

BAB IV

DESKRIPSI PEKERJAAN

4.1 Analisa Sistem

Berdasarkan hasil analisis sistem yang sedang berjalan saat ini pada CV. Langgeng Jaya, sistem yang ada di CV. Langgeng Jaya ini belum terkomputerisasi dengan baik. Oleh karena itu dirancanglah sebuah sistem yang merupakan hasil pembahasan mengenai sistem yang seperti apa yang diinginkan oleh CV. Langgeng Jaya, yaitu Aplikasi Penjualan yang menangani transaksi penjualan.

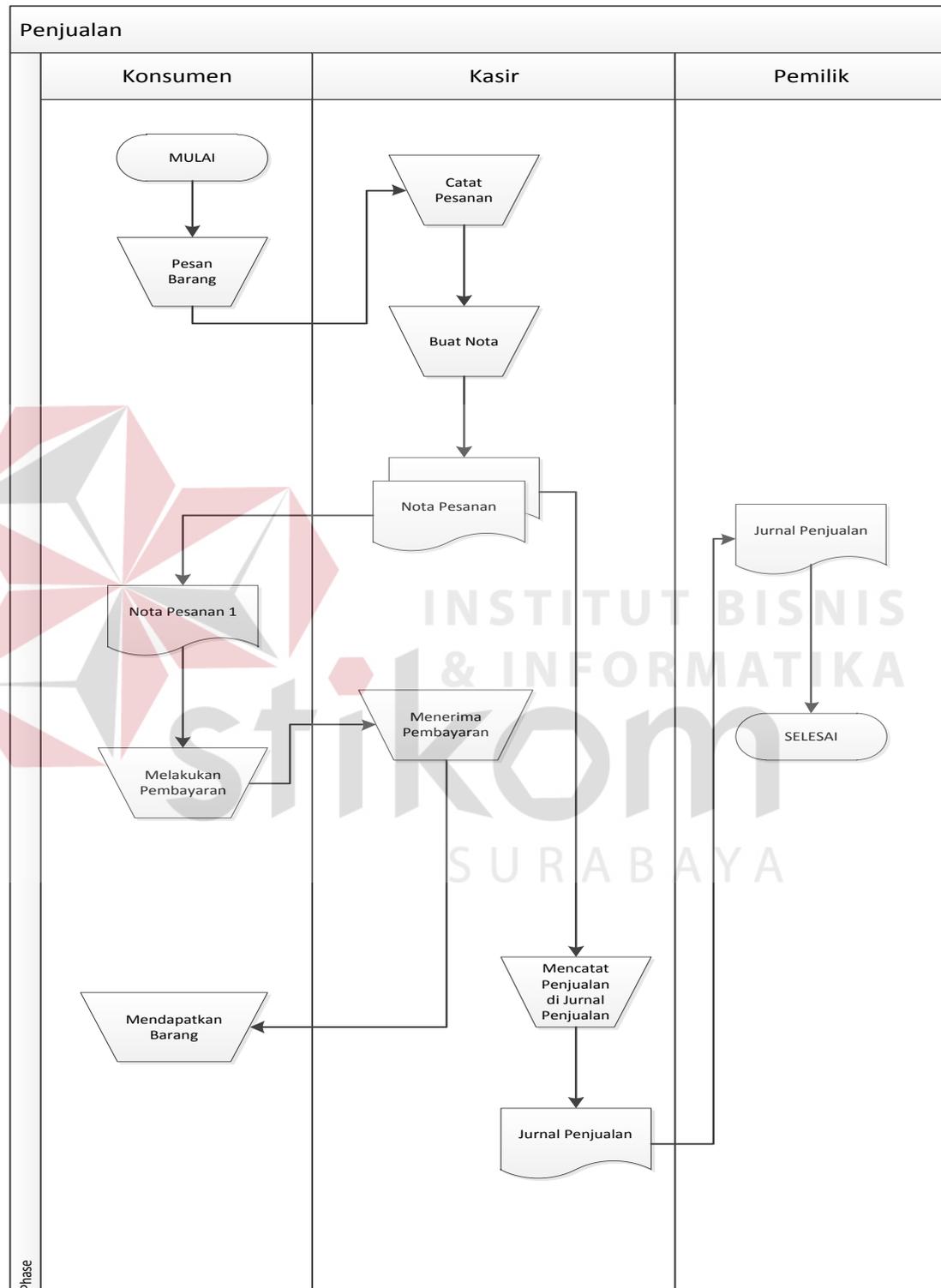
4.2 Perancangan Sistem

Perancangan sistem dimaksudkan untuk membantu menyelesaikan masalah pada sistem yang sedang berjalan saat ini sehingga dapat menjadi lebih baik dengan adanya sistem terkomputerisasi. Perancangan sistem menggunakan beberapa bahasa pemodelan untuk mempermudah analisa terhadap sistem. Pemodelan sistem yang digunakan adalah *Document Flow*, *System Flow*, *Data Flow Diagram*, dan *Entity Relationship Diagram*.

3.1 4.2.1 Document Flow

Berikut ini adalah *document flow* dari aplikasi penjualan minyak goreng. *Document flow* pertama menjelaskan tentang penjualan minyak goreng ke pada konsumen. Dimulai dari konsumen melakukan pemesanan minyak goreng ke bagian kasir, lalu bagian kasir melakukan pencatatan pemesanan barang konsumen dan membuat bukti dua pembayaran yang diserahkan konsumen dan

diarsipkan. Lalu bagian kasir membuat laporan yang akan diserahkan pemilik perusahaan. Untuk lebih detailnya dapat dilihat pada gambar 4.1.



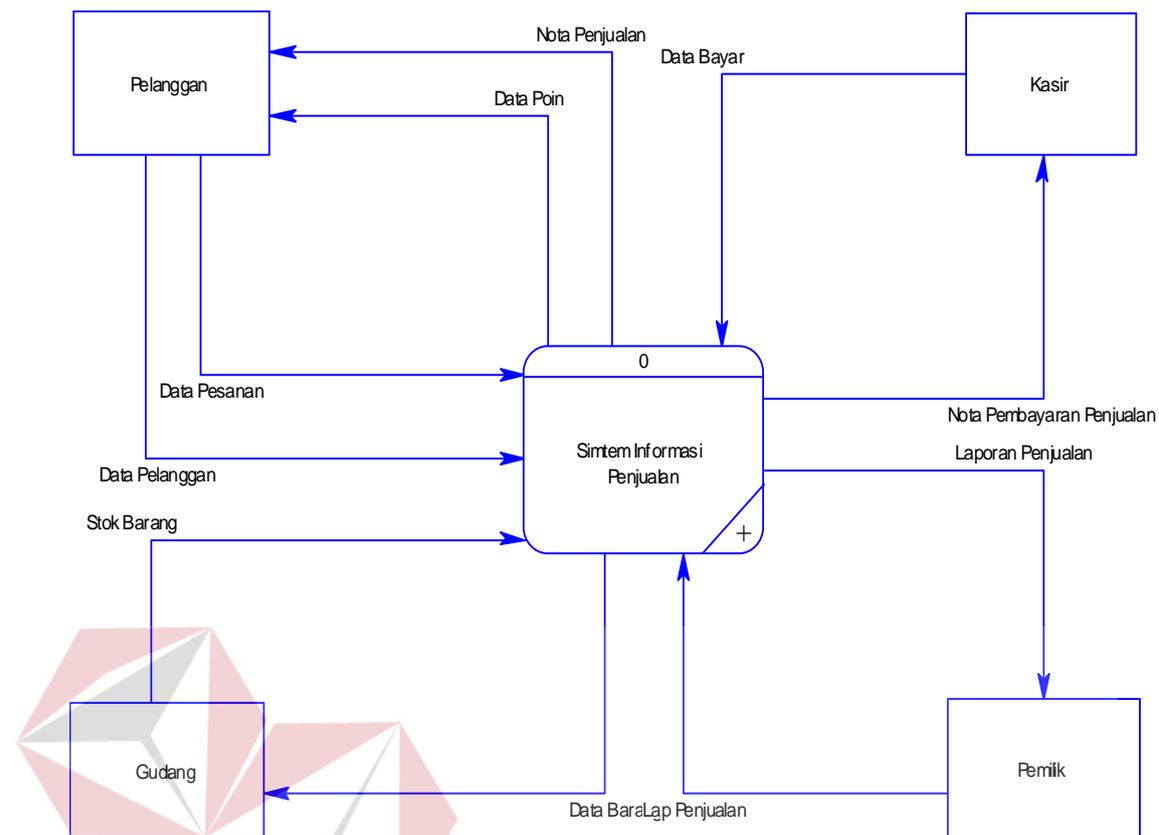
Gambar 4.1. Document Flow Penjualan

3.2 4.2.2 Data Flow Diagram

Data Flow Diagram (DFD) menggambarkan aliran data yang terjadi di dalam sistem, sehingga dengan dibuatnya *data flow diagram* ini akan terlihat arus data yang mengalir dalam sistem. *Data flow diagram* dimulai dari pembuatan *context diagram*, kemudian *data flow diagram* level 0 hingga level terendah dari proses yang dibutuhkan. Pada *data flow diagram* tergambar proses-proses yang berlangsung terhadap sistem. Selain itu, dapat juga terlihat *entity* luar yang berhubungan dengan sistem dengan memberikan *input* kepada sistem lalu sistem juga akan memberikan *output* kepada *entity* tersebut. Terdapat juga *data store* yang digunakan sistem untuk menyimpan dan membaca data untuk kebutuhan *input* dan *output* terhadap proses yang berjalan.

a. Context Diagram

Context diagram sistem informasi penjualan terdapat empat entitas yaitu : pelanggan, gudang, pemilik dan kasir. Masing – masing dari *entity* tersebut memberikan *input* dan sistem mengeluarkan *output* yang berupa laporan. Pada proses rancang bangun sistem informasi penjualan dimulai dari proses pemesanan dari pelanggan, kemudian sistem melakukan cek barang yang ada di gudang dan apabila barang tersebut ada di gudang maka akan terjadi proses penjualan. Untuk lebih detailnya dapat dilihat pada gambar 4.2

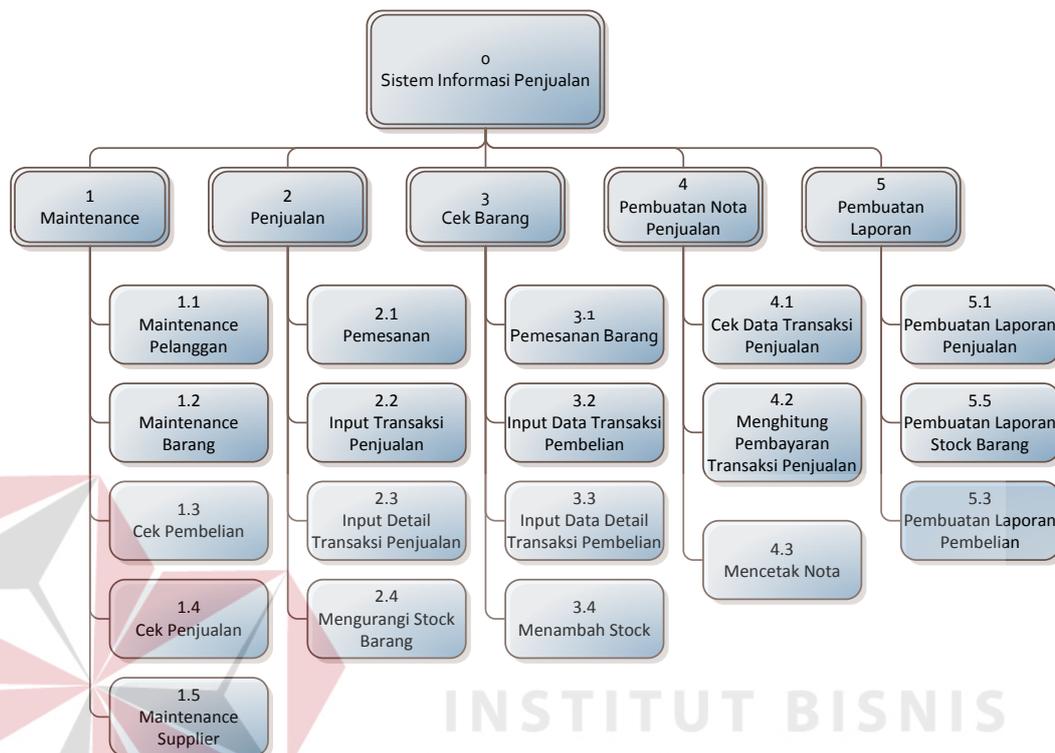


Gambar 4.2 Context Diagram Aplikasi Penjualan

b. Diagram Berjenjang

Diagram berjenjang merupakan alat perancangan sistem yang dapat menampilkan seluruh proses yang terdapat pada suatu aplikasi tertentu dengan jelas dan terstruktur. Pada rancangan sistem informasi penjualan barang ini terdiri dari 5 (lima) proses utama yaitu *maintenance*, penjualan, cek barang, pembuatan nota penjualan dan pembuatan laporan. Masing-masing dari proses tersebut akan dijabarkan kembali ke dalam beberapa sub proses. Dari diagram berjenjang berikut ini akan terlihat masing-masing sub *level* dari *Data Flow Diagram* (DFD). Seluruh proses yang terbentuk merupakan penjabaran dari masing-masing proses di atasnya dimana proses mengelola data master, melakukan transaksi dan mencetak laporan dapat diturunkan (*decomposition*)

lagi jadi beberapa sub proses. Adapun secara garis besar, diagram berjenjang yang membangun rancangan aplikasi digambarkan seperti pada Gambar 4.3.



Gambar 4.3 Diagram Berjenjang Penjualan

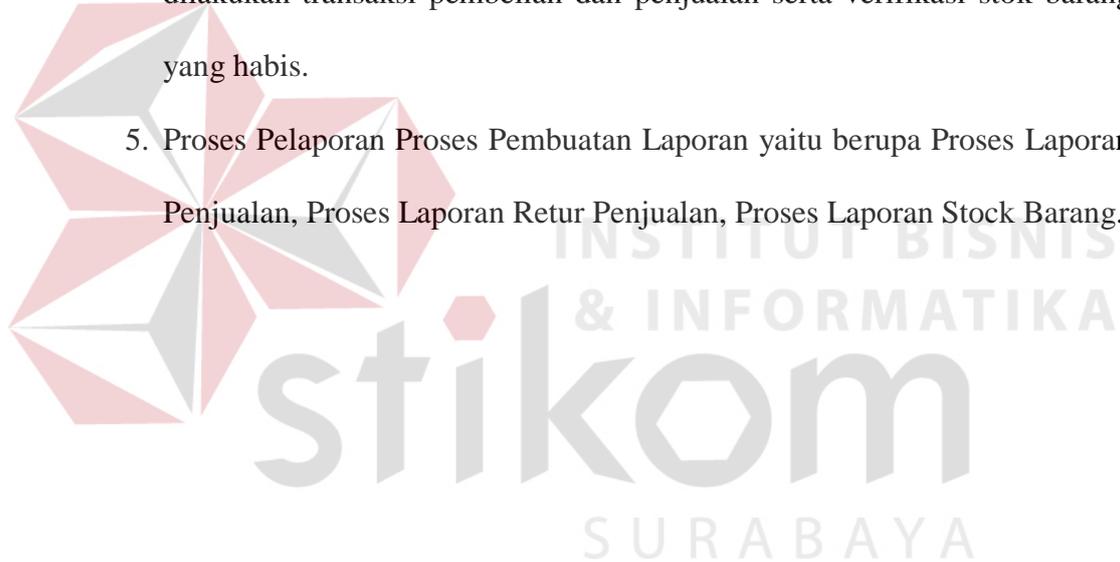
c. Data Flow Diagram Level 0

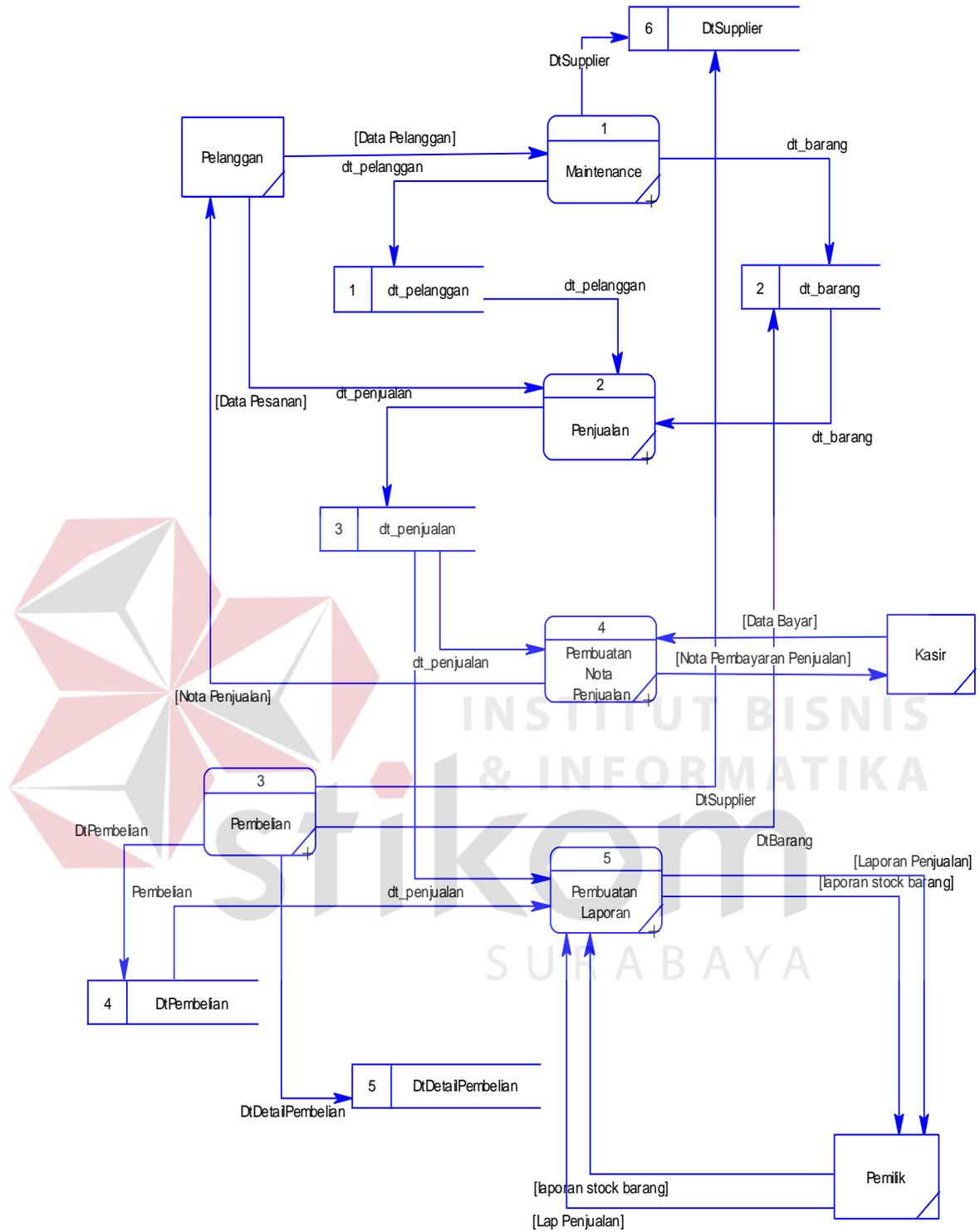
Setelah *Context Diagram* sistem informasi penjualan

didekomposisikan maka akan didapat DFD level 0 yang terdiri dari lima subproses, yaitu :

1. Proses *maintenance* proses ini digunakan untuk input pelanggan, *input* data *supplier*, data barang, data pelanggan, cek penjualan dan cek pembelian. Dimana data-data ini digunakan sebagai masukan pada proses lainnya. Dalam proses ini yang terlibat entitas *supplier* dan pelanggan.

