

## **BAB IV**

### **DESKRIPSI PEKERJAAN**

#### **4.1. Prosedur Kerja Praktik**

Kerja praktek yang dilakukan pada CV golden Smart terhitung dilakukan sejak tanggal 10 November 2014 sampai dengan 10 Desember 2014. Ditempatkan pada kantor CV. Golden Smart yang tepatnya berada di kota Sidoarjo. Selama berada pada CV Golden Smart saya banyak dibantu oleh bapak Nanda Lutfian untuk melihat proses-proses apa saja yang terjadi pada perusahaan tersebut, mulai dari gudang sampai dengan proses penjualan barang. Karena yang saya kerjakan adalah membuat aplikasi sistem informasi pembelian dan penjualan maka saya banyak diarahkan pada melihat proses-proses bagaimana alur barang dan proses penjualan dilakukan.

Kerja praktek ini dilakukan selama 160 jam dengan perkiraan waktu sebagai berikut, masuk pada hari kerja pada hari Senin-Jumat yang dilakukan selama 7 jam perharinya. Dalam kerja praktek ini, diharuskan menemukan permasalahan yang ada, mempelajari serta memberikan solusi bagi masalah yang timbul.

Langkah-langkah yang harus dilakukan dalam penyusunan laporan kerja praktek ini adalah sebagai berikut:

1. Observasi

Dengan mengadakan pengamatan secara langsung untuk mengetahui sistem yang digunakan oleh CV. Golden Smart, ini dilakukan untuk mendapatkan informasi dan data yang berhubungan dengan penyelesaian masalah, selain itu juga untuk mengetahui langkah-langkah apa yang dilakukan oleh perusahaan untuk mengembangkan usahanya.

2. Wawancara

Mengadakan tanya jawab tentang masalah yang dihadapi kemudian mencari solusi atas masalah yang dihadapi.

### 3. Studi Kepustakaan

Dilakukan dengan mencari informasi dari berbagai literatur yang berhubungan dengan kegiatan kerja praktek dan perancangan aplikasi.

## 4.2. Analisis Sistem

CV. Golden Smart merupakan perusahaan yang bergerak dalam bidang penjualan barang, proses bisnis pada CV. Golden Smart dimulai dari pelanggan yang hendak membeli barang memberikan informasi kepada kasir serta membayar barang yang diinginkan, setelah itu kasir memberikan struk kepada pelanggan. Kemudian kasir memberikan daftar barang yang telah dibeli oleh pelanggan kepada divisi gudang yang nantinya divisi gudang akan melakukan *packing* barang. Setelah proses ini pihak CV. Golden Smart memberikan penawaran layanan kepada pelanggan berupa jasa antar barang, pelanggan dapat menggunakan atau tidak menggunakan layanan ini. Agar dapat memenuhi kebutuhan pelanggan, CV. Golden Smart dalam hal ini divisi gudang melakukan cek stok barang setiap hari. Kemudian divisi gudang memberikan laporan stok barang kepada pihak keuangan, di sini pihak keuangan akan mengumpulkan data-data stok barang dan mengevaluasi barang-barang mana saja yang harus dibeli. Apabila barang yang dibutuhkan sudah dicatat, pihak keuangan memberikan daftar stok barang pada pihak gudang dan pihak gudang akan membuatkan PO (*purchase order*) yang akan diberikan kepada supplier, kemudian setelah barang diterima oleh pihak gudang, struk pembelian akan diserahkan kepada pihak keuangan sebagai tanda bukti pembelian.

Masalah yang terjadi dalam proses penjualan dimana kasir tidak dapat mengetahui stok barang yang ada di gudang sehingga pihak kasir harus bertanya lagi kepada pihak gudang perihal stok barang, hal ini membutuhkan waktu tunggu yang lama bagi pelanggan. Dalam

pembelian barang pun terdapat masalah yaitu stok barang yang ada digudang tidak ter-*update* dari divisi penjualan ke bagian gudang, begitu pula sebaliknya sehingga divisi gudang harus melakukan pengecekan ulang untuk mengetahui jumlah stok barang yang ada saat ini. Hal ini juga dialami oleh pihak keuangan yang tidak mengetahui langsung jumlah barang yang terjual dan jumlah barang yang terdapat di dalam gudang, sebab itu pihak keuangan membutuhkan data-data yang cukup banyak mengenai data stok per hari.

Berkaitan dengan hal tersebut, kami akan merekomendasikan pembuatan perangkat lunak/*software* sistem informasi pembelian dan penjualan. Dimulai dari pembuatan *document Flow* penjualan tunai dan pembelian tunai kemudian dilanjutkan dengan pembuatan *system flow* penjualan tunai dan pembelian tunai hingga pembuatan desain *database* sampai dengan membuat aplikasi. Dengan dibuatkannya aplikasi ini pemberian informasi stok barang dari gudang kepada pihak kasir bisa lebih cepat, begitu pula sebaliknya kasir langsung bisa memberikan informasi barang apa saja yang sudah terjual kepada pihak gudang.

#### 4.3. Perancangan Sistem

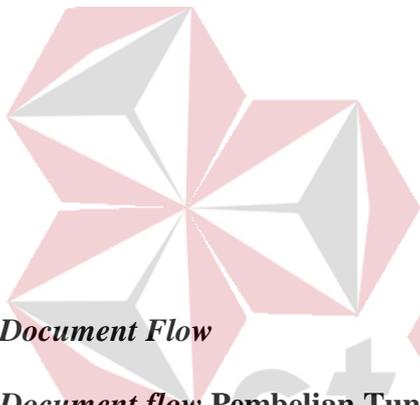
Setelah mengetahui dan mengenali latar belakang, tujuan, ruang lingkup dan proses yang sudah dikaji secara keseluruhan dari sistem tersebut, maka dapat disusun dan dilaksanakan tahap demi tahap langkah-langkah sebagai berikut.

1. Memahami sistem yang akan digunakan pada komputer dan merancang *document flow* yang meliputi apa dan bagaimana dari tiap-tiap *input*, proses dan *output*.
2. Menyusun *file-file* dan struktur *file* yang untuk membuat *system flow*.
3. Membuat desain sistem merupakan tahap pengembangan setelah analisis yaitu dengan membuat *Context Diagram*,
4. Membuat *Data Flow Diagram (DFD)* hasil *decompose* dari *context diagram*.

5. Membuat model ERD atau *Conceptual Data Model (CDM)*.
6. Membuat *Physical Data Model (PDM)* adalah hasil dari generate dari CDM.

