

PERANCANGAN SISTEM INFORMASI PERSEDIAAN BARANG PADA PT. GARUDA INDAWA TANGERANG

DIAN APRIADI

STMIK Mahakarya

JL. TAMPAK SIRING RAYA BLOK KJE NO. 2 KALIDERES
JAKARTA BARAT 11840

Abstract

The program of inventory information system is made for helping and accelerating the data collection, requesting goods, ordering goods, and returning goods and also making the report fastly, exactly and accurately so that the informations which are given more qualified. In making the system, the writer identified whether the information system of inventory can give the best result or still need the suggestions for repairing that system in increasing data collection proses, the processing and the out put, especially in processing the data information system of inventory at PT.Garuda Indawa Tangerang. The purposes of the research are for increasing writer's knowledge in case of the research it self, programming and implementing a application program based in title, and increasing the efficiency and affectivities the information system that has already. With repairing the result to information system that has already, so can help the data processing of good supplies become more qualified and be better. The system that was suggested in making the computerization system are by using specification. Intel Pentium II 350 GHZ, as a main supporter of data processing, Random Acces Memory (RAM) 64 MB, as a helper of the processing and as a main supporter in the report of data inventory.

Key Word: Information System Of Inventory

A. PENDAHULUAN

1. Latar Belakang Masalah

Dengan meningkatnya kebutuhan akan produk sepatu dalam negeri, dan besarnya minat konsumen terhadap sepatu itu sendiri, dengan segala cara dan usaha semaksimal mungkin pihak perusahaan berusaha memenuhi kebutuhan tersebut. Untuk itu informasi secara cepat, tepat dan akurat sangat diperlukan sebagai bahan pengambilan keputusan. Komputer merupakan salah satu sarana untuk dapat membantu dalam sistem pengolahan data tersebut, akibatnya sistem informasi dengan komputer akan mempunyai nilai lebih, dari sistem yang

diolah secara manual, dapat dikatakan bahwa sistem informasi berbasis komputer akan menunjukkan efisiensi produktifitas yang tinggi.

Salah satu penerapan teknologi komputer adalah dalam pendataan input dan output suatu barang untuk memberikan kemudahan dan keakuratan data yang akan dikerjakan. Dalam pengolahan data *input* dan *output* barang menggunakan sistem pengolahan yang berbasis komputer, sehingga dapat memungkinkan data yang di dapat pun lebih cepat, akurat, dan tepat, maka semuanya itu membantu mempercepat penyelesaian tugas pengolahan data, hasil pendataan barang dan dalam pelaksanaan pengerjaannya.

2 Perumusan Masalah

Masalah dalam penelitian ini adalah sebagai berikut:

- a) Mengidentifikasi apakah 125elati informasi pengolahan data barang *input* dan *output* sudah memenuhi target yang akan dicapai.
- b) Apakah memerlukan usulan dalam peningkatan pengolahan, pendataan terutama yang berhubungan dengan 125elati informasi pengolahan data barang yang ada di persediaan barang .
- c) Dengan perbaikan hasil terhadap 125elati informasi yang ada maka dapat membantu dan mempercepat dalam pendataan barang yang akan dikerjakan dengan tepat dan cepat.
- d) Untuk lebih mempermudah dalam melakukan pekerjaan, dan mempercepat proses pengambilan keputusan yang membutuhkan waktu yang 125elative singkat.

B. LANDASAN TEORI

1. Konsep Dasar Sistem

Menurut: *Jogiyanto HM*, dalam buku *Analisis & Desain Sistem Informasi* menjelaskan definisi sistem adalah:

“ Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan. Berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu”.

(Jogiyanto HM ; 1993 :1)

Definisi diatas dapat dibagi kedalam dua kelompok, yaitu yang menekankan pada prosedur data dan yang menekankan pada

komponen atau elemennya. Kedua kelompok definisi ini adalah benar dan tidak bertentangan, yang berbeda adalah pendekatannya, didalam suatu sistem dapat terdiri dari subsistem atau sistem bagian, sehingga subsistem-subsistem dalam sistem tidak dapat berdiri sendiri. Karena untuk mencapai tujuan atau sasaran, subsistem-subsistem harus berinteraksi dan berhubungan membentuk satu kesatuan yang utuh.

2 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu komponen-komponen (*components*), batas sistem (*boundary*), lingkungan luar sistem (*environment*), Masukan (*input*), keluaran (*output*), pengolah (*proces*), dan sasaran (*objectives*), atau tujuan (*goal*).

Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi. Komponen-komponen sistem atau elemen-elemen sistem berupa suatu sub sistem atau bagian dari sistem.

Batas Sistem

Batas sistem (*Boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya atau dengan lingkungan luarnya. Batas sistem ini menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

Lingkungan Luar Sistem

Lingkungan luar sistem (*environment*) dari suatu sistem adalah apapun yang ada di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem yang dapat bersifat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara.

Penghubung Sistem

Merupakan media penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.

Masukan Sistem

Merupakan (*input*) adalah energi yang dimasukan kedalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Masukan perawatan adalah energi yang dimasukan agar sistem tersebut dapat beroperasi, sedangkan masukan signal adalah energi yang dapat diproses untuk didapatkan keluaran.

Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

Keluaran Sistem

Keluaran (*output*) sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan keluaran dapat berupa tampilan dilayar monitor yaitu dalam bentuk laporan, grafik tabel dan keluaran lainnya adalah hasil cetakan laporan media kertas.

Sasaran Sistem

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran maka operasi sistem itu tidak ada gunanya, suatu sistem dikatakan berhasil jika mengenai sasaran atau tujuannya.

3. Konsep Sistem Informasi

Sistem informasi merupakan salah satu sistem manusia dan mesin yang saling berinteraksi dalam menyediakan informasi untuk melayani kegiatan dari fungsi operasi, manajemen, dan pengambilan keputusan. Informasi yang dibutuhkan oleh manajemen akan sangat berbeda.