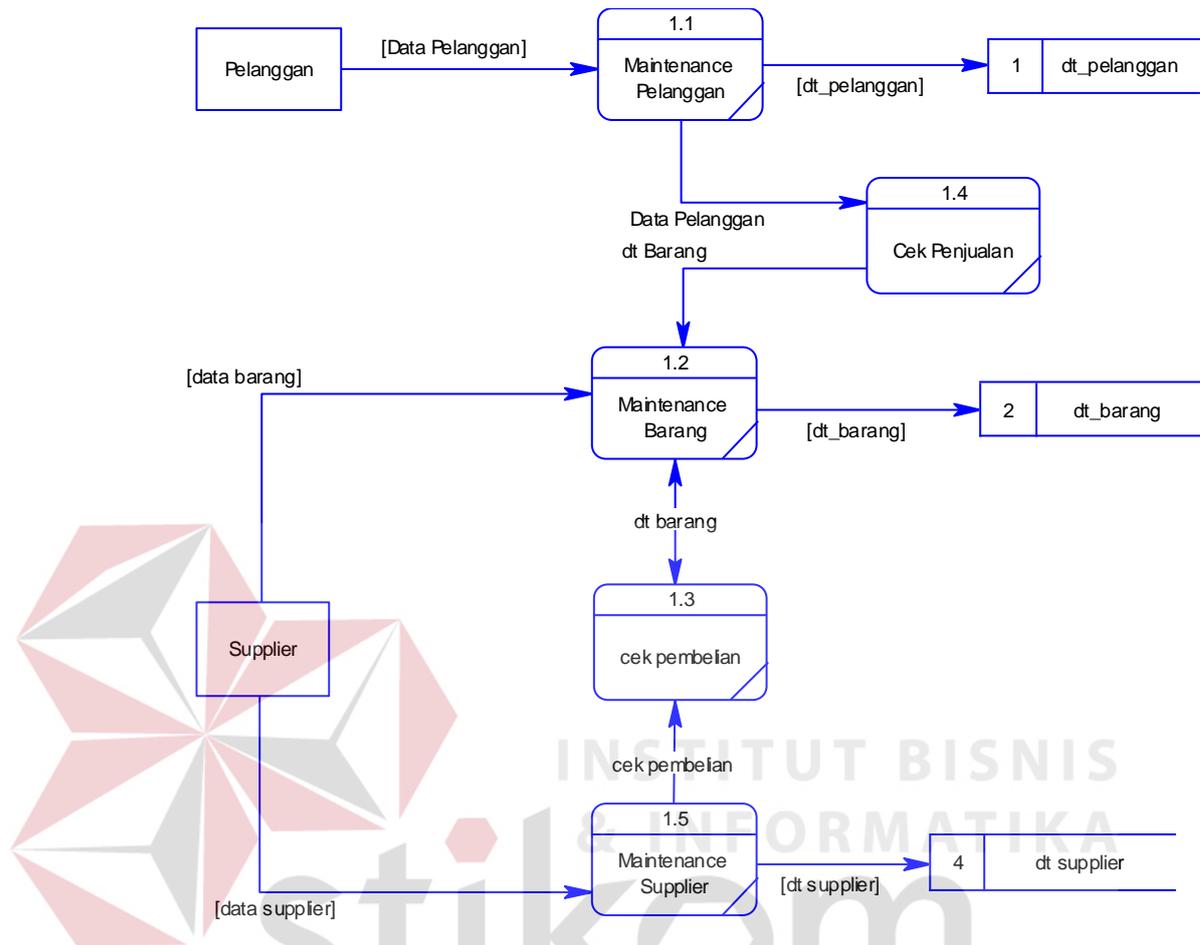
2. Proses Transaksi Penjualan, entitas dalam proses ini adalah pelanggan memberukan inputan berupa data pesanan yang akan dibeli. Dari data-data tersebut sistem memproses transaksi penjualan ke pada kasir. Kemudian sistem memproses pemesanan dari kasir ke bagian gudang, kemudian bagian pembelian melakukan pengurangan stok barang.
3. Proses pembuatan nota pembayaran pemesanan merupakan proses untuk mencetak nota pembayaran per transaksi penjualan.
4. Proses cek stok barang merupakan proses perhitungan stok sebelum dilakukan transaksi pembelian dan penjualan serta verifikasi stok barang yang habis.
5. Proses Pelaporan Proses Pembuatan Laporan yaitu berupa Proses Laporan Penjualan, Proses Laporan Retur Penjualan, Proses Laporan Stock Barang.





Gambar 4.4. Data Flow Diagram Level 0

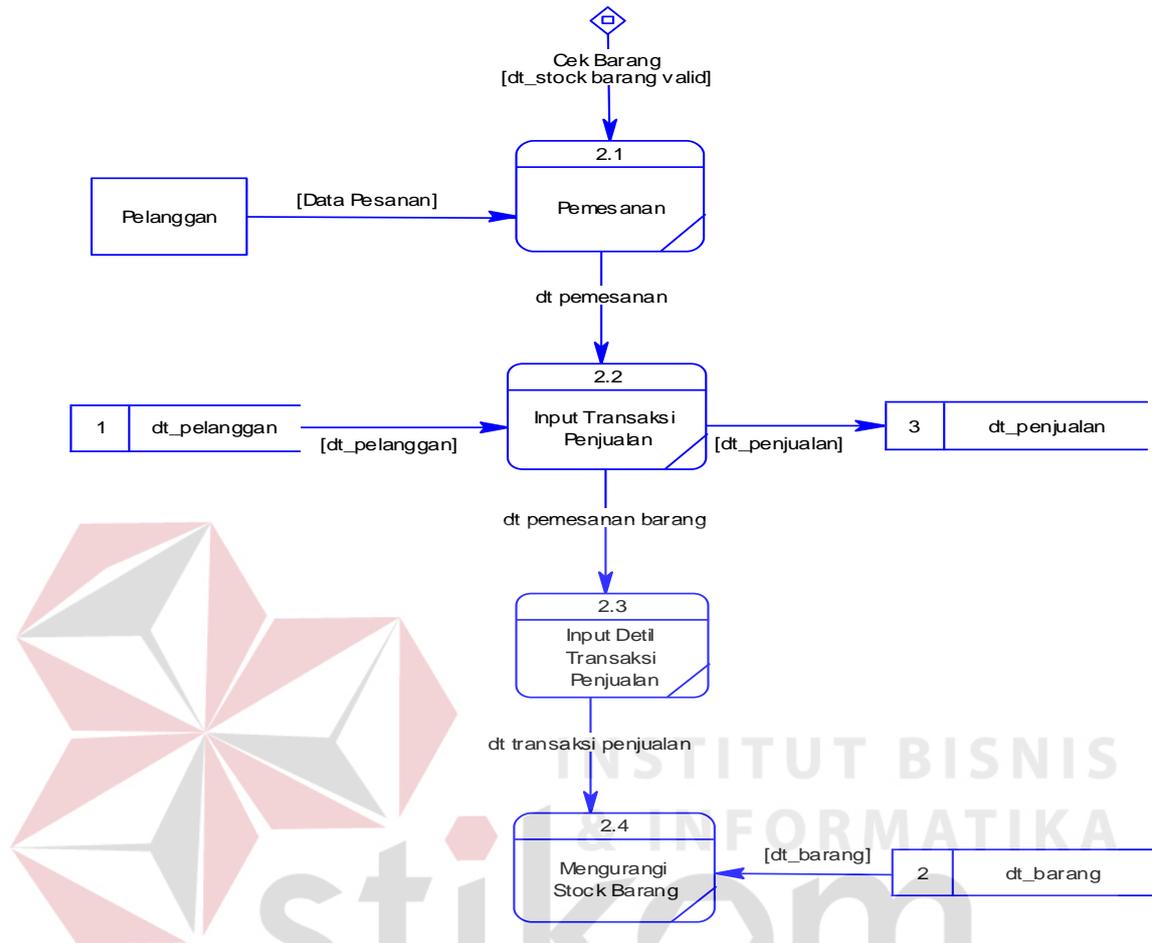
d. DFD Level 1 Sub Proses Maintenance



Gambar 4.5. DFD Level 1 Subproses Maintenance

Pada DFD Level 1 subproses *maintenance* data terdapat lima proses, yaitu *maintenance* data pelanggan, *maintenance* data supplier, *maintenance* data barang, mengecek penjualan dan mengecek pembelian. Dalam proses pengolahan data awal yang dimaksud adalah pengolahan untuk mengisi tabel master. Data-data master yang telah terbentuk tetap dapat dilakukan *update* sewaktu-waktu atas isinya melalui proses update data.

e. DFD Level 1 Subproses Transaksi Penjualan



Gambar 4.6. DFD Level 1 Subproses Transaksi Penjualan

Berikut merupakan subsistem transaksi penjualan. Terdapat beberapa proses di dalam subsistem transaksi penjualan yaitu :

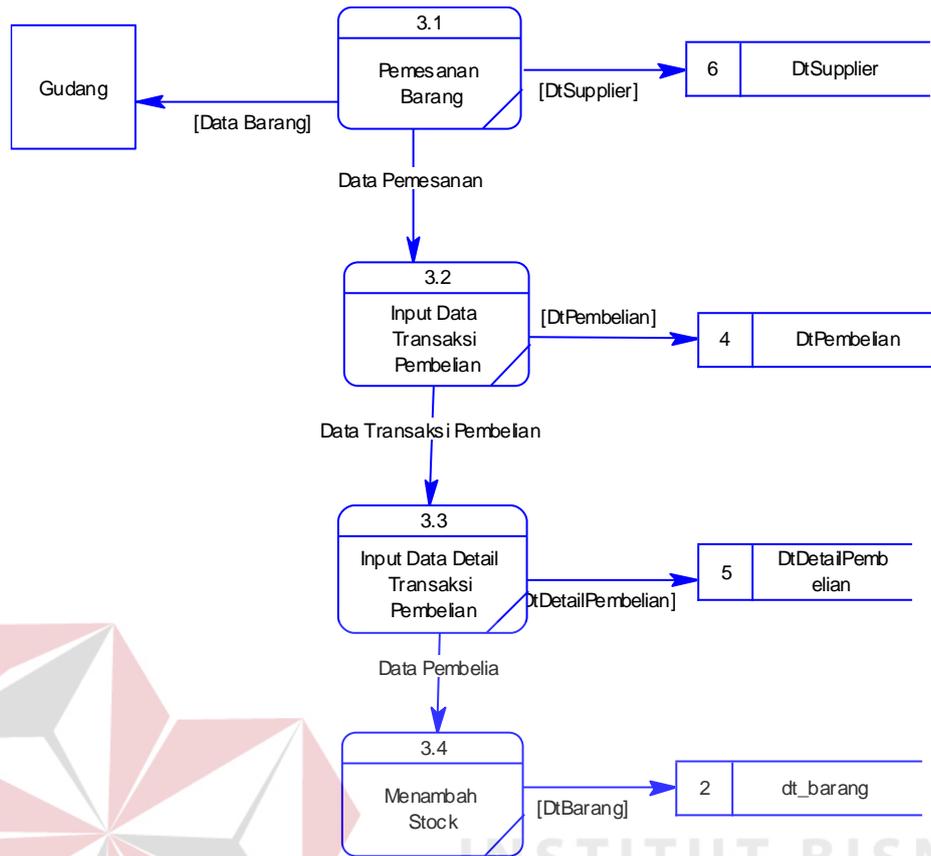
1. Proses pemesanan barang merupakan proses yang mencaatat pemesanan pelanggan kemudian mengirimkannya ke bagian penjualan. Dalam proses ini terdapat proses pemesanan dan proses penyampaian data pesanan penjualan