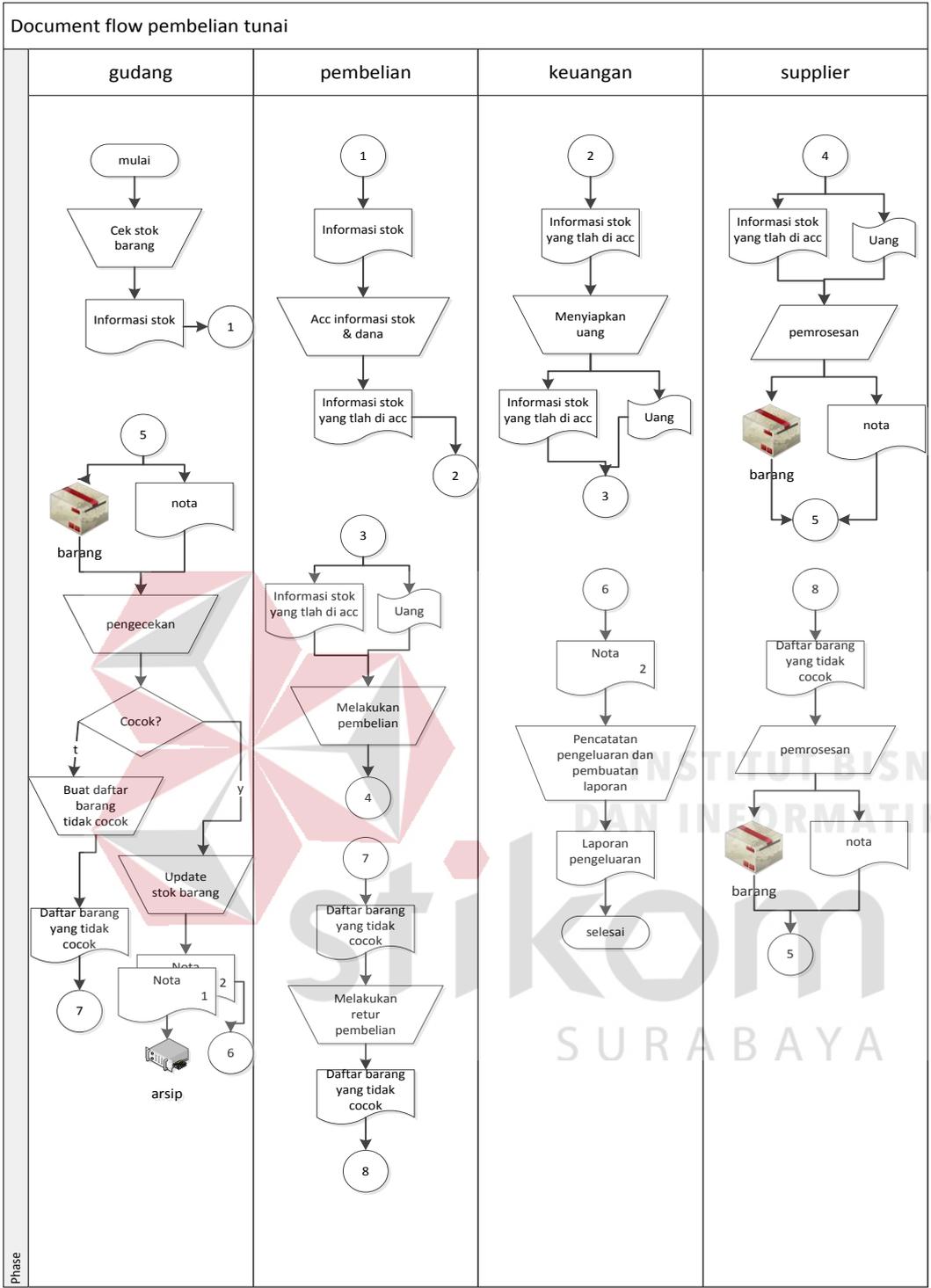
#### **4.4. *Document Flow***

##### **4.4.1. *Document flow* Pembelian Tunai (lama)**



INSTITUT BISNIS  
DAN INFORMATIKA

stikom  
SURABAYA

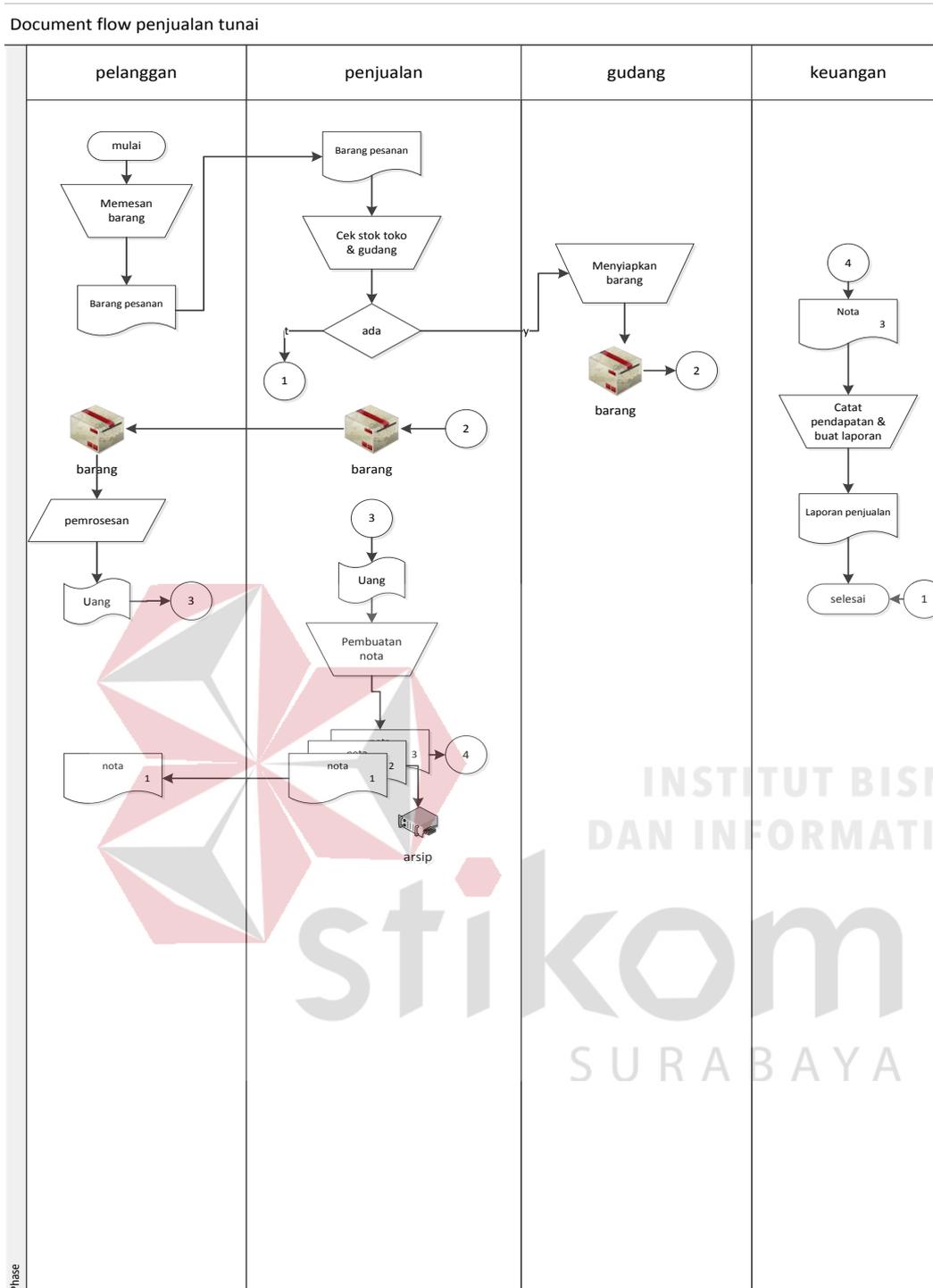


Gambar 4.1. Document flow pembelian tunai

Pada gambar 4.1 proses *document flow* ini dimulai dari divisi gudang melakukan cek stok barang yang ada di gudang yang menghasilkan dokumen informasi stok barang, kemudian dokumen tersenut diberikan kepada pihak pembelian untuk dilakukan acc dan

mengetahui jumlah dana yang diperlukan, setelah itu dokumen informasi stok yang telah di acc diberikan kepada pihak keuangan, kemudian pihak keuangan akan menyiapkan dana yang diperlukan. Daftar stok dan dana tersebut kemudian akan diberikan kepada pihak pembelian yang untuk melakukan proses pembelian barang kepada pihak *supplier*. Kemudian pihak *supplier* akan menyiapkan barang pesanan dan nota yang akan langsung dikirimkan ke bagian gudang, barang dan nota tersebut kemudian akan dilakukan pengecekan. Jika barang yang dikirimkan tidak cocok, maka akan dibuatkan daftar barang yang tidak cocok yang akan di berikan kepada pihak pembelian, dan pihak pembelian akan melakukan retur pembelian yang langsung dikirimkan ke pihak *supplier*. Kemudian pihak *supplier* kembali menyiapkan barang dan nota untuk dikirimkan ke pihak gudang. setelah itu pihak gudang kembali memeriksa atau melakukan pengecekan, setelah barang cocok dengan nota maka pihak gudang akan melakukan *update* stok barang dan dibuatkan nota rangklap dua, nota pertama untuk disimpan oleh pihak gudang dan nota kedua akan di berikan kepada pihak keuangan untuk dilakukan pencatatan pengeluaran dan pembuatan laporan.

#### **4.4.2. Document Flow Penjualan Tunai (lama)**



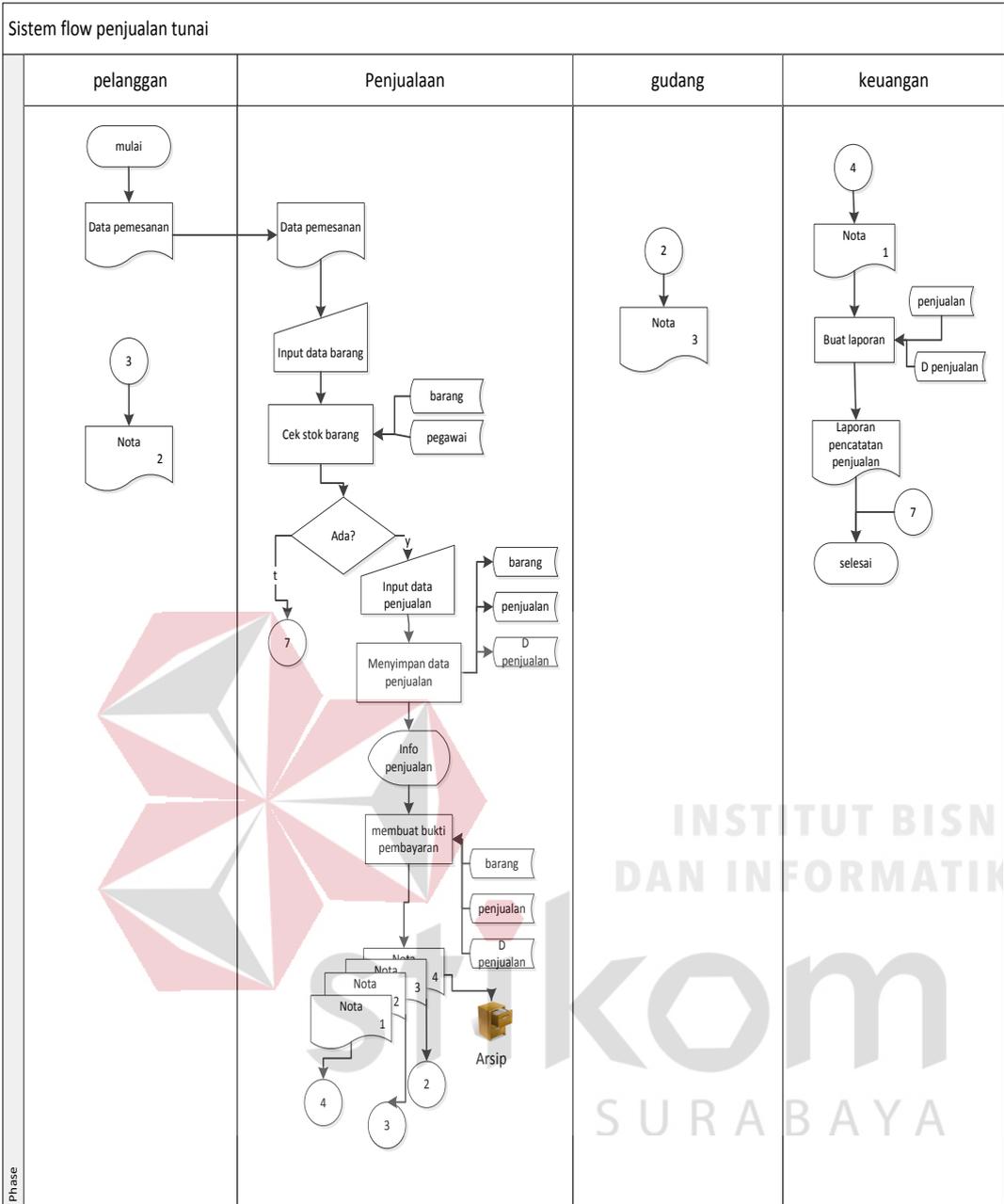
Gambar 4.2. Document flow penjualan tunai

Proses alur *Document Flow* penjualan tunai dimulai dari pelanggan melakukan pemesanan barang kepada kasir, kemudian kasir melakukan cek stok barang dan gudang, jika barang tidak ada maka proses akan selesai. jika barang ada maka pihak gudang akan

menyiapkan barang, kemudian pihak gudang akan mengirimkan barang tersebut ke pihak kasir dan pihak kasir akan memberikan barang tersebut ke pelanggan. Setelah itu pelanggan akan melakukan pembayaran, setelah uang diberikan ke kasir kemudian kasir akan membutuhkan nota tiga rangkap, nota pertama akan di berikan ke pelanggan untuk bukti pembayaran, nota kedua akan disimpan oleh kasir dan nota ketiga akan diberikan kepada pihak keuangan untuk melakukan pencatatan pendapatan dan pembuatan laporan keterangan selengkapnya bisa dilihat pada gambar 4.2.

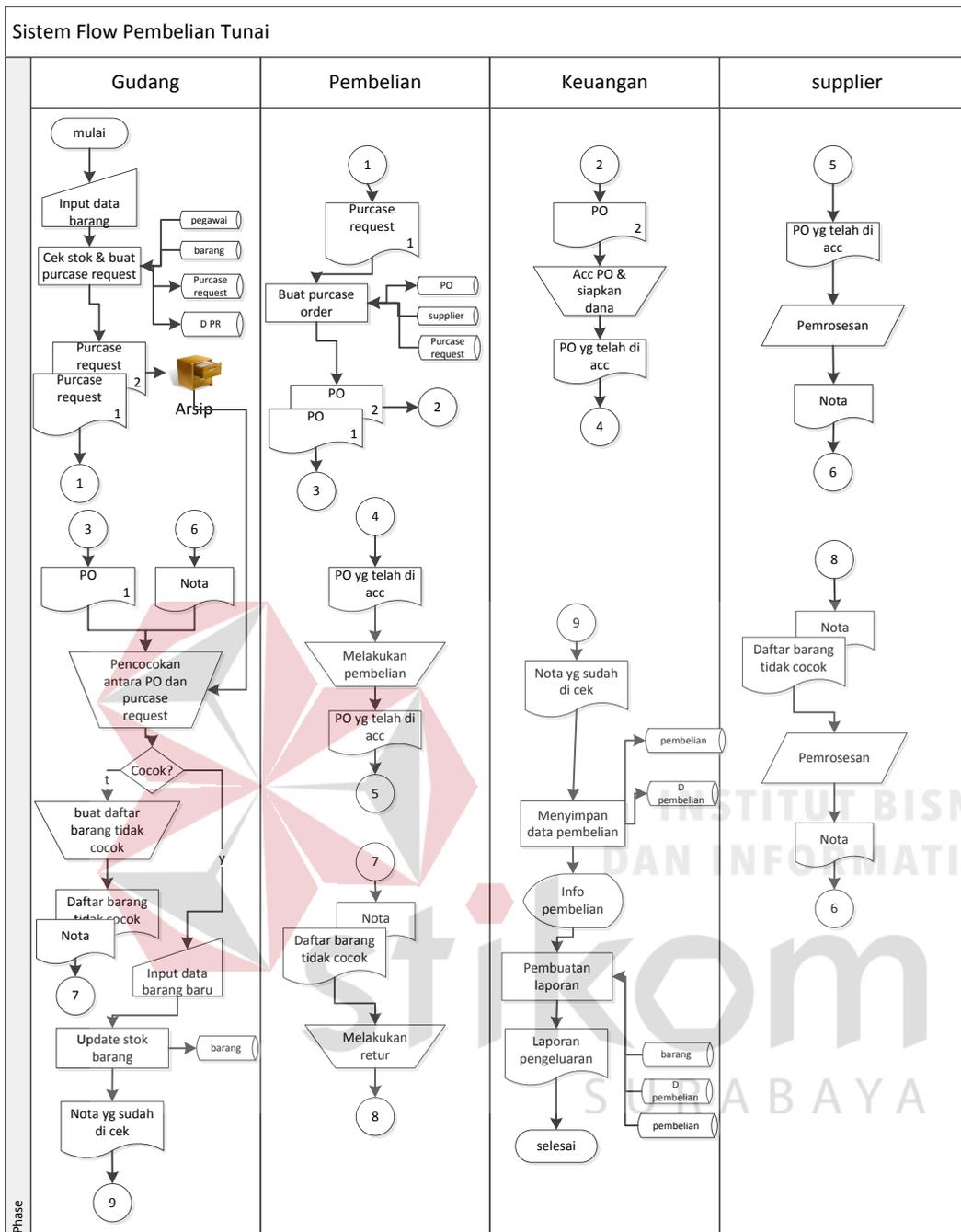
#### **4.4.3. *System flow* Penjualan Tunai (usulan)**

