BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1. Analisis Sistem Yang Berjalan

Analisis sistem yang berjalan pada bengkel Hel's Motor bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah yang dihadapi sistem untuk dijadikan landasan usulan perancangan analisis sistem yang sedang berjalan yang dilakukan berdasarkan urutan kejadian yang ada pada bengkel Hel's Motor.

4.1.1. Analisis Dokumen

Analisis dokumen merupakan kegiatan pengumpulan informasi mengenai dokumen-dokumen yang digunakan dalam suatu sistem. Tujuan dari analisis dokumen adalah mengetahui dan memahami dokumen-dokumen apa saja yang terlibat dan mengalir dalam suatu sistem yang sedang berjalan. Dokumen yang digunakan pada sistem informasi bengkel Hel's Motor adalah sebagai berikut :

1. Nama Dokumen : Faktur Penjualan

   Sumber : *Supplier*

   Rangkap : 2

   Fungsi : Sebagai bukti pembayaran pembelian *Spare Parts* dari *Supplier*

   Item Data : No_Faktur, Tanggal, Sales, No_CO, Tgl_CO, Tgl_jatuh_tempo, Repick, Pembeli, Nama, Alamat, NPWP, No, Kode_Barang, Part_no, Uraian, Jumlah_pcs_koli,, Harga_satuan, Reg, Discount_cam, Rp, Harga_diskon, Jumlah_harga, Penjual, pembeli, Harga_jual, Pot_harga, Pot_tunai, DPP, PPN, Total.

2. Nama Dokumen : Nota Penjualan

   Sumber : Bag Administrasi

   Rangkap : 2

   Fungsi : Sebagai bukti pembayaran penjualan *Spare Parts* ke *customer*

   Item Data : Tgl, Type_motor, no_polisi, Mekanik, Qty, Nama_barang, Harga _satuan, jumlah, Total_jumlah

3. Nama Dokumen : Laporan Harian

   Sumber : Bag. Administrasi

   Rangkap : -

   Fungsi : Pencatatan keuangan harian

   Item Data : No, Pendapatan_bengkel, Spare_part, Service, Counter, Lain_lain, Pengeluaran, Saldo, Total

4. Nama Dokumen : Order Pembelian

   Sumber : Bag.Administrasi

   Rangkap : -

   Fungsi : Pemesanan barang kepada *supplier*

   Item Data : Kepada_yth, Alamat, Tlp, Tanggal, No, Nama_spart, Code_spart, Jumlah, Ttd_pemilik

4.1.2. Analisis Prosedur yang Sedang Berjalan

Prosedur kerja yang terlibat dalam sistem pembelian dan penjualan *Spare Parts* di bengkel Hel's Motor adalah sebagai berikut.

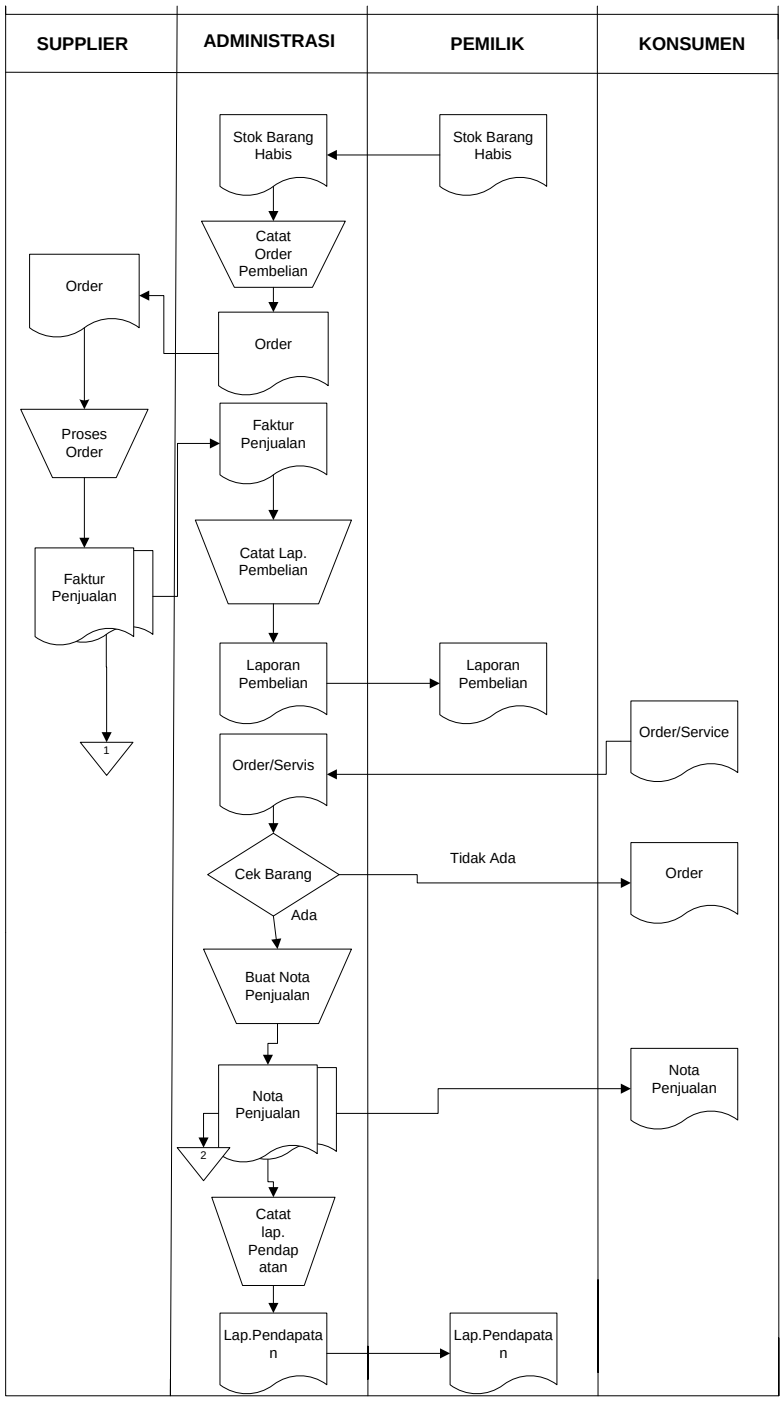
Prosedur pembelian dan penjualan yang sedang berjalan, sebagai berikut :

1. Pemilik Bengkel melaporkan stok barang habis
2. Bag. Administrasi melakukan pembelian barang ke *supplier* kemuadian *supplier* memberikan faktur penjualan kepada bag.administrasi

3. Bag.Administrasi mencatat laporan pembelian dan dilaporkan kepada pemilik.
4. Konsumen melakukan orderan/*service* ke bag.administrasi, kemudian pengecekan barang ada atau tidaknya barang tersebut.
5. Bila barang ada Bag.administrasi membuat nota penjualan yang akan diserahkan ke konsumen dan arsip bengkel.
6. Bag.Administrasi mencatat laporan pendapatan harian bengkel dan diserahkan ke pemilik bengkel

4.1.2.1 Flow Map

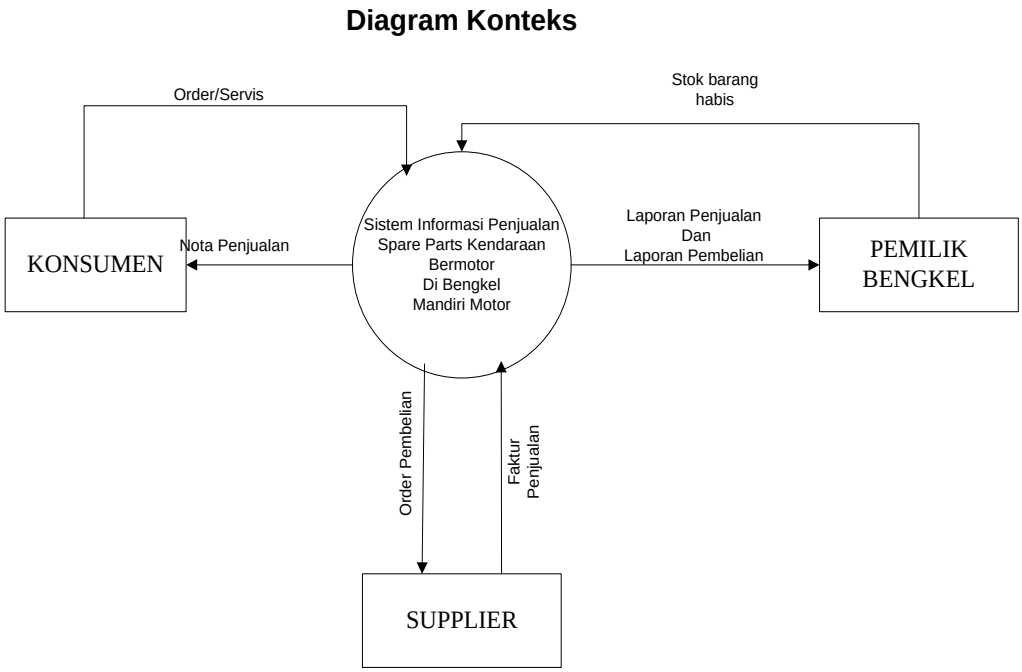
Di bawah ini merupakan gambar flow map prosedur pembelian dan penjualan pada Hel’s Motor saat ini :



Gambar 4.1 Flow map Pembelian dan Penjualan Spare Part yang Berjalan

4.1.2.2 Diagram Konteks

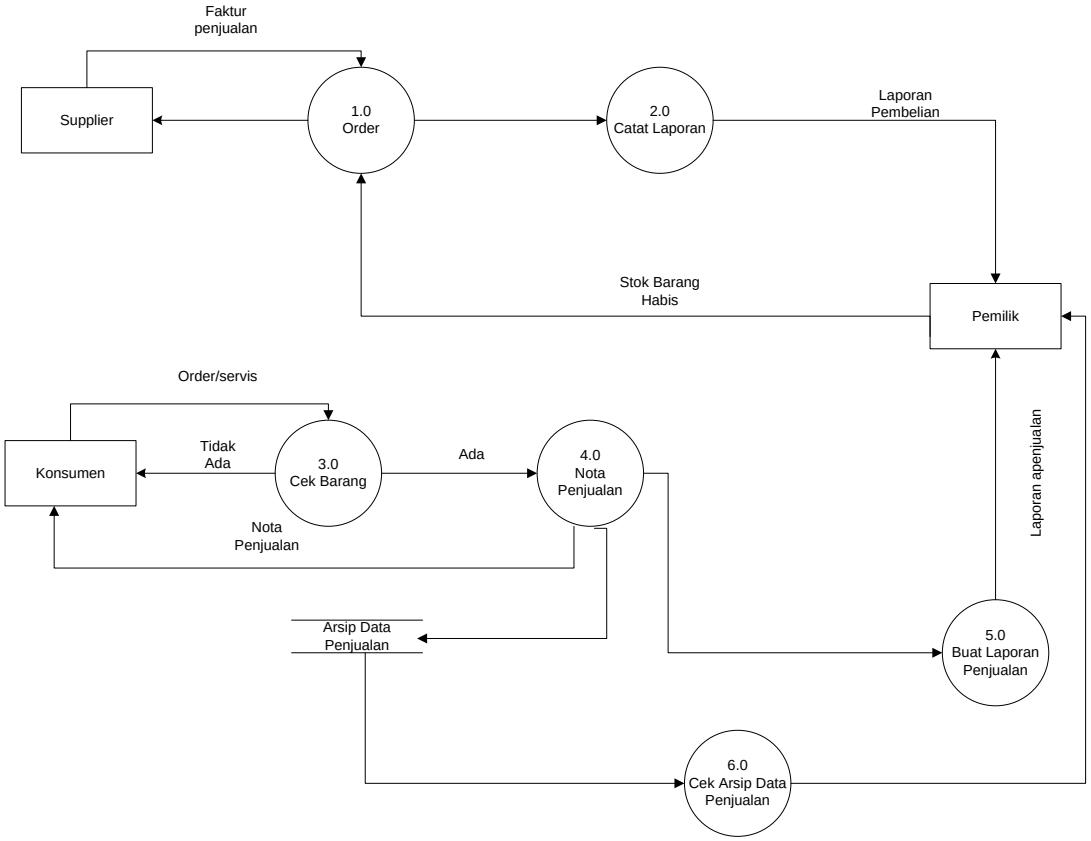
Diagram Konteks penjualan dan pembelian Spare Parts kendaraan bermotor sebagai berikut



Gambar 4.2. *Diagram Konteks*

4.1.2.3 Data Flow Diagram (DFD)

Data Flow Diagram yang sedang berjalan di bengkel Hel's Motor sebagai berikut :



Gambar 4.3. *Data Flow Diagram* Yang Sedang Berjalan

4.1.3. Evaluasi Sistem yang Sedang Berjalan

Berdasarkan hasil pengamatan dan penelitian dari sistem informasi penjualan *spare parts* di bengkel Hel's Motor yang sedang berjalan, maka penulis mengevaluasi sistem tersebut sebagai berikut :

Tabel 4.1. Evaluasi Sistem yang sedang berjalan

No	Permasalahan	Rencana Perancangan / Solusi
1.	Untuk proses transaksi pembelian *Spare Parts* yang masih mengandalkan *nota* pembelian.	Dengan adanya sistem *database* dibagian pembelian dapat mempermudah proses pengolahan data pembelian menjadi lebih cepat dan *efisien*.
2.	Untuk proses transaksi penjualan *Spare Parts* yang masih mengandalkan *nota* penjualan.	Dengan adanya sistem *database* dibagian penjualan dapat mempermudah proses pengolahan data penjualan menjadi lebih cepat dan *efisien*.
3.	Untuk Proses *Stock* barang yang masih sering melihat pada *etalase-etalase*.	Dengan adanya sistem *database* dibagian *Stock* barang dapat mempermudah proses pengolahan data *Stock* barang berkurang dan bertambah.

4.	Untuk proses laporan penjualan dan pembelian *Spare Parts* yang masih mengandalkan buku.	Dengan adanya sistem database dapat mempermudah proses pengolahan data laporan penjualan dan pembelian *Spare Parts* menjadi lebih cepat.

4.2. Perancangan Sistem

Perancangan sistem adalah suatu tahapan untuk mendesain program dengan menggunakan bentuk operasi berupa sumber-sumber diagram alir sistem. Perancangan ini dibuat berdasarkan permasalahan-permasalahan pada sistem yang telah diuraikan sebelumnya pada tahap analisis sistem

4.2.1 Tujuan Perancangan Sistem

Tujuan perancangan sistem ini adalah untuk memberikan penjelasan kepada pemakai program atau User. Dengan demikian pembuatan program ini diharapkan dapat membantu mengatasi kekurangan-kekurangan yang ada. Adapun perancangan yang diusulkan merupakan langkah untuk lebih mengefektifkan dan mengefisienkan penyampaian penjualan dan pembelian yang dilakukan secara *offline*.

4.2.2 Gambaran Umum Sistem yang Diusulkan

Sistem yang diusulkan merupakan suatu langkah untuk lebih mengefektifkan penyampaian penjualan dan pembelian pada bengkel Hel’s Motor. Sistem yang diusulkan merupakan sistem yang berbasis *web*, dan dijalankan melalui media

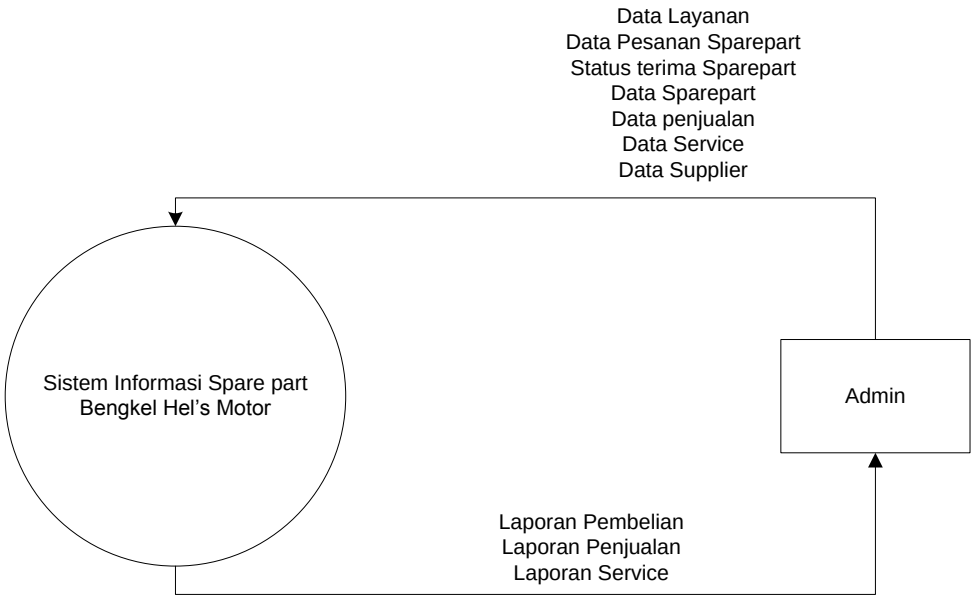
internet. Tetapi sistem ini tidak sepenuhnya menggantikan sistem yang lama, karena bagaimanapun sistem penjualan konvensional tetap diperlukan. Sehingga dapat dikatakan sistem ini merupakan sistem tambahan agar kinerja bengkel lebih maksimal.

4.2.3 Perancangan Prosedur yang Diusulkan

Rancangan sistem yang diusulkan bukan untuk menggantikan sistem yang lama tapi merupakan penambahan sistem yaitu membuat program secara *offline* agar kinerja bengkel lebih maksimal. Perancangan dibuat untuk meminimalkan kekurangan, kelemahan, dan mengatasi masalah yang di hadapi dan sistem informasi yang dirancang diharap akan memberi solusi alternatif baru yang dapat memaksimalkan penyampaian penjualan dan pembelian pada bengkel Hel’s Motor.

4.2.3.1 Diagram Konteks

Diagram konteks adalah sebuah diagram sederhana yang menggambarkan hubungan antara entity luar, masukan dan keluaran dari sistem. Adapun diagram konteks dapat dilihat pada gambar 4.4 :

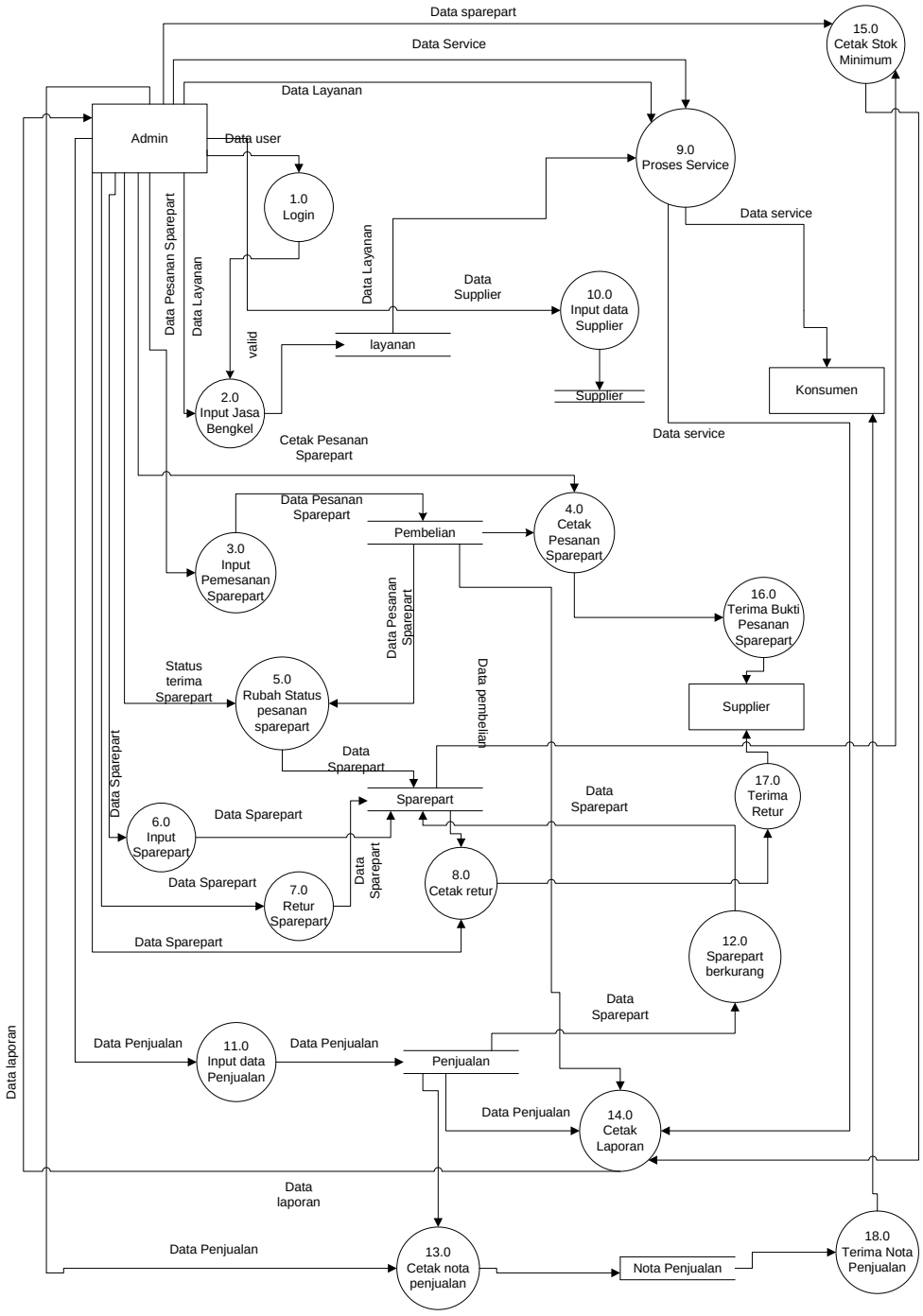


Gambar 4.4. *Diagram Konteks* Yang Diusulkan

4.2.3.2 Data Flow Diagram

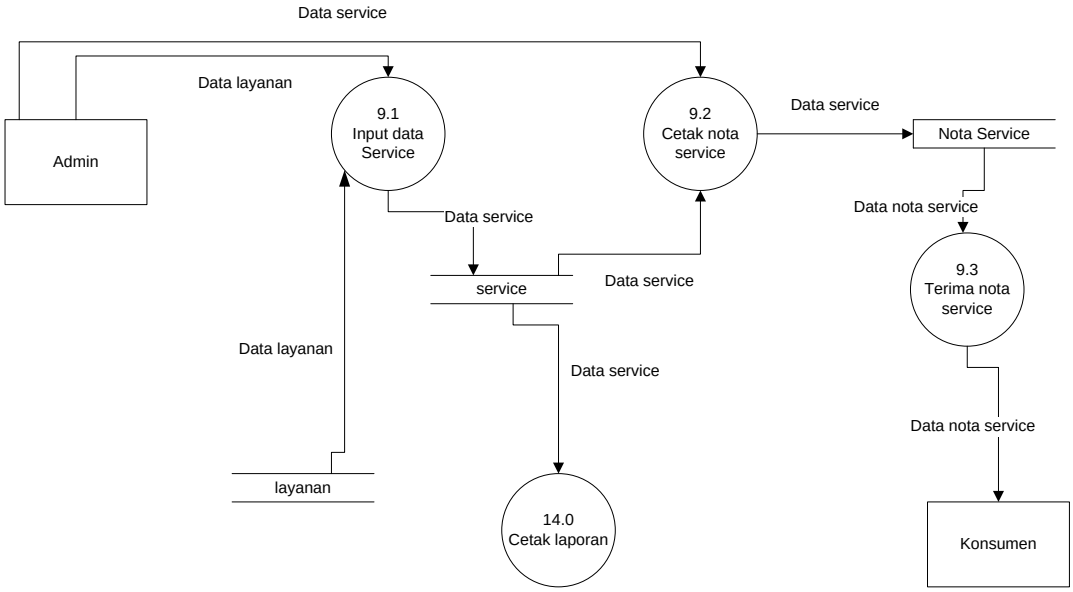
Diagram aliran data merupakan teknik yang menjelaskan keadaan sebenarnya yang dibuat untuk menggambarkan darimana asal dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. Adapun DFD yang diusulkan dapat dilihat pada gambar di bawah ini.

DFD LEVEL 1



Gambar 4.5. *Data Flow Diagram* level 1 Yang Diusulkan

DFD Level 2 Proses 9.0 proses service



Gambar 4.6. *Data Flow Diagram* level 2 Yang Diusulkan

4.2.3.3 Kamus Data

Kamus data adalah suatu kamus yang berfungsi untuk mendeskripsikan data dan aliran informasi yang ada di Data Flow Diagram, kamus data digunakan dalam perancangan basis data yang dipakai dalam pembuatan program aplikasi.

Kamus Data DFD level 1 dan level 2 Penjualan *offline* pada bengkel Hel's Motor adalah sebagai berikut:

1. Nama arus data : Data User

   Deskripsi : Berisi tentang Pengecekan login

Aliran data : Admin – Proses1.0,

Atribut : username, password, nama_lengkap,status

2. Nama arus data : Data Layanan

Deskripsi : Berisi tentang data Layanan bengkel

Aliran data : Admin – Proses2.0, Proses2.0 - F.Layanan, Flayanan-proses9.0, admin – proses9.0

Atribut : Kode_layanan, nama_layanan, jenis_layanan, harga_layanan

3. Nama arus data : Data Pembelian

Deskripsi : Berisi tentang data pembelian dari *Supplier*

Aliran data : fpembelian – proses 13.0.

Atribut : kode_pembelian, tgl_pembelian, supplier, total_pembelian

4. Nama arus data : Data Sparepart

Deskripsi : Berisi tentang data *Sparepart*

Aliran data : Admin – Proses6.0, Proses6.0 – F.sparepart, admin – proses8.0, proses8.0 – fsprepart, proses5.0 – fsparepart.

Atribut : kode_sparepart, Nama_sparepart, Harga_sparepart, Stok_sparepart

5. Nama arus data : Status Terima *Sparepart*

   Deskripsi : Berisi tentang status pengiriman barang

   Aliran data : Admin– proses5.0

   Atribut : Kode_pembelian, Tgl_pembelian, Supplier, Total_pembelian, Status_pembelian.

6. Nama arus data : Data Pesanan *Sparepart*

   Deskripsi : Berisi tentang pesanan pembelian barang.

   Aliran data : admin – proses3.0, proses3.0 – fpembelian, fpembelian – proses4.0, fpembelian – proses5.0.

   Atribut : Kode_pembelian, Tgl_pembelian, Supplier, Total_pembelian, Status_pembelian

7. Nama arus data : Data penjualan

   Deskripsi : Berisi tentang penjualan *sparepart*

   Aliran data : Admin – proses11.0, proses11.0 – fpenjualan, Admin – proses13.0, fpenjualan – proses13.0

8. Nama arus data : Data *Supplier*

   Deskripsi : Berisi tentang data *supplier*

   Aliran data : Admin – proses20.0, proses20.0 – fsupplier

9. Nama arus data : Data Service level 2

   Deskripsi : Berisi data service konsumen

Aliran data : Admin – proses9.0, proses9.0 – fservice, admin – proses9.0, fservice – proses9.0.

Atribut : No_service, Tgl_service, Keluhan, Montir, No_polisi, Type_motor

4.2.4 Perancangan Basis Data

Perancangan basis data adalah langkah untuk menentukan basis data yang diharapkan dapat mewakili seluruh kebutuhan pengguna. Basis data didefinisikan sebagai kumpulan data yang berupa suatu penyampaian informasi yang lengkap dengan jenis *record* yang mempunyai spesifikasi yang sama, sedangkan yang dimaksud dengan *record* adalah kumpulan data yang berisikan *field* - *field* yang berbeda, jumlah *record* pada umumnya terbatas.

Perancangan basis data dapat diartikan yaitu menciptakan atau merancang kumpulan data yang terhubung dan disimpan secara bersama - sama. Basis data itu sendiri dapat diartikan sebagai:

1. Himpunan kelompok data yang saling berhubungan satu sama lain.
2. Kumpulan data yang disimpan secara bersama - sama untuk memenuhi berbagai kebutuhan.

4.2.4.1. Normalisasi

Proses Perubahan perancangan basis data yang menghasilkan skema table dalam bentuknormal.

Untuk mendapatkan proses normalisasi yang normal, memerlukan tahapan sebagai berikut ;

1. Bentuk Unnormalisasi (Tidak Normal)

Pada tahap ini tabel belum normal dengan mencantumkan semua atribut yang ada.

Admin **:**{ No_Transaksi, Kode_Onderdil, Jumlah_Onderdil, Harga_Onderdil , SubTotal_Transaksi, Kode_Pembelian, Kode_Sparepart , Jumlah_Pembelian, Harga_Pembelian, SubTotal_Pembelian, No_Penjualan, Kode_Sparepart , Jumlah_Penjualan , Harga_Sparepart, SubTotal_Penjualan, Kode_Retur, Kode_Sparepart , Jumlah_Retur, SubTotal_Retur, No_Service, Kode_Service , Harga_Service, Kode_Layanan, Nama_Layanan, Jenis_Layanan, Harga_Layanan, Kode_Pembelian, Tgl_Pembelian, Supplier , Total_Pembelian, Status_Beli , No_Penjualan, Tgl_Penjualan, Total_Penjualan, Kode_Retur, Tgl_Retur, Kode_Pembelian, Total_Retur, No_Service, Tgl_Service, No_Polisi , Total_Service, Keluhan,

Montir , Konsumen , Jenis_Service, Type_Motor , Kode_Sparepart, Nama_Sparepart, Harga_Sparepart , Stock_Sparepart, Kode_supplier, Kode_supplier, Nama_supplier, alamat_supplier, Telp_supplier No_Transaksi, Tgl_Transaksi, No_Polisi, No_Identitas, Total_Transaksi, User_Name, Password, Nama_Lengkap, Status }

2. Bentuk Normal ke – 1

Bentuk normal pertama yaitu dengan menghilangkan kesamaan atribut pada bentuk tidak normal

Admin **:** {No_Transaksi, Kode_Onderdil, Jumlah_Onderdil, Harga_Onderdil , SubTotal_Transaksi, Kode_Pembelian, Kode_Sparepart , Jumlah_Pembelian, Harga_Pembelian, SubTotal_Pembelian, No_Penjualan, Kode_Sparepart , Jumlah_Penjualan , Harga_Sparepart, SubTotal_Penjualan, Kode_Retur, Jumlah_Retur, SubTotal_Retur, No_Service, Kode_Service , Harga_Service, Kode_Layanan, Nama_Layanan, Jenis_Layanan, Harga_Layanan,

Tgl_Pembelian, Supplier , Total_Pembelian, Status_Beli , No_Penjualan, Tgl_Penjualan, Total_Penjualan, Kode_Retur, Tgl_Retur, Total_Retur, Tgl_Service, No_Polisi , Total_Service, Keluhan, Montir , Konsumen , Jenis_Service, Type_Motor , Nama_Sparepart, Stock_Sparepart, Kode_supplier, Nama_supplier, alamat_supplier, Telp_supplier, Tgl_Transaksi, No_Polisi, No_Identitas, Total_Transaksi, User_Name, Password, Nama_Lengkap, Status }

3. Bentuk normal kedua (2NF)

Sebuah relasi dalam bentuk normal II, jika relasi tersebut dalam bentuk normal I serta seluruh atribut (bukan *primary key*) tergantung secara fungsional sepenuhnya pada *primary key*.

Sparepart :{ kode_sparepart*, Nama_sparepart,
Harga_sparepart, Stok_sparepart, No_transaksi**,
kode_onderdil**, Jumlah_onderdil,
harga_onderdil, Subtotal_transaksi,
kode_pembelian**, kode_sparepart**,
Jumlah_pembelian, SubTotal_pembelian }

Pembelian :{ kode_pembelian*, tgl_pembelian, supplier, total_pembelian, kode_pembelian**, kode_sparepart**, Jumlah_pembelian, SubTotal_pembelian }

Layanan :{ kode_layanan*, nama_layanan, jenis_layanan, harga_layanan, No_transaksi**, kode_service**, Harga_service }

Service :{ No_service*, tgl_service, no_polisi, keluhan, Jenis_service, montir, No_polisi, Kode_service**, harga_service, No_service }

Retur :{Kode_retur*, Tgl_retur, Kode_pembelian**, Total_retur, Kode_retur**,kode_sparepart**, Jumlah_retur, Subtotal_retur}

Supplier :{ Kode_supplier*, Nama_supplier, alamat_supplier, Telp_supplier}

4. Bentuk normal ketiga (3NF)

Suatu relasi dalam normal III, jika relasi tersebut sudah dalam bentuk normal II dan setiap atribut tidak tergantung secara transitif pada *primary key*.

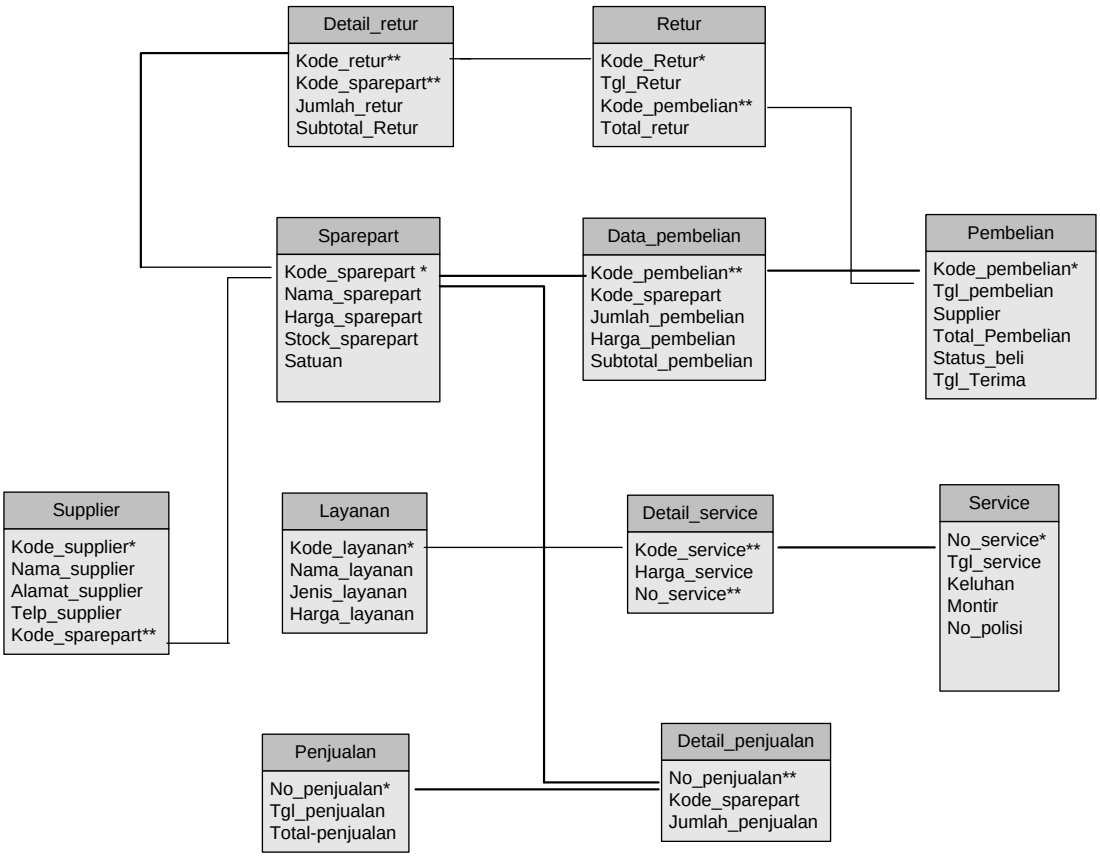
Sparepart :{ kode_sparepart*, Nama_sparepart, Harga_sparepart, Stok_sparepart }

Data_Pembelian :{ kode_pembelian**, kode_sparepart**, Jumlah_pembelian, SubTotal_pembelian,}

Pembelian	:{ kode_pembelian*, tgl_pembelian, supplier, total_pembelian}
Layanan	:{ kode_layanan*, nama_layanan, jenis_layanan, harga_layanan }
Service	:{ No_service*, Tgl_service, keluhan, jenis_service, Montir, No_polisi}
Detail_service	:{ No_transaksi**, kode_service**, Harga_service }
Penjualan	:{ No_penjualan*, Tgl_penjualan, Total_penjualan}
Detail_penjualan	:{ No_penjualan**, kode_sparepart, jumlah_sparepart}
Retur	:{ Kode_retur*, Tgl_retur, Kode_pembelian**, Total_retur}
Detail_Retur	:{ Kode_Retur**, Kode_sparepart**, Jumlah_retur, Subtotal_retur}
Supplier	:{ Kode_supplier*, Nama_supplier, alamat_supplier, Telp_supplier}

4.2.4.2 Relasi Tabel

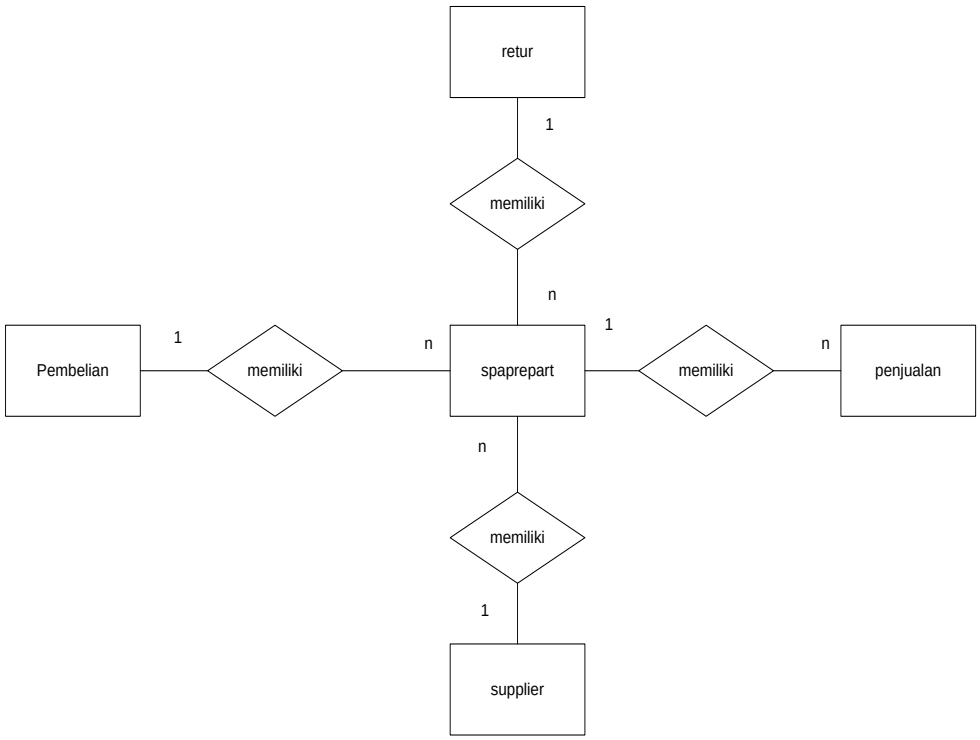
Relasi Antar Tabel adalah bagan yang menggambarkan hubungan antar tabel satu dengan tabel yang lain sehingga membentuk Basis Data. Selain tabel yang berelasi, ada beberapa tabel yang tidak berelasi dengan tabel manapun, tabel tersebut digunakan untuk melengkapi sistem penjualan yang akan dibangun.



Gambar 4.7 Relasi Tabel

4.2.4.3 *Entity Relationship Diagram*

ERD adalah pemodelan data utama yang membantu mengorganisasikan data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas. ERD digunakan untuk memodelkan struktur data dan hubungan antar data. Berikut ini adalah ERD yang diusulkan untuk bengkel Hel's Motor :



Gambar 4.8 *ERD*

4.2.4.4 Struktur File

Struktur file digunakan dalam perancangan sistem, karena struktur file ini akan menentukan struktur fisik database yang menunjukkan struktur dari elemen data yang menyatakan panjang elemen data dan jenis datanya. Struktur merupakan urutan isi atau data-data item yang terdaftar pada sebuah *record.* File yang dipakai untuk sistem informasi pada bengkel Hel's Motor adalah sebagai berikut :

Tabel 4.2 Struktur File detail_onderdil

Nama File : detail_onderdil

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	No_transaksi	Char	20	FK	No transaksi onderdil
2	Kode_onderdil	Char	20	FK	Kode onderdil
3	Jumlah_onderdil	Char	4		Jumlah onderdil
4	Harga_onderdil	Char	20		Harga jual onderdil
5	Subtotal_transaksi	Char	20		Jumlah total keseluruhan

Tabel 4.3 Struktur File detail_pembelian

Nama File : detail_pembelian

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	Kode_pembelian	Char	20	FK	No pembelian dari supplier
2	Kode_sparepart	Char	20	FK	Kode onderdil
3	Jumlah_pembelian	Char	5		Jumlah onderdil
4	Harga_pembelian	Char	20		Harga jual onderdil
5	Subtotal_pembelian	Char	20		Jumlah total keseluruhan

Tabel 4.4 Struktur File detail_service

Nama File : detail_service

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	No_transaksi	Char	20	FK	No transaksi service
2	Kode_service	Char	20	FK	Kode service
3	Harga_service	Char	20		Harga service

Tabel 4.5 Struktur detail penjualan

Nama File : detail_penjualan

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	No_Penjualan	Char	20	FK	No penjualan kepada konsumen
2	Kode_sparepart	Char	20	FK	Kode onderdil
3	Jumlah_Penjualan	Char	5		Jumlah onderdil
4	Harga_Sparepart	Char	20		Harga jual onderdil
5	SubTotal_Penjualan	Char	20		Jumlah total keseluruhan

Tabel 4.6 Struktur File detail retur

Nama File : detail_retur

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	Kode_Retur	Varchar	20	FK	Kode retur barang
2	Kode_Sparepart	Varchar	20	FK	Kode sparepart yang di retur
3	Jumlah_Retur	Varchar	20		Jumlah barang yang diretur
4	SubTotal_Retur	Varchar	20		Jumlah keseluruhan retur

Tabel 4.7 Struktur File penjualan

Nama File : penjualan

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	No_penjualan	Varchar	20	PK	No penjualan
2	Tgl_penjualan	Varchar	20		Tanggal penjualan
3	Total_penjualan	Varchar	20		Total harga penjualan

Tabel 4.8 Struktur File layanan

Nama File : layanan

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	Kode_layanan	Varchar	20	PK	Kode layanan untuk konsumen
2	Nama_layanan	Varchar	30		Nama layanan yang diinginkan kosumen
3	Jenis_layanan	Varchar	20		Jenis layanan berat/ringan
4	Harga_layanan	Varchar	20		Harga layanan sesuai jenisnya

Tabel 4.9 Struktur File pembelian

Nama File : pembelian

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	Kode_pembelian	Varchar	20	PK	Kode pembelian dari supplier
2	Tgl_pembelian	Varchar	30		Tgl pembelian
3	Supplier	Varchar	20		Nama supplier
4	Total_pembelian	Varchar	20		Jumlah yang dibelanjakan
5	Tgl_Terima	Varchar	20		Tanggal datangnya barang

Tabel 4.10 Struktur File sparepart

Nama File : sparepart

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	Kode_sparepart	Varchar	20	PK	Kode sparepart bengkel
2	Nama_sparepart	Varchar	30		Tgl pembelian
3	Harga_sparepart	Varchar	20		Nama supplier
4	Stock_sparepart	Varchar	5		Jumlah yang dibelanjakan
5	Kode_supplier	Varchar	20		Kode Supplier

Tabel 4.11 Struktur File transaksi

Nama File : transaksi

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	No_transaksi	Char	20	PK	No transaksi
2	Tgl_transaksi	Char	20		Tanggal transaksi
3	No_polisi	Char	20		Plat nomor konsumen
4	No_identitas	Char	20		No/nama identitas konsumen
5	Total_transaksi	Char	20	FK	Total harga penjualan

Tabel 4.12 Struktur File retur

Nama File : retur

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	Kode_Retur	Varchar	20	PK	Kode retur
2	Tgl_Retur	Varchar	20		Tanggal retur
3	Kode_Pembelian	Varchar	20	FK	Kode pembelian barang yang diretur
4	Total_Retur	Varchar	20		Total jumlah retur

Tabel 4.13 Struktur File Service

Nama File : service

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	No_Service	Varchar	20	PK	No service
2	Tgl_Service	Varchar	20		Tanggal tservice
3	No_Polisi	Varchar	10		No polisi konsumen
4	Total_Service	Varchar	20		Total harga service
5	Keluhan	Varchar	50		Keluhan konsumen
6	Montir	Varchar	30		Nama montir yang mengerjakan
7	Konsumen	Varchar	30		Nama konsumen
8	Type_Motor	Varchar	30		Tipe motor konsumen

Tabel 4.14 Struktur File Supplier

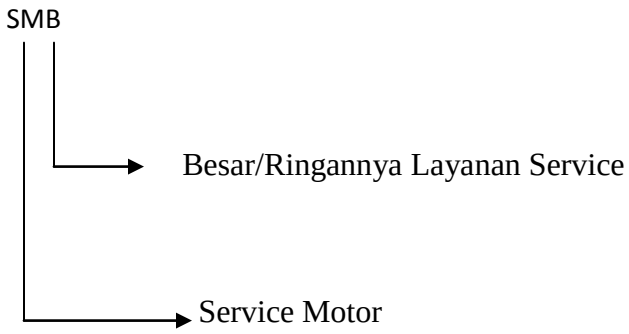
Nama File : supplier

No	Nama Field	Tipe Data	Size	Key	Keterangan
1	Kode_supplier	Varchar	20	PK	Kode supplier
2	Nama_supplier	Varchar	20		Nama supplier
3	Alamat_supplier	Varchar	10		Alamat supplier
4	Telp_supplier	Varchar	20		Nomor tlp supplier

4.2.4.5. Kodifikasi

Pengkodean digunakan untuk mengklasifikasikan data dan membantu dalam mengidentifikasi objek data.

1. Kode data layanan adalah 3 digit, yaitu untuk menentukan kode data layanan *service* yang akan dipilih.



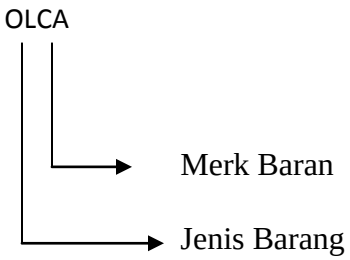
Contoh:

SMB

SM: Service Motor

B : Besar

2. Kode Data *Sparepart* adalah 4 digit, yaitu 2 digit pertama didepan untuk menentukan kode *data sparepart* , 2 digit terakhir menandakan nama merk barang.



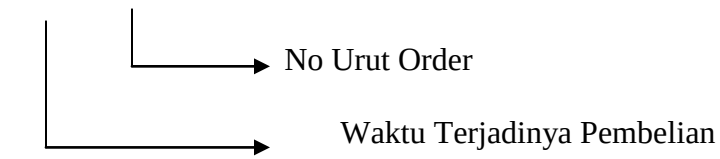
Contoh:

OLCA

OL: Produk Oli

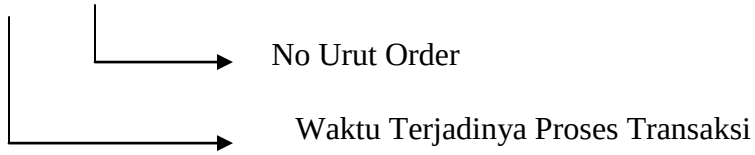
CA: Bermerk Castrol Activ

3. No data Pembelian adalah 10 digit, 6 digit awal adalah untuk menetukan thn/bln/tgl 4 digit terakhirnya menentukan nomor urut Format :



4. No data transaksi adalah 10 digit, 6 digit awal adalah untuk menetukan thn/bln/tgl 6 digit terakhirnya menentukan nomor urut.

   1206070001

5. Kode data *Supplier* adalah 4 digit, yaitu untuk menentukan kode data *supplier*.

   Contoh:

   PTJK

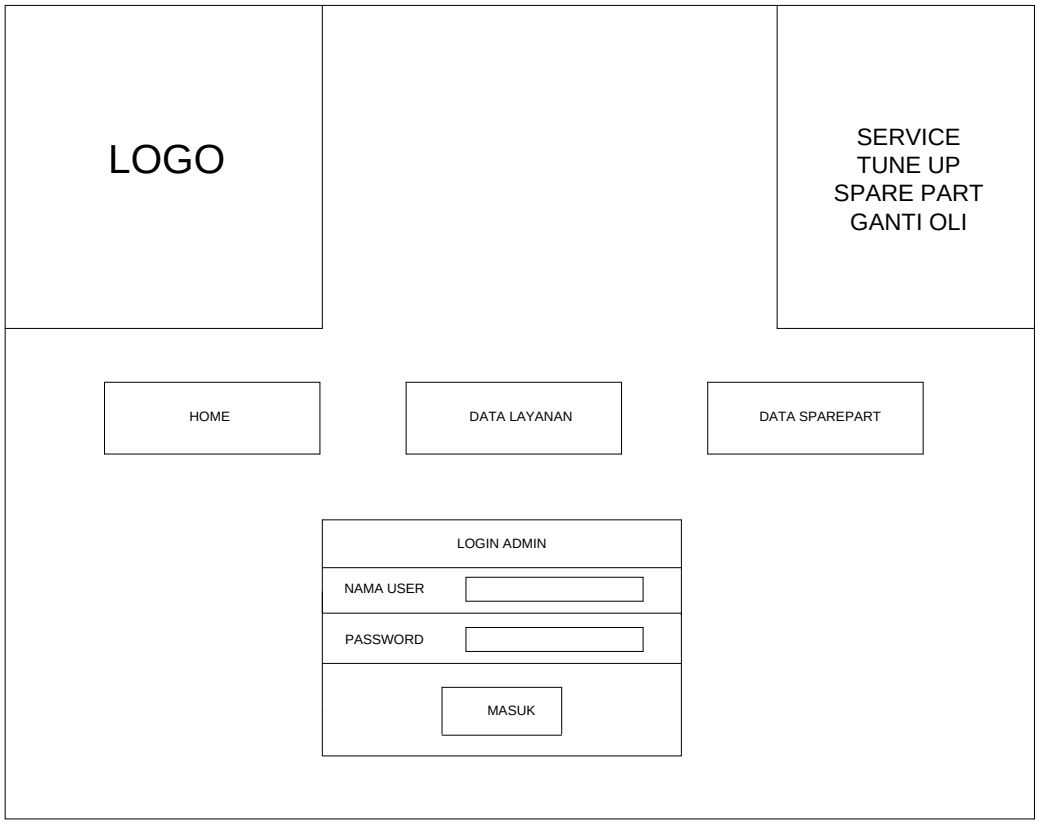
   SM: PT

   JK : Jaya Kusuma

4.2.5 Perancangan Antar Muka

Perancangan antarmuka merupakan tahap yang harus dikerjakan pada proses pembuatan sebuah program atau aplikasi. Antarmuka atau *interface* adalah suatu bagian yang berhubungan langsung dengan pengguna aplikasi. Perancangan

antarmuka bertujuan agar program atau aplikasi yang dihasilkan terlihat lebih menarik dan mudah dimengerti pada saat dioperasikan. Berikut ini adalah tampilan antarmuka yang dirancang pada sistem *website* penjualan dan pembelian *offline* bengkel Hel's Motor..



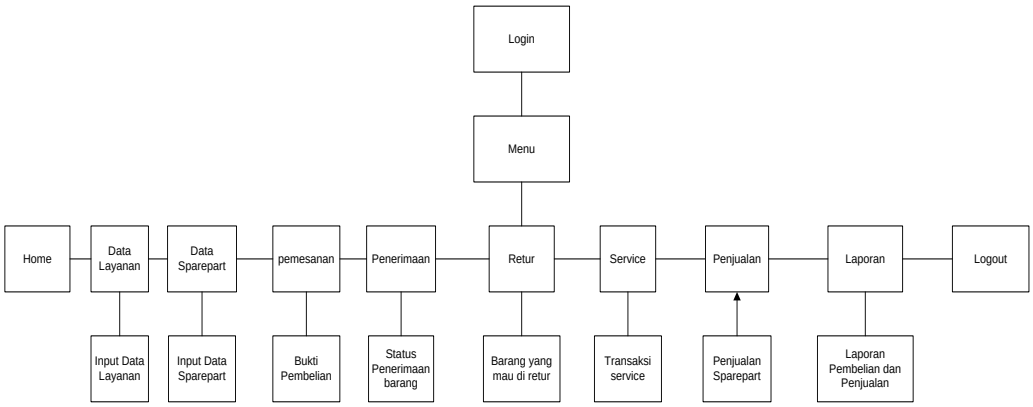
Gambar 4.9 Rancangan Antarmuka Website (*Admin*)

4.2.5.1. Struktur Menu

Struktur menu adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program komputer sehingga pada saat menjalankan program komputer, *user/admin* tidak mengalami kesulitan dalam

memilih menu - menu yang diinginkan. Pada perancangan ini dibuat menu yang dapat mengintegrasikan seluruh data dalam suatu sistem dan disertai dengan instruksi yang ada pada pilihan menu tersebut.

Adapun menu tersebut dapat dilihat pada gambar sebagai berikut:

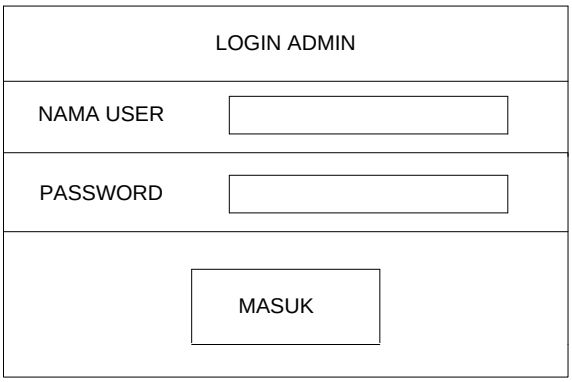


Gambar 4.10 Struktur Menu

4.2.5.2. Perancangan Input

Perancangan Input merupakan awal mula dari penerapan aplikasi untuk proses penginputan data. Untuk memperjelas bentuk dari perancangan input, berikut pemaparan dan fungsi dari setiap tampilan yang telah dibuat :

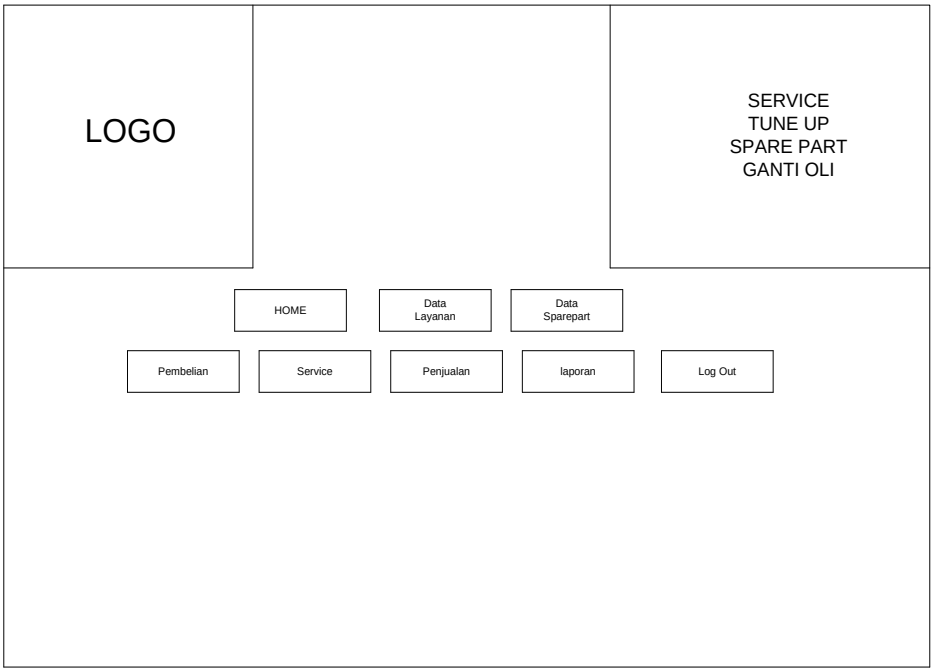
1. Login (User)



Gambar 4.11. Form Login

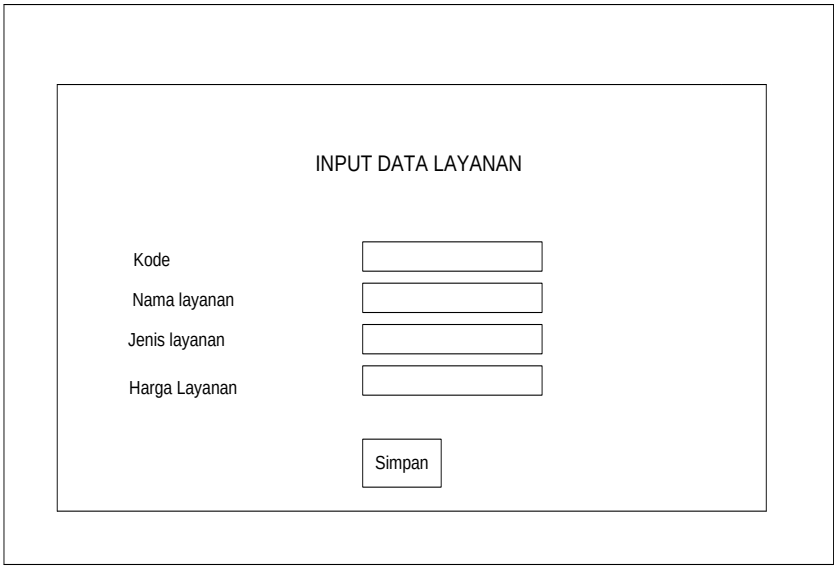
Login merupakan input yang dimasukkan oleh user berupa *username* dan *password*. Hanya *user/admin* yang dapat melakukan login.

2. Tampilan Utama



Gambar 4.12. Form Tampilan Utama

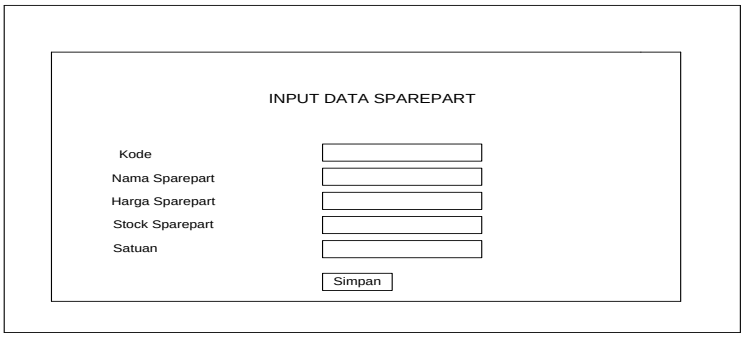
3. Input Data Layanan



Gambar 4.13. Form Data Layanan

Proses penginputan data layanan adalah memasukan layanan beserta harga apa saja yang akan dijual ke konsumen.

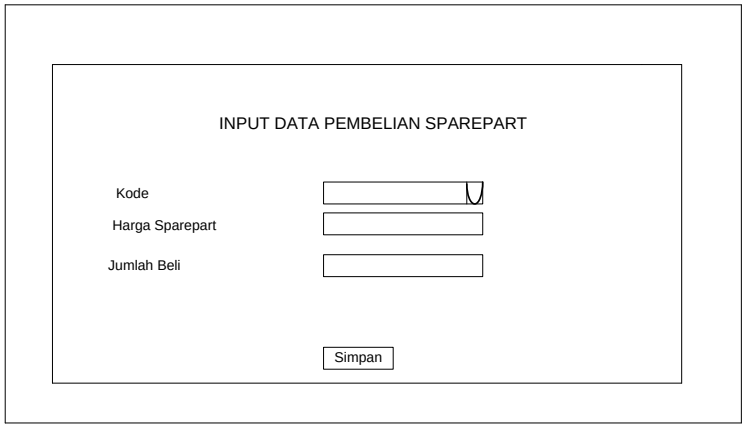
4. Input Data Sparepart



Gambar 4.14. Form Data Sparepart

Proses Input data sparepart adalah tempat menyimpan *stock* data barang bengkel yang tersedia.

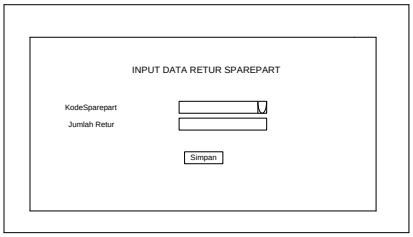
5. Pemesanan



Gambar 4.15. Form Data Pemesanan

Proses data pemesanan ini adalah untuk membuat pesanan barang yang akan dibeli kepada supplier.

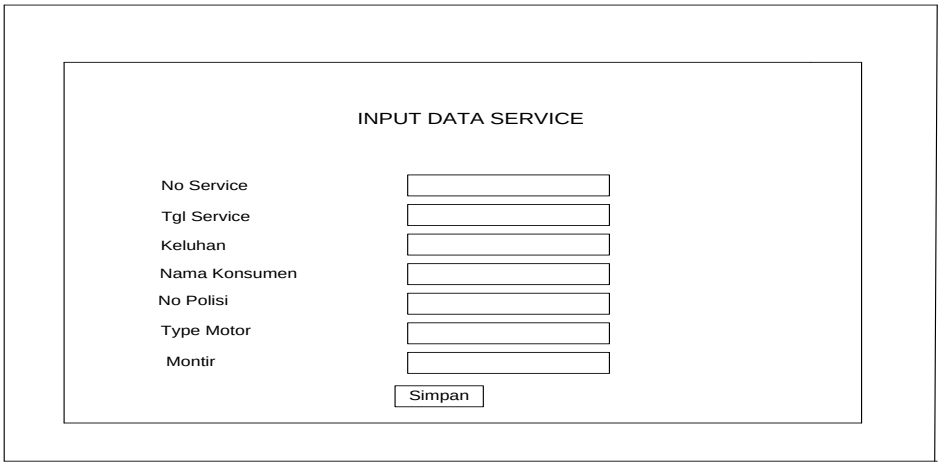
6. Retur



Gambar 4.16. Form Data Retur

Proses dimana admin melakukan input data barang yang akan di retur.

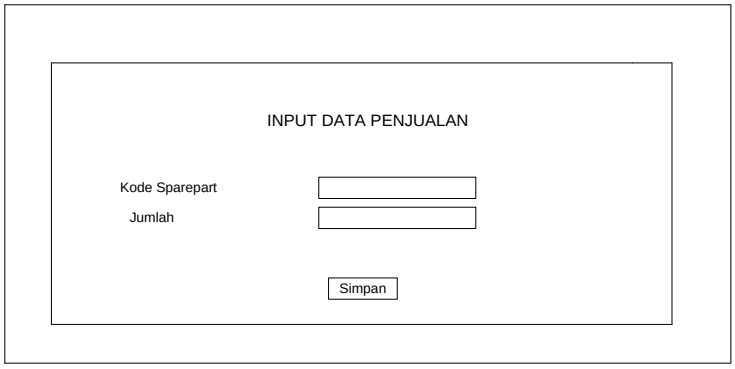
7. Service



Gambar 4.17. Form Data Service

Proses dimana admin melakukan input data konsumen.

8. Penjualan



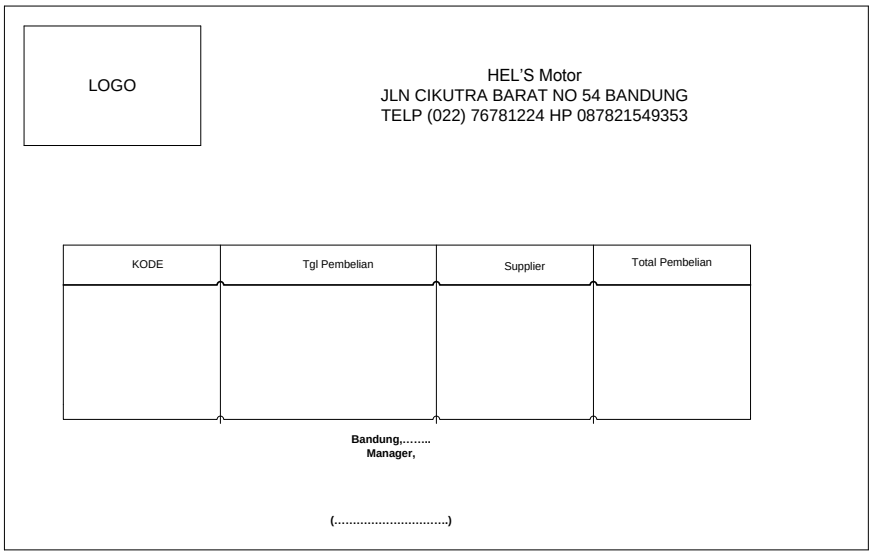
Gambar 4.18. Form Data Penjualan

Proses dimana admin melakukan input penjulan data sparepart kepada konsumen

4.2.5.3 Perancangan Output

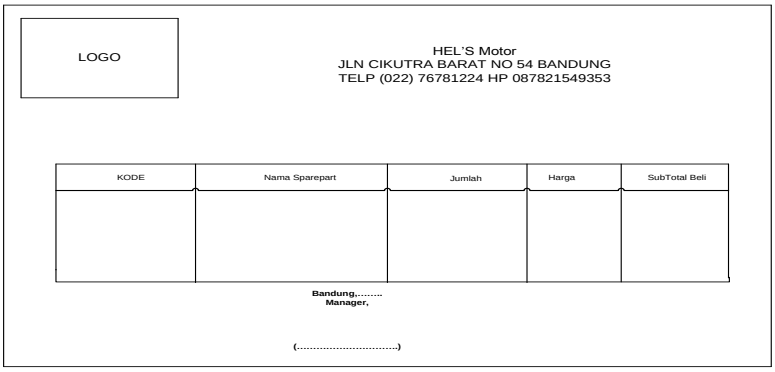
Seperti halnya rancangan input, rancangan output juga merupakan rencana pembuatan antarmuka untuk keluaran sebuah program. Berikut adalah *output* atau keluaran yang dirancang untuk sistem penjualan *offline* bengkel Hel's Motor.

1. Laporan Pembelian



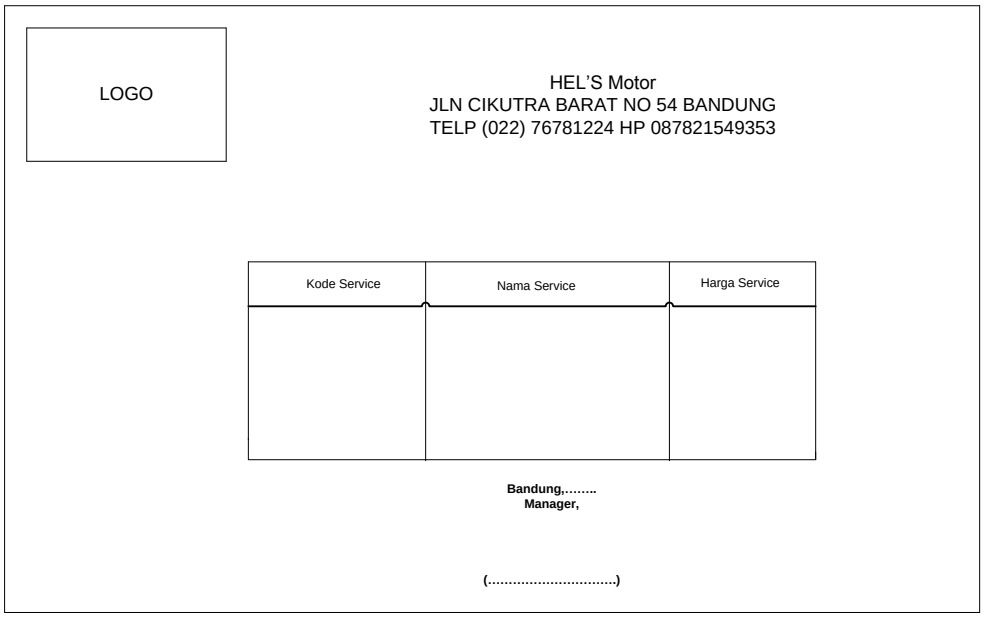
Gambar 4.19. Form laporan Pembelian

2. Laporan penjualan



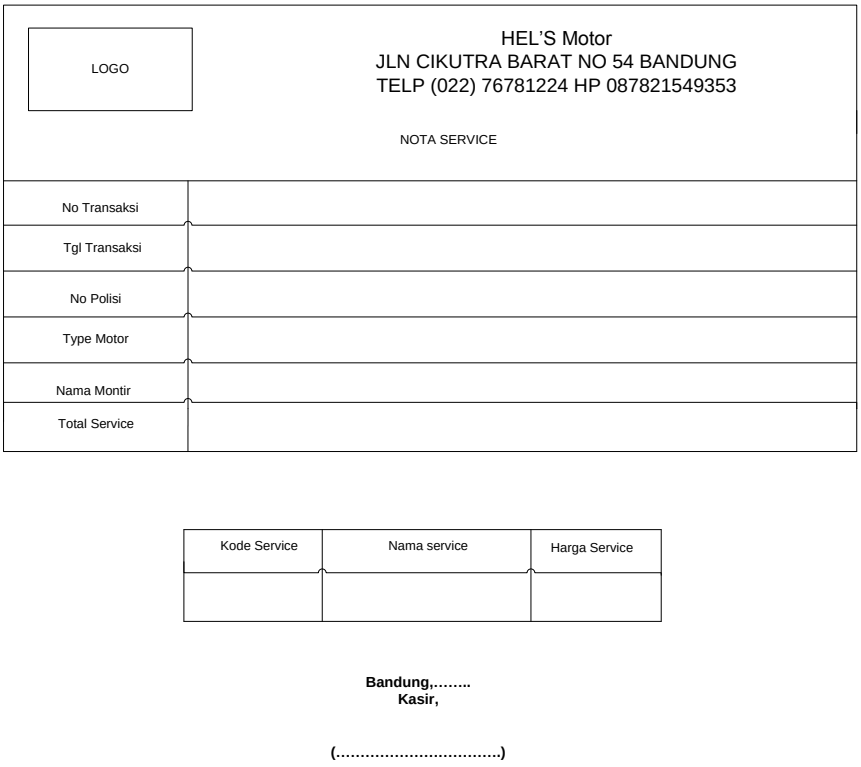
Gambar 4.20. Form laporan Penjualan

3. Laporan Service



Gambar 4.21. Form laporan Service

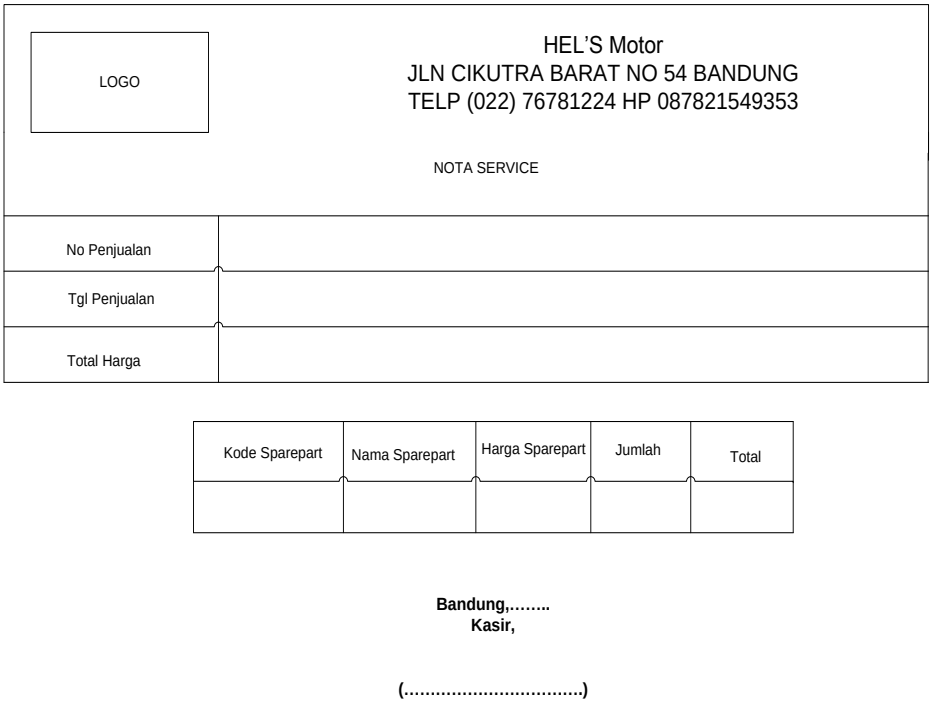
4. Nota Service



Gambar 4.22. Form Nota Service

Nota yang akan konsumen terima setelah melakukan transaksi servie motor.

5. Nota Penjualan

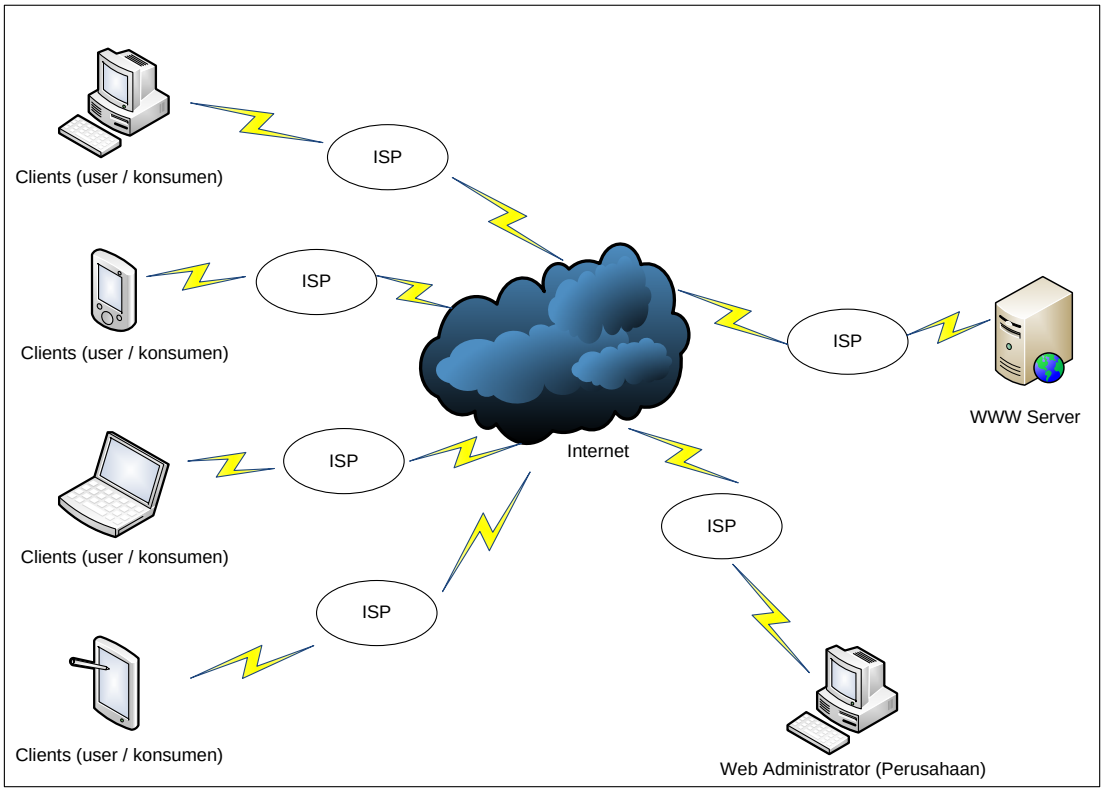


Gambar 4.23. Form Nota Penjualan

Nota yang akan konsumen terima setelah membeli sparepart.

4.2.6 Perancangan Arsitektur Jaringan

Arsitektur jaringan merupakan pemilihan topologi jaringan dan protokol yang tepat agar kinerja perangkat lunak yang dibangun berjalan maksimal. Pada aplikasi desktop arsitektur jaringan sangat mempengaruhi kinerja perangkat lunak. Sedangkan untuk aplikasi berbasis (*website*), arsitektur jaringan merupakan sesuatu yang optional karena *website* dapat dijalankan selama memiliki koneksi internet dan *internet browser*.



Gambar 4.24 Arsitektur Jaringan