2. Proses *input* transaksi penjualan merupakan proses yang mengelola data transaksi penjualan untuk diproses dan disimpan dalam *data store* data transaksi penjualan.
3. Proses *input* detail transaksi penjualan merupakan proses yang mengelola inputen data detail transaksi penjualan untuk diproses dan disimpan dalam *data store* data detail transaksi penjualan
4. Proses pengurangan stok barang merupakan proses pengurangan stok barang berdasarkan pesanan barang dari pelanggan

f. DFD Level 1 Subproses Pembelian

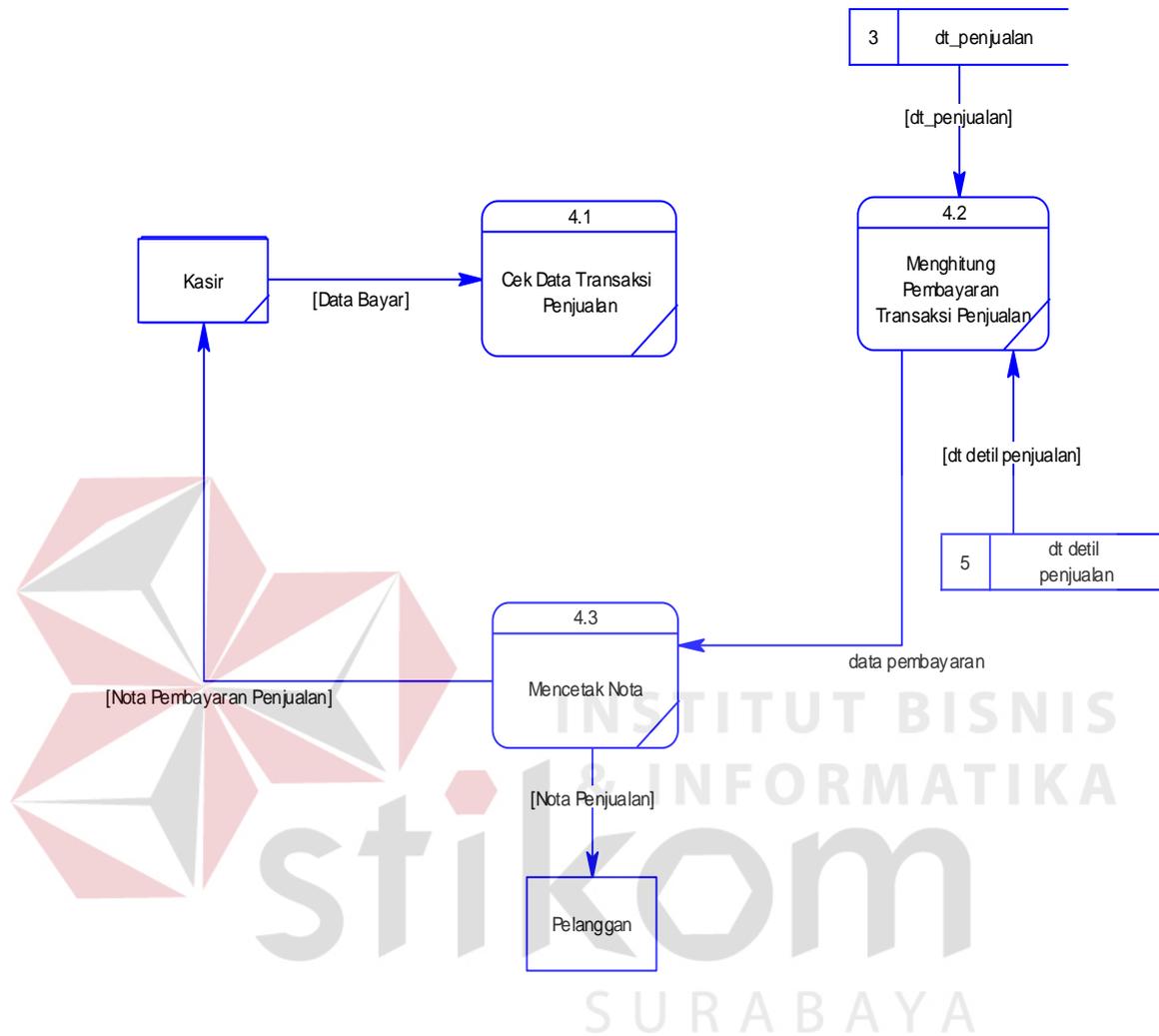
Berikut merupakan subsistem transaksi pembelian barang. Terdapat empat proses di dalam subsistem pembelian barang yaitu :

1. Proses pemesanan barang merupakan proses yang menentukan dan jenis jumlah stok barang yang dipesan kepada *supplier*.
2. Proses *input* data transaksi pembelian stok barang merupakan proses yang mengelola data transaksi pembelian stok barang untuk diproses dan disimpan dalam *data store* transaksi pembelian barang.
3. Proses *input* data detail transaksi pembelian merupakan proses yang mengelola data detail transaksi pembelian bahan baku diproses dan disimpan dalam *data store* data detail transaksi pembelian.
4. Proses penambahan stok barang merupakan proses penambahan jumlah stok barang berdasarkan barang yang dibeli. Pada proses ini terjadi stok masuk ke gudang sehingga stok barang bertambah sesuai dengan jumlah pembelian barang.



Gambar 4.7. DFD Level 1 Subproses Pembelian

g. DFD Level 1 Subproses Pembuatan Nota Penjualan



Gambar 4.8. DFD Level 1 Subproses Pembuatan Nota Penjualan

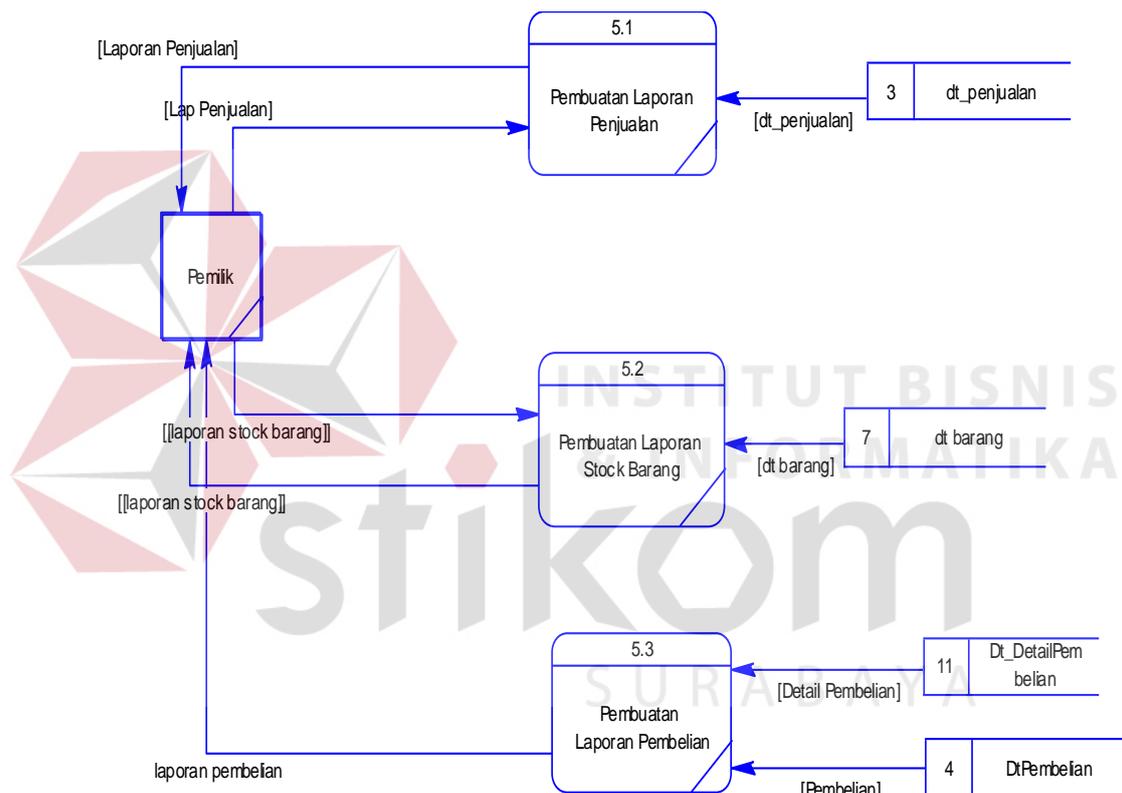
Subsistem pembuatan nota. Terdapat tiga proses di dalam subsistem pembuatan laporan yaitu :

1. Proses cek data transaksi penjualan merupakan proses cek kesamaan data penjualan dengan data pesanan
2. Proses menghitung pembayaran transaksi penjualan merupakan proses yang menghitung jumlah pesanan barang yang di pesan pelanggan. Hasil perhitungan juga termasuk diskon

3. Proses mencetak nota pembayaran pemesanan merupakan proses yang menghitung stok setelah proses transaksi penjualan

h. DFD Level 1 Subproses Pembuatan Laporan

Berikut ini adalah DFD level 1 sub proses pembuatan laporan yang terdiri dari beberapa proses yaitu yaitu pembuatan laporan penjualan, pembuatan laporan stok barang dan pembuatan laporan pembelian.



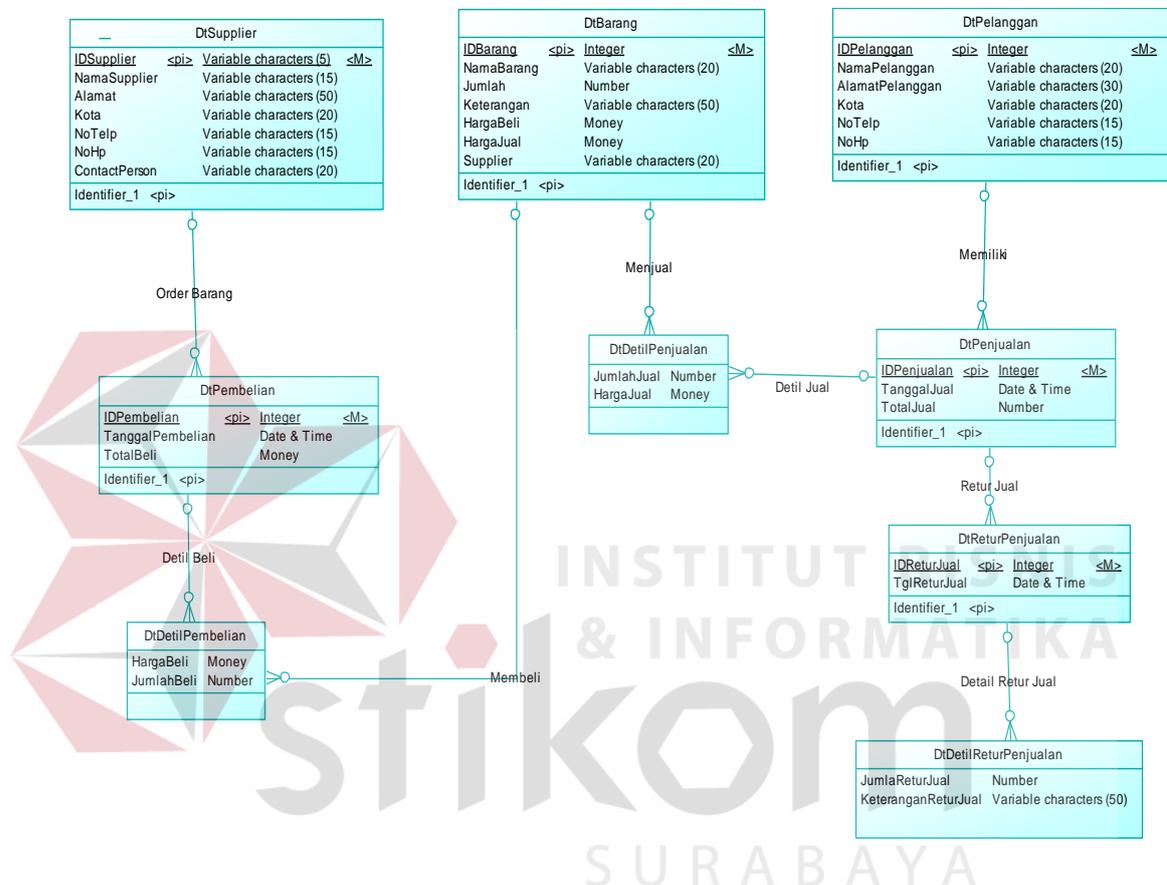
Gambar 4.9. DFD Level 1 Subproses Pembuatan Laporan

3.3 4.2.3 Entity Relationship Diagram

Entity relationship diagram (ERD) dari Aplikasi Penjualan yang terdiri dari *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM) dijelaskan pada gambar 4.24 dan gambar 4.25.

a. *Conceptual Data Model (CDM)*

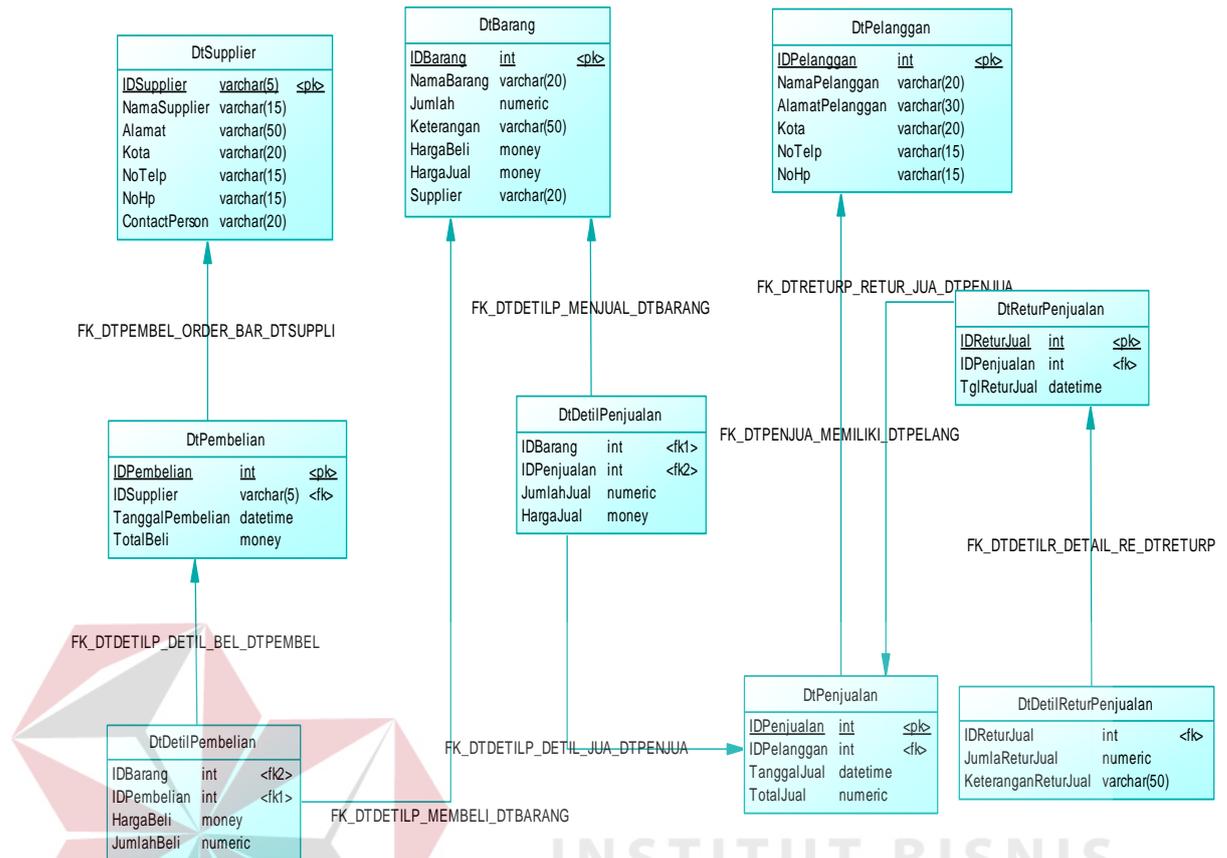
Sebuah *Conceptual Data Model (CDM)* menggambarkan secara keseluruhan konsep struktur basis data yang di rancang untuk aplikasi sebagaimana terlihat dibawah ini.



Gambar 4.10. Conceptual Data Model Aplikasi Penjualan

b. *Physical Data Model (PDM)*

Sebuah *Physical Data Model (PDM)* menggambarkan secara detail konsep rancangan struktur basis data yang dirancang untuk suatu aplikasi. PDM merupakan hasil generate dari CDM. Pada PDM tergambar jelas tabel-tabel penyusun basis data beserta *field-field* yang terdapat pada setiap tabel sebagai mana pada gambar di bawah ini.



Gambar 4.11. Physical Data Model Aplikasi Penjualan

3.4 4.2.4 Struktur Tabel

Suatu rancangan *database* harus disesuaikan dengan DFD dan ERD yang telah dibuat, dimana *database* tersebut harus sesuai dengan kebutuhan informasi yang diperlukan oleh *user*.

Adapun tabel-tabel yang digunakan dalam aplikasi ini adalah sebagai berikut :

a. Nama Tabel : IDSupplier

Primary Key : IDSupplier

Foreign Key : -

Fungsi : Untuk menyimpan data Supplier.

Tabel 4.1. Tabel Supplier

| No. | Field | Type | Length | Description |
|-----|---------------|---------|--------|-----------------|
| 1 | IDSupplier | Int | | Identitas |
| 2 | NamaSupplier | Varchar | | Nama Supplier |
| 3 | Alamat | Varchar | 50 | Alamat Supplier |
| 4 | Kota | varchar | 20 | Asal Kota |
| 5 | NoTelp | varchar | 15 | Komunikasi |
| 6 | NoHp | varchar | 15 | Komunikasi |
| 7 | ContactPerson | varchar | 15 | Komunikasi |

b. Nama Tabel : Pelanggan

Primary Key : IDPelanggan

Foreign Key : -

Fungsi : Untuk menyimpan data Pelanggan.

Tabel 4.2. Tabel Pelanggan

| No. | Field | Type | Length | Description |
|-----|---------------|---------|--------|------------------|
| 1 | IDPelanggan | Int | | Identitas |
| 2 | NamaPelanggan | Varchar | 20 | Nama Pelanggan |
| 3 | Alamat | Varchar | 30 | Alamat Pelanggan |
| 4 | Kota | Varchar | 20 | Asal Kota |
| 5 | NoTelp | varchar | 15 | Komunikasi |
| 6 | NoHp | Varchar | 15 | Komunikasi |

c. Nama Tabel : Barang

Primary Key : IDBarang

Foreign Key : -

Fungsi : Untuk menyimpan Data Barang.

Tabel 4.3. Tabel Barang

| No. | Field | Type | Length | Description |
|-----|------------|---------|--------|------------------|
| 1 | IDBarang | Int | | Identitas Barang |
| 2 | NamaBarang | Varchar | 20 | Nama Barang |
| 3 | Jumlah | Numeric | | Jumlah Barang |
| 4 | Keterangan | varchar | 50 | Keterangan |
| 5 | HargaBeli | money | | Harga Beli |
| 6 | HargaJual | money | | Harga Jual |
| 7 | Supplier | Varchar | 20 | Nama Supplier |

d. Nama Tabel : Penjualan

Primary Key : IDPenjualan, IDPelanggan

Foreign Key : -

Fungsi : Untuk menyimpan Data Penjualan Barang.

Tabel 4.4. Tabel Penjualan

| No. | Field | Type | Length | Description |
|-----|-------------|---------|--------|---------------------|
| 1 | IDPenjualan | Int | | Identitas Penjualan |
| 2 | IDPelanggan | Int | | Identitas Pelanggan |
| 3 | TglJual | Date | | Tanggal |
| 4 | TotalJual | Numeric | | Total Bayar |

e. Nama Tabel : Detil Penjualan

Primary key : IDPenjualan, IDBarang

Foreign Key : -

Fungsi : Untuk menyimpan semua Detail Penjualan.

Tabel 4.5. Tabel Detail Penjualan

| No. | Field | Type | Length | Description |
|-----|-------------|---------|--------|---------------------------|
| 1 | IDPenjualan | int | | Identitas Penjualan |
| 2 | IDBarang | int | | Identitas Barang |
| 3 | IDPenjualan | int | | Identitas Penjualan |
| 4 | JumlahJual | numeric | | Jumlah Barang yang dibeli |
| 5 | Harga | money | | Harga Barang |

f. Nama Tabel : Pembelian

Primary Key : IDPembelian, IDSupplier

Foreign Key :

Fungsi : Untuk menyimpan Data Pembelian.

Tabel 4.6. Tabel Pembelian

| No. | Field | Type | Length | Description |
|-----|------------------|------|--------|--------------------|
| 1 | IDPembelian | int | | Identitas Pembeli |
| 2 | IDSupplier | int | | Identitas Supplier |
| 3 | TanggalPembelian | date | | Tanggal Pembelian |
| 4 | TotalBeli | int | | Total Bayar |

g. Nama tabel : Detail Pembelian

Primary Key : IDPembelian, IDBarang, IDSupplier

Foreign Key : -

Fungsi : Untuk menyimpan Data Detail Pembelian.

Tabel 4.7. Tabel Detail Pembelian

| No. | Field | Type | Length | Description |
|-----|-------------|---------|--------|-------------------------|
| 1 | IDPembelian | int | | Identitas Pembelian |
| 2 | IDBarang | int | | Identitas Barang |
| 3 | IDSupplier | int | | Identitas Supplier |
| 4 | HargaBeli | money | | Harga Beli |
| 5 | JumlahBeli | numeric | | Jumlah Barang yg dibeli |

h. Nama tabel : Retur Penjualan

Primary Key : IDRetuPenjualan, IDPenjualan

Foreign Key : -

Fungsi : Untuk menyimpan Retur Penjualan.

Tabel 4.8. Tabel Retur Penjualan

| No. | Field | Type | Length | Description |
|-----|------------------|------|--------|---------------------------|
| 1 | IDReturPenjualan | int | | Identitas Retur Penjualan |
| 2 | IDPenjualan | int | | Identitas Penjualan |
| 3 | IDPelanggan | int | | Identitas Pelanggan |
| 4 | Tanggal | Date | | Tanggal |

i. Nama Tabel : Detil Retur Penjualan

Primary Key : IDReturJual, IDPenjualan, IDBarang

Foreign Key : -

Fungsi : Untuk menyimpan data detil retur penjualan.

Tabel 4.9. Tabel Detil Retur Penjualan

| No. | Field | Type | Length | Description |
|-----|---------------------|---------|--------|----------------------------|
| 1 | IDReturJual | int | | Identitas Retur |
| 2 | IDPenjualan | int | | Identitas Penjualan |
| 3 | IDBarang | int | | Identitas Barang |
| 4 | JumlahReturJual | numeric | | Jumlah Barang yang diretur |
| 5 | KeteranganReturJual | varchar | | Keterangan barang retur |

3.5 4.2.5 Desain I/O (Input/Output)

Berikut ini merupakan rancangan desain *input/output* dari Sistem Informasi Penjualan pada CV. Langgeng Jaya.

a. Form Menu Utama

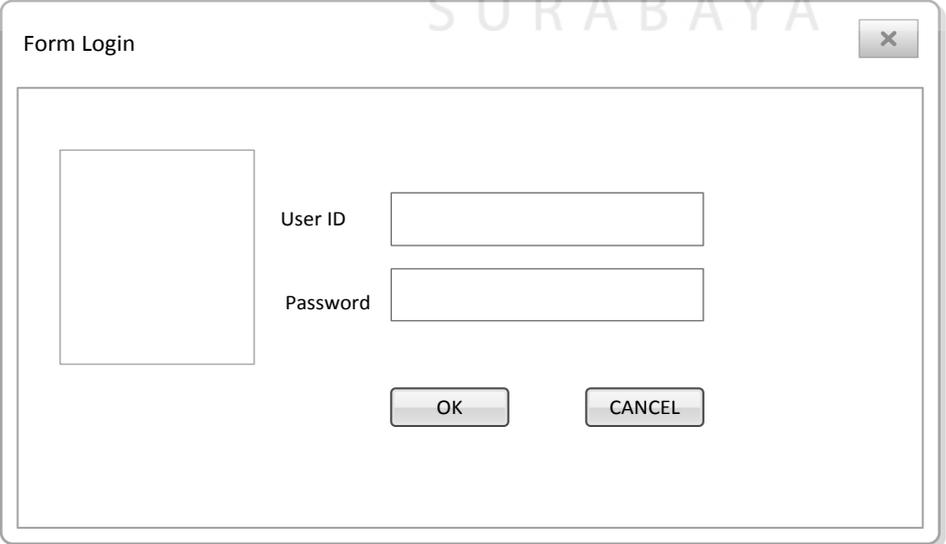
Form Menu Utama merupakan *form* induk dari aplikasi yang dibuat, dimana seluruh *form* yang lain dapat dipanggil melalui *form* tersebut. Desain Form Menu Utama dapat dilihat pada gambar 4.12.



Gambar 4.12. Desain Form Menu Utama

b. Form Login

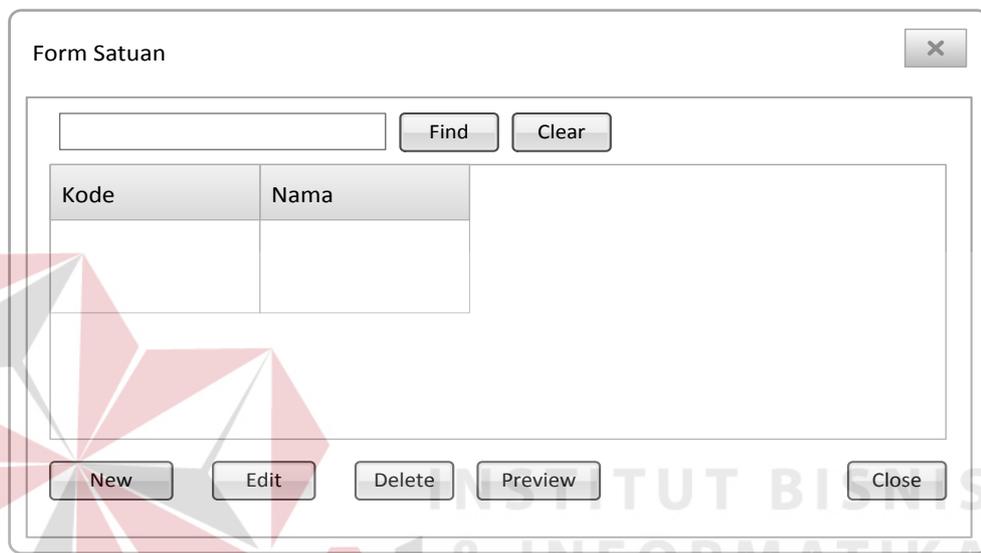
Form Login digunakan sebagai autentifikasi *user* yang ingin masuk ke dalam sistem, dimana *user* dibagai ke dalam 2 *group*, yaitu : Administrator, dan Manajer. Masing-masing *group* memiliki hak akses yang berbeda-beda. Desain Form Login dapat dilihat pada gambar 4.13.



Gambar 4.13. Desain Form Login

c. Form Satuan

Form Satuan digunakan untuk menambah, mengubah, dan menghapus data kode dan nama satuan. Desain Form Satuan mempunyai lima *button* yaitu *new*, *edit*, *delete*, *preview* dan *close*. Form Satuan dapat dilihat pada Gambar 4.14.



| Kode | Nama |
|------|------|
| | |

Gambar 4.14. Desain Form Satuan

d. Form Kategori

Form Kategori digunakan untuk menambah, mengubah, dan menghapus data kode, nama satuan dan keterangan. Desain Form Kategori mempunyai lima *button* yaitu *new*, *edit*, *delete*, *preview* dan *close*. Form Kategori dapat dilihat pada Gambar 4.15.

Form Kategori

Find Clear

| Kode | Nama | Keterangan |
|------|------|------------|
| | | |

New Edit Delete Preview Close

Gambar 4.15. Desain Form Kategori

e. Form Kontak

Form Kontak digunakan untuk menambah, mengubah, dan menghapus data kontak *Supplier*, *Customer*, Pegawai. Desain Form Kontak mempunyai lima *button* yaitu *new*, *edit*, *delete*, *preview* dan *close*. Form Kontak dapat dilihat pada Gambar 4.16.

Form Kontak

Find Clear

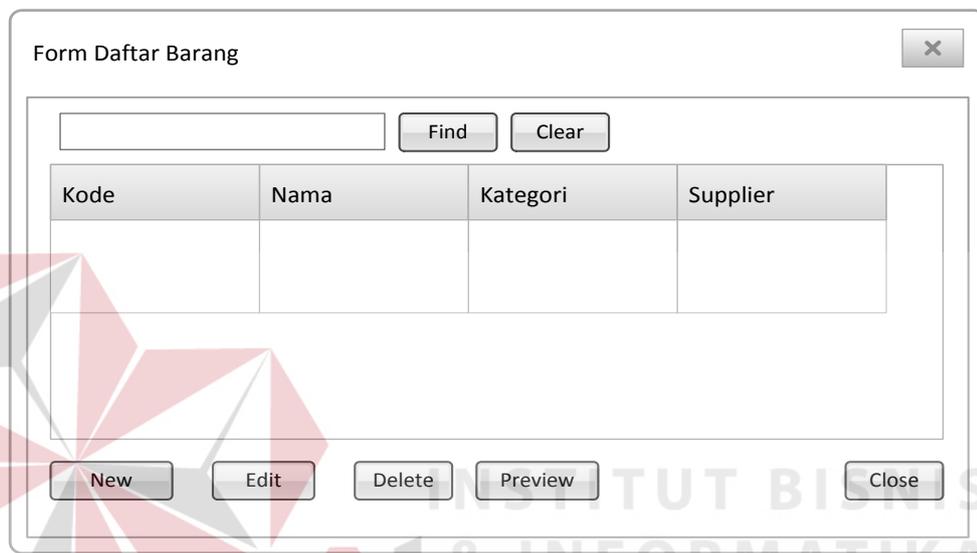
| Kode | Nama | Alamat | Telp |
|------|------|--------|------|
| | | | |

New Edit Delete Preview Close

Gambar 4.16. Desain Form Kontak

f. Form Daftar Barang

Form Daftar Barang digunakan untuk menambah, mengubah, dan menghapus data kode, nama, kategori dan *supplier*. Desain Form Daftar Barang mempunyai lima *button* yaitu *new*, *edit*, *delete*, *preview* dan *close*. Form Daftar Barang dapat dilihat pada Gambar 4.17.



| Kode | Nama | Kategori | Supplier |
|------|------|----------|----------|
| | | | |

Gambar 4.17. Desain Form Daftar Barang

g. Form Daftar Pembelian

Form Daftar Pembelian digunakan untuk menambah, mengubah, dan menghapus data kode, *supplier*, tanggal, total dan keterangan. Desain Form Daftar Pembelian mempunyai enam *button* yaitu *new*, *edit*, *delete*, *print faktur*, *refresh* dan *close*. Form Daftar Pembelian dapat dilihat pada Gambar 4.18.

Form Daftar Pembelian

Tanggal s/D

| Kode | Supplier | Tanggal | Total | Keterangan |
|------|----------|---------|-------|------------|
| | | | | |
| | | | | |

Gambar 4.18. Desain Form Daftar Pembelian

h. Form Daftar Penjualan

Form Daftar Penjualan digunakan untuk menambah, mengubah, dan menghapus data kode penjualan, tanggal penjualan, tanggal tempo pembayaran, *customer* dan total penjualan. Desain Form Daftar Penjualan mempunyai enam *button* yaitu *new*, *edit*, *delete*, *print faktur*, *refresh* dan *close*. Form Daftar Penjualan dapat dilihat pada Gambar 4.19.

Form Daftar Penjualan

Tanggal s/D

| Kode | Tanggal | Tgl Tempo | Customer | Total |
|------|---------|-----------|----------|-------|
| | | | | |

Gambar 4.19. Desain Form Daftar Penjualan

4.3 Implementasi Sistem

Implementasi sistem ini akan menjelaskan detail dari aplikasi Sistem Informasi Penjualan, mulai dari spesifikasi *hardware* / *software* pendukung, cara installasi program, serta fitur-fitur yang terdapat pada aplikasi.

3.6 4.3.1 Kebutuhan Sistem

Sistem yang digunakan untuk menjalankan aplikasi ini terdiri dari *hardware* dan *software* pendukung. Adapun *hardware* dan *software* pendukung yang digunakan adalah sebagai berikut :

Spesifikasi *hardware* pendukung terdiri dari :

1. Processor Core 2 Duo atau lebih tinggi.
2. Memory 2 GB atau lebih tinggi.
3. Hard Disk minimal 10 GB.

Spesifikasi *software* pendukung terdiri dari :

1. Sistem Operasi Microsoft Windows 7 Professional Edition / Ultimate.
2. Microsoft Visual Studio .NET 2010.
3. CRRuntime 32 bit.
4. Microsoft SQL Server 2008 Express Editon.
5. .NET Framework 4.0 atau lebih tinggi.