Proses alur *system flow* penjualan tunai dimulai dari pelanggan memberikan data pemesanan kepada pihak penjualan, kemudian pihak penjualan akan melakukan *input* data barang dan melakukan cek stok barang yang diambil dari tabel barang untuk melihat stok barang dan tabel pegawai untuk melakukan *login*. Jika setelah dilakukan pengecekan, barang tidak ada maka proses akan selesai dan jika barang yang dicek ada maka pihak penjualan akan melakukan *input* data penjualan kemudian akan dilanjutkan dengan menyimpan data penjualan pada tabel barang, tabel penjualan dan detail penjualan. Setelah menyimpan data pihak penjualan akan membuat bukti pembayaran berupa nota rangkap empat yang diambil dari tabel barang, tabel penjualan dan detail penjualan. Nota pertama akan di berikan pada pihak keuangan, nota kedua akan diberikan kepada pelanggan sebagai bukti pembelian, nota ketiga akan diberikan kepada pihak gudang dan nota keempat akan disimpan sebagai arsip. selanjutnya pihak keuangan yang sudah menerima nota akan membuat laporan pencatatan penjualan yang diambil dari tabel penjualan dan detail penjualan gambar selengkapnya bisa dilihat pada gambar 4.3.



Gambar 4.3. System flow penjualan tunai

4.4.4. System flow Pembelian Tunai (usulan)



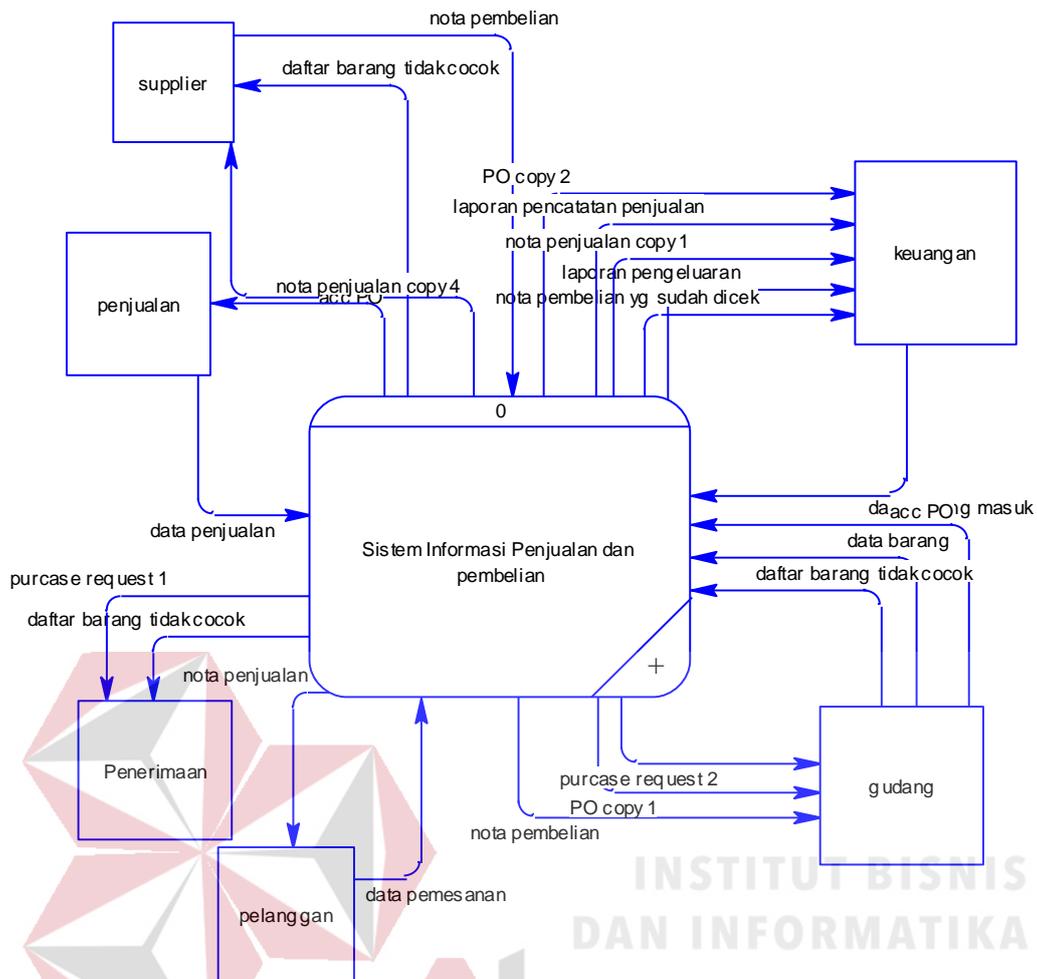
Gambar 4.4. *System Flow* pembelian tunai

Alur proses *system flow* pada gambar 4.4, pembelian tunai dimulai dari pihak gudang melakukan *input* data barang kemudian melakukan cek stok dan membuat *Purchase Request* yang diambil dari tabel barang, dan tabel pegawai untuk melakukan *login*, selanjutnya akan disimpan kedalam tabel *purcse request* dan detail *purcse request*. Proses tersebut akan menghasilkan dokumen *Purchase Request* rangkap dua, dokumen kedua akan diarsipkan

yang nantinya akan digunakan untuk mencocokkan barang yang dibeli dari *supplier*, dan dokumen pertama akan diberikan kepada pihak pembelian. Selanjutnya pihak pembelian akan membuatkan *purchase order* yang diambil dari tabel *supplier* dan tabel *Purchase Request* kemudian akan disimpan pada tabel *purchase order*. Dari proses tersebut akan dihasilkan dokumen PO dua rangkap, dokumen PO pertama akan diberikan ke divisi gudang dan dokumen kedua akan diberikan kepada pihak keuangan untuk dilakukan acc PO dan menyiapkan dana. Dokumen PO yang telah di acc akan diberikan lagi ke pihak pembelian yang digunakan untuk melakukan pembelian barang pada pihak *supplier*. Setelah melakukan pembelian, *supplier* akan membuatkan nota, kemudian nota akan dikirimkan kepada pihak gudang. selanjutnya gudang akan melakukan pencocokan antara dokumen PO dan *Purchase Request*, jika setelah dicek barang tidak cocok maka pihak gudang akan membuatkan daftar barang yang tidak cocok. Kemudian daftar barang yang tidak cocok beserta nota akan diberikan kepada pihak pembelian untuk melakukan retur pembelian. setelah melakukan retur pembelian, pihak *supplier* akan mengirimkan nota kepada pihak gudang yang kemudian akan dilakukan pencocokan kembali. Jika cocok pihak gudang akan memasukkan data barang baru dan melakukan *update* stok barang yang akan disimpan pada tabel barang. Kemudian nota yang sudah dicek dan dicocokkan akan diberikan kepada pihak keuangan untuk selanjutnya akan melakukan penyimpanan data pembelian yang disimpan pada tabel pembelian dan detail pembelian. setelah itu pihak keuangan akan melakukan pembuatan laporan keuangan yang diambil dari tabel barang, tabel pembelian dan tabel detail pembelian.

#### **4.5. Perancangan Basis Data Sistem Informasi Pembelian dan Penjualan**

##### **4.5.1. Context diagram**

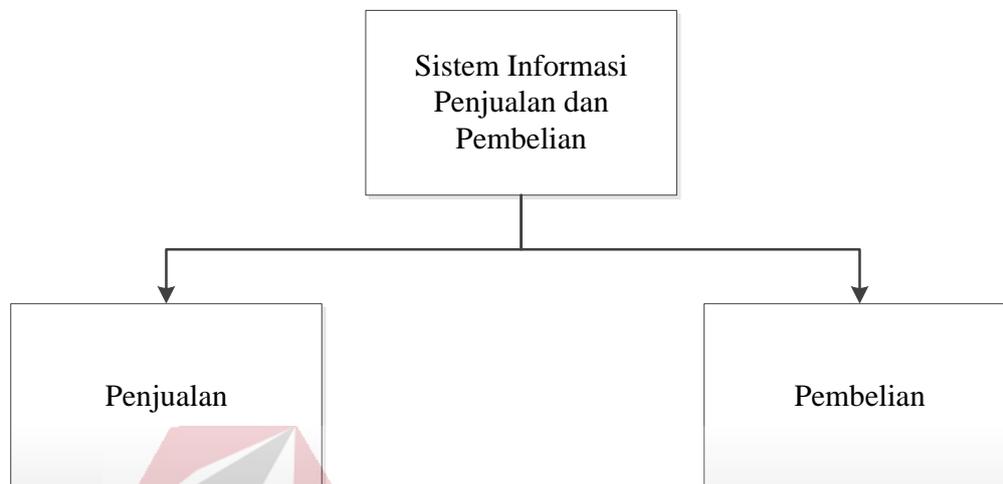


Gambar 4.5. *Context diagram* sistem informasi pembelian dan penjualan

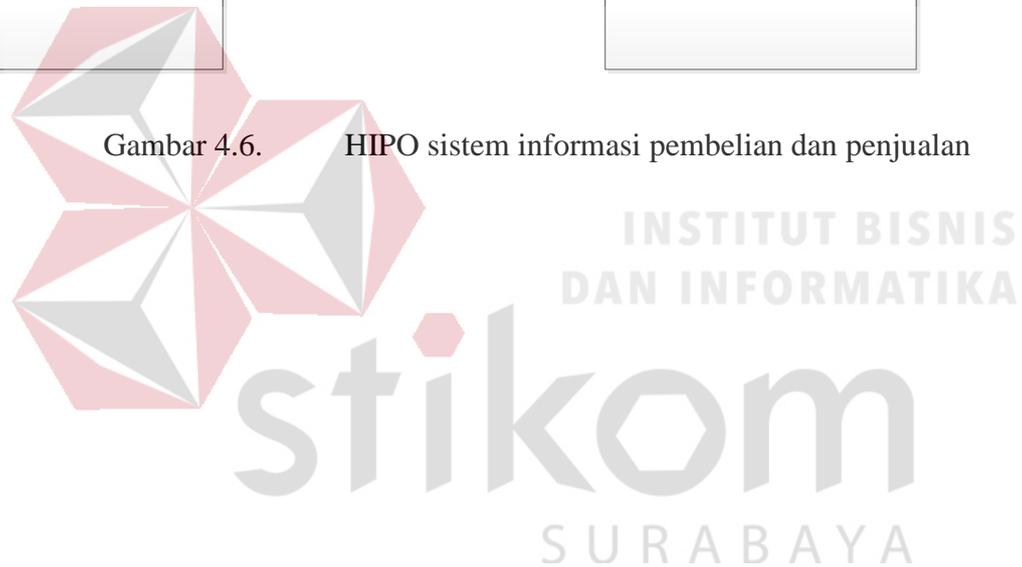
Pada gambar 4.5 *Context diagram* sistem informasi pembelian dan penjualan merupakan gambar sistem informasi pembelian dan penjualan pada CV. Golden Smart, secara garis besar terdapat empat *internal entity* yaitu penjualan, gudang, pembelian dan keuangan. Sedangkan *external entity* yaitu pelanggan dan *supplier*

#### 4.5.2. HIPO Sistem Informasi Pembelian dan Penjualan

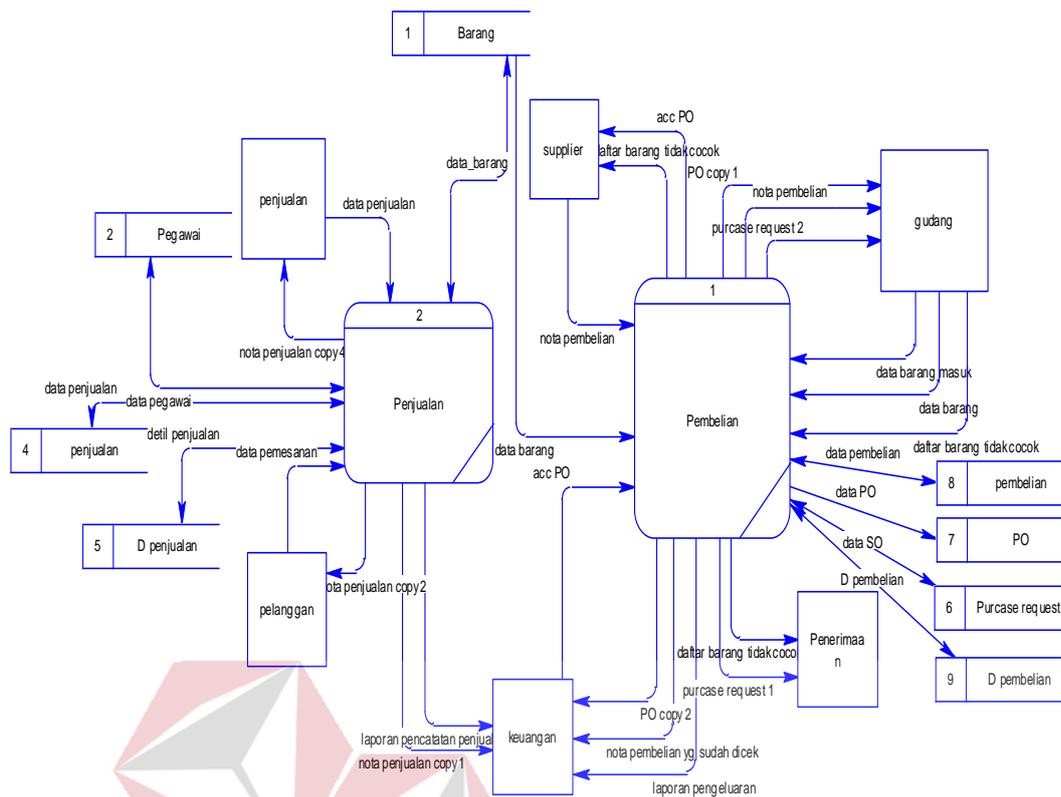
Berikut ini merupakan gambar diagram berjenjang dari sistem informasi pembelian dan penjualan CV. Golden Smart



Gambar 4.6. HIPO sistem informasi pembelian dan penjualan



#### 4.5.3. DFD Sistem Informasi Pembelian dan penjualan



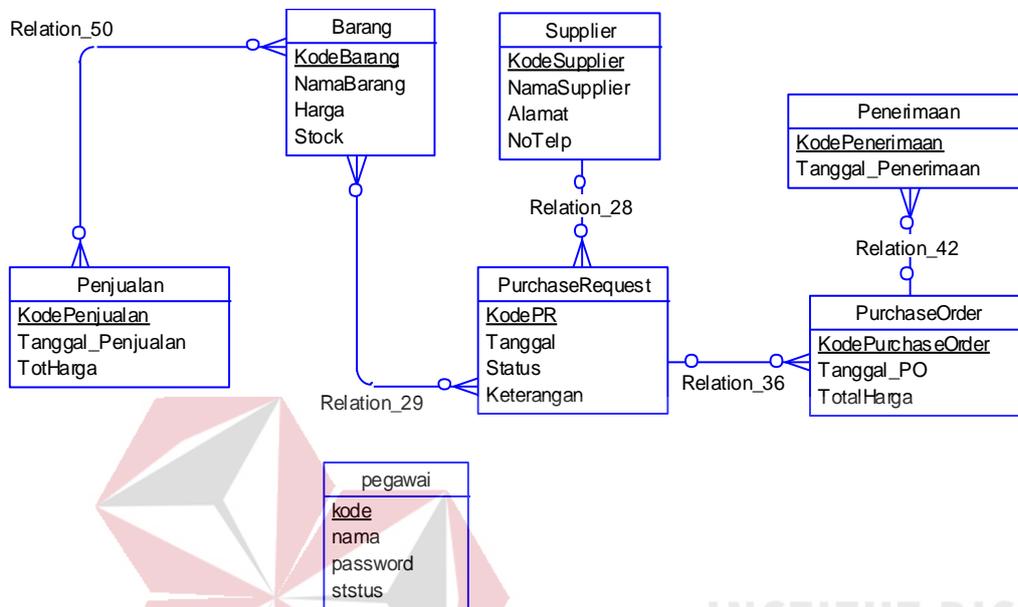
Gambar 4.7. DFD Sistem informasi pembelian dan penjualan

Setelah membuat *context diagram* dari sistem informasi pembelian dan penjualan pada CV. Golden Smart, untuk selanjutnya *context diagram* tersebut akan dibagi menjadi sub-sub proses yang lebih kecil. Untuk penjelasan *context diagram* dapat dilihat pada gambar 4.7. Hasil *decompose* dari *context diagram* disebut DFD level 0, dan DFD level 0 itu sendiri terdiri dari dua proses utama yaitu proses pembelian dan proses penjualan.

#### 4.5.4. Entity Relationship Diagram

##### A. Conceptual Data Model

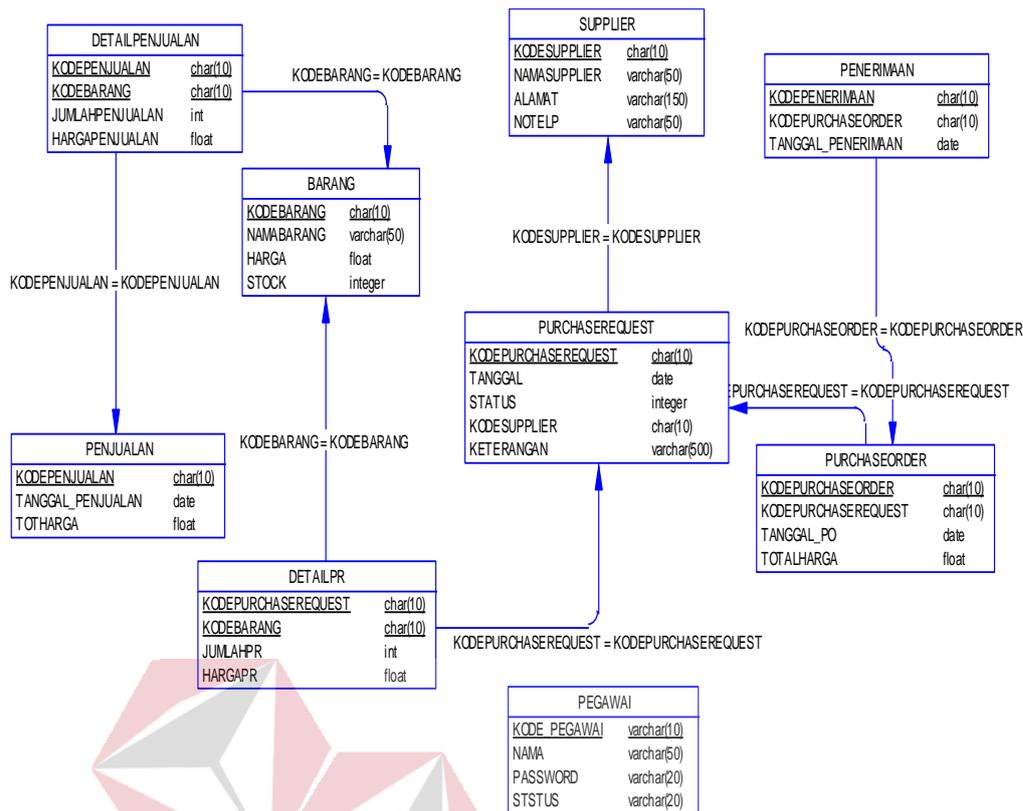
*Conceptual Data Model (CDM)* dari sistem informasi pembelian dan penjualan pada CV. Golden Smart terdapat empat belas tabel. Masing-masing tabel mempunyai relasi ke tabel yang lain seperti pada gambar 4.8.



Gambar 4.8. *Conceptual Data Model* sistem informasi pembelian dan penjualan

**B. Physical Data Model**

*Physical Data Model (PDM)* adalah hasil dari *generate* dari CDM. Data tabel pada PDM inilah yang akan digunakan pada saat membuat aplikasi. PDM dari sistem informasi perawatan mobil dapat dilihat pada gambar 4.9.



Gambar 4.9. *physical data model* sistem informasi pembelian dan penjualan

#### 4.5.5. Struktur Data

Struktur basis data yang diperlukan dalam pembuatan Sistem Informasi pembelian dan penjualan adalah :

a. Tabel Pegawai

Nama : Pegawai

Primary Key : Kode\_Pegawai

Foreign Key : -

Fungsi : Menyimpan data pegawai, digunakan untuk *login*

Tabel 4.1. Tabel Pegawai

No	Nama Field	Tipe	Lebar	Constraint	Foreign Key
----	------------	------	-------	------------	-------------

		Data			Tabel	Kolom
1.	Kode_Pegawai	Varchar	10	PK		
2.	Nama_Supplier	Varchar	50	Not Null		
3.	Password	Varchar	20	Not Null		
4.	Status	Varchar	20	Not Null		

b. Tabel *Supplier*

Nama : *Supplier*

Primary Key : *Kode\_Supplier*

Foreign Key : -

Fungsi : Menyimpan data *supplier*, digunakan untuk membuat PO pembelian.

Tabel 4.2. Tabel *Supplier*

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom
1.	<i>Kode_Supplier</i>	Varchar	15	PK		
2.	<i>Nama_Supplier</i>	Varchar	50	Not Null		
3.	Alamat	Varchar	100	Not Null		
4.	NoTelp	Int		Not Null		

c. Tabel barang

Nama : *Barang*

Primary Key : *Kode\_Barang*

Foreign Key : -

Fungsi : Menyimpan data Data Barang.