Sistem dalam sebuah basis data mempunyai 4 tipe yaitu level operasional, level pengetahuan (*knowledge*), level manajemen, dan level strategis.

a Sistem dengan Level Operasional

Sistem informasi yang memonitor langsung aktivitas dasar

- dan transaksi-transaksi dari organisasi.
- b. Sistem dengan Level *Knowledge* (pengetahuan)
Sistem informasi yang mendukung pengetahuan dan data pekerjaan dalam suatu pekerja.
- c. Sistem dengan Level Manajemen
Sistem informasi yang mendukung monitoring, pengendalian, pengambilan keputusan, dan aktifitas dari manajer tingkat menengah.
- d. Sistem dengan Level Strategis
Sistem informasi yang mendukung aktifitas perencanaan-perencanaan jangka panjang dari manajemen senior (tingkat atas).

4. Diagram Arus Data

Diagram arus data merupakan alat yang cukup populer, karena dapat menggambarkan arus data didalam sistem dengan terstruktur dan jelas. Lebih lanjut diagram arusa data juga merupakan dokumentasi dari sistem yang baik. Diagram arus data sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang ingin dikembangkan secara logika tanpa mempertimbangkan dimana data tersebut mengalir. Keuntungan menggunakan diagram arus data adalah memudahkan pemakai yang kurang menguasai komputer, mengerti sistem yang akan dikerjakan atau dikembangkan.

Bentuk Diagram Arus Data

Diagram arus data mempunyai 2 bentuk, yaitu diagram arus data fisik dan diagram arus data logika.

a. Diagram Arus Data Fisik

Dengan menggunakan Diagram Arus Data Fisik (DADF), bagaimana proses sistem yang ada akan lebih dapat digambarkan dan dikomunikasikan kepada pemakai sistem, sehingga analisa sistem akan dapat memperoleh gambaran yang jelas, bagaimana sistem tersebut bekerja. Diagram arus data fisik lebih tepat digunakan untuk menggambarkan sistem yang ada. Penekanan dari diagram arus data fisik adalah bagaimana proses-proses dari sistem diterapkan (dengan cara apa, oleh siapa, dan dimana) termasuk proses-proses manual.

Untuk memperoleh gambaran bagaimana sistem yang ada diterapkan, DADF harus memuat sebagai berikut :

- a) Proses-proses manual juga digambarkan
 - b) Nama dan arus data juga harus menunjukkan fakta penerapannya semacam nomor formulir dan medianya.
 - c) Simpanan data dapat menunjukkan simpanan non komputer
 - d) Nama dari simpanan data harus menunjukkan nama tipe penerapannya apakah secara manual atau komputerisasi.
 - e) Proses harus menunjukkan nama dari pemroses yaitu orang, departemen, sistem komputer atau nama program komputer yang mengeksekusi program tersebut.
- b. Diagram Arus Data Logika
- Diagram arus data logika lebih tepat digunakan untuk menggambarkan sistem yang akan diusulkan. DADL tidak menekankan pada bagaimana sistem diterapkan, tetapi penekanannya hanya pada logika dari kebutuhan-kebutuhan sistem. Karena sistem yang diusulkan belum tentu diterima oleh pemakai sistem dan biasanya yang diusulkan terdiri dari beberapa alternatif, maka penggambaran dari sistem logika akan lebih mengena dan menghemat waktu penggambarannya apabila dibandingkan dengan DADF dan biasanya proses DADL digambarkan hanya merupakan proses-proses komputer saja.
- c. Simbol-Simbol Diagram Arus Data
- Beberapa simbol yang dipakai diagram arus data untuk maksud mewakili:
- i) Kesatuan Luar
Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* kepada lingkungan luarnya.
 - ii) Arus Data
Arus data diberi simbol anak panah, arus data ini mengalir diantara proses-proses, simpanan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.
 - iii) Proses
Suatu proses adalah kegiatan atau kerja yang dilakukan

oleh orang, mesin atau komputer dari hasil suatu arus data yang akan dihasilkan oleh suatu proses. Suatu proses ditunjukkan oleh lingkaran atau empat persegi panjang tegak dengan sudut-sudutnya tumpul.

iv) Simpanan Data

Simpanan data merupakan simpanan dari data yang berupa *file* atau data base di sistem komputer, arsip atau catatan manual, tabel atau acuan manual dan agenda atau catatan buku. Disimbolkan dengan sepasang garis horizontal paralel tertutup di salah satu ujungnya.

C. METODOLOGI PENELITIAN

1 Metode Penelitian

Analisa sistem yang digunakan adalah analisa sistem yang terstruktur atau terarah dimana transformasi *input* dan *output*nya dinyatakan dengan diagram arus data (DFD) yang diperoleh dengan mengamati kebutuhan informasinya. Pembuatan diagram arus data dilihat berdasarkan aliran informasi dari sistem yang sudah ada dalam tahap ini menggunakan notasi-notasi untuk menggambarkan arus data dari sistem sangat membantu dalam proses komunikasi dengan pemakai (*user*) sistem untuk memahami sistem secara logika tanpa mempertimbangkan perangkat komputer dimana data tersebut mengalir atau dimana data tersebut disimpan untuk proses pengembangan selanjutnya.

2. Analisa Sistem

Masukan yang digunakan untuk proses persediaan barang di PT Garuda Indawa adalah sebagai berikut:

a. Analisa perangkat lunak (*software*) dan perangkat keras (*hardware*)

Pada sistem yang sedang berjalan belum adanya suatu perangkat lunak (*software*) yang begitu berarti, adapun spesifikasi dari perangkat lunak pada sistem yang sedang berjalan diantaranya adalah:

1. *Microsoft Windows* sebagai sistem operasi penunjang utama program *Office* pengolahan data
2. *MS Office* yang digunakan untuk pengetikan ulang dan pemindahan *file* dari buku besar ke dalam komputer

yang selanjutnya akan dicetak sebagai bahan laporan kegiatan.

b. Analisa Tenaga Kerja

Pengolahan data persediaan barang pada PT Garuda Indawa, dilakukan oleh staf-staf (pegawai) yang telah ditunjuk sebelumnya, untuk melaksanakan pencatatan barang persediaan yang masuk dan yang keluar, sampai dengan kegiatan pembuatan laporan.