4.3.2 Cara Instalasi Program

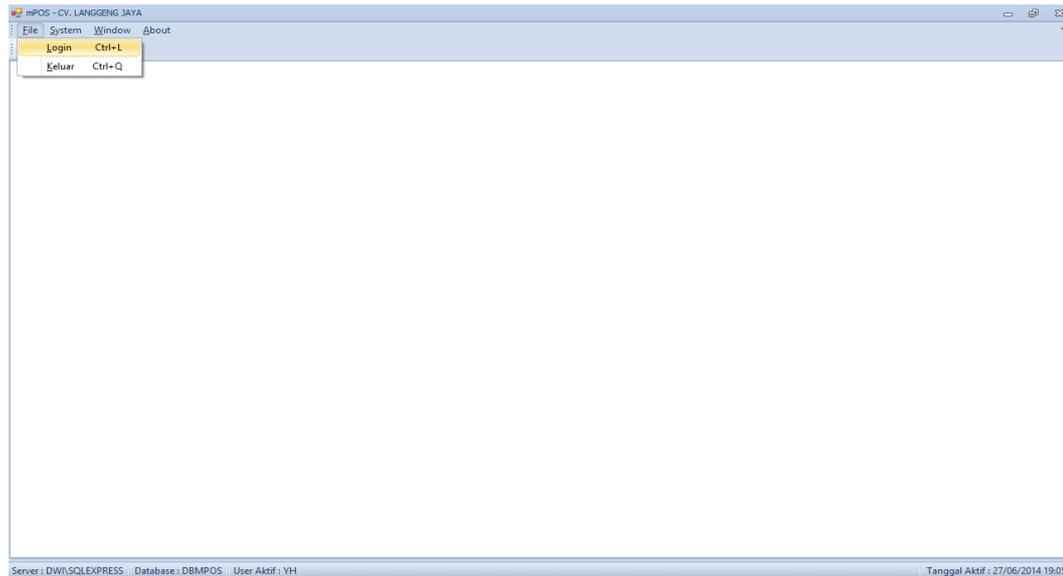
Langkah pertama untuk melakukan instalasi program ini adalah melakukan instalasi .NET Framework 4.0 dan CRRuntime 32 bit agar program dapat berjalan tanpa perlu melakukan instalasi Microsoft Visual Studio 2010. Selanjutnya adalah melakukan instalasi Microsoft SQL Server 2010 Express Edition sebagai akses *database*. Tahap terakhir adalah melakukan instalasi program aplikasi Sistem Informasi Penjualan (*setup.exe*). Setelah semua tahap selesai dilakukan maka program telah siap untuk digunakan.

4.3.3 Implementasi Sistem

Dibawah ini merupakan petunjuk penggunaan fitur-fitur yang terdapat pada Sistem Informasi Penjualan pada CV. Langgeng Jaya.

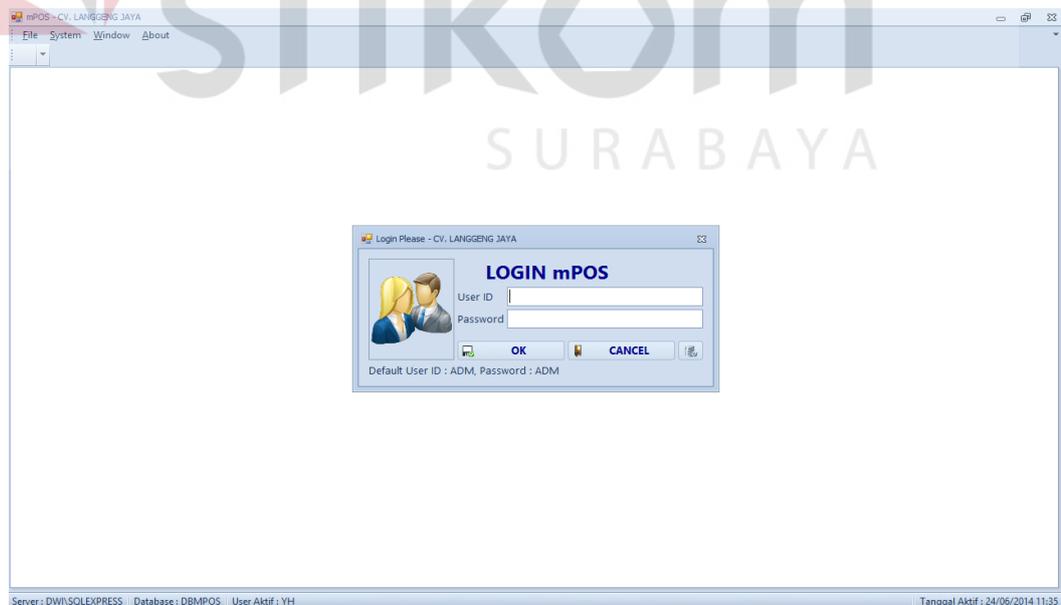
a. Form Utama

Inilah halaman yang pertama kali akan ditampilkan ketika *user* membuka sistem informasi penjualan. Status pertama kali ketika belum masuk kedalam sistem adalah **Anonymous** seperti tampak ditampilan status sebelah kiri atas. Form Utama dapat dilihat pada Gambar 4.20.



Gambar 4.20. Form Utama

Agar bisa masuk ke dalam sistem informasi penjualan. User harus melakukan login dengan memasukkan *username* dan *password* di dalam tampilan login. Untuk menampilkan tampilan login, harus menekan tombol login di tampilan status sehingga tampilannya akan menjadi seperti Gambar 4.21.



Gambar 4.21. Form Login

b. Form Satuan

Form Satuan digunakan untuk menambah, mengubah, dan menghapus data Satuan. Tampilan Form Satuan dapat dilihat pada gambar 4.22.

| Kode | Nama | Is Aktif |
|------|--------|-------------------------------------|
| PCS | PCS | <input checked="" type="checkbox"/> |
| LSN | LUSIN | <input checked="" type="checkbox"/> |
| CRT | CARTON | <input checked="" type="checkbox"/> |
| DUS | DUS | <input checked="" type="checkbox"/> |

Gambar 4.22. Form Satuan

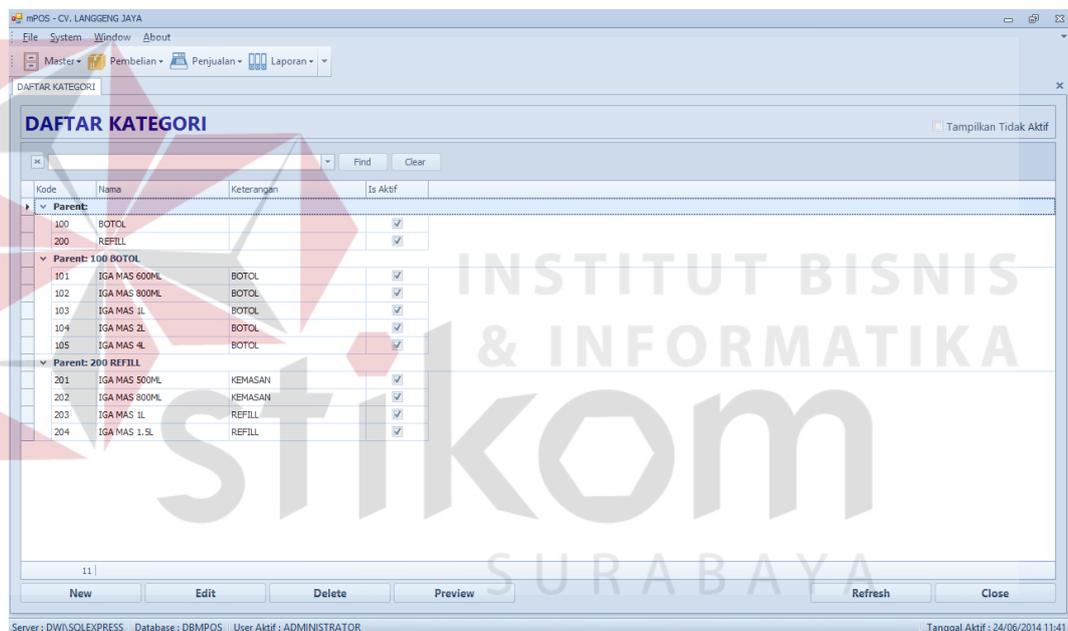
Untuk dapat mengisi Data Satuan, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Satuan. *textbox* “Kode”. Untuk menyimpan Satuan, *textbox* “Nama” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Satuan, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Pengaturan Data Satuan. Setelah perubahan data selesai dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data. Untuk menghapus Data Satuan, *user* harus memilih terlebih dahulu data

yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan dihapus. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Satuan dengan menekan tombol “Delete”. Tombol “Close” berfungsi untuk menutup Form Data Satuan.

c. Form Kategori

Form Kategori digunakan untuk menambah, mengubah, dan menghapus data Kategori. Tampilan Form Kategori dapat dilihat pada gambar 4.23.



Gambar 4.23. Form Kategori

Untuk dapat mengisi Data Kategori, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Kategori. *textbox* “Kode”. Untuk menyimpan kode kategori, *textbox* “Nama” dan *textbox* “Keterangan” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Satuan, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada

datagridview yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Data Kategori. Setelah perubahan data selesai dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data. Untuk menghapus Data Kategori, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan dihapus. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Kategori dengan menekan tombol “Delete”. Tombol “Close” berfungsi untuk menutup Form Data Kategori.

d. Form Kontak

Form Kontak digunakan untuk menambah, mengubah, dan menghapus data Kontak *supplier, customer, sales*. Tampilan Form Kategori dapat dilihat pada gambar 4.24.

| Kode | Nama | Alamat | Kota | Telp | HP | Contact Person | Keterangan | JT Supplier | JT Customer | Limit Hutang | Limit Piutang | Is Aktif | Is Supplier | Is Customer |
|------|--------------|--------------------------|----------|------|--------------|----------------|------------|-------------|-------------|--------------|---------------|-------------------------------------|-------------------------------------|-------------------------------------|
| YH | YANTO H | SIMOWAU INDAH BLOK B-46B | SIDOARJO | | 085777109441 | | | 0,00 | 0,00 | 0,00 | 0,00 | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| WA | WINDI ASTRID | SIDOARJO | SIDOARJO | | | | | 0,00 | 0,00 | 0,00 | 0,00 | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| BD | BLDT | NGANJUK | NGANJUK | | | | | 0,00 | 0,00 | 0,00 | 0,00 | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| S001 | SUNCO | RUNGKUT | SURABAYA | | | | | 14,00 | 0,00 | 0,00 | 0,00 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

Gambar 4.24. Form Kontak

Untuk dapat mengisi Data Kontak, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Kontak. *textbox* “Kode”. Untuk menyimpan kode kategori, *textbox* “Nama” untuk menyimpan nama kontak, *textbox* “Alamat” untuk menyimpan data alamat, *textbox* “Kota” untuk menyimpan kota, *textbox* “NoTelp” untuk menyimpan no telp , *textbox* “NoHP” untuk menyimpan no hp dan *textbox* “Keterangan” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Kontak, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Data Kontak. Setelah perubahan data selesai dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data.

Untuk menghapus Data Kontak, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan dihapus. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Kontak dengan menekan tombol “Delete”. Tombol “Close” berfungsi untuk menutup Form Data Kontak.

e. Form Barang

Form Barang digunakan untuk menambah, mengubah, dan menghapus data Barang. Tampilan Form Barang dapat dilihat pada gambar 4.25.

| Barcode | Kategori | Kode | Nama | Harga Beli | Supplier1 | Supplier2 | Supplier3 | Qty Max | Qty Min | Is Aktif | Konversi Beli | Harga Beli Netto | Saldo Stok | Satuan | Konversi Jual | Mark Up | HPP |
|----------|---------------|---------|---------------|--------------|-----------|-----------|-----------|----------|---------|-------------------------------------|---------------|------------------|------------|--------|---------------|---------|-----|
| 10200018 | IGA MAS 80... | 1020001 | IGA MAS 1L | 5.000.000... | SUNCO | | | 1.000,00 | 10,00 | <input checked="" type="checkbox"/> | 40,00 | 125.000,00 | | CRT | 1,00 | 27,27 | 870 |
| 10100011 | IGA MAS 60... | 1010001 | IGA MAS 600ML | 4.000.000... | SUNCO | | | 1.000,00 | 10,00 | <input checked="" type="checkbox"/> | 100,00 | 40.000,00 | | CRT | 1,00 | 5,00 | 50 |

Gambar 4.25. Form Barang

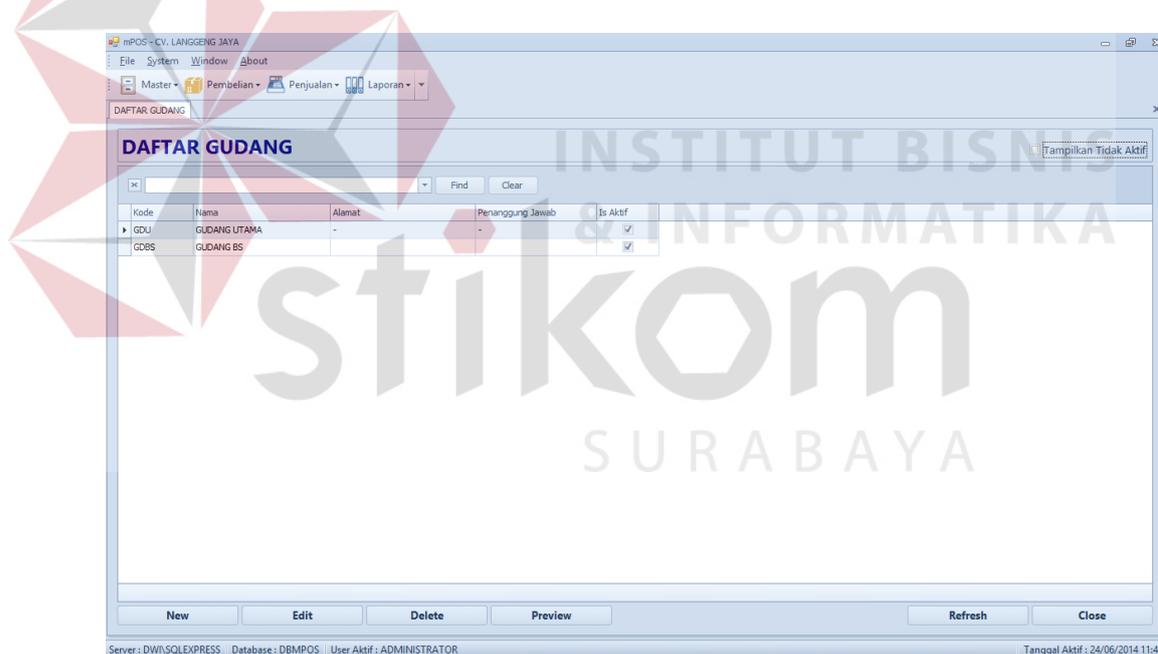
Untuk dapat mengisi Data Barang, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Barang. *textbox* “Kode”. Untuk menyimpan kode barang, *textbox* “Nama”, *textbox* “Kategori” untuk menyimpan data kategori barang, *textbox* “Supplier” untuk menyimpan data supplier, *textbox* “Harga Jual” untuk menyimpan harga jual, dan *textbox* “Keterangan” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Barang, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Data Kategori. Setelah perubahan data selesai

dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data. Untuk menghapus Data Barang, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan dihapus. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Barang dengan menekan tombol “Delete”. Tombol “Close” berfungsi untuk menutup Form Data Barang.

f. Form Gudang

Form Gudang digunakan untuk menambah, mengubah, dan menghapus data Gudang. Tampilan Form Gudang dapat dilihat pada gambar 4.26.



Gambar 4.26. Form Gudang

Untuk dapat mengisi Data Gudang, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Gudang. *textbox* “Kode”. Untuk menyimpan kode barang, *textbox* “Nama” untuk menyimpan nama gudang,

textbox “Alamat” untuk menyimpan alamat gudang dan *textbox* “Penanggung Jawab” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Gudang, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Data Gudang. Setelah perubahan data selesai dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data. Untuk menghapus Data Gudang, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan dihapus. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Gudang dengan menekan tombol “Delete”. Tombol “Close” berfungsi untuk menutup Form Data Gudang.

g. Form Pembelian

Form Pembelian digunakan untuk menambah, mengubah, dan menghapus data Pembelian. Tampilan Form Pembelian dapat dilihat pada gambar 4.27.

The screenshot displays the 'DAFTAR PEMBELIAN' window in the mPOS application. At the top, there are menu options: 'Master', 'Pembelian', 'Penjualan', and 'Laporan'. Below the title bar, the window title is 'DAFTAR PEMBELIAN'. There are two date pickers: 'Tanggal' set to '24-06-2014' and 's/d' set to '24-06-2014'. A search bar with 'Find' and 'Clear' buttons is present. The main area contains a table with the following data:

| Kode | Supplier | Tanggal | Tgl Tempo | Reff | Total | Keterangan | Tgl Entri | User Entri | Tgl Edit | User Edit |
|------------------|----------|------------|------------|------|--------------|------------|------------|---------------|------------|---------------|
| BL/2014/VI/00002 | SUNCO | 24-06-2014 | 08-07-2014 | | 5.000.000,00 | | 24-06-2014 | ADMINISTRATOR | 24-06-2014 | ADMINISTRATOR |

Below the table, there is a status bar showing '1' and 'SUM=5.000.000,00'. At the bottom, there are buttons: 'New', 'Edit', 'Delete', 'Preview', 'Print Faktur', 'Refresh', and 'Close'. The status bar at the very bottom indicates 'Server: DW\SQLEXPRESS Database: DBMPOS User Aktif: ADMINISTRATOR' and 'Tanggal Aktif: 24/06/2014 12:09'.

Gambar 4.27. Form Pembelian

Untuk dapat mengisi Data Pembelian, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Pembelian. *textbox* “Kode”. Untuk menyimpan kode barang, *textbox* “Supplier”, *textbox* “Tanggal”, *textbox* “Total” dan *textbox* “Keterangan” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Pembelian, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Data Pembelian. Setelah perubahan data selesai dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data. Untuk menghapus Data Pembelian, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris

pada *datagridview* yang akan dihapus. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Pembelian dengan menekan tombol “Delete”. Tombol “Print Faktur” berfungsi untuk mencetak faktur pembelian dan terdapat tombol “Close” berfungsi untuk menutup Form Data Pembelian.

h. Form Penjualan Customer

Form Penjualan digunakan untuk menambah, mengubah, dan menghapus data Penjualan. Tampilan Form Penjualann dapat dilihat pada gambar 4.28.

| Kode | Tanggal | Tgl Tempo | Customer | Ref | Salesman | Keterangan | Total | Bayar | Sisa | Total Poin | Tgl Entri | User Entri | Tgl Edit | User Edit |
|------------------|------------|------------|------------|-----|----------|------------|--------------|-------|--------------|------------|------------|---------------|------------|---------------|
| 3L/2014/VI/00002 | 24-06-2014 | 01-07-2014 | WINDI A... | | BUDI | | 3.901.800,00 | 0,00 | 3.901.800,00 | 19,00 | 24-06-2014 | ADMINISTRATOR | 24-06-2014 | ADMINISTRATOR |

Gambar 4.28. Form Penjualan Customer

Untuk dapat mengisi Data Penjualan, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Pembelian. *textbox* “Kode”. Untuk menyimpan kode barang, *textbox* “Tanggal”, *textbox* “Customer”, *combobox* “Sales” *textbox* “Total” dan *textbox* “Keterangan” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Penjualan, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Data Penjualan. Setelah perubahan data selesai dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data. Untuk menghapus Data Penjualan, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan dihapus. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Penjualan dengan menekan tombol “Delete”. Tombol “Print Faktur” berfungsi untuk mencetak faktur Penjualan dan terdapat tombol “Close” berfungsi untuk menutup Form Data Penjualan.

i. Form Retur Penjualan

Form Retur Penjualan digunakan untuk menambah, mengubah, dan menghapus data Daftar Retur Penjualan. Tampilan Form Penjualan dapat dilihat pada gambar 4.29.

| Kode | Customer | Tanggal | Tgl Tempo | Total | Tgl Entri | User Entri | Tgl Edit | User Edit | Keterangan |
|---------------|------------|------------|------------|-----------|------------|---------------|------------|---------------|-------------|
| R3/2014/00002 | WINDI A... | 24-06-2014 | 24-06-2014 | 42.000,00 | 24-06-2014 | ADMINISTRATOR | 24-06-2014 | ADMINISTRATOR | botol bocor |

Gambar 4.29. Form Retur Penjualan

Untuk dapat mengisi Data Retur Penjualan, terlebih dahulu *user* menekan tombol “New” untuk mengaktifkan *groupbox* Data Retur Penjualan. *textbox* “Kode”. Untuk menyimpan kode barang, *textbox* “Customer”, *textbox* “Tanggal”, *textbox* “Total” dan *textbox* “Keterangan” harus diisi terlebih dahulu kemudian tekan tombol “Simpan”.

Untuk mengubah Data Retur Penjualan, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan diubah. Setelah *field input* terisi data sesuai dengan data yang terpilih pada *datagridview*, tekan tombol “Edit” kemudian *user* dapat melakukan perubahan data pada Data Retur Penjualan. Setelah perubahan data selesai dilakukan, tekan tombol “Simpan” untuk melakukan perubahan data. Untuk menghapus Data Retur Penjualan, *user* harus memilih terlebih dahulu data yang akan diubah pada *datagridview*. Pilih data dengan cara memilih baris pada *datagridview* yang akan dihapus. Setelah *field input*

terisi data sesuai dengan data yang terpilih pada *datagridview*, *user* dapat melakukan penghapusan data pada Data Retur Penjualan dengan menekan tombol “Delete”. Tombol “Print Faktur” berfungsi untuk mencetak faktur Retur Penjualan dan terdapat tombol “Close” berfungsi untuk menutup Form Data Retur Penjualan.

j. Faktur Penjualan

Faktur Penjualan digunakan untuk mencetak Faktur atau Nota yang diberikan kepada Customer sebagai tanda terima telah melakukan pembelian barang.

Tampilan Faktur Penjualan dapat dilihat pada gambar 4.30.

| Kode Barang | Nama Barang | Qty | Isi | Harga Jual | Jumlah |
|-------------|---------------|--------|------|------------|---------|
| 1020001 | IGA MAS 1L | 10PCS | 1,00 | 1.400 | 14.000 |
| 1010001 | IGA MAS 600ML | 5PCS | 1,00 | 4.800 | 24.000 |
| 1010001 | IGA MAS 600ML | 100PCS | 1,00 | 4.800 | 480.000 |

| | | | | | |
|---------------|------------|---------|---|-------|---------|
| Dengan Hormat | Bag. Admin | P O I N | 2 | TOTAL | 542.000 |
| | | | | BAYAR | 0 |
| | | | | SISA | 542.000 |

Gambar 4.30. Faktur Penjualan