Tabel 4.3. Tabel *Barang*

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom

1.	Kode_Barang	Varchar	15	PK		
3	Nama_Barang	Varchar	50	Not Null		
4	Harga	Int		Not Null		
5	Stok	Int				

d. Tabel penjualan

Nama : Penjualan

Primary Key : No\_Penjualan

Foreign Key : -

Fungsi : Menyimpan data Data Penjualan

Tabel 4.4. Tabel Penjualan

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom
1.	No_Penjualan	Varchar	15	PK		
2.	Tanggal	DateTime		Not Null		
3	Total	Int		Not Null		

e. Tabel Detail penjualan Barang

Nama : Detail Penjualan Barang

Primary Key : -

Foreign Key : No\_Penjualan, Kode\_Barang

Fungsi : Menyimpan data Data Penjualan Barang

Tabel 4.5. Tabel Detail Penjualan

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom
1.	Kode_penjualan	Varchar	10	FK	Penjualan	No_Penjualan

2.	Kode_Barang	Varchar	10	FK	Barang	Kode_Barang
3	Jumlah	Int		Not Null		
4	Harga	Int		Not Null		

f. Tabel *Purchase Request*

Nama : *Purchase Request*

Primary Key : Kode\_PR

Foreign Key : -

Fungsi : Menyimpan data Data *Purchase Request*

Tabel 4.6. Tabel *Purchase Request*

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom
1.	Kode_PR	Char	10	PK		
2.	Tanggal	DateTime				
3	Status	Integer				
4	Kode_suplier	Char	10			
5	Keterangan	Varchar				

g. Tabel Detail *Purchase Request*

Nama : Detail *Purchase Request*

Primary Key : -

Foreign Key : Kode\_PR, Kode\_Barang

Fungsi : Menyimpan data Detail *Purchase Request*

Tabel 4.7. Tabel Detail *Purchase Request*

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom
1.	Kode_PR	Varchar	15	FK	PR	Kode_PR

2.	Kode_Barang	Varchar	15	FK	Barang	Kode_Barang
3	Harga	Int		Not Null		
4	Jumlah	Int		Not Null		

h. Tabel PO

Nama : Purchase order

Primary Key : kode\_PO

Foreign Key : Kode\_Supplier

Fungsi : Menyimpan data Data PO Pembelian

Tabel 4.8. Tabel PO

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom
1.	Kode_PO	Char	10	PK		
2.	Kode_PR	Char	10	FK	Toko	Kode_Supplier
4	Tanggal_PO	DateTime		Not Null		
5	Total Harga	Fload		Not Null		

i. Tabel Penerimaan

Nama : Penerimaan

Primary Key : Kode\_Penerimaan

Foreign Key : Kode\_PO

Fungsi :Menyimpan data-data barang yang sudah msauk ke gudang

Tabel 4.9. Tabel Penerimaan

No	Nama Field	Tipe Data	Lebar	Constraint	Foreign Key	
					Tabel	Kolom
1.	Kode_Penerimaan	Char	10	PK		

2.	Kode_PO	Char	10	FK	PO Pembelian	No_PO
3	Tanggal	DateTime		Not Null		

## 4.5 Desain *Input* dan *Output*

Desain *input* dan *output* adalah sebuah rancangan yang berupa sebuah *form* untuk memasukkan data dan laporan sebagai informasi yang dihasilkan dari pengolahan sebuah data. Desain *input* dan *output* ini akan digunakan untuk membuat sebuah rancangan aplikasi.

### 4.6.1 Desain Layar Dialog

#### a. *Form Login*

*form login* ini digunakan *user* yang memiliki hak akses untuk masuk ke dalam aplikasi. *User* terlebih dahulu harus memasukkan *user name* dan *password*, jika *textbox username* dan *password* sudah terisi kemudian *user* harus menekan tombol OK untuk masuk ke dalam aplikasi dan jika ingin membatalkan maka tekan tombol *cancel* dan bisa dilihat pada gambar 4.10.

Gambar 4.10. *Form Login*

#### b. *Form Master Barang*

Dalam gambar 4.11 *Form form master* barang digunakan untuk menyimpan data barang yang masuk pada proses pembelian barang dan juga bisa digunakan untuk melakukan pencarian nama barang yang dibutuhkan *user*.

Form master barang

Cari

Data barang

Gambar 4.11. *Form Master Barang*

c. *Form Master Supplier*

*Form master supplier* digunakan untuk memasukan data *supplier* ke dalam *database*. Sehingga bisa digunakan untuk keperluan permintaan pemesanan barang seperti yang terlihat pada gambar 4.12.

Form Master supplier

Cari

Data supplier

Gambar 4.12. *from Master Supplier*

d. *Master Request* pembelian

Pada gambar 4.13 *form* request pembelian digunakan untuk melakukan permintaan pembelian oleh gudang, sedangkan untuk data barang dan data *supplier* diambil dari *master* barang dan *master supplier*. *User* harus mengisi data *supplier* dan data barang terlebih dahulu agar bisa dilakukan acc oleh divisi pembelian.

Form Request Pembelian

Kode request pembelian

Tanggal request pembelian

Kode supplier

Nama supplier

Kode barang

Nama barang

Harga

Jumlah

Daftar request pembelian

Gambar 4.13. *Form Request* Pembelian

e. *Form* Penjualan Barang

*Form form* penjualan digunakan oleh kasir/divisi penjualan untuk melakukan transaksi penjualan seperti yang terlihat pada gambar 4.14. Di dalam *form* penjualan barang *user* akan mengisi data barang terlebih dahulu yang bisa dilakukan dengan menekan tombol cari, dan kemudian menekan tombol tambah untuk memasukkan data penjualan kedalam daftar pada *form* penjualan tersebut.

Form penjualan

Kode penjualan

Tanggal

Kode barang

Nama barang

Harga

Jumlah

Daftar penjualan barang

Gambar 4.14. *Form Penjualan Barang*

f. *Form Acc Purchase Request*

Pada gambar 4.15 *form acc Purchase Request* digunakan untuk melakukan persetujuan atas permintaan barang yang dilakukan oleh divisi gudang. di dalam *form* ini terdapat dua kondisi, permintaan ditolak atau diterima. Untuk melihat daftar *request* pembelian dengan cara memilih nomer *purchase reques* yang untuk melihat daftar barang yang di-*request*

Form acc purchase request

No. Purcase request   No. Purcase order

Tanggal purchase request  Tgl purchase orde

Daftar purchase request

Keterangan

Gambar 4.15. *Form Acc Purchase Request*

g. *Form Pembelian*

*Form* pembelian digunakan untuk melihat daftar pembelian yang sudah dilakukan oleh divisi pembelian. Dan juga bisa melihat detail barang apa saja yang akan dibeli dan dapat dilihat pada gambar 4.16.

Form Pembelian

Tanggal awal

Tanggal akhir

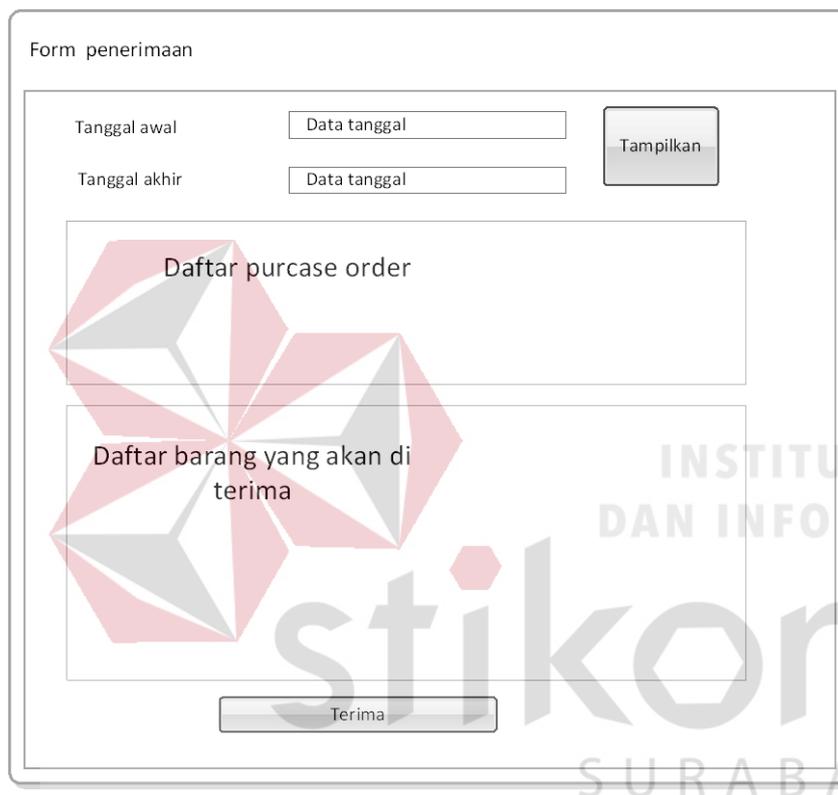
Daftar purchase order

Daftar purchase order secara detail

Gambar 4.16. *Form Pembelian*

#### h. *Form* Penerimaan Barang

Pada gambar 4.17, *form* penerimaan barang berfungsi untuk melakukan penerimaan daftar barang yang masuk setelah dilakukan pembelian barang. Barang yang sudah masuk ke gudang kemudian *diinput*-kan ke dalam *form* penerimaan yang bertujuan untuk *update* stok barang.



Gambar 4.17. Gambar *Form* Penerimaan

#### 4.5.6. Desain *Output*

##### i. Laporan Penjualan

*Form report* penjualan berfungsi menampilkan report penjualan berdasarkan tanggal yang diminta *user*, *form* ini akan menampilkan daftar barang, harga dan total harga barang secara keseluruhan pada tanggal yang sudah di tentukan, seperti yang terlihat pada gambar 4.18

Laporan Penjualan

Tanggal Awal

Tanggal Akhir






Kode Penjualan :  
 Tanggal :

Kode barang	Nama barang	Jumlah	Harga	Total

Gambar 4.18. Laporan Penjualan

j. Laporan Pembelian

Pada gambar 4.19 *form report* penjualan berfungsi menampilkan *report* penjualan berdasarkan tanggal yang diminta *user*, *form* ini akan menampilkan daftar barang, harga dan total harga barang secara keseluruhan pada tanggal yang sudah di tentukan

Laporan Pembelian

Tanggal Awal

Tanggal Akhir





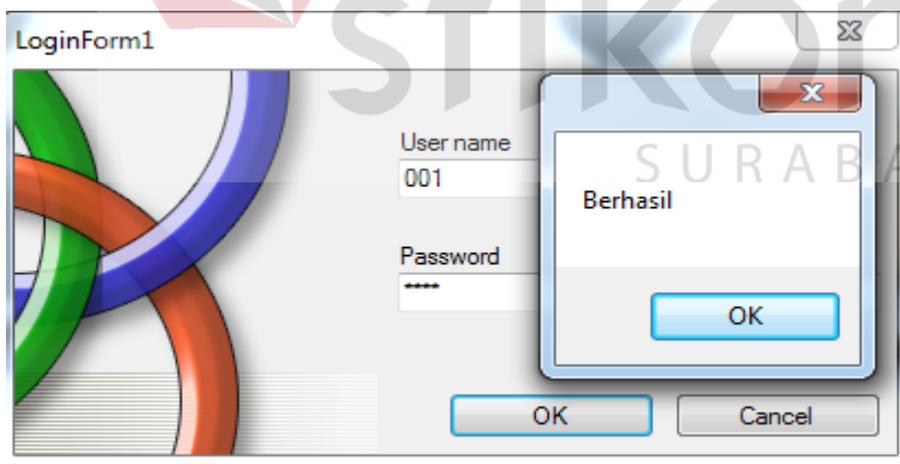

Kode Pembelian :  
Tanggal :

Kode barang	Nama barang	Jumlah	Harga	Total

Gambar 4.19. Laporan Pembelian

#### 4.6. Testing Aplikasi

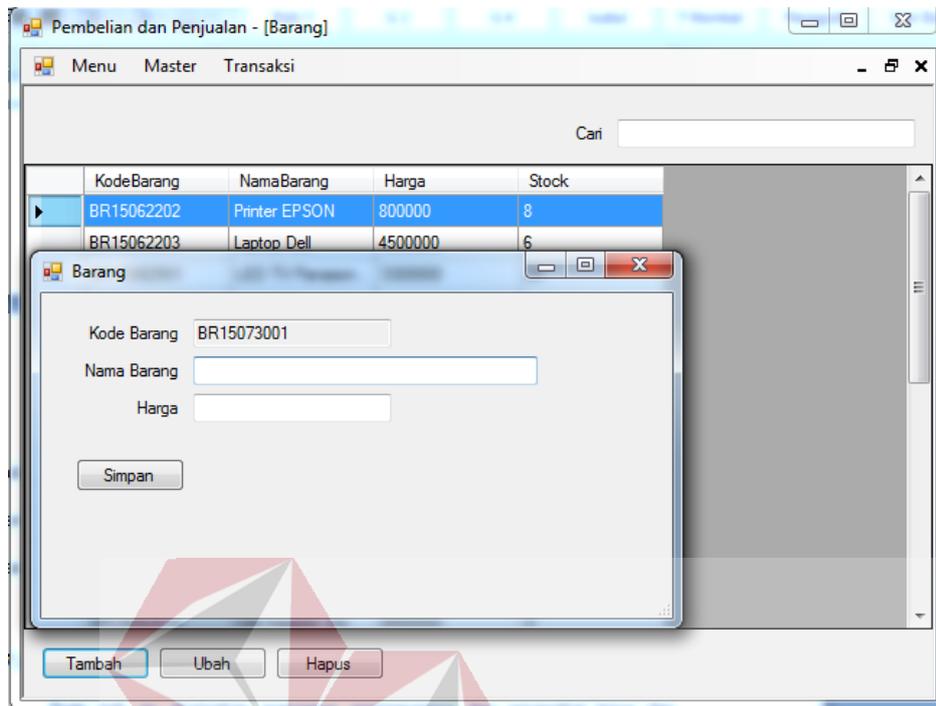
##### a. Login



Gambar 4.20. Login Aplikasi

Pada gambar 4.20 tombol *login* berhasil di jalankan setelah memasukkan password yang benar. Kemudian akan muncul notifikasi “berhasil”. Jika OK maka akan masuk ke aplikasi.

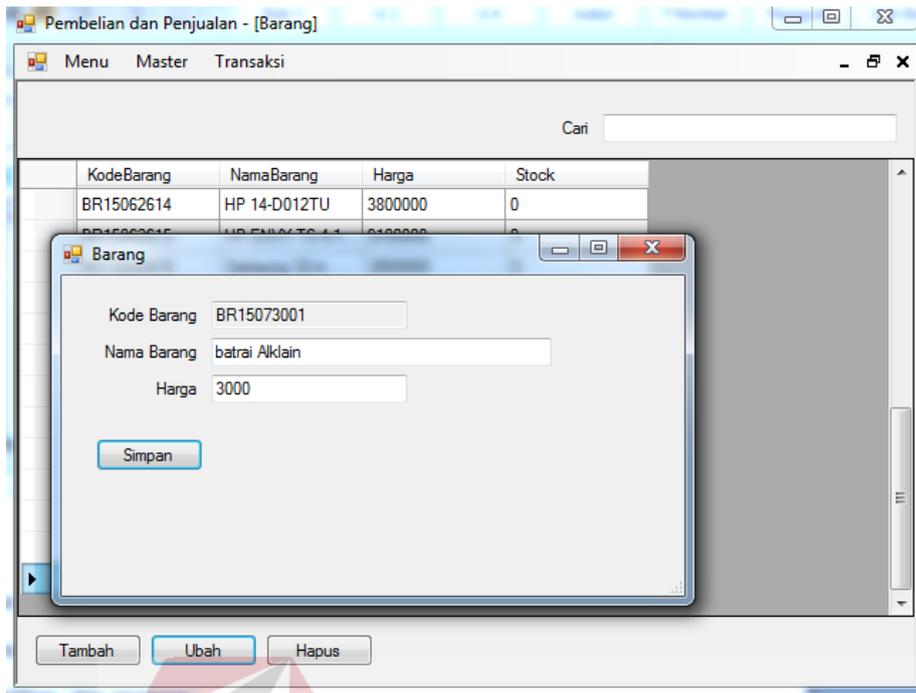
b. Tambah Barang



Gambar 4.21. Tambah Barang

Pada gambar 4.21, jika nama barang ditambahkan dan kemudian menekan tombol simpan maka data akan langsung terimpan ke dalam *database* dan muncul notifikasi “data tersimpan”

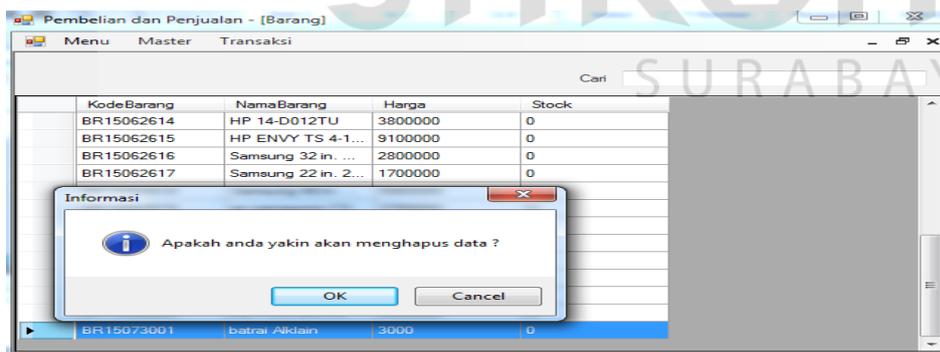
c. Ubah Barang



Gambar 4.22. Tambah Barang

Pada gambar 4.22, *form* ubah barang berfungsi untuk mengubah nama dan harga barang. Jika sudah memasukkan data yang diubah selanjutnya tekan tombol simpan, dan data akan tersimpan ke dalam *database*.

d. Hapus data

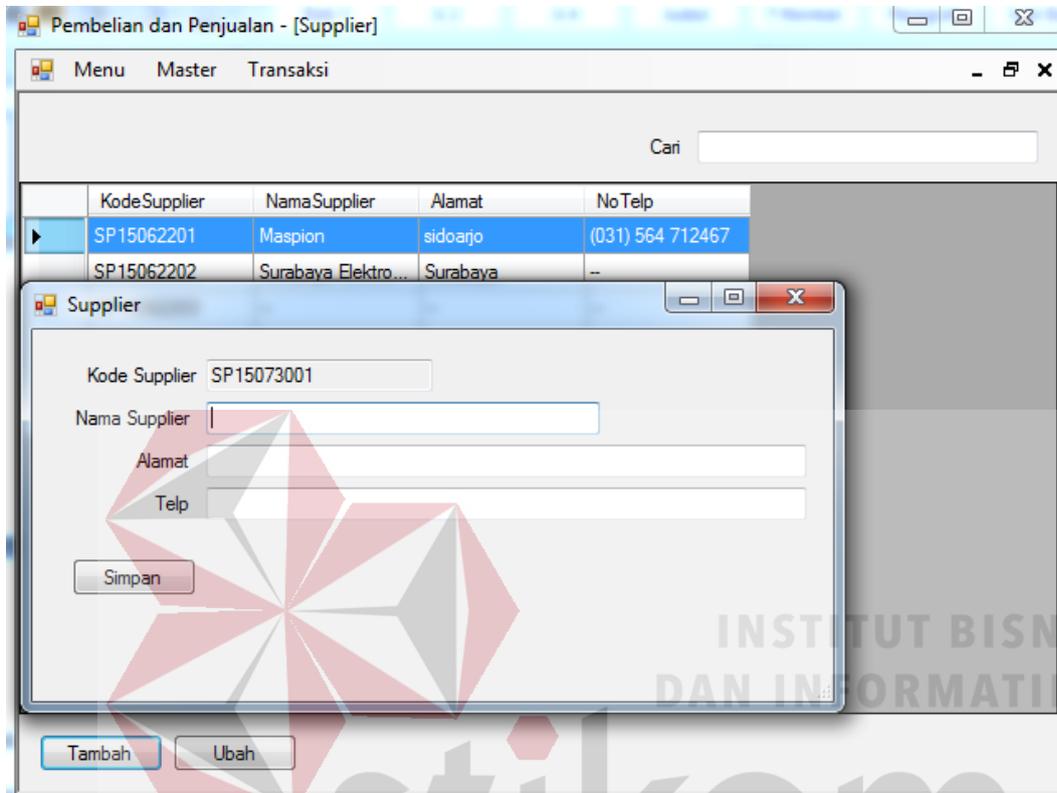


Gambar 4.23. Hapus Barang

Pada gambar 4.23, proses penghapusan dimulai dari melakukan *select* pada nama barang yang akan dihapus, kemudian tekan tombol hapus yang menghasilkan notifikasi

“apakah anda yakin akan menghapus data” jika menekan tombol ok maka data otomatis terhapus dari *database*.

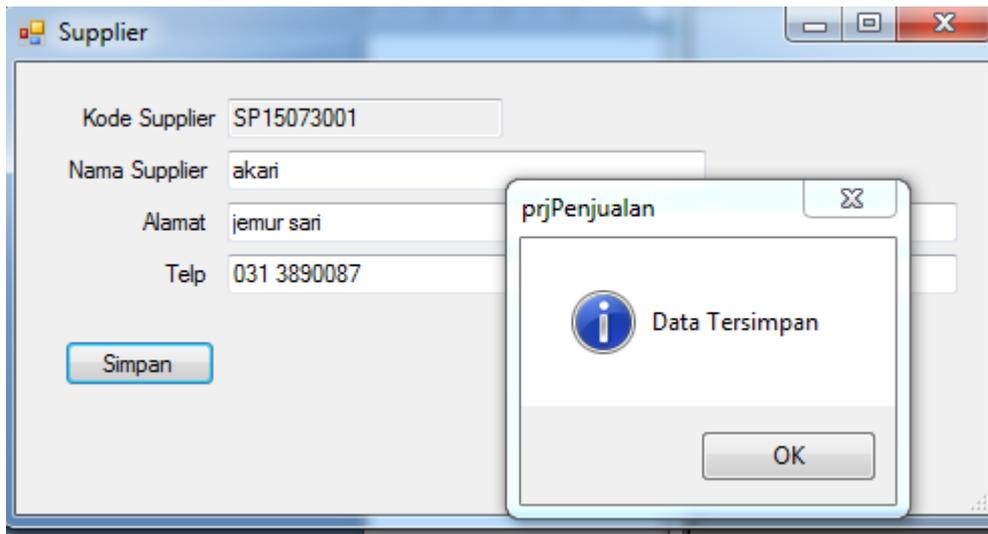
e. Tambah Suplier



Gambar 4.24. Tambah *Supplier*

Pada gambar 4.24, untuk menambahkan data *supplier* terlebih dahulu menekan tombol tambah, kemudian mengisi data *supplier*. Setelah menekan tombol simpan maka data otomatis tersimpan ke dalam *database*.

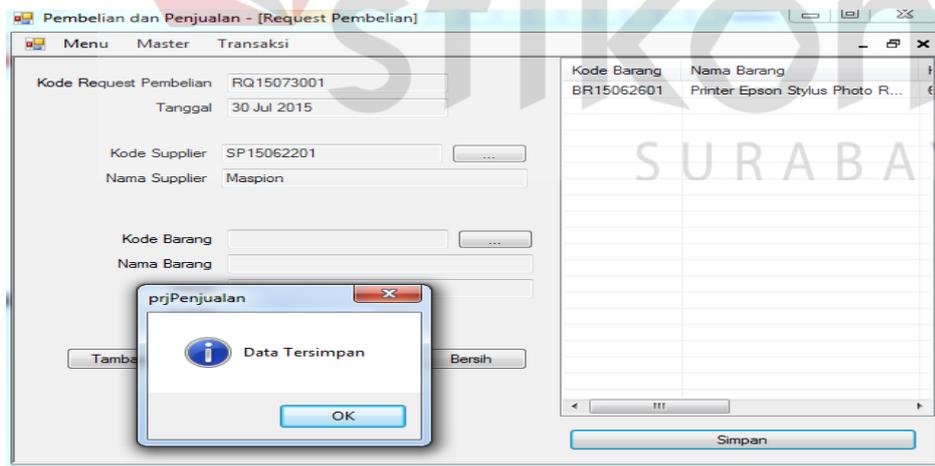
f. Ubah *Supplier*



Gambar 4.25. Ubah *Supplier*

Pada gambar 2.25 ubah *supplier* dimulai dari menekan memilih *supplier* yang akan diubah. Kemudian mengisikan data tentang *supplier* setelah data lengkap selanjutnya menekan tombol *simpan*, akan muncul notifikasi “data tersimpan” artinya data sudah tersimpan pada *database*.

g. Simpan Data Transaksi Request Pembelian

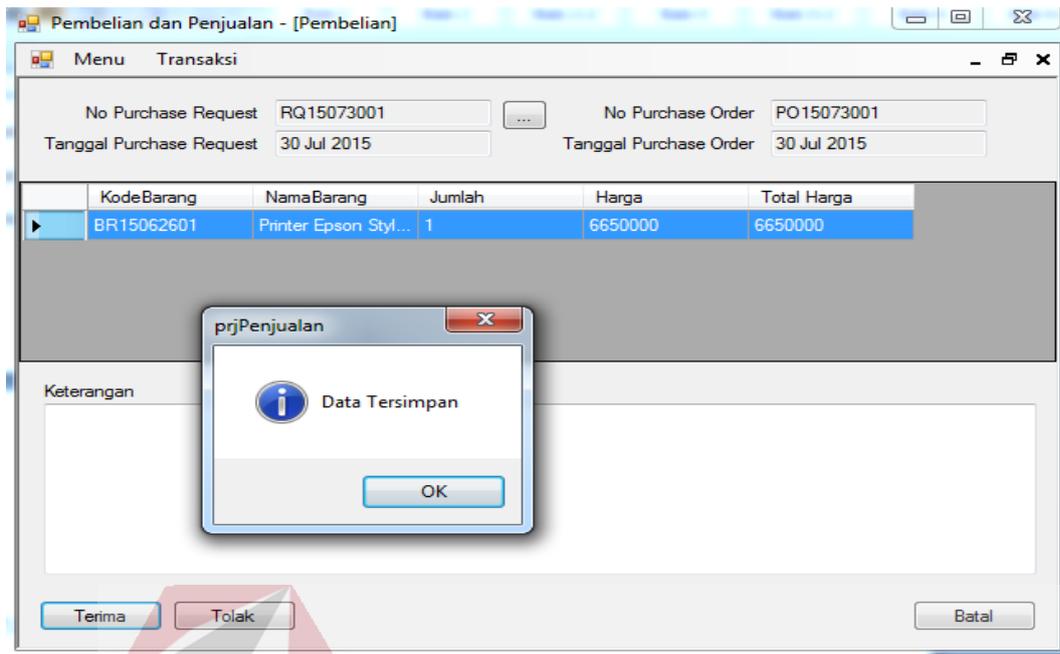


Gambar 4.26. Simpan *Request* pembelian

Pada gambar 4.26 untuk menyimpan data *request* pembelian *user* harus melengkapi data-data barang yang akan di minta. Kemudian tekan tombol *tambah*, jika berhasil data akan



i. *Acc Purchase Request*

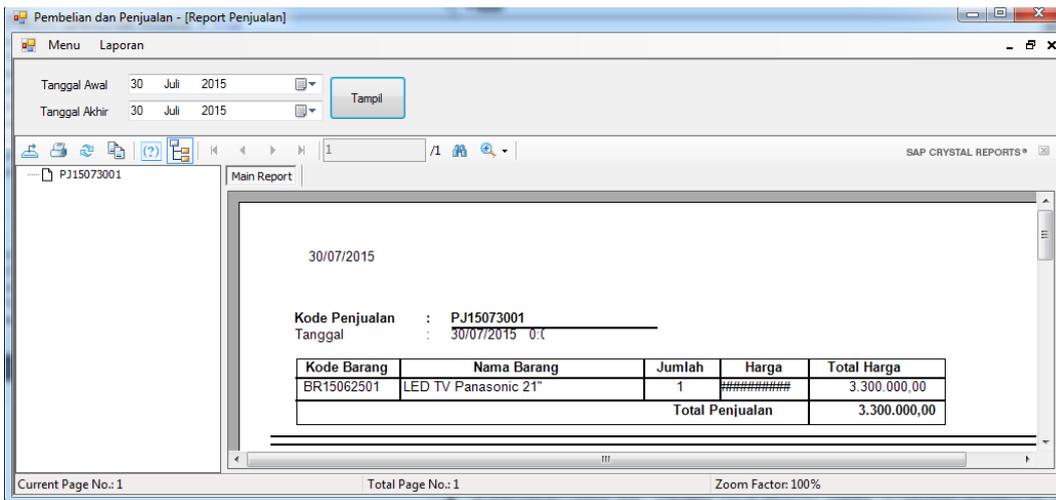


Gambar 4.28. *Acc Purchase Request*

Pada gambar 4.28, proses acc request dimulai dengan memilih no. *Purchase Request* kemudian menampilkan daftar barang yang akan dibeli, jika menerima permintaan maka menekan tombol terima dan kemudian muncul notifikasi "data tersimpan". Dan jika menolak maka harus menulis keterangan barang apa saja yang di tolak pada kolom keterangan.

SURABAYA

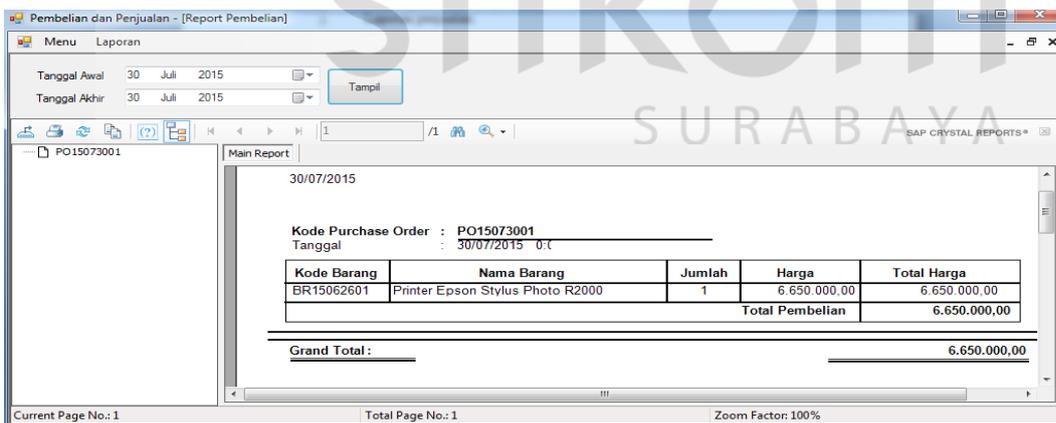
j. Laporan penjualan



Gambar 4.29. Laporan Penjualan

Pada gambar 4.29 laporan penjualan menampilkan hasil dari penjualan yang dilakukan. Untuk menampilkan laporan, *user* harus memilih tanggal yang diinginkan terlebih dahulu, kemudian menekan tombol tampil yang secara otomatis akan menampilkan laporan penjualan.

k. Laporan Pembelian

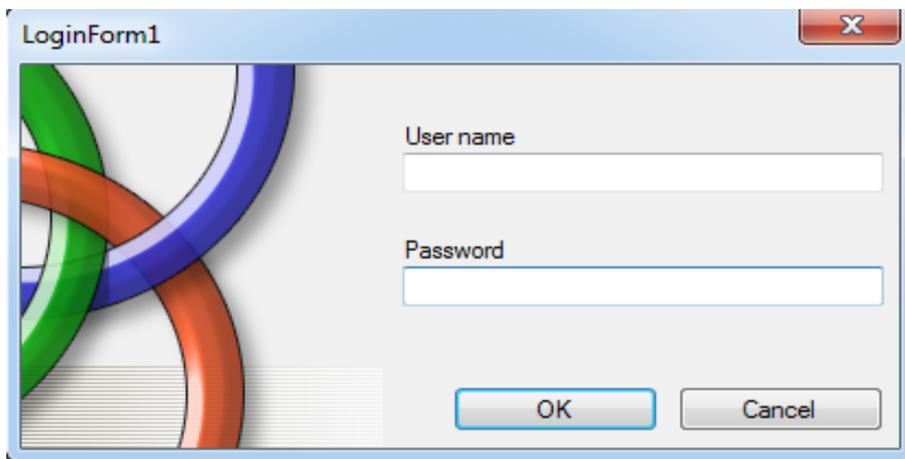


Gambar 4.30. Laporan Pembelian

Pada gambar 4.30 untuk menampilkan laporan pembelian terlebih dahulu memilih tanggal yang diinginkan. Kemudian menekan tombol tampil yang otomatis akan menampilkan laporan pembelian.

## 4.7. Implementasi Sistem

### a. *Login*



Gambar 4.31. Halaman *Login*

Pada Gambar 4.20 *Form login* digunakan untuk masuk ke dalam aplikasi. *User* harus memasukkan *username* dan *password*. Jika *textbox username* dan *password* telah diisi, maka *user* harus menekan tombol *OK*. Dan jika ingin keluar, harus menekan tombol *cancel*. *Form* ini sendiri digunakan oleh 3 *user* yaitu kasir/divisi penjualan, divisi pembelian, divisi gudang dan divisi keuangan, yang masing-masing *user* mempunyai *privilege*. Sebagai contoh divisi penjualan dapat membuka akses ke *master* barang dan transaksi penjualan, divisi gudang dapat mengakses halaman *request* pembelian, divisi pembelian dapat mengakses halaman *acc request* pembelian, halaman pembelian dan halaman penerimaan. Sedangkan untuk divisi keuangan bisa mengakses halaman laporan penjualan dan laporan pembelian.

b. Menu Utama



Gambar 4.32. Gambar Halaman Menu Utama

Pada gambar 4.21 halaman ini digunakan dengan cara memilih menu apa yang akan dipakai. Seperti menu *setup master* dan transaksi yang terletak di divisi atas. Setiap Hasil *login* akan muncul menu sesuai dengan *user* pengguna. Misal *Login* divisi pembelian maka hanya akan muncul menu untuk melakukan acc request pembelian, halaman pembelian dan sebagainya yang merupakan wewenang divisi pembelian.

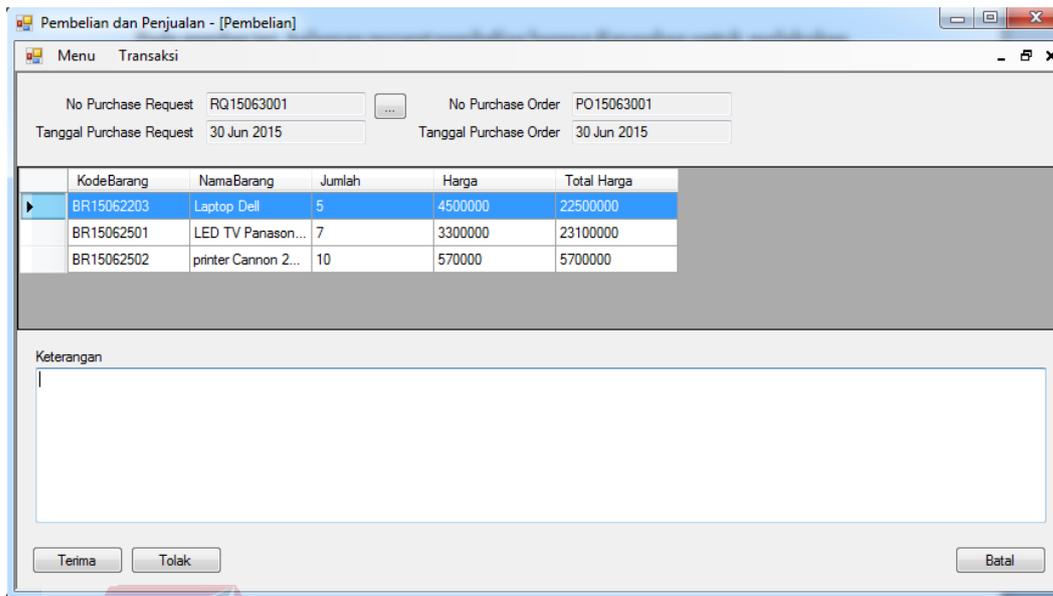
c. *Request* Pembelian Barang

Kode Barang	Nama Barang	Harga	Jumlah	Total Harga
BR15062602	Printer Epson L1800	625000	7	4375000
BR15062603	Printer Epson B510DN	5875000	7	41125000
BR15062611	Dell Inspiron 14z	5500000	5	27500000

Gambar 4.33. Halaman *Request* Pembelian

Pada gambar 4.22, halaman *request* pembelian barang digunakan untuk melakukan permintaan barang, dimana divisi gudang harus memasukkan kode *supplir* yang secara otomatis nama *supplier* akan keluar, kemudian memasukkan kode barang yang secara otomatis nama barang, harga dan jumlah akan keluar. Tombol tambah berfungsi menambah daftar barang yang akan yang akan di-*request*. tombol simpan akan di gunakan untuk menyimpan data *request* pembelian yang nantinya siap untuk di lakukan acc pada halama acc *request* pembelian. penamaan kode *request* RQ15063002, RQ di ambil dari inisal *request*, 15 diambil dari tahun saat itu, 06 diambil dari bulan saat itu, 30 diambil dari tanggal saat itu dan 02 merupakan transaksi ke 2.

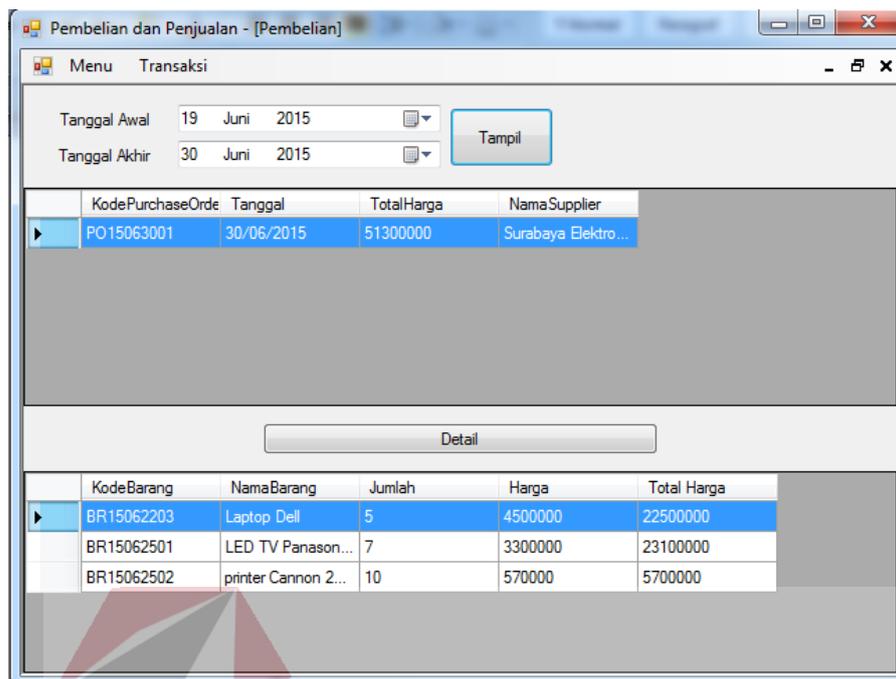
d. Halaman Acc *Request* Pembelian



Gambar 4.34. Halaman Acc *Request* Pembelian

Pada gambar 4.23, halaman acc *request* pembelian berfungsi untuk melakukan persetujuan terhadap request permintaan barang yang sebelumnya diajukan oleh pihak gudang. pihak pembelian pertama kali akan memilih no *Purchase Request* untuk melihat daftar barang yang sudah ada pada tabel *Purchase Request* dan secara otomatis akan tampil pada daftar barang. Tombol terima berfungsi Jika pihak pembelian menerima *request* pembelian dan bisa dilihat pada halaman pembelian, tombol tolak berfungsi menolak 1 transaksi *request* pembelian, harus ada keterangan dari divisi Pembelian dan menuliskan data apa yang harus direvisi. Penamaan nomer *purchase order* diambil berdasarkan inisial *purchase order*, kemudian di urutkan berdasarkan YYMMDD saat itu

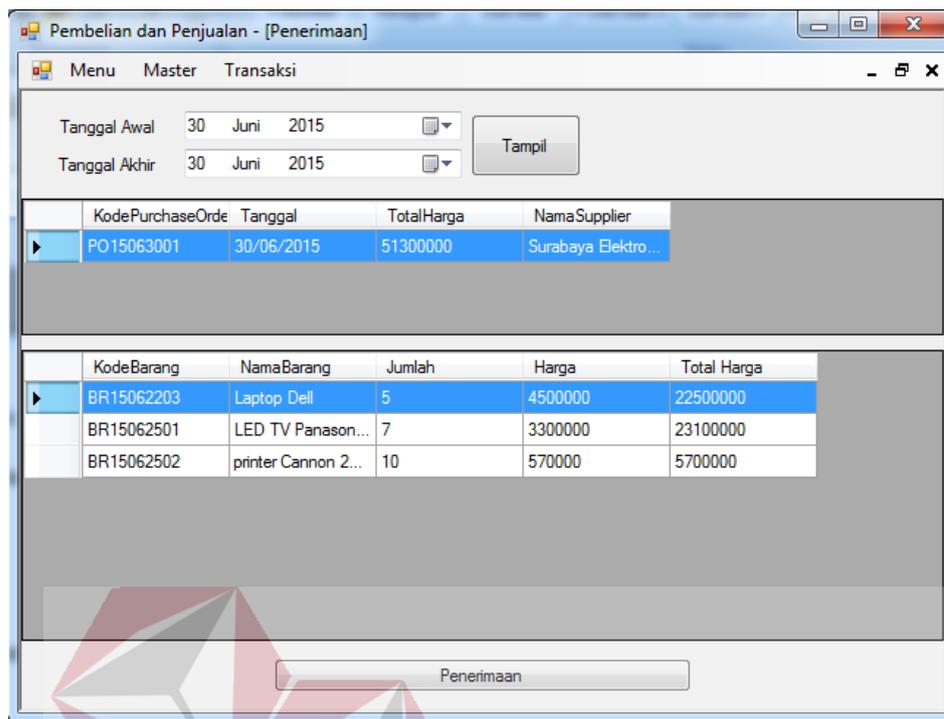
e. Halaman Pembelian



Gambar 4.35. Halaman Pembelian

Halaman pembelian pada berfungsi untuk mengetahui daftar pembelian apa saja yang dilakukan berdasarkan *input*-an tanggal yang diinginkan *user*. Tombol detail berfungsi melihat barang secara detail dari daftar PO dan bisa dilihat pada gambar 4.24.

f. Halaman Penerimaan Barang



Gambar 4.36. Halaman Penerimaan Barang

Pada gambar 4.25, halaman penerimaan berfungsi untuk proses menerima barang yang sudah masuk dalam gudang, gudang sebagai *user* akan melakukan penerimaan jika barang yang sudah dipesan sampai pada gudang. tombol penerimaan berfungsi untuk menyimpan data barang yang sudah sampai ke gudang dan secara otomatis stok pada gudang akan ter-*update*.



h. Halaman Report Penjualan

Perencanaan dan Penjualan - [report: Penjualan]

Menu Laporan

Tanggal Awal: 19 Juni 2015  
Tanggal Akhir: 30 Juni 2015

Tampilkan

SAP CRYSTAL REPORTS

Main Report

Kode Penjualan : PJ15063002  
Tanggal : 30/06/2015 0:00

Kode Barang	Nama Barang	Jumlah	Harga	Total Harga
BR15062202	Printer EPSON	1	800.000,00	800.000,00
Total Penjualan				800.000,00

Kode Penjualan : PJ15063003  
Tanggal : 30/06/2015 0:00

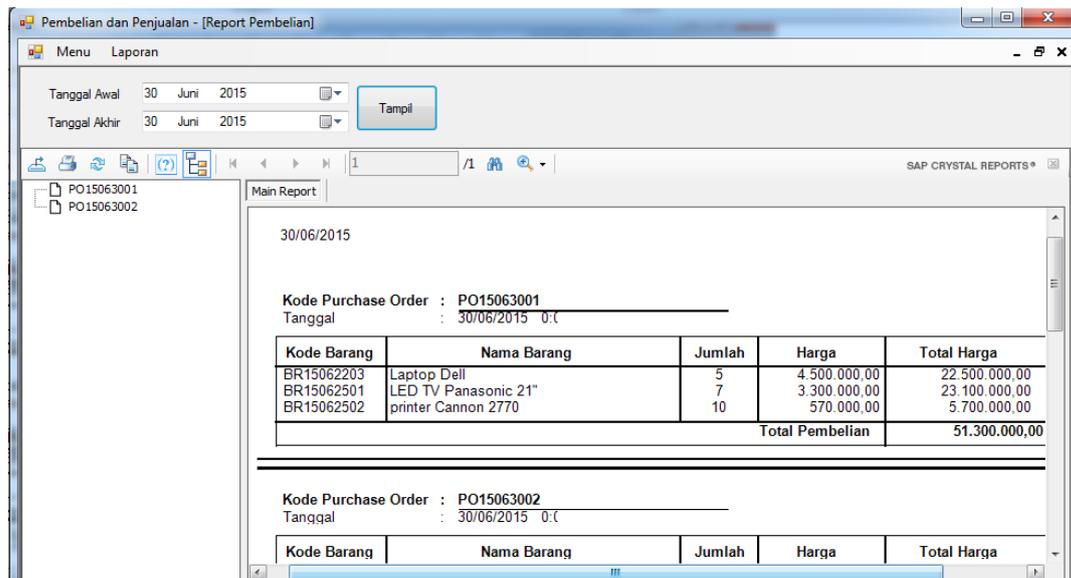
Kode Barang	Nama Barang	Jumlah	Harga	Total Harga
BR15062502	printer Cannon 2770	1	570.000,00	570.000,00
Total Penjualan				570.000,00

Grand Total: 6.085.000,00

Gambar 4.38. Report Penjualan

Didalam halaman *report* penjualan berfungsi sebagai output dari hasil penjualan berupa laporan, untuk menampilkan *report* penjualan pihak keuangan harus memasukkan rentang tanggal yang diinginkan seperti yang terlihat pada gambar 4.27

i. Report pembelian



The screenshot shows a SAP Crystal Reports window titled 'Pembelian dan Penjualan - [Report Pembelian]'. The report is for the date 30/06/2015. It displays two purchase orders: PO15063001 and PO15063002. The first table shows items for PO15063001:

Kode Barang	Nama Barang	Jumlah	Harga	Total Harga
BR15062203	Laptop Dell	5	4.500.000,00	22.500.000,00
BR15062501	LED TV Panasonic 21"	7	3.300.000,00	23.100.000,00
BR15062502	printer Cannon 2770	10	570.000,00	5.700.000,00
Total Pembelian				51.300.000,00

The second table shows items for PO15063002, but only the header is visible:

Kode Barang	Nama Barang	Jumlah	Harga	Total Harga
-------------	-------------	--------	-------	-------------

Gambar 4.39. Report Pembelian

Pada gambar 4.28 *report* pembelian merupakan *output* dari transaksi pembelian. Untuk menampilkan *report* pembelian harus memasukkan rentang tanggal yang diinginkan oleh pihak keuangan sebagai *user*. Didalam laporan tersebut ditampilkan berdasarkan kode PO masing-masing transaksi agar dapat membedakan brapa kali transaksi penjualan dilakukan.

#### 4.8. Kebutuhan Perangkat Lunak

Pada bab ini dijelaskan mengenai implementasi dari perangkat keras dan lunak yang harus dipersiapkan oleh pengguna. Untuk perangkat keras, minimal pengguna harus mempersiapkan spesifikasi sebagai berikut:

1. Intel Pentium Core 2 Duo CPU 3.00 Ghz
2. Memory 2 GB Ram
3. VGA 512 MB

Kebutuhan minimum perangkat lunak untuk aplikasi ini adalah sebagai berikut :

1. Microsoft Windows XP Profesional
2. Microsoft SQL Server 2008

