS
`foto_tiga`,`bersamamotor`.`kendaraan`.`kondisi` AS
`kondisi`,`bersamamotor`.`kendaraan`.`harga_jual` AS
`harga_jual`,`bersamamotor`.`pembelian`.`harga_beli` AS
```

```
`harga_beli`,`bersamamotor`.`pembelian`.`nama_pemilik` AS
`nama_pemilik`,`bersamamotor`.`kendaraan`.`bahan_bakar` AS
`bahan_bakar`,`bersamamotor`.`kendaraan`.`no_bpkb` AS `no_bpkb` from
(`bersamamotor`.`pembelian` join `bersamamotor`.`kendaraan`) where
(`bersamamotor`.`pembelian`.`no_rangka` =
`bersamamotor`.`kendaraan`.`nomor_rangka`);


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


--

-- Structure for view `detail_penjualan`

--

DROP TABLE IF EXISTS `detail_penjualan`;

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL
SECURITY DEFINER VIEW `bersamamotor`.`detail_penjualan` AS select
`bersamamotor`.`penjualan`.`no_invoice` AS
`no_invoice`,`bersamamotor`.`penjualan`.`tgl_invoice` AS
`tgl_invoice`,`bersamamotor`.`penjualan`.`nama_cust` AS
`nama_cust`,`bersamamotor`.`penjualan`.`alamat` AS
`alamat`,`bersamamotor`.`penjualan`.`telp` AS
```

```
`telp`,`bersamamotor`.`penjualan`.`no_rangka` AS
`no_rangka`,`bersamamotor`.`kendaraan`.`nomor_mesin` AS
`nomor_mesin`,`bersamamotor`.`kendaraan`.`jenis` AS
`jenis`,`bersamamotor`.`kendaraan`.`merk` AS
`merk`,`bersamamotor`.`kendaraan`.`type` AS
`type`,`bersamamotor`.`kendaraan`.`warna` AS
`warna`,`bersamamotor`.`kendaraan`.`thn_pembuatan` AS
`thn_pembuatan`,`bersamamotor`.`kendaraan`.`model` AS
`model`,`bersamamotor`.`kendaraan`.`isi_silinder` AS
`isi_silinder`,`bersamamotor`.`kendaraan`.`no_polisi` AS
`no_polisi`,`bersamamotor`.`kendaraan`.`foto` AS
`foto`,`bersamamotor`.`kendaraan`.`foto_dua` AS
`foto_dua`,`bersamamotor`.`kendaraan`.`foto_tiga` AS
`foto_tiga`,`bersamamotor`.`kendaraan`.`kondisi` AS
`kondisi`,`bersamamotor`.`kendaraan`.`harga_beli` AS
`harga_beli`,`bersamamotor`.`kendaraan`.`no_bpkb` AS
`no_bpkb`,`bersamamotor`.`penjualan`.`harga_jual` AS
`harga_jual`,`bersamamotor`.`penjualan`.`dp` AS
`dp`,`bersamamotor`.`penjualan`.`sisa_pembayaran` AS
`sisa_pembayaran`,`bersamamotor`.`penjualan`.`mediator` AS
`mediator`,`bersamamotor`.`kendaraan`.`nama_pemilik` AS
`nama_pemilik`,`bersamamotor`.`kendaraan`.`bahan_bakar` AS
`bahan_bakar`,`bersamamotor`.`penjualan`.`pembayaran` AS
```

```
`pembayaran`,`bersamamotor`.`penjualan`.`bank_pembiayaan` AS
`bank_pembiayaan` from (`bersamamotor`.`penjualan` join
`bersamamotor`.`kendaraan`) where (`bersamamotor`.`penjualan`.`no_rangka` =
`bersamamotor`.`kendaraan`.`nomor_rangka`);
```

4.4.5 Implementasi Antar Muka

Antar muka sebuah perangkat lunak adalah hal yang sangat penting, karena program yang baik adalah program yang dapat menciptakan media yang baik pula antara *user* dan sistem. Oleh karena itu berikut merupakan gambaran implementasi antar muka pada Sistem Informasi pengelolaan transaksi jual beli yang digambarkan sesuai dengan klasifikasi hak akses dari setiap *user*.

4.4.5.1 Implementasi Antar Muka Halaman Utama

Tabel 4.10 Implementasi Antar Muka Halaman Utama

Menu	Deskripsi	Menu File
Home	File untuk halaman utama home	Index.php
Profil Perusahaan	File untuk melihat halaman profil perusahaan	Profil.php
Lihat Kendaraan	File untuk melihat halaman data kendaraan	Kendaraankonsumen.php
About	File untuk melihat halaman *contack person* perusahaan	About.php

4.4.5.2 Implementasi Antar Muka Halaman Pimpinan

Tabel 4.11 Implementasi Antar Muka Halaman Pimpinan

Menu	Deskripsi	Menu File
Home Pimpinan	File halaman utama Pimpinan	Homemanajer.php
Laporan Penjualan	Sub menu untuk melihat laporan penjualan kendaraan	Lap_penjualan.php
Laporan Pembelian	Sub menu untuk melihat laporan pembelian kendaraan	Lap_pembelian.php
Laporan Pendapatan	Sub menu untuk melihat laporan pendapatan	Lap_pendapatan.php
Lihat Kendaraan	Sub menu untuk melihat laporan kendaraan	Lap_kendaraan.php
Logout	File untuk keluar dari sistem	Logout.php

4.4.5.3 Implementasi Antar Muka Halaman Administrasi

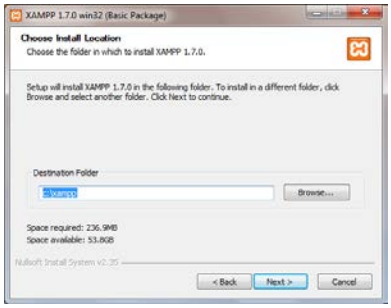
Tabel 4.12 Implementasi Antar Muka Halaman Administrasi

Menu	Deskripsi	Menu File
Home Administrasi	File halaman utama Admin	Homeadmin.php
Data Kendaraan	Sub menu untuk melihat data kendaraan	Kendaraan.php
Data Pembelian	Sub menu untuk menginput pembelian kendaraan baru & kendaraan bekas	Pembelian.php
Data Penjualan	Sub menu untuk menginput data penjualan kendaraan	Penjualan.php
Laporan	Sub menu untuk melihat laporan	Laporan.php
Logout	File untuk keluar dari sistem	Logout.php

4.4.6 Implementasi Instalasi Program

Untuk menjalankan aplikasi sistem informasi penjualan di Bersama Motor ini, cukup menginstal perangkat lunak yang akan digunakan, pada perancangan ini menggunakan perangkat lunak xampp-win32-1.5.7.exe. adapun hal yang harus dilakukan adalah sebagai barikut :

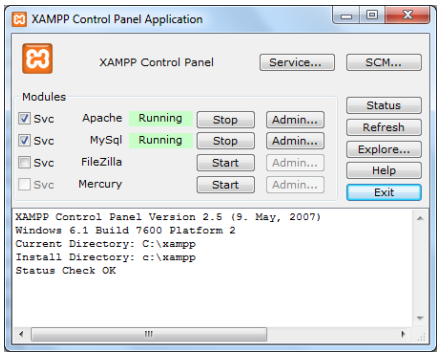
1. Tahap Instal xampp



Gambar 4.19 Halaman Instalasi XAMPP.

Klik dua kali pada aplikasi maka akan tampil gambar seperti gambar di atas. Kemudian pilih letak file installer xampp pada sistem. Jika tempat telah sesuai dengan keinginan maka langsung klik tombol *next*.

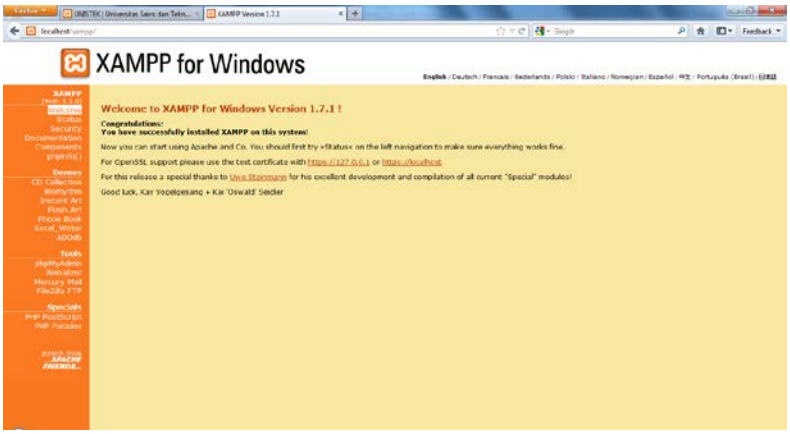
2. Jalankan paket xampp



Gambar 4.20 Halaman XAMPP Control Panel.

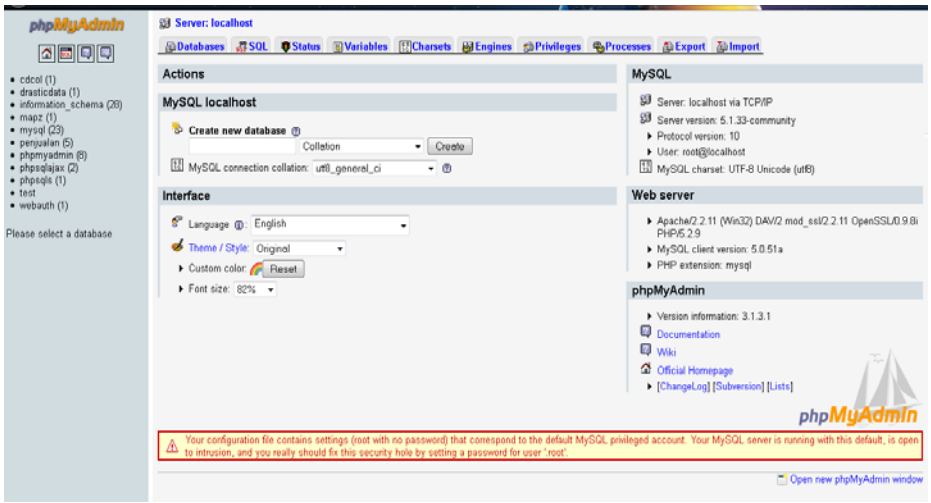
Setelah aplikasi selesai diinstal lalu klik dua kali shortcut xampp yang ada di dekstop maka muncul tampilan seperti diatas. klik di tombol *start* Apache dan start MySql.

3. Menghubungkan database dengan aplikasi



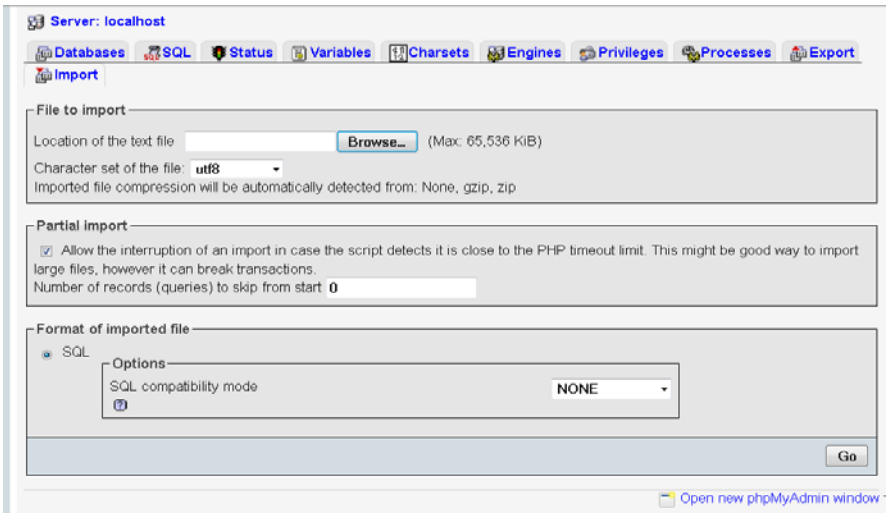
Gambar 4.21 Halaman Utama XAMPP.

Untuk menghubungkan database dan aplikasi, klik phpmyadmin maka akan muncul tampilan seperti gambar diatas, lalu klik import untuk mengambil database yang akan dihubungkan ke aplikasi.



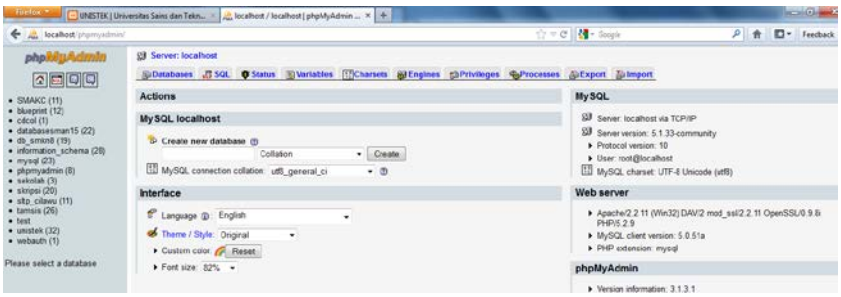
Gambar 4.22 Tampilan phpMyAdmin

Klik browse untuk mencari file database, setelah mendapatkan file database yang diinginkan klik GO



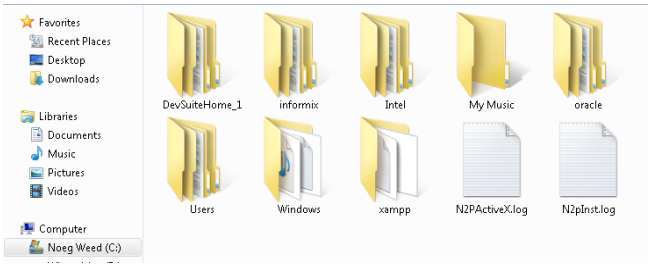
Gambar 4.23 Tampilan Import database

Setelah berhasil maka file database anda akan tersimpan di xampp, nama database akan tertera di tampilan sebelah kiri dari aplikasi xampp ini

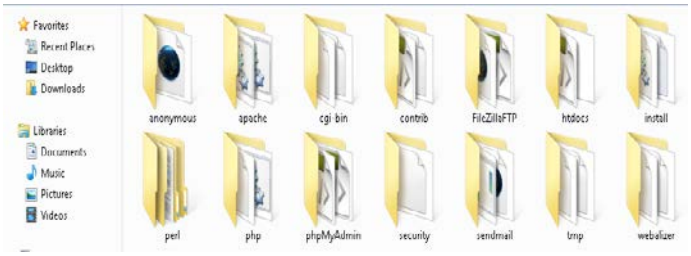


Gambar 4.24 Tampilan phpMyAdmin setelah import database

Simpan aplikasi Sistem Informasi pemesanan dan penjualan di Bersama Motor di folder htdocs yang terletak di folder xampp. Letak folder xampp tergantung dari dimana anda meng*install* aplikasi xampp tersebut.

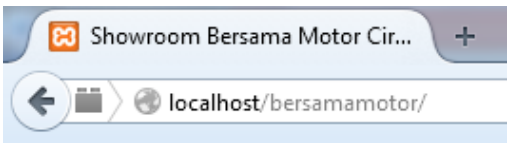


Gambar 4.25 Tampilan letak folder xampp



Gambar 4.26 Tampilan letak folder htdocs

Kemudian ketikkan localhost/namafolder atau localhost/namafile.php pada web browser anda kemudian tekan enter, setelah itu aplikasi dapat dijalankan.



Gambar 4.27 Alamat aplikasi pada web browser

4.4.7 Penggunaan Program

1. Halaman Utama

Halaman ini adalah halaman utama Bersama Motor. Dimana *customer* bisa melihat website Bersama Motor.



Gambar 4.28 Halaman Utama

2. Halaman Profil Perusahaan

Klik menu profil perusahaan, pada halaman ini menjelaskan tentang profil perusahaan beserta visi misi showroom Bersama Motor.



Gambar 4.29 Halaman Profil Perusahaan

3. Halaman Lihat Kendaraan

Klik menu lihat kendaraan, pada halaman ini *customer* bisa melihat katalog kendaraan di Bersama Motor.



Gambar 4.30 Halaman Lihat Kendaraan

4. Halaman *About*

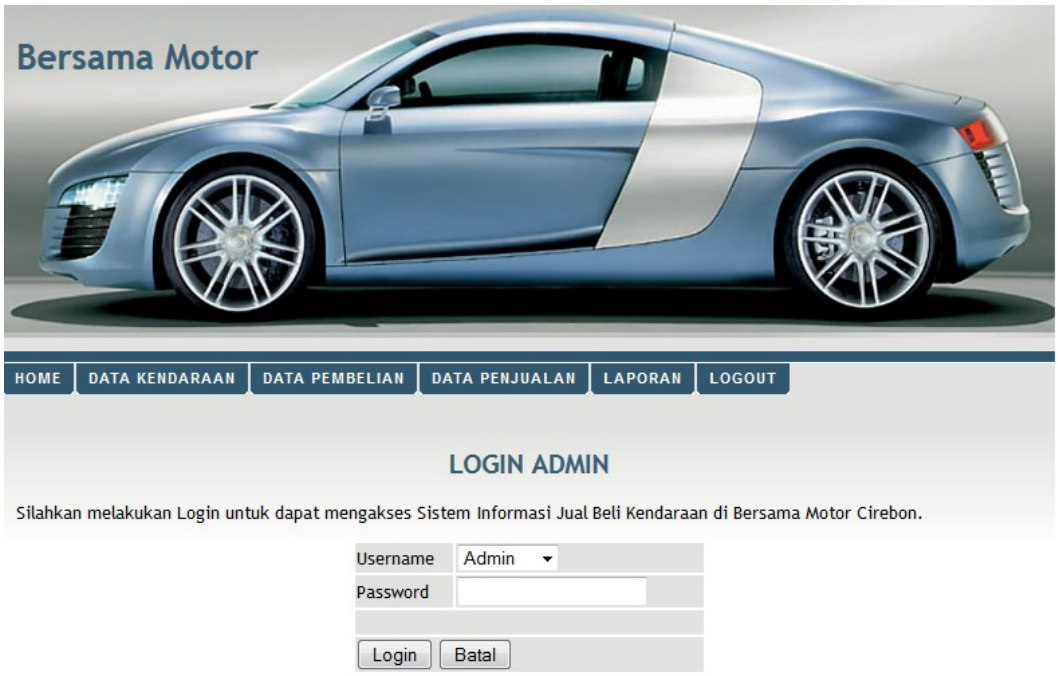
Klik menu about, pada halaman ini *customer* bisa melihat informasi kontak Bersama Motor.



Gambar 4.31 Halaman *About*

5. Halaman *Login*

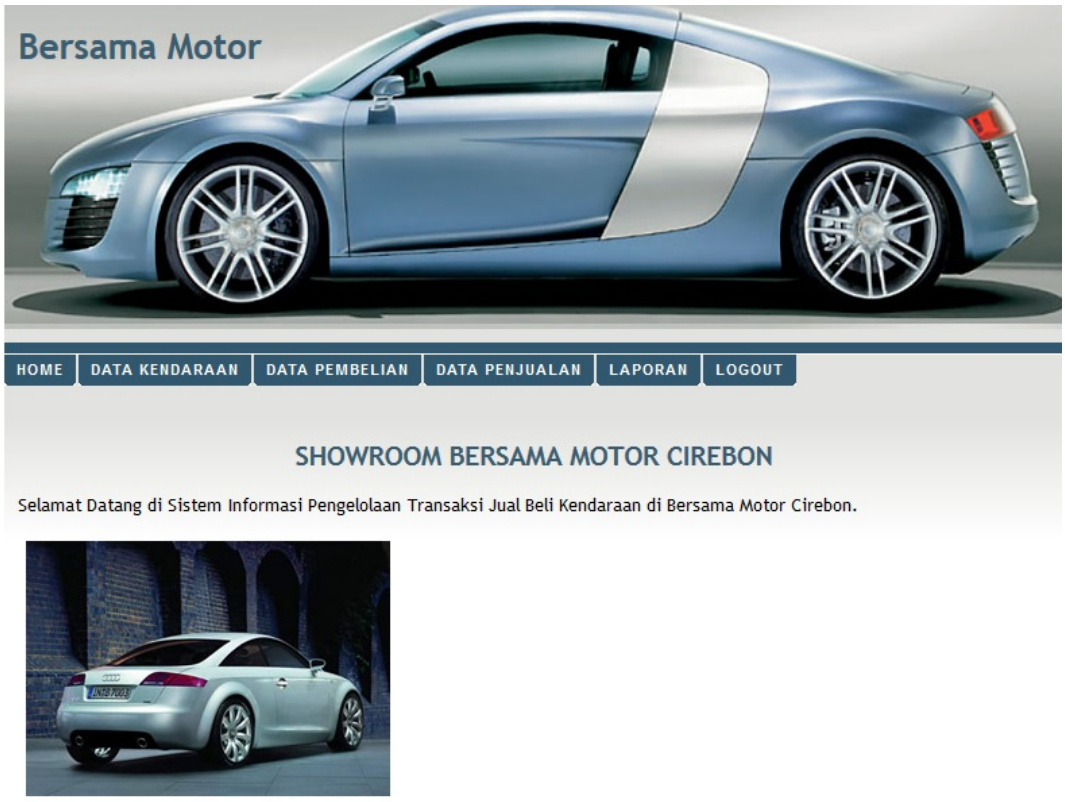
Pada halaman ini merupakan halaman tentang *user* dimana administrasi dan pimpinan harus melakukan pengisian data *username* dan *password* untuk bisa mengakses semua fitur yang ada pada sistem.



Gambar 4.32 Halaman *Login*

6. Halaman Utama Administrasi

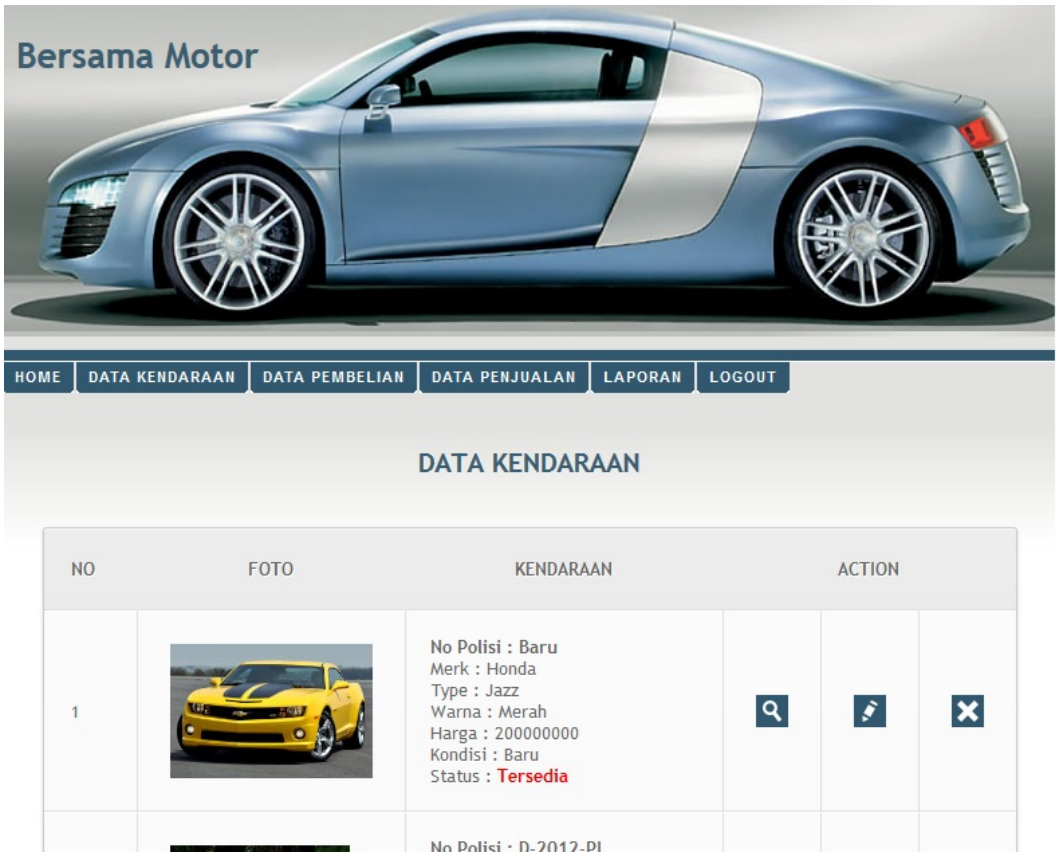
Halaman ini merupakan halaman utama ketika administrasi berhasil melakukan *login* dan membuka aplikasi sistem informasi pengelolaan transaksi jual beli kendaraan bermotor.



Gambar 4.33 Halaman Utama Administrasi

7. Halaman Kendaraan

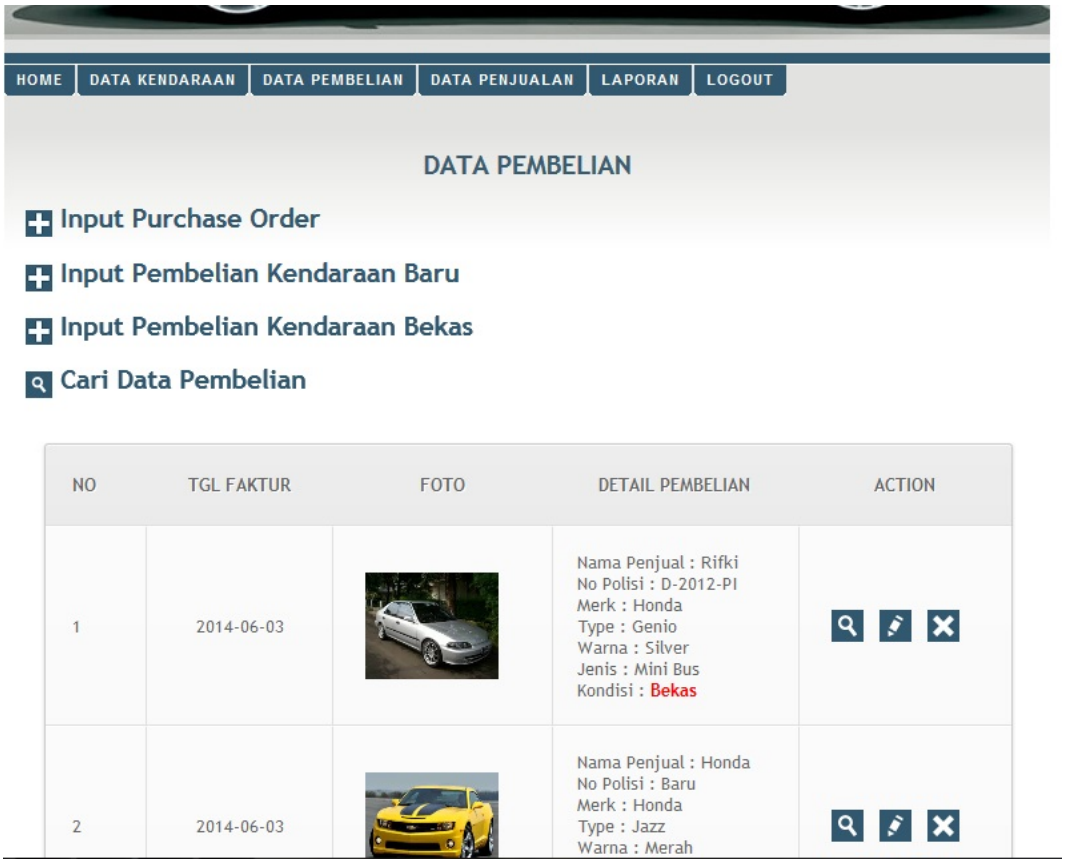
Halaman ini merupakan halaman data kendaraan pada administrasi, dimana semua data kendaraan terdapat dihalaman ini.



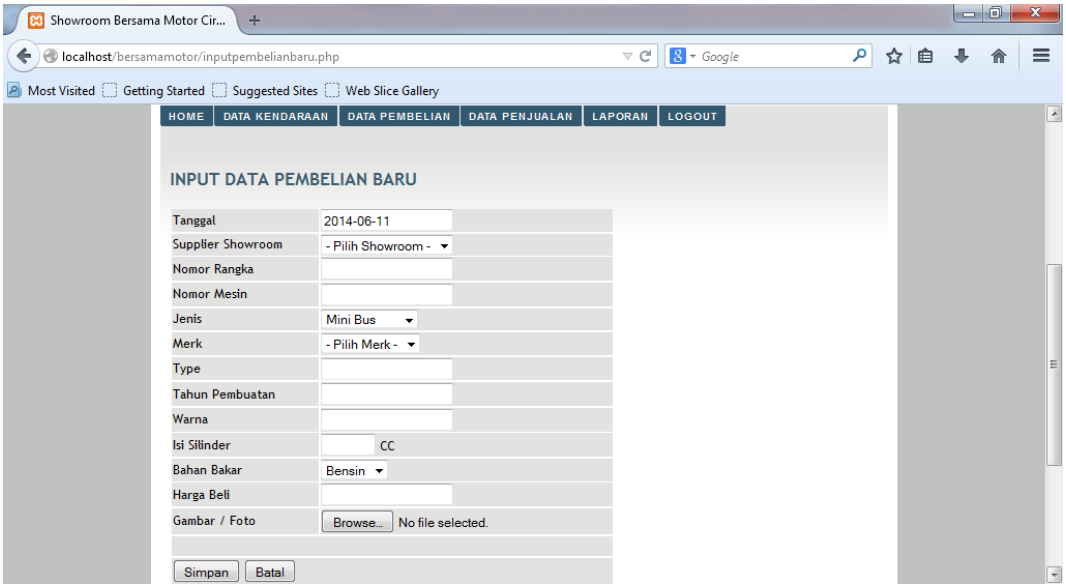
Gambar 4.34 Halaman Kendaraan

8. Halaman Pembelian

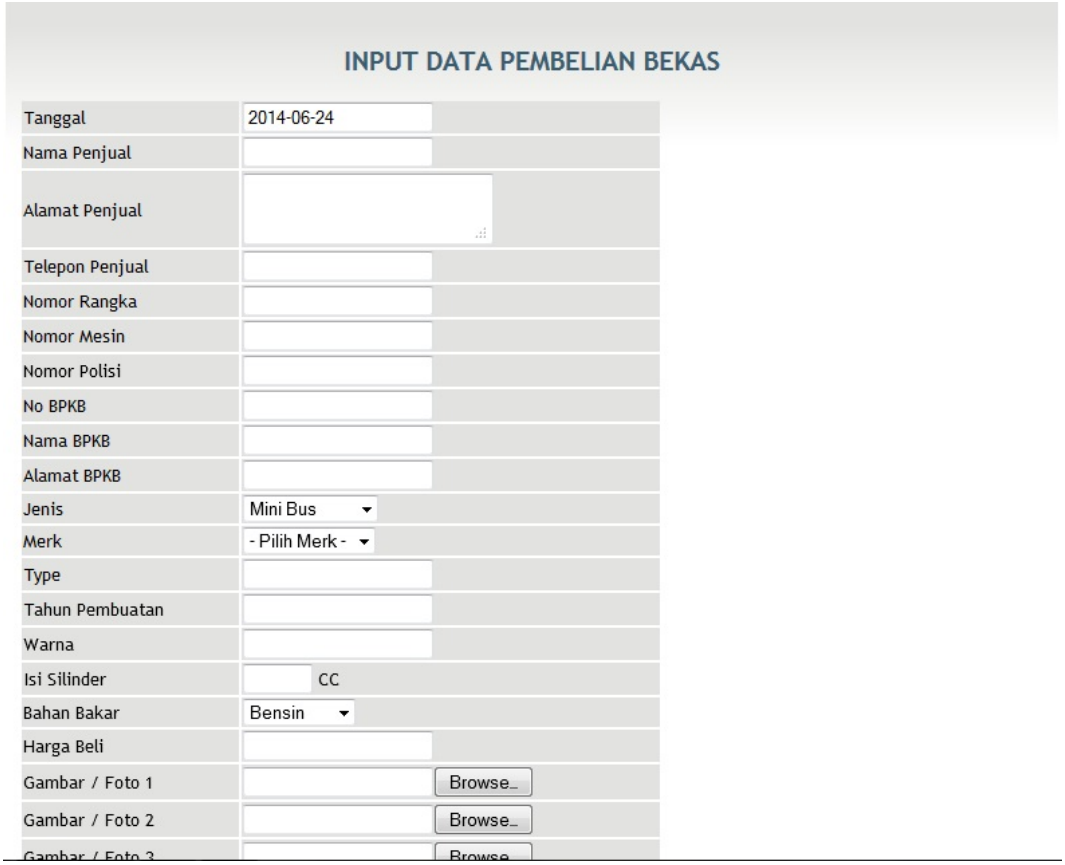
Pada halaman ini administrasi dapat melihat data pembelian dan bisa mengklik menu input pembelian kendaraan bekas, input pembelian kendaraan baru, *input purchese order*, dan cari data pembelian.



Gambar 4.35 Halaman Pembelian



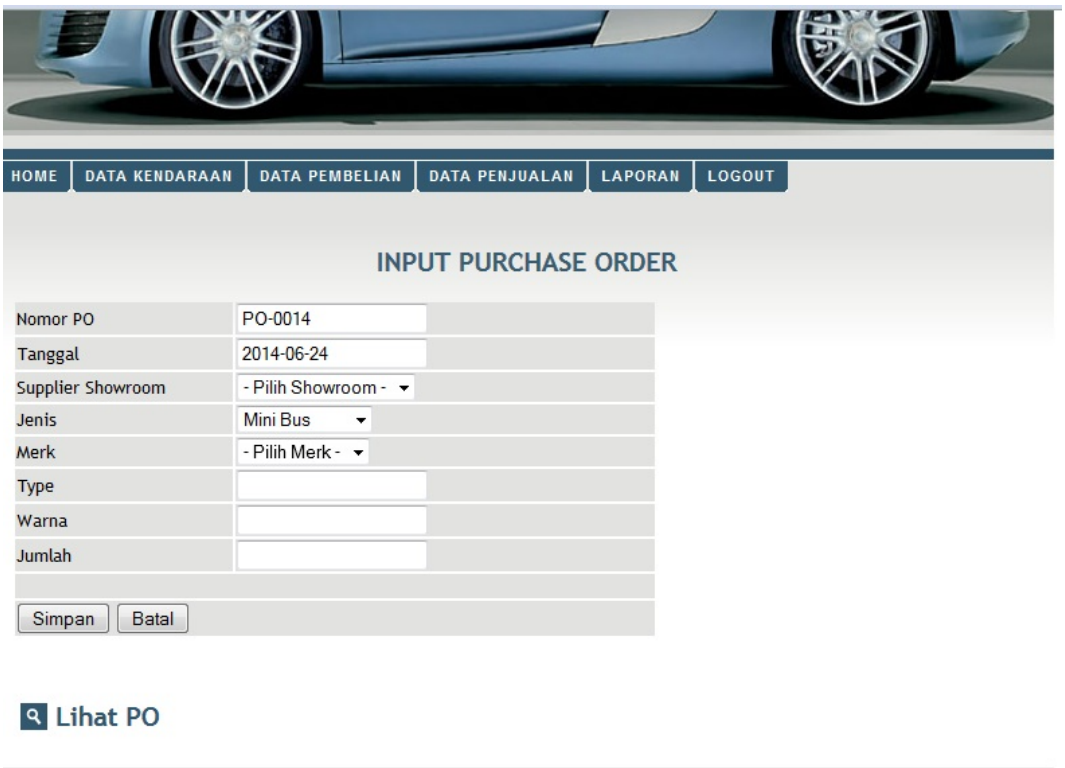
Gambar 4.36 Halaman *Input* Pembelian Kendaraan Baru



Gambar 4.37 Halaman *Input* Pembelian Kendaraan Bekas

9. Halaman *Purchase Order*

Halaman ini adalah halaman administrasi dapat melihat *purchase order* dan bisa mengklik menu input *purchase order*.



Gambar 4.38 Halaman *Input Purchase Order*

10. Halaman Cari Data Pembelian

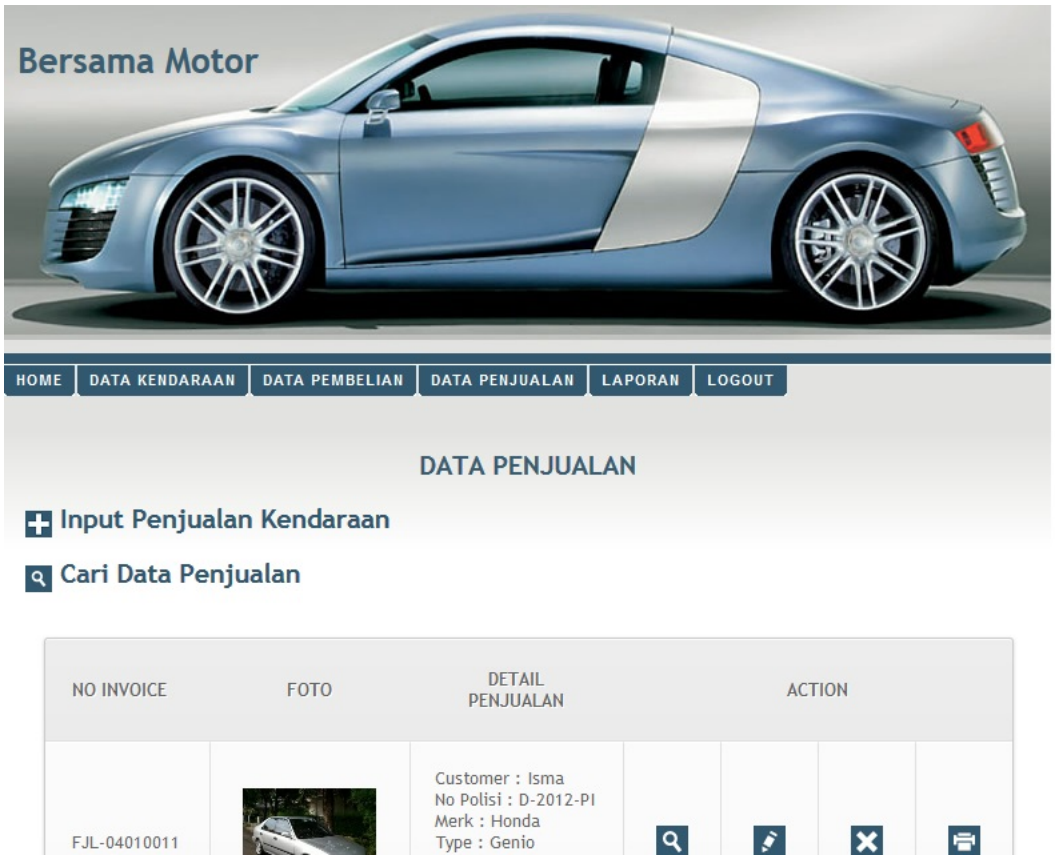
Halaman ini adalah halaman administrasi dapat mencari data pembelian dengan mengisikan no.faktur atau merk kendaraan.



Gambar 4.39 Halaman Cari Data Pembelian

11. Halaman Penjualan

Halaman ini adalah halaman administrasi dapat melihat data penjualan dan bisa mengklik menu *input* penjualan kendaraan.



Gambar 4.40 Halaman Penjualan



Gambar 4.41 Halaman *Input* Penjualan

12. Halaman Cari Data Penjualan

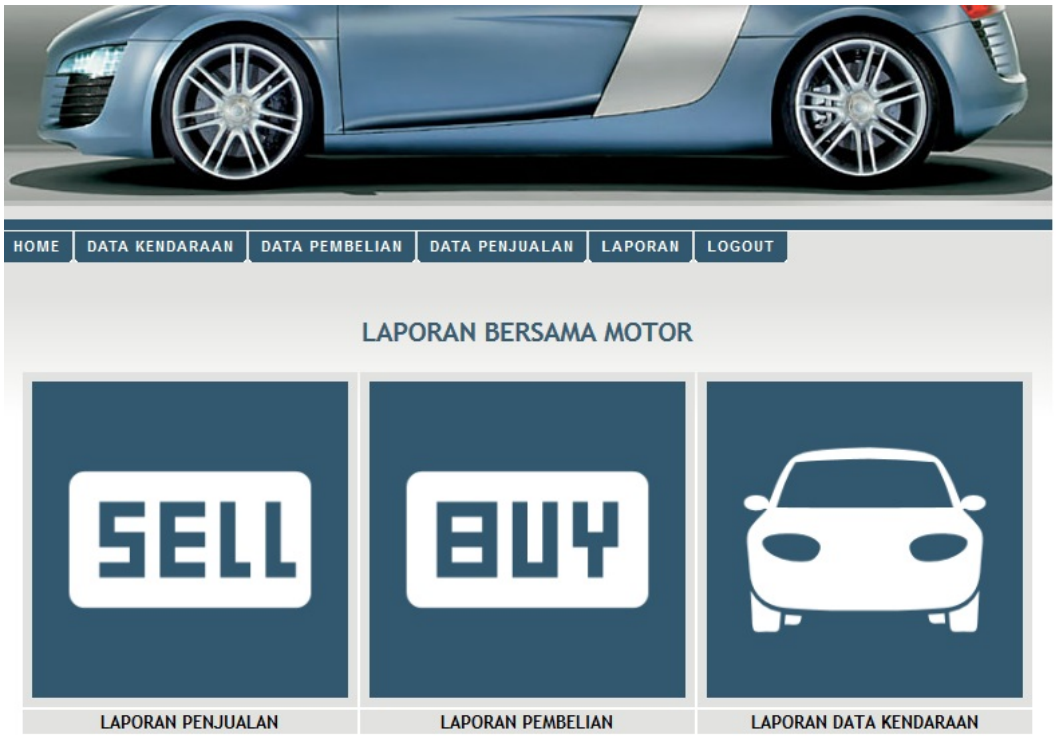
Halaman ini adalah halaman administrasi dapat melakukan pencarian data penjualan dengan mengisikan no.*invoice* atau merk kendaraan.



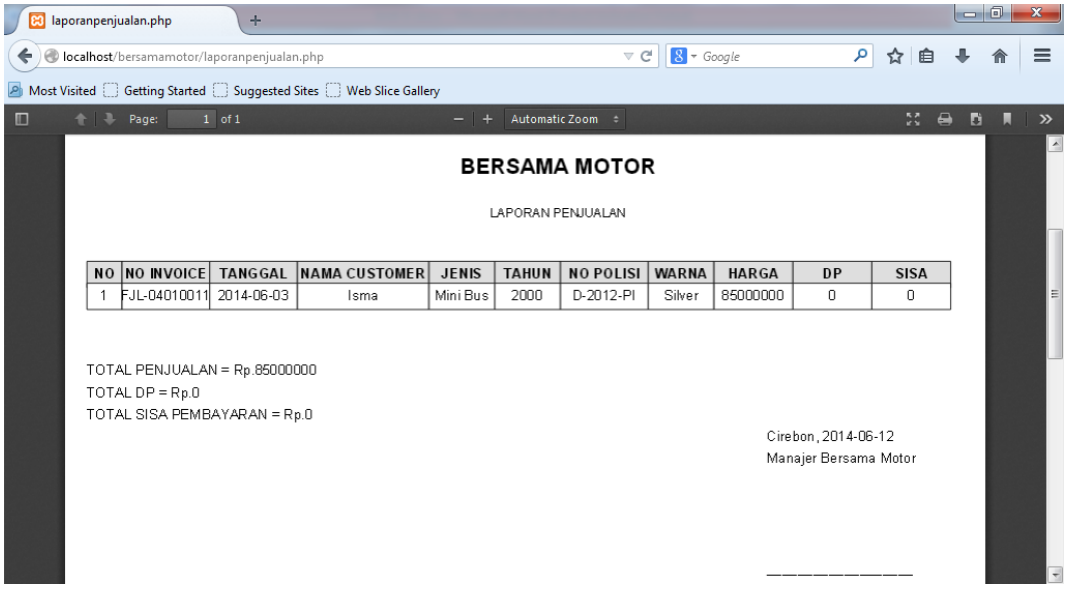
Gambar 4.42 Halaman Cari Data Penjualan

13. Halaman Laporan

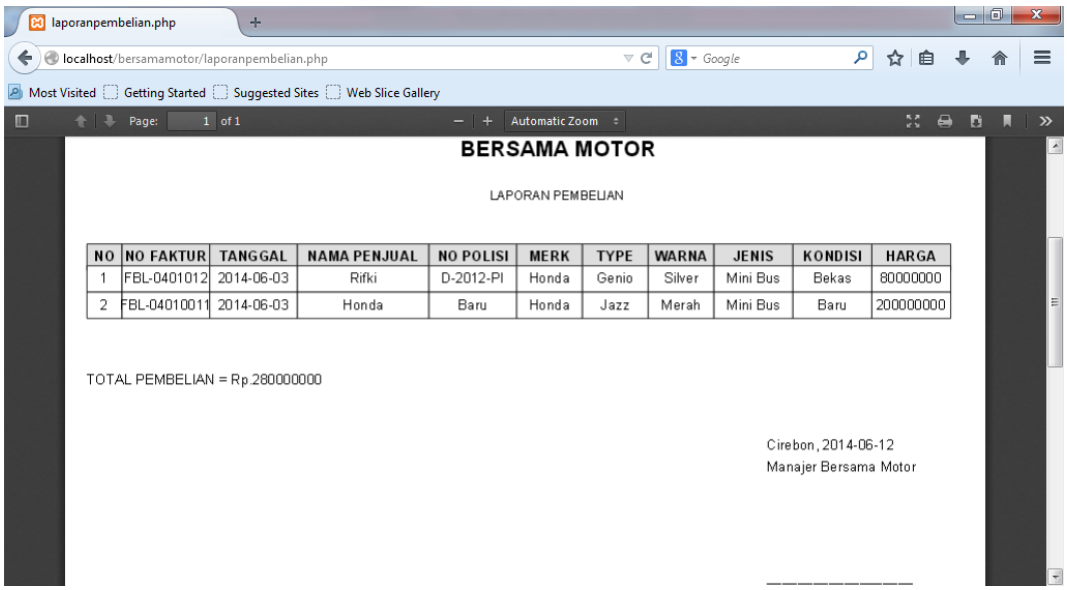
Halaman ini adalah halaman administrasi dapat melihat dan mencetak laporan pembelian, laporan penjualan, dan laporan kendaraan.



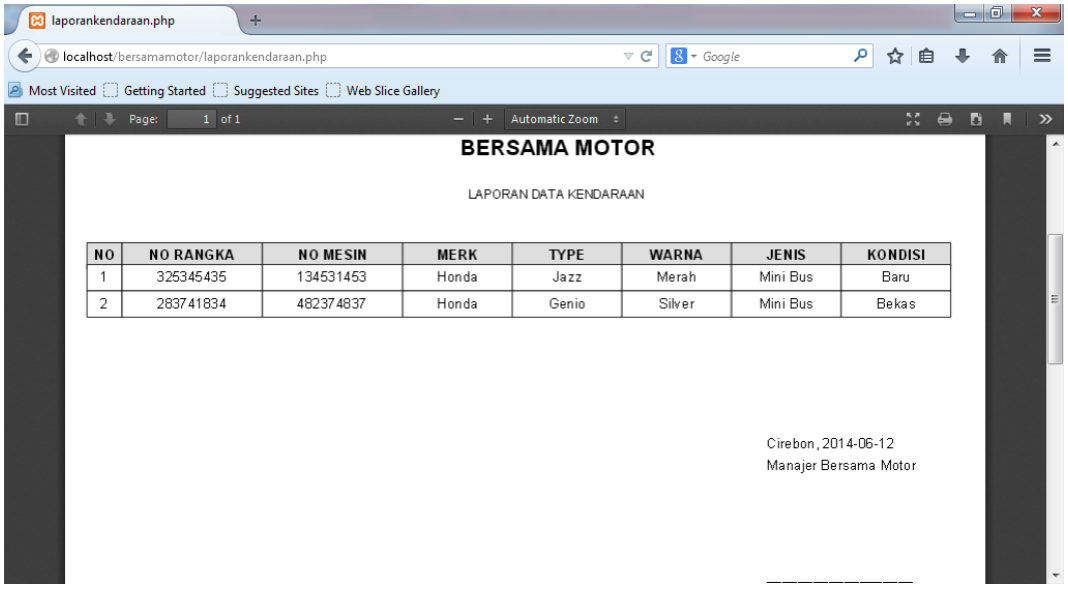
Gambar 4.43 Halaman Laporan



Gambar 4.44 Halaman Laporan Penjualan



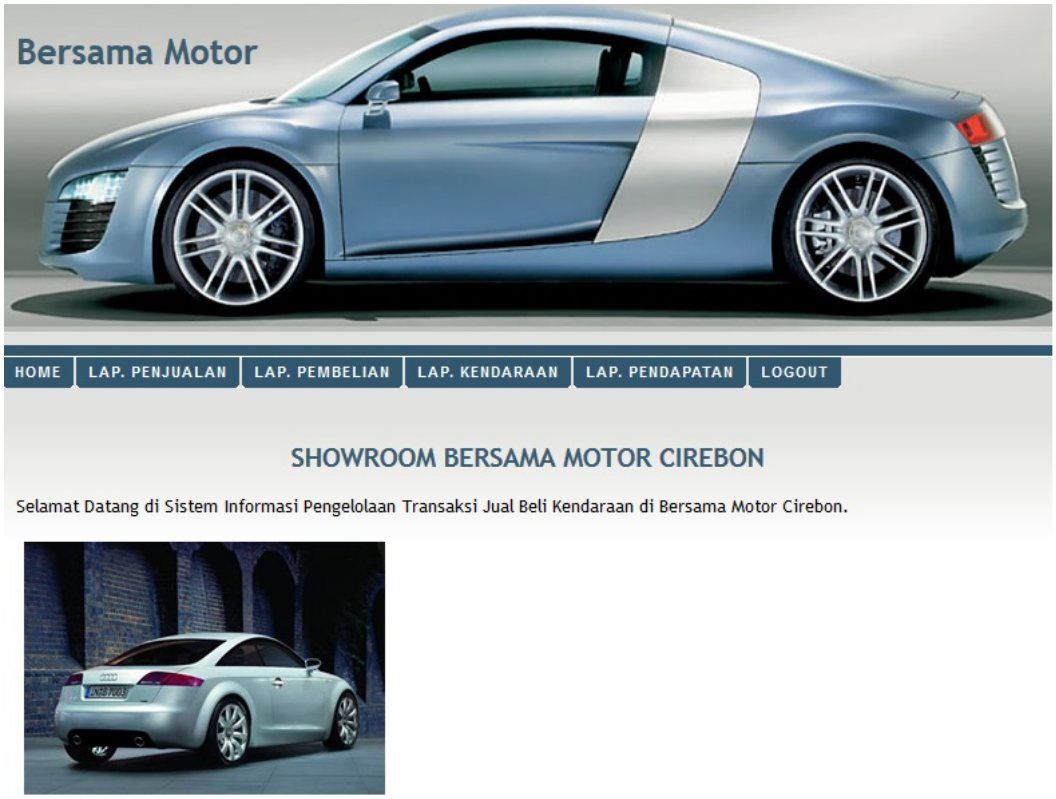
Gambar 4.45 Halaman Laporan Pembelian



Gambar 4.46 Halaman Laporan Kendaraan

14. Halaman Utama Pimpinan

Halaman ini adalah halaman utama ketika pimpinan berhasil login.



Gambar 4.47 Halaman Utama Pimpinan

15. Halaman Laporan Penjualan

Halaman ini adalah halaman dimana pimpinan dapat melihat semua laporan penjualan kendaraan yang terjadi.



Gambar 4.48 Halaman Laporan Penjualan

16. Halaman Laporan Pembelian

Halaman ini adalah halaman dimana pimpinan dapat melihat semua laporan pembelian kendaraan yang terjadi.



HOME | LAP. PENJUALAN | LAP. PEMBELIAN | LAP. KENDARAAN | LAP. PENDAPATAN | LOGOUT

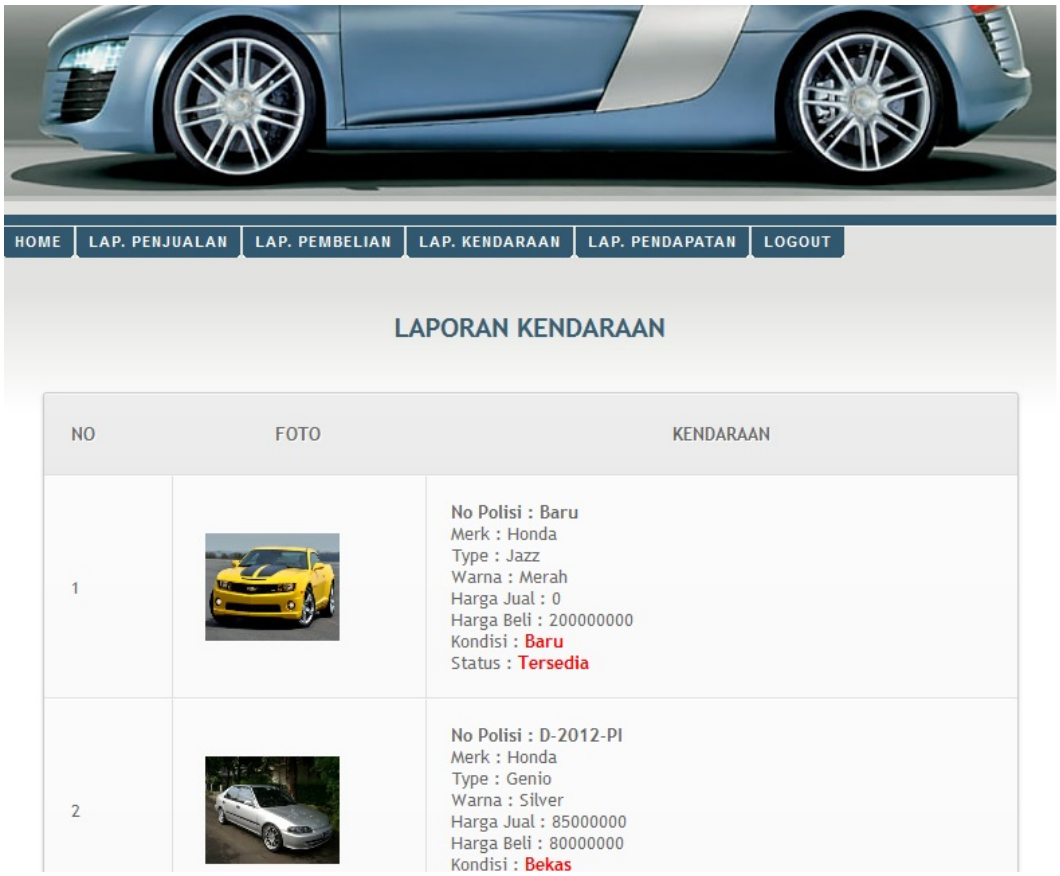
LAPORAN PEMBELIAN

NO FAKTUR	TANGGAL	NAMA PENJUAL	JENIS	MERK	TYPE	TAHUN	NOPOL	WARNA	HARGA BELI	KONDISI
FBL-0401012	2014-06-03	Rifki	Mini Bus	Honda	Genio	2000	D-2012-PI	Silver	85000000	Bekas
FBL-04010011	2014-06-03	Honda	Mini Bus	Honda	Jazz	2014	Baru	Merah	0	Baru
FBL-0201013	2014-06-23	Honda	Mini Bus	Toyota	Honda City	2000	-	Biru	0	Baru
FBL-0402013	2014-06-24	Dedi	Sedan	Honda	Honda City	2000	A 8989 DF	Biru	0	Bekas
FBL-0402014	2014-06-24	gff	Sedan	Honda	RUSH	2009	5t5t	HITAM	0	Bekas

Gambar 4.49 Halaman Laporan Pembelian

17. Halaman Laporan Kendaraan

Halaman ini adalah halaman dimana pimpinan dapat melihat semua laporan kendaraan yang pernah dibeli dan sudah dijual.



Gambar 4.50 Halaman Laporan Kendaraan

18. Halaman Laporan Pendapatan

Halaman ini adalah halaman dimana pimpinan dapat melihat laporan pendapatan yang dihasilkan.



Gambar 4.51 Halaman Laporan Pendapatan

4.5 Pengujian

Pengujian program adalah pengujian dimana *user* memasukan data ke dalam sistem, sekaligus pengecekan data jika *user* salah memasukan data maka aplikasi tidak dapat membaca data yang diinputkan kedalam sistem informasi pengelolaan transaksi jual beli kendaraan ini.

4.5.1 Rencana Pengujian

Tabel 4.13 Rencana Pengujian

Kelas Uji	Butir Uji	Tingkat Pengujian	Jenis Penguji
Login	Hak Akses Setiap User	Sistem	Black Box
Data Kendaraan	Pengisian Data Kendaraan	Sistem	Black Box
Data Pembelian	Pengisian Data Pembelian	Sistem	Black Box
Data Penjualan	Pengisian Data Penjualan	Sistem	Black Box
Purchese Order	*Pengisian Purchese Order*	Sistem	Black Box
Cari Data Pembelian	Pencarian Data Pembelian	Sistem	Black Box
Cari Data Penjualan	Pencarian Data Penjualan	Sistem	Black Box
Laporan	Pengisian Data Laporan sesuai data yang dimasukan	Sistem	Black Box

4.5.2 Kasus dan Hasil Pengujian

Berikut ini uraian hasil pengujian dengan teknik pengujian black box berdasarkan *requirement* pada rencana pengujian :

1. **Pengujian *Login***

Pengujian *login* pimpinan yang dapat melihat data kendaraan, data pembelian, data penjualan, laporan penjualan, laporan pembelian, laporan data kendaraan, dan pendapatan .

Pengujian *login* Administrasi yang mempunyai hak akses penuh untuk melakukan pengelolaan data pembelian, data penjualan, pencarian data pembelian, pencarian data penjualan, dan *purchese order.*

Tabel 4.14 Hasil Pengujian Login

Kasus dan Hasil Pengujian (Data Normal)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Status : Pimpinan **Username :** pimpinan **Password :** pimpinan (Data Valid)	Dapat melihat data kendaraan, data pembelian, data penjualan,laporan penjualan, laporan pembelian, laporan kendaraan, dan pendapatan.	Dapat melihat data kendaraan, data pembelian, data penjualan,laporan penjualan, laporan pembelian, laporan kendaraan, dan pendapatan.	[X] Diterima [ ] Ditolak
Status : Admin **Username :** admin **Password :** admin (Data Valid)	Dapat melihat dan menginput data pembelian, data penjualan, dan data kendaraan.	Dapat mengolah data penjualan, data pembelian, dan data kendaraan.	[X] Diterima [ ] Ditolak

2. Pengujian Data Pembelian

Pengujian ini adalah pengujian dengan memasukan pengisian data pembelian kendaraan dengan form data pembelian ke dalam database.

Tabel 4.15 Hasil Pengujian Data Pembelian

Kasus dan Hasil Pengujian (Data Normal)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua Data Diisi	Data dapat dimasukan ke dalam database pembelian.	Data tersimpan sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Pengujian (Data Salah)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Form tidak diisi semua atau hanya sebagian	Data tidak masuk ke dalam database, muncul pesan “Data Gagal disimpan”	Sesuai dengan yang diharapkan data tidak masuk ke dalam database	[X] Diterima [ ] Ditolak

3. Pengujian Data Penjualan

Pengujian ini adalah pengujian dengan memasukan pengisian data penjualan kendaraan dengan form data penjualan ke dalam database.

Tabel 4.16 Hasil Pengujian Data Penjualan

Kasus dan Hasil Pengujian (Data Normal)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua Data Diisi	Data dapat dimasukan ke dalam database penjualan.	Data tersimpan sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Pengujian (Data Salah)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Form tidak diisi semua atau hanya sebagian	Data tidak masuk ke dalam database, muncul pesan "Data Gagal disimpan"	Sesuai dengan yang diharapkan data tidak masuk ke dalam database	[X] Diterima [ ] Ditolak

4. Pengujian Data Kendaraan

Pengujian ini adalah pengujian dengan menampilkan seluruh data kendaraan dengan form data kendaraan.

Tabel 4.17 Hasil Pengujian Data Kendaraan

Kasus dan Hasil Pengujian (Data Normal)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua Data Diisi	Dapat menampilkan data kendaraan	Menampilkan data kendaraan sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak

5. Pengujian *Purchase Order*

Pengujian ini adalah pengujian dengan memasukan pengisian data *purchase order* dengan form data pembelian ke dalam database.

Tabel 4.18 Hasil Pengujian *Purchase Order*

Kasus dan Hasil Pengujian (Data Normal)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua Data Diisi	Dapat menginput *Purchase Order* dan menampilkan *Purchase Order*	Dapat menginput *Purchase Order* dan menampilkan *Purchase Order*	[X] Diterima [ ] Ditolak
Kasus dan Hasil Pengujian (Data Salah)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Form tidak diisi semua atau hanya sebagian	muncul pesan “Data Gagal disimpan”	Sesuai dengan yang diharapkan muncul pesan data gagal disimpan	[X] Diterima [ ] Ditolak

6. Pengujian Cari Data Pembelian

Pengujian ini adalah pengujian pencarian data pembelian dengan memasukan no. faktur atau merk kendaraan.

Tabel 4.19 Hasil Cari Data Pembelian

Kasus dan Hasil Pengujian (Data Normal)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua Data Diisi	Dapat melakukan pencarian data pembelian berdasarkan no. Faktur dan merk	Menampilkan data pembelian sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Pengujian (Data Salah)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Form tidak diisi	Muncul pesan data tidak tersedia	Sesuai dengan yang diharapkan muncul pesan data tidak tersedia	[X] Diterima [ ] Ditolak

7. Pengujian Cari Data Penjualan

Pengujian ini adalah pengujian pencarian data penjualan dengan memasukan no. *invoice* atau merk kendaraan.

Tabel 4.20 Hasil Cari Data Penjualan

Kasus dan Hasil Pengujian (Data Normal)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua Data Diisi	Dapat melakukan pencarian data penjualan berdasarkan no. *invoice* dan merk	Menampilkan data penjualan sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Pengujian (Data Salah)			
Data Masukan	**Data Diharapkan**	**Pengamatan**	**Kesimpulan**
Form tidak diisi	Muncul pesan data tidak tersedia	Sesuai dengan yang diharapkan muncul pesan data tidak tersedia	[X] Diterima [ ] Ditolak

4.5.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji *sample* di atas dapat ditarik kesimpulan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil sesuai dengan yang diharapkan.