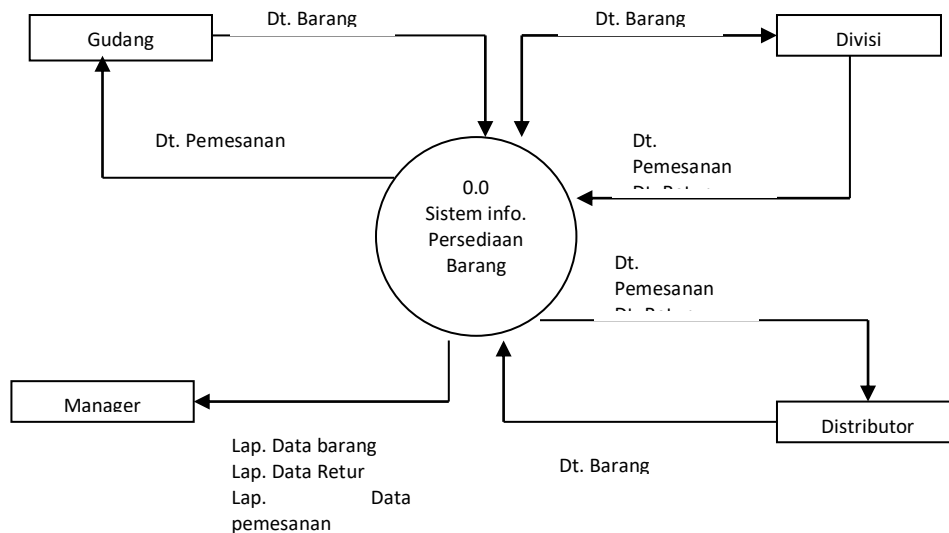
c. Analisa Kebutuhan Sistem

Analisa kebutuhan sistem yang akan dicapai dalam pengolahan data persediaan barang pada PT Garuda Indawa, pada analisa yang sedang berjalan masih mempunyai kelemahan sehingga membutuhkan suatu sistem yang dapat mengatasi permasalahan tersebut yaitu dengan sistem komputerisasi, dengan sistem ini akan mempermudah dalam pengolahan data, salah satunya dalam pengecekan data laporan kegiatan hanya dengan menampilkan program yang telah di buat sedemikian rupa oleh penulis, maka data-data tersebut akan ditampilkan kembali secara cepat, sehingga keakuratan data akan lebih diprioritaskan dan didalam pembuatan laporan tidak mempersulit pengumpul data karena data sudah tersimpan dalam *file-file* yang ada dikomputer.

3. Rancangan Sistem

Rancangan sistem yang sedang berjalan dibuat dengan menggunakan *Diagram Konteks* dan *Data Flow Diagram* (DFD) level I, hal ini dilakukan agar rancangan dari sistem yang sedang berjalan dapat lebih dimengerti dan dipahami. Untuk lebih jelasnya dapat dilihat pada gambar 1 dan 2 sebagai berikut:

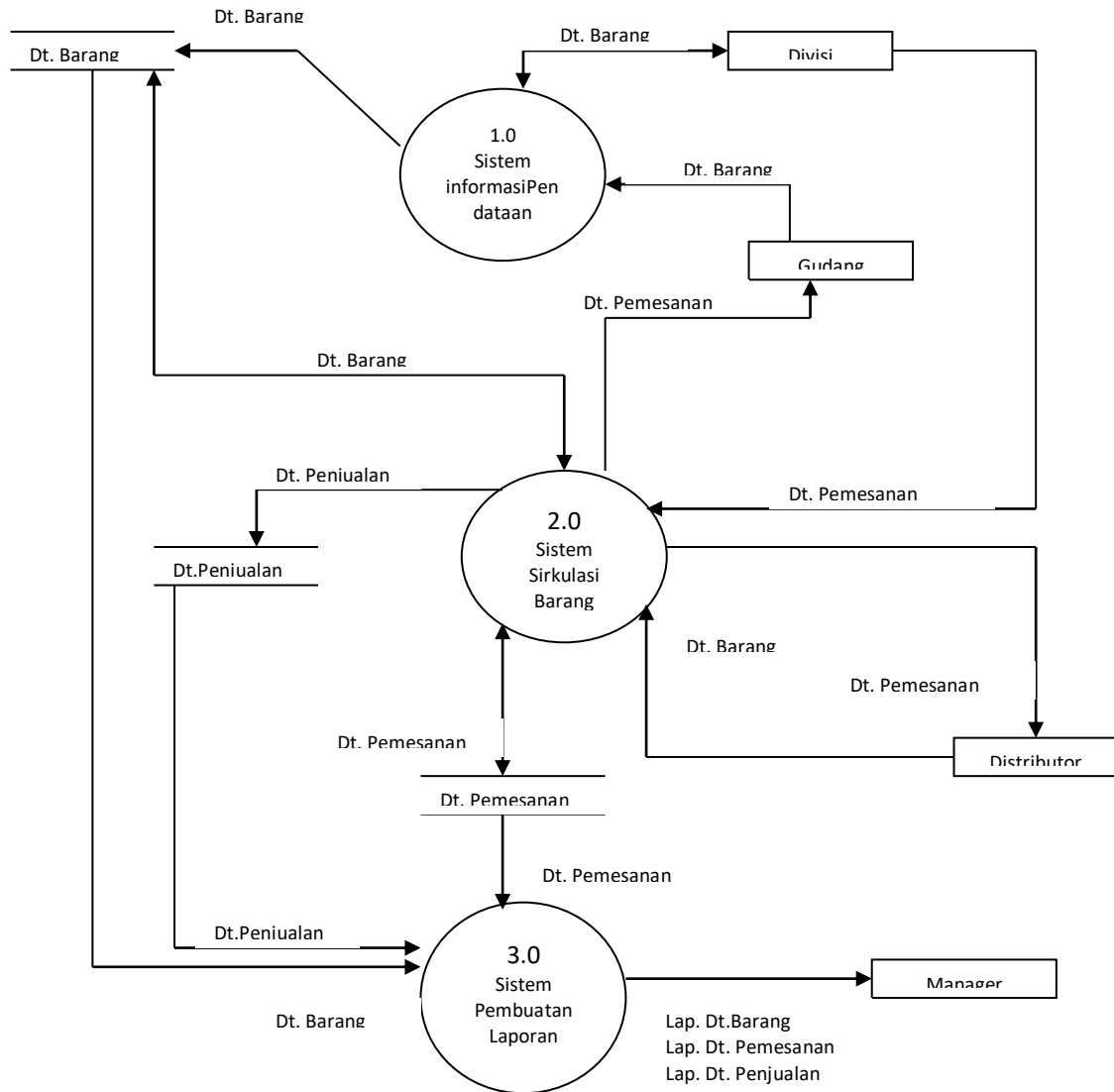
a. Diagram Konteks



Gambar 1 Diagram Konteks Sistem Informasi Persediaan Barang

b. Data Flow Diagram (DFD) level I

Proses pada *Data Flow Diagram* (DFD) level I Sistem Informasi Persediaan Barang pada PT Garuda Indawa, menjelaskan secara lebih terperinci tentang pengolahan data persediaan barang yang dimulai dari proses 1.0 sampai proses 3.0. *Data Flow Diagram* (DFD) level I dapat dilihat pada gambar 2 berikut:



Gambar 2 DFD level I Sistem Informasi Persediaan Barang

D. PEMBAHASAN DAN HASIL PENELITIAN

1. Implementasi Sistem

Pada implementasi sistem ini mencakup Rancangan Basis Data, dan Analisis Implementasi. Perancangan Basis Data meliputi

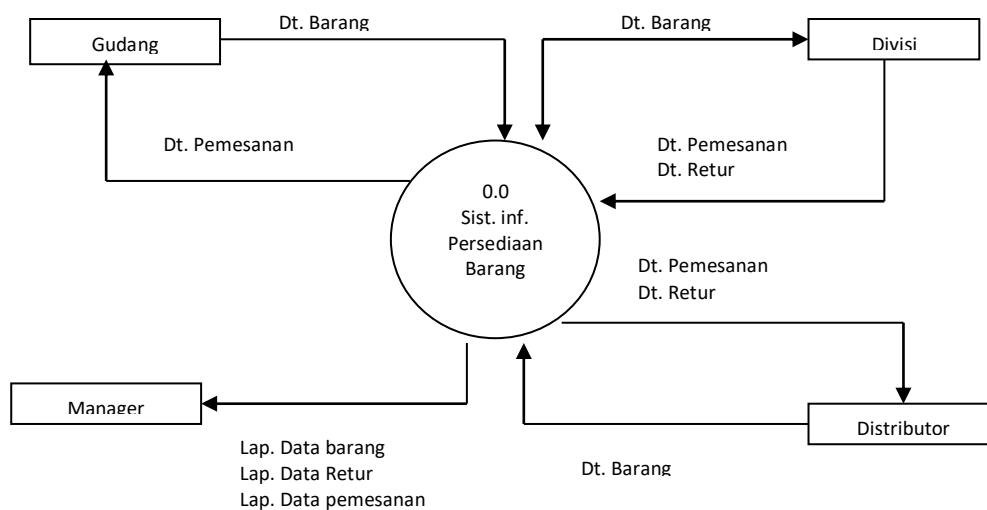
Diagram Konteks, Rancangan Model Proses dengan menggunakan *Data Flow Diagram*, Normalisasi dan Struktur Basis Data, sedangkan untuk Analisis Implementasi meliputi Perancangan Antar Muka.

2 Perancangan Basis Data

Perancangan Basis Data untuk sistem yang diusulkan ini disusun mulai dari diagram konteks dan dikembangkan lagi kedalam bentuk relasi *Data Flow Diagram (DFD)* yang dihasilkan pada analisa data, kemudian dilakukan tahap normalisasi untuk menghilangkan elemen-elemen yang berulang-ulang dalam file serta mengurangi kesalahan seminimal mungkin dalam basis data tersebut yang selanjutnya diubah kedalam bentuk *E-RD (Entity Relationship Diagram)*. Melalui langkah-langkah diatas basis data yang dihasilkan menjamin ketepatan waktu, hemat dalam pembiayaan dan pemeliharanya.

Diagram Konteks

Pada Diagram *konteks* ini, proses aliran data dijelaskan secara keseluruhan, sehingga kita dapat mengetahui pada diagram *konteks* mempunyai empat entitas yaitu ; bagian gudang memberikan data pemesanan barang dan data barang, Divisi memberikan data barang, data retur, dan data pemesanan barang, Manager memberikan laporan data barang, laporan data retur, dan laporan data pemesanan. Distributor memberikan data pemesanan, data retur, dan data barang. Untuk lebih jelasnya dapat dilihat pada gambar 3



Gambar 3. Diagram *Konteks* Sistem Informasi Persediaan Barang

3. Normalisasi

Perancangan basis data sangat diperlukan, agar kita dapat memiliki basis data yang baik dan efisien dalam penggunaan ruang penyimpanan, cepat dalam pengaksesan dan mudah dalam melakukan manipulasi (penambahan, pengubahan dan penghapusan)

Dalam merancang basis data, kita dapat melakukan dengan dua cara yaitu:

- a. Menerapkan teknik Normalisasi terhadap struktur tabel yang ada
- b. Membuat model *Entity Relationship*.

Definisi Normalisasi

Normalisasi adalah proses pengelompokan data kedalam bentuk tabel atau relasi atau file untuk menyatakan entitas dan hubungan mereka sehingga terwujud satu bentuk *database* yang mudah untuk dimodifikasi. Pada proses normalisasi selalu diuji pada beberapa kondisi. Apakah ada kesulitan selama proses *delete*, *insert*, atau *update*, dalam basis data yang disebut sebagai penyimpangan-penyimpangan atau dikenal dengan *Anomaly*.

Macam-Macam Penyimpangan (*Anomaly*)

- a. *Insertion Anomaly*
Yaitu *Error* atau kesalahan yang terjadi sebagai akibat operasi menyisipkan *tuple/record* pada sebuah relasi.
- b. *Deletion Anomaly*
Yaitu *error* atau kesalahan yang terjadi sebagai akibat operasi penghapusan terhadap *tuple/record* dari sebuah relasi.
- c. *Update Anomaly*
Yaitu *error* atau kesalahan yang terjadi sebagai akibat operasi perubahan *tuple/record* dari sebuah relasi.
- d. *Composite Key*
Adalah *key* yang terdiri dari dua *atribut* atau lebih, dimana *atribut-atribut* tersebut bila berdiri sendiri tidak menjadi identitas *record*, tetapi bila dirangkaikan menjadi satu kesatuan yang dapat mengidentifikasi *record* secara *unique*.

e. *Foreign Key*

Adalah *non key atribut* pada sebuah relasi lainnya. *Foreign key* biasanya digunakan sebagai penghubung antara *record-record* dari kedua relasi tersebut.

4. Perancangan Basis Data

Pengembangan Sistem Informasi Persediaan Barang di PT Garuda Indawa menggunakan 4 buah tabel yaitu sebagai berikut:

Tabel 1 *File Basis Data Persediaan Barang*

No	Nama File Basis Data	Keterangan
1	Data Barang.DB	<i>File basis data yang berisikan tentang data persediaan barang yang ada, kode barang, item, quantity, color, dan size (lihat pada tabel 4.2)</i>
2	Data Permintaan.DB	<i>File basis data yang berisikan tentang data permintaan barang dari pembeli yang berisi tentang pemesanan barang (lihat pada tabel 4.3)</i>
3	Data Order.DB	<i>File basis data yang berisikan tentang data order antar bagian yang diberikan kepada bagian produksi dan eksekusi akhir (lihat pada tabel 4.4)</i>
4	Data Retur.DB	<i>File basis data yang berisikan tentang data pengembalian barang dari konsumen (lihat pada tabel 4.5)</i>

1. Data Barang .DB

Data Barang ini adalah tabel data untuk menyimpan data Barang

Nama tabel : Data Barang.DB

Panjang *record* : 100 *byte*

Field kunci : Kode Barang

Fungsi : Untuk mencatat data barang

Tabel 2 Data Persediaan Barang.DB

<i>Field</i>	<i>Data Type</i>	<i>Field size</i>	<i>Description</i>
Kd_ Barang*	<i>Text</i>	10	Menampung data barang dan sebagai <i>primary</i>

			<i>key</i>
Item	<i>Text</i>	25	Menampung data item
Quantity	<i>Text</i>	25	Menampung data Quanttity
Color	<i>Text</i>	25	Menampung data clor
Size	<i>Text</i>	15	Menampung data size
Junlah record		100	

2. Data Permintaan.DB

Data Permintaan ini adalah tabel data untuk menyimpan data-data Permintaan

Nama tabel : Data Permintaan.DB

Panjang *record* : 70 *byte*

Field kunci : Kode Permintaan

Fungsi : Untuk mencatat data-data Permintaan

Tabel 3 Data Permintaan Barang. DB

<i>Field</i>	<i>Data Type</i>	<i>Field size</i>	<i>Description</i>
Kode permintaan*	<i>Text</i>	10	Menampung data permintaan dan sebagai <i>primary key</i>
Kode barang	<i>Text</i>	10	Menampung data permintaan
Tanggal permintaan	<i>Text</i>	15	Menampung data tanggal permintaan
No order	<i>Text</i>	10	Menampung data order
Jumlah barang	<i>Text</i>	25	Menampung data jumlah barang
Panjang <i>record</i>		70	

3. Data Order.DB

Data order adalah tabel data untuk menyimpan data-data order barang antar departemen barang.

Nama tabel : Data Order.DB

Panjang *record* : 100 *byte*

Field kunci : Kode Order

Fungsi : Order barang antar departemen

Tabel 4 Data Order.DB

<i>Field</i>	<i>Data Type</i>	<i>Field size</i>	<i>Description</i>
Kode Order	<i>Text</i>	10	Menampung data order dan sebagai <i>primary key</i>
Kd_Barang	<i>Text</i>	10	Menampung kode Barang
Nama Departemen	<i>Text</i>	25	Menampung data departemen
Tanggal order	<i>Text</i>	15	Menampung data tanggal order
Jumlah barang	<i>Text</i>	25	Menampung data jumlah barang
Stok	<i>Text</i>	15	Menampung data stok
Jumlah record		100	

4. Data Retur.DB

Data retur adalah tabel data untuk menyimpan data-data retur barang.

Nama tabel : Data Retur.DB

Panjang record : 195 byte

Field kunci : No retur

Fungsi : Retur barang

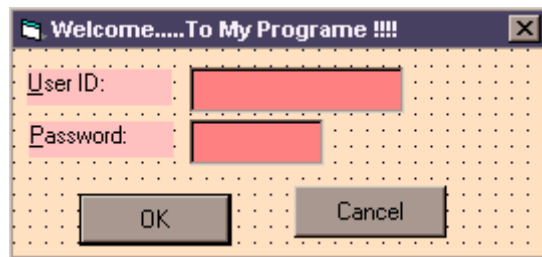
Tabel 5 Data Retur Barang. DB

<i>Field</i>	<i>Data Type</i>	<i>Field size</i>	<i>Description</i>
No retur	<i>Text</i>	10	Menampung data retur dan sebagai <i>primary key</i>
Tanggal retur	<i>Text</i>	15	Menampung tanggal retur
Nama Pembeli	<i>Text</i>	25	Menampung data pembeli
Jalan	<i>Text</i>	25	Menampung data jalan
Kota	<i>Text</i>	25	Menampung data kota
NPWP	<i>Text</i>	15	Menampung data

			NPWP
Kode barang	<i>Text</i>	10	Menampung data barang
Item	<i>Text</i>	25	Menampung data item
Kwantum	<i>Text</i>	15	Menampung data kwantum
Harga satuan	<i>Text</i>	15	Menampung harga satuan
Harga satuan yang dikembalikan	<i>Text</i>	15	Menampung harga satuan yang dikembalikan
Jumlah <i>record</i>		195	

5. Perancangan Antar Muka Form Login Sistem

Rancangan antar muka untuk *login* sistem seperti yang ditunjukkan dalam gambar 5.1



Gambar 4 Rancangan Form Antar Muka *Login* Sistem

Pada rancangan antar muka *login* sistem ini digunakan agar data yang ada dalam program tersimpan dengan aman. Untuk lebih jelasnya dari rancangan antar muka *form login* tersebut seperti pada keterangan di bawah ini :

1. *User ID* : Untuk menuliskan nama pembuat program
2. *Password* : Untuk menuliskan nama *password* yang akan dipakai oleh *User*
3. *Command Button OK* : Untuk melanjutkan proses aplikasi
4. *Command Button Cancel* : Untuk membatalkan perintah

Form Input Data Barang

Gambar 5 Rancangan Form Antar Muka Input Data Barang

Pada *form input* data ini di gunakan untuk memasukan data barang. Untuk lebih jelasnya dapat kita lihat pada keterangan di bawah ini :

- Kode Barang
Untuk memasukan data kode barang
- Item
Untuk menuliskan jenis barang atau nama barang
- Quantity*
Untuk menuliskan jumlah barang
- Color*
Untuk menuliskan data color/warna
- 1. *Size* : Untuk Menuliskan data *Size*/ukuran barang
- 2. *Timer*: Untuk memberi waktu pada animasi label
- 3. *Adodc1* : Sebagai koneksi data dengan tabel data
- 4. *First* : Sebagai tombol perintah untuk memindahkn ke data pertama
- 5. *Next* : Sebagai perintah untuk memindahkan data selanjutnya
- 6. *Last* : Sebagai perintah untuk memindahkan data ke terakhir

7. *Add* : Sebagai perintah untuk menambahkan data
8. *Edit* : Sebagai perintah untuk merubah data
9. *Previous* : Sebagai perintah untuk memindahkan ke data sebelumnya
10. *Cancel* : Sebagai tombol perintah batal
11. *Delete* : Sebagai tombol perintah untuk melakukan penghapusan
12. *Save* : Sebagai tombol perintah untuk menyimpan data
13. *Exit* : Sebagai tombol perintah untuk keluar dari tampilan yang sedang dibuka

Form Data Permintaan

Rancangan antar muka untuk melihat data Permintaan seperti yang ditunjukkan dalam gambar 6

Gambar 6 Rancangan Form Antar Muka Data Permintaan

- Kode Permintaan : Untuk menuliskan kode permintaan
 Kode barang : Untuk menuliskan kode barang
 Tanggal permintaan : Untuk menuliskan tanggal permintaan
 No order : Untuk menuliskan no order
 Jumlah barang : Untuk menuliskan banyaknya jumlah barang yang dipesan

6. Implementasi Antar Muka *Form* Data Order

Implementasi antar muka *form* menu utama *entry* data order dapat dilihat pada gambar 7.

Gambar 7 Implementasi Antar Muka Data Order

Pembuatan *Form* Menu utama *Entry* Data Order

Pada pembuatan *form* menu utama *entry* data order barang hampir sama dengan pembuatan *form* data permintaan, yang membedakan hanyalah pada fungsi *label* dan penulisan *caption*, jumlah *label* yang dibutuhkan dan penulisan *caption*, yaitu:

- Label ke-1, *caption* : Kode order
- Label ke-2, *caption* : Kode barang
- Label ke-3, *caption* : Nama Departemen
- Label ke-4, *caption* : Tanggal order
- Label ke-5, *caption* : Jumlah barang
- Label ke-6, *caption* : Stok

Adapun kode program untuk *form* data order ini adalah:

```
Private Sub Form_Load()
```

```
save.Enabled = False
```

```
cancel.Enabled = False
```

```
add.Enabled = True
```

```
edit.Enabled = True
```

```
first.Enabled = True
```

```
last.Enabled = True
```

```
nex.Enabled = True
previous.Enabled = True
delete.Enabled = True
End Sub
```

- *CommandButton1, Caption :&First*
 Berfungsi sebagai tombol perintah untuk menampilkan data pertama.
 Masukan kode program di bawah ini:

```
Private Sub first_Click()
If Not ado_order.Recordset.RecordCount = 0 Then
ado_order.Recordset.MoveFirst
End If
End Sub
```
- *CommandButton2, Caption :&Next*
 Berfungsi sebagai tombol perintah untuk menampilkan data berikutnya.
 Masukan kode program di bawah ini:

```
Private Sub nex_Click()
ado_order.Recordset.MoveNext
If ado_order.Recordset.EOF Then
MsgBox (" Data Telah Diakhir"), vbOKOnly, "Message"
ado_order.Recordset.MoveLast
End If
End Sub
```
- *CommandButton3, Caption :&Last*
 Berfungsi sebagai tombol perintah untuk data sebelumnya.
 Masukan kode program di bawah ini:

```
Private Sub last_Click()
If Not ado_order.Recordset.RecordCount = 0 Then
ado_order.Recordset.MoveLast
End If
End Sub
```
- *CommandButton4, Caption :&Add*
 Berfungsi sebagai tombol perintah untuk menambahkan data.
 Masukan kode program di bawah ini:

```
Private Sub add_Click()
    ado_order.Recordset.AddNew
    Text1.Enabled = True
    Text2.Enabled = True
    Text3.Enabled = True
    Text4.Enabled = True
    Text5.Enabled = True
    Text6.Enabled = True
    Text1.SetFocus
    add.Enabled = False
    nex.Enabled = False
    first.Enabled = False
    last.Enabled = False
    previous.Enabled = False
    edit.Enabled = False
    delete.Enabled = False
    cancel.Enabled = True
    save.Enabled = True
End Sub
```

- *CommandButton5, Caption :&Previous*
 Berfungsi sebagai tombol perintah untuk menampilkan data berikutnya.
 Masukan kode program di bawah ini:

```
Private Sub previous_Click()
    ado_order.Recordset.MovePrevious
    If ado_order.Recordset.BOF Then
        MsgBox ("Data Telah Diawal"), vbOKOnly, "Message"
    End If
    End Sub
```
- *CommandButton6, Caption :&Edit*
 Berfungsi sebagai tombol perintah untuk mengubah data.
 Masukan kode program di bawah ini:

```
Private Sub edit_Click()
    ado_order.Recordset.Update
    nex.Enabled = False
    first.Enabled = False
    last.Enabled = False
```

```
previous.Enabled = False
delete.Enabled = False
cancel.Enabled = True
save.Enabled = True
edit.Enabled = False
add.Enabled = False
Text1.Enabled = True
Text2.Enabled = True
Text3.Enabled = True
Text4.Enabled = True
Text5.Enabled = True
Text6.Enabled = True
Text1.SetFocus
End Sub
```

- *CommandButton7, Caption :&Cancel*
 Berfungsi sebagai tombol perintah untuk membatalkan perintah.
 Masukkan kode program di bawah ini:

```
Private Sub cancel_Click()
ado_order.Recordset.CancelUpdate
ado_order.Recordset.MoveFirst
Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False
Text6.Enabled = False
add.Enabled = True
edit.Enabled = True
first.Enabled = True
last.Enabled = True
nex.Enabled = True
previous.Enabled = True
delete.Enabled = True
save.Enabled = False
cancel.Enabled = False
End Sub
```

- *CommandButton8, Caption :&Delete*

Berfungsi sebagai tombol perintah untuk menghapus data.
Masukan kode program di bawah ini:

```
Private Sub delete_Click()
If MsgBox("YAKIN NEH MAU DI HAPUS", vbYesNo,
"Konfirmasi") = vbYes Then
ado_order.Recordset.delete
ado_order.Recordset.MoveFirst
MsgBox ("Data kembali ke Awal"), vbOKOnly, "Message"
End If
End Sub
```

- *CommandButton9, Caption :&Save*

Berfungsi sebagai tombol perintah untuk menyimpan data.
Masukan kode program di bawah ini:

```
Private Sub save_Click()
ado_order.Recordset.save
ado_order.Recordset.MoveLast
save.Enabled = False
cancel.Enabled = False
add.Enabled = True
edit.Enabled = True
first.Enabled = True
last.Enabled = True
nex.Enabled = True
previous.Enabled = True
delete.Enabled = True
Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False
Text6.Enabled = False
End Sub
```

- *CommandButton10, Caption :&Exit*

Berfungsi sebagai tombol perintah untuk keluar dari tampilan form.
Masukan kode program di bawah ini:

```
Private Sub exit_Click()
MDIForm1.Show
Me.Hide
End Sub
```

- *CommandButton11, Caption :&Timer*
Berfungsi sebagai tombol perintah untuk animasi *text*.
Masukan kode program di bawah ini:
Private Sub Timer1_Timer()
If l1.Left > -2520 Then
l1.Left = l1.Left - 500
Else
l1.Left = 6840
End If
End Sub

Analisis implementasi *form data order*

Analisis implementasi dari implementasi antar muka *form data order* hampir sama dengan analisis implementasi *form data permintaan*, yang membedakan hanyalah pada nama dari setiap *label*-nya saja.

Implementasi Antar Muka *Form Data Retur*

Implementasi antar muka *form menu utama entry data retur* dapat dilihat pada gambar 8.

Gambar 8 Implementasi Antar Muka Data Retur

Pembuatan *Form* Menu utama *Entry* Data retur

Pada pembuatan *form* menu utama *entry* data retur barang hampir sama dengan pembuatan *form* data order, yang membedakan hanyalah pada fungsi label dan penulisan *caption*, jumlah *label* yang dibutuhkan dan penulisan *caption*, yaitu:

- *Label ke-1, caption* :No retur
- *Label ke-2, caption* :Tanggal retur
- *Label ke-3, caption* : Nama pembeli
- *Label ke-4, caption* : Jalan
- *Label ke-5, caption* : Kota
- *Label ke-6, caption* : NPWP
- *Label ke-7, caption* : Kode barang
- *Label ke-8, caption* : Item
- *Label ke-9, caption* :Kwantum
- *Label ke-10, caption* : Harga satuan
- *Label ke-11, caption* : Harga satuan yang dikembalikan

Adapun kode program untuk *form* data retur ini adalah:

```
Private Sub Form_Load()
```

```
save.Enabled = False
```

```
cancel.Enabled = False
```

```
add.Enabled = True
```

```
edit.Enabled = True
```

```
first.Enabled = True
```

```
last.Enabled = True
```

```
nex.Enabled = True
```

```
previous.Enabled = True
```

```
delete.Enabled = True
```

```
End Sub
```

- *CommandButton1, Caption* :&First

Berfungsi sebagai tombol perintah untuk menampilkan data pertama.

Masukan kode program di bawah ini:

```
Private Sub first_Click()
```

```
If Not ado_retur.Recordset.RecordCount = 0 Then
```

```
ado_retur.Recordset.MoveFirst
```

```
End If
```

```
End Sub
```

- *CommandButton2, Caption :&Next*
 Berfungsi sebagai tombol perintah untuk menampilkan data berikutnya.
 Masukkan kode program di bawah ini:

```
Private Sub nex_Click()
ado_retur.Recordset.MoveNext
If ado_retur.Recordset.EOF Then
MsgBox (" Data Telah Diakhir"), vbOKOnly, "Message"
ado_retur.Recordset.MoveLast
End If
End Sub
```
- *CommandButton3, Caption :&Last*
 Berfungsi sebagai tombol perintah untuk data sebelumnya.
 Masukkan kode program di bawah ini:

```
Private Sub last_Click()
If Not ado_retur.Recordset.RecordCount = 0 Then
ado_retur.Recordset.MoveLast
End If
End Sub
```
- *CommandButton4, Caption :&Add*
 Berfungsi sebagai tombol perintah untuk menambahkan data.
 Masukkan kode program di bawah ini:

```
Private Sub add_Click()
ado_retur.Recordset.AddNew
Text1.Enabled = True
Text2.Enabled = True
Text3.Enabled = True
Text4.Enabled = True
Text5.Enabled = True
Text6.Enabled = True
Text7.Enabled = True
Text8.Enabled = True
Text9.Enabled = True
Text10.Enabled = True
Text11.Enabled = True
Text1.SetFocus
add.Enabled = False
nex.Enabled = False
first.Enabled = False
```

- ```
last.Enabled = False
previous.Enabled = False
edit.Enabled = False
delete.Enabled = False
cancel.Enabled = True
save.Enabled = True
End Sub
```
- *CommandButton5, Caption :&Previous*  
Berfungsi sebagai tombol perintah untuk menampilkan data berikutnya.  
Masukan kode program di bawah ini:  

```
Private Sub previous_Click()
ado_retur.Recordset.MovePrevious
If ado_retur.Recordset.BOF Then
MsgBox ("Data Telah Diawal"), vbOKOnly, "Message"
ado_retur.Recordset.MoveFirst
End If
End Sub
```
  - *CommandButton6, Caption :&Edit*  
Berfungsi sebagai tombol perintah untuk mengubah data.  
Masukan kode program di bawah ini:  

```
Private Sub edit_Click()
ado_retur.Recordset.Update
nex.Enabled = False
first.Enabled = False
last.Enabled = False
previous.Enabled = False
delete.Enabled = False
cancel.Enabled = True
save.Enabled = True
edit.Enabled = False
add.Enabled = False
Text1.Enabled = True
Text2.Enabled = True
Text3.Enabled = True
Text4.Enabled = True
Text5.Enabled = True
Text6.Enabled = True
Text7.Enabled = True
```

```
Text8.Enabled = True
Text9.Enabled = True
Text10.Enabled = True
Text11.Enabled = True
Text1.SetFocus
End Sub
```

- *CommandButton7, Caption :&Cancel*  
 Berfungsi sebagai tombol perintah untuk membatalkan perintah.  
 Masukkan kode program di bawah ini:  

```
Private Sub cancel_Click()
ado_retur.Recordset.CancelUpdate
ado_retur.Recordset.MoveFirst
Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False
Text6.Enabled = False
Text7.Enabled = False
Text8.Enabled = False
Text9.Enabled = False
Text10.Enabled = False
Text11.Enabled = False
add.Enabled = True
edit.Enabled = True
first.Enabled = True
last.Enabled = True
nex.Enabled = True
previous.Enabled = True
delete.Enabled = True
save.Enabled = False
cancel.Enabled = False
End Sub
```
- *CommandButton8, Caption :&Delete*  
 Berfungsi sebagai tombol perintah untuk menghapus data.  
 Masukkan kode program di bawah ini:  

```
Private Sub delete_Click()
If MsgBox("YAKIN NEH MAU DI HAPUS", vbYesNo,
```

```
"Konfirmasi") = vbYes Then
ado_retur.Recordset.delete
ado_retur.Recordset.MoveFirst
MsgBox ("Data kembali ke Awal"), vbOKOnly, "Message"
End If
End Sub
```

○ *CommandButton9, Caption :&Save*

Berfungsi sebagai tombol perintah untuk menyimpan data.  
Masukan kode program di bawah ini:

```
Private Sub save_Click()
ado_retur.Recordset.save
ado_retur.Recordset.MoveLast
save.Enabled = False
cancel.Enabled = False
add.Enabled = True
edit.Enabled = True
first.Enabled = True
last.Enabled = True
nex.Enabled = True
previous.Enabled = True
delete.Enabled = True
Text1.Enabled = False
Text2.Enabled = False
Text3.Enabled = False
Text4.Enabled = False
Text5.Enabled = False
Text6.Enabled = False
Text7.Enabled = False
Text8.Enabled = False
Text9.Enabled = False
Text10.Enabled = False
Text11.Enabled = False
End Sub
```

○ *CommandButton10, Caption :&Exit*

Berfungsi sebagai tombol perintah untuk keluar dari tampilan *form*.

Masukan kode program di bawah ini:

```
Private Sub exit_Click()
```

```
MDIForm1.Show
Me.Hide
End Sub
```

- *CommandButton11, Caption :&Timer*  
 Berfungsi sebagai tombol perintah untuk animasi *text*.  
 Masukkan kode program di bawah ini:  
 Private Sub Timer1\_Timer()  
 If 11.Left > -2520 Then  
 11.Left = 11.Left - 500  
 Else  
 11.Left = 6840  
 End If  
 End Sub

## E. KESIMPULAN DAN SARAN

### 1. Kesimpulan

Dengan menyelesaikan pembuatan aplikasi dan menganalisa permasalahan yang terjadi selama perancangan sistem informasi persediaan barang di PT. Garuda Indawa Tangerang, maka penulis dapat mengambil kesimpulan sebagai berikut:

1. Perancangan Sistem Informasi Persediaan Barang merupakan suatu perancangan sistem yang bertujuan untuk membuat aplikasi tentang pengolahan data persediaan yang ada di gudang. Dalam hal ini mencakup beberapa kegiatan yaitu pengadaan barang, permintaan barang, order barang, dan retur barang, sampai pembuatan laporan kegiatan yang dilakukan.
2. Implementasi Perancangan Sistem Informasi Persediaan Barang di buat dengan menggunakan *Database* yang dibuat terlebih dahulu pada *Microsoft Acces* dan dihubungkan melalui fasilitas *Adodc* yang sudah ada dalam paket *Visual Basic 6.0*.
3. Didalam pembuatan sistem aplikasi persediaan barang ini diperlukan suatu analisis tentang kinerja sistem. Hal ini bertujuan untuk mengetahui akan kesalahan dan kekurangan sistem agar dapat diperbaharui kearah yang lebih baik.

### 2. Saran

Saran dalam penelitian ini yang dapat peneliti berikan adalah :

1. Penulis berharap untuk pengembangan selanjutnya *software* (perangkat lunak) didalam pembuatan program aplikasi akan lebih canggih, sehingga dalam pembuatan sistem akan menghasilkan program aplikasi yang berkualitas serta menjadi lebih baik dan sempurna.
2. Program yang dibuat dapat dijalankan dalam sistem jaringan. Adapun untuk keamanan data kita lakukan dengan menggunakan sistem keamanan dengan menggunakan *Password* yang lebih canggih, seperti *password* berbentuk suara atau menggunakan sidik jari pengguna (*user*), sehingga orang lain diluar sistem tidak dapat mengakses program dan data menjadi lebih aman.

#### DAFTAR PUSTAKA

- Jefkins, Frank, 1995, "Public Relations", Erlangga, Jakarta.
- Hartono, Jogiyantom, 1999, "Pengenalan Komputer ", Andi Offset, Yogyakarta.
- Kurniadi, Adi, 1999, "Pemrograman Microsoft Visual Basic 6.0 ", PT. Elex Media Komputindo Gramedia, Jakarta.
- Sutanta, Edhy, 2004, "Sistem Basis Data ", Konsep dan Peranannya dalam sistem Informasi Manajemen, Andi Offset, Yogyakarta.
- Fatansyah, Ir, 2004, "Basis Data", Informatika Bandung, Bandung.
- Manullang, M, 2001, "Dasar-Dasar Manajemen", Gadjah Mada University Press. Yogyakarta.