

BAB II

PEMBAHASAN

2.1. Teori Pendukung

A. Konsep Dasar Pemrograman

Menurut Kurniadi (2004:346), “ program adalah sekelompok instruksi yang dijalankan oleh komputer untuk melakukan tugas-tugas tertentu. Bahasa pemrograman komputer merupakan sarana komunikasi yang menghubungkan antara manusia dengan komputer ”.

Menurut Sutabri (2003:1), “ Dasar program adalah suatu urutan instruksi-instruksi dalam bahasa komputer yang disusun secara logis dan sistematis yang bertujuan untuk memecahkan suatu masalah dan memudahkan pekerjaan yang diinginkan oleh pemakai (*user*) ”.

Bahasa pemrograman komputer dikelompokkan menjadi dua kelompok besar, yaitu bahasa pemrograman tingkat rendah (*Low Level Language*) dan bahasa pemrograman tingkat tinggi (*High Level Language*). Dengan bahasa pemrograman komputer, manusia dapat membuat sebuah program yang dimanfaatkan untuk membantu menyelesaikan permasalahannya.

Saat ini setelah munculnya sistem operasi dan lingkungan *Windows*, bahasa pemrograman lebih ditekankan dalam hal tampilannya (*visual*). Sarana

pengembangannya pun bersifat grafis, sehingga pemakai hanya perlu menggeser gambar-gambar saja dengan *mouse*. Oleh sebab itu, banyak bahasa pemrograman dinamai dengan “*visual*” didepannya. Contohnya *Visual FoxPro*, *Visual C++*, dan *Visual Basic* sendiri dan lain sebagainya.

Selain itu, setiap bahasa pemrograman yang berbasis *visual* kini tidak lagi menggunakan orientasi linear (segaris) di dalam pembuatan programnya, melainkan dengan berorientasi pada objek-objek yang terpisah. Pemrograman tersebut dikenal dengan nama ***Objek Oriented Programming (OOP)***. OOP memiliki konsep *Modulator Programming*, dimana kode-kode program letaknya tersebar di dalam modul-modul (objek-objek) yang terpisah.

B. Teori Merancang Program

Teknik pemrograman yang baik memiliki ciri-ciri atau karakteristik sebagai berikut :

1. Mengandung algoritma pemecahan masalah yang tepat, benar, sederhana, standar dan efektif.
2. Memiliki struktur logika dan struktur program yang benar dan mudah dipahami.
3. Membutuhkan biaya *testing*, pemeliharaan dan pengembangan yang rendah.
4. Memiliki dokumentasi yang baik.

Untuk menyusun suatu program komputer ada beberapa langkah yang harus dilakukan oleh pemrogram atau *programmer* yaitu :

1. Menganalisa Masalah

Untuk menemukan dan memahami persoalan, kemudian dikembangkan dalam proses urutan logika untuk menyelesaikan masalah dalam bentuk algoritma.

2. Mendefinisikan Masalah

Langkah ini dilakukan untuk menentukan banyak *input*, *ouput* yang diinginkan dan sebagai gambaran tentang data yang akan diproses serta informasi yang dihasilkan. penjelasan tentang hal ini diperlukan yang sesuai dengan kebutuhan.

3. Menyusun *Flowchart*

Flowchart ini disusun dengan menggunakan simbol-simbol untuk menceritakan aktifitas data yang akan diolah menjadi informasi.

4. Menyusun Kode

Merupakan pengkodean dari algoritma yang dibuat, diterjemahkan ke dalam bentuk *statement-statement* yang sesuai dengan bahasa pemrograman yang digunakan.

5. Melakukan *Test Program*

Dari proses logika yang dibuat, diperiksa apakah program tersebut sudah benar dan bebas dari unsur kesalahan atau masih harus diperbaiki kembali. Semua kesalahan yang terjadi diperbaiki agar program komputer dapat dijalankan dan memberi hasil seperti yang diharapkan.

6. Dokumentasi Program

Dokumentasi yang dihasilkan adalah berupa catatan-catatan atau keterangan-keterangan yang dibuat untuk menjelaskan urutan suatu program, yang berkaitan dengan proses data yang akan dilakukan pada program tersebut. Dengan adanya dokumentasi yang lengkap, maka dapat diketahui fungsi dari suatu program, bentuk *input*, *process* serta susunan *output*nya.

C. Definisi Program *Microsoft Visual Basic*

Microsoft Visual Basic 6.0 adalah bahasa pemrograman yang digunakan untuk membuat suatu aplikasi dalam *Microsoft Windows* yang berbasis grafis ***Graphical User Interface (GUI)***, *visual basic* adalah salah satu program aplikasi yang termasuk dalam bahasa pemrograman tingkat tinggi dan berbasis OOP. *Visual Basic* memudahkan pemrograman untuk berinteraksi langsung dengan elemen-elemen untuk setiap bentuk pemrograman. *Visual Basic* juga merupakan pemrograman terkendali kejadian (*even-driven programming*), artinya program menunggu sampai dengan adanya respon dari pemakai berupa *event* atau kejadian tertentu (seperti tombol diklik, menu dipilih, dan lain-lain). Dan ketika *event* terdeteksi, kode yang berhubungan dengan *event* tersebut (*procedur event*) akan dijalankan.

Beberapa kemampuan dari *Visual Basic* diantaranya seperti :

1. Untuk membuat program aplikasi berbasis *Windows*.
2. Untuk membuat objek-objek pembantu seperti misalnya kontrol *Active*, *File Help*, aplikasi internet, dan sebagainya.

3. Menguji program (*debugging*) dan menghasilkan program akhir berekstensi EXE yang bersifat *Executable*, atau dapat langsung dijalankan.

BASIC (*Beginner's All Purpose Symbolic Instruction Code*) adalah suatu bahasa pemrograman “kuno” yang merupakan awal dari bahasa-bahasa pemrograman tingkat tinggi lainnya. BASIC dirancang tahun 1950-an dan ditujukan untuk dapat digunakan oleh para *programmer* pemula. *Visual Basic* masih dapat mempertahankan sintaks atau format penulisan yang pernah dipakai oleh BASIC.

Visual Basic mempunyai beberapa kelebihan dari pemrograman linear (BASIC), diantaranya seperti :

1. Lebih Cepat (*Faster*)

Tidak seperti pemrograman linear, *Visual Basic* tidak perlu mengetikkan kode program untuk setiap objek. Untuk membuat objek dalam *Visual Basic* hanya perlu memiliki dan mengklik pilihan-pilihan yang disediakan dengan menggunakan *mouse*, sehingga waktu untuk membuat program aplikasi pun menjadi lebih singkat.

2. Resiko Kesalahan Kecil (*Small Risk on Error*)

Karena lebih sedikit mengetik, secara otomatis kesalahan juga relatif lebih kecil. Sintaks pemrograman juga tidak perlu dihafalkan lagi, karena semuanya telah disediakan di dalam menu pilihan yang tinggal dipilih sesuai dengan kebutuhan. Demikian juga waktu *debugging* akan terpotong sangat besar, karena setiap objek tidak perlu di *debug* setiap kali digunakan

3. Daur Ulang (*Program Recycle*)

Setiap objek dapat digunakan berulang-ulang dalam program yang sama maupun program yang lain. Dengan mudah objek bisa digunakan ulang, dipanggil atau disisipkan dimana saja.

4. Perawatan Lebih Mudah (*Easier Maintenance*)

Dalam BASIC seandainya ingin merubah sifat dari sebuah modul program, tentunya harus merubah seluruh program dan semua yang ikut dipengaruhi oleh modul tersebut. Tetapi di dalam *Visual Basic*, perubahan pada suatu desain atau modul program tidak perlu melakukan perubahan besar-besaran pada program secara keseluruhan. Karena setiap objek berdiri sendiri dan tidak perlu berhubungan dengan lingkungannya, maka perubahan yang dilakukan oleh objek tersebut tidak mempengaruhi komponen lain dalam sistem yang ditempatinya.

Selain dari bermacam kemampuan dan keuntungan, *Visual Basic* juga handal dalam mengolah dan mengelola basis data atau *database*. *Database* merupakan kumpulan data yang terdiri atas satu atau lebih tabel yang terintegrasi satu sama lain, dimana setiap *user* diberi wewenang (*otoritas*) untuk dapat mengakses (mengubah, menghapus, menganalisa dan memperbaiki) data dalam tabel tersebut

Adapun sistem yang mengatur sebuah *database* dinamakan ***Database Management System (DBMS)***. DBMS itu sendiri merupakan Kumpulan *software* yang mengkoordinasikan semua kegiatan yang berhubungan dengan basis data agar data dapat diakses/dipakai oleh pemakai (*user*). Tujuannya adalah efisiensi dan kenyamanan dalam memperoleh dan menyimpan informasi di dalam basis data.

Menurut Jogiyanto (2005a:711), “ *database* merupakan kumpulan data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya ”.

Pembuatan program tentunya tidak terlepas dari tahapan-tahapan yang harus dikerjakan secara terstruktur untuk membantu program dalam menyelesaikan programnya dengan baik.

Langkah-langkah pembuatan suatu program *Visual Basic* mulai dari pertama sampai dengan program tersebut dapat dijalankan, adalah sebagai berikut :

1. Terlebih dahulu di dalam proses pemrograman digambarkan bentuk *flowchart* untuk membantu memudahkan program yang rumit.
2. Membuat *source* program dengan menggunakan software *Microsoft Visual Studio 6.0* dan mengklik *Visual Basic 6.0* kemudian memilih *Standar Exe* dari menu pilihan untuk memulai pembuatan program, yang akan tampak dilayar terminal dan hasilnya disimpan di *diskette* maupun *harddisk*.
3. Bahasa pemrograman *Visual Basic* merupakan *interpreter* untuk menterjemahkan *source* program yang disimpan di media penyimpanan dan apabila terjadi suatu kesalahan maka otomatis akan muncul akan kesalahan tersebut dan harus segera diperbaiki sebelum kompilasi dengan *compiler* yang akan menghasilkan program yang siap dijalankan (*file executable*).
4. *Executable* program yang dapat dijalankan, apabila pada langkah ini masih ada kesalahan lagi yang tidak dapat dideteksi oleh *compiler* tetapi muncul sewaktu *executable* dijalankan. Dan kesalahan ini disebut *runtime error*, yang terjadi

bukan karena kesalahan penulisan program tetapi terjadi karena kesalahan proses-proses lainnya. Bila terjadi kesalahan seperti ini maka *source* program harus diperbaiki kembali, maka kompilasi diulang kembali.

D. Peralatan Pendukung

Penggunaan peralatan pendukung sangat penting untuk menggambarkan dan menjelaskan tentang alur atau cara kerja dari program itu sendiri. Dalam penulisan ini peralatan pendukung yang digunakan adalah :

1. Normalisasi

Normalisasi menurut Jogiyanto (2005b:403) adalah “ proses pengelompokkan data elemen menjadi tabel-tabel yang menunjukkan *entity* dan relasinya”. Normalisasi dikemukakan pertama kali oleh E.F. Codd tahun 1970 mengemukakan tentang tingkat normalisasi. Bentuk normalisasi tidak hanya terdiri dari tiga tingkatan saja tetapi ada bentuk BCNF, bentuk keempat dan bentuk kelima. Semakin tinggi tingkatan normalisasi maka dianggap lebih baik dari tingkatan di bawahnya.

Dalam perspektif normalisasi, sebuah basis data dapat dikatakan baik, jika setiap tabel yang menjadi unsur pembentuk basis data tersebut juga telah berada dalam keadaan baik dan normal.

Pada normalisasi ini ada beberapa konsep yang harus diketahui seperti atribut kunci dan ketergantungan kunci. Ada beberapa macam kunci (*key function*) yang

digunakan untuk proses pencarian, penyaringan, hapus dan lain sebagainya yang biasa digunakan di dalam pengolahan *database*, yaitu sebagai berikut :

a. Kunci Calon (*Candidate Key*)

Adalah satu atribut atau satu set minimal atribut yang mengidentifikasi secara unik suatu kejadian yang spesifik dari suatu *entity*.

b. Kunci Utama (*Primary Key*)

Adalah satu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian yang spesifik, akan tetapi juga dapat mewakili setiap kejadian dari suatu *entity*.

c. Kunci Alternatif (*Alternate Key*)

Adalah kunci kandidat yang tidak dipakai sebagai *primary key*.

d. Kunci Tamu (*Foreign Key*)

Adalah satu atribut atau satu set minimal atribut yang mewakili satu hubungan (*relationship*) yang menunjukkan ke induknya.

e. *Super Key*

Adalah himpunan dari atribut atau lebih *entitas* yang dapat digunakan untuk mengidentifikasi secara unik sebuah *entitas* di dalam *entitas set*.

Adapun beberapa tahapan normalisasi file adalah sebagai berikut :

a. Bentuk Tidak Normal (*Unnormalized Form*)

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti suatu format tertentu. Dapat saja data tidak lengkap atau terduplikasi.

Data dikumpulkan apa adanya sesuai dengan saat menginput.

b. Bentuk Normal Kesatu (1 NF / *First Normal Form*)

Bentuk relasi 1 NF jika dan hanya jika dari setiap relasi atributnya bersifat *atomic*. Atom adalah zat terkecil yang masih memiliki sifat induknya, bila dipecah lagi maka ia tidak memiliki sifat induknya.

Ciri-ciri 1 NF :

- 1) Setiap data dibentuk dalam *flat file*, data dibentuk dalam satu record demi satu record nilai dari *field* berupa *atomic value*.
- 2) Tidak ada set atribut yang berulang atau bernilai ganda.
- 3) Tiap *field* hanya satu pengertian.

c. Bentuk Normal Kedua (2 NF / *Second Normal Form*)

Bentuk normal kedua mempunyai syarat yaitu bentuk data telah memenuhi kriteria bentuk normal kesatu. Atribut bukan kunci haruslah bergantung secara fungsional pada kunci utama/*primary key*. Sehingga untuk membentuk normal kedua haruslah sudah ditentukan kunci-kunci *field*. Kunci *field* harus unik dan dapat mewakili atribut lain yang menjadi anggotanya.

d. Bentuk Normal Ketiga (3NF / *Third Normal Form*)

Untuk menjadi normal ketiga maka relasi harus memenuhi syarat normal kedua dan semua atribut bukan primer tidak punya hubungan transitif. Dengan kata lain, setiap atribut bukan kunci haruslah bergantung hanya pada *primary key* secara menyeluruh.

e. *Boyce Codd Normal Form* (BCNF)

Untuk menjadi BCNF, relasi harus dalam bentuk kesatu dan setiap atribut harus bergantung fungsi pada atribut *super key*.

f. Bentuk Normal Keempat (4 NF / *Fourth Normal Form*)

Relasi R adalah bentuk normal keempat jika dan hanya jika relasi tersebut juga termasuk BCNF dan semua ketergantungan *multivalued* adalah juga ketergantungan fungsional.

g. Bentuk Normal Kelima (5 NF / *Fifth Normal Form*)

Dari normalisasi keempat dilakukan dengan menghilangkan ketergantungan *join* yang bukan merupakan kunci kandidat.

Dalam tugas akhir ini sampai kepada bentuk normal ketiga karena masih ada kerangkapan data pada normal kedua.

2. HIPO (*Hierarchy Plus Input-Process-Output*)

Menurut Al Fatta (2007:147) menyatakan bahwa “HIPO merupakan teknik untuk mendokumentasikan pengembangan suatu sistem yang dikembangkan oleh IBM”. Akan tetapi sekarang, HIPO juga banyak digunakan sebagai alat disain dan teknik dokumentasi dalam siklus pengembangan sistem.

3. Bagan Alir (*Flowchart*)

Flowchart menurut Jogiyanto (2005c:753) adalah “ bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika”. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

Pada waktu akan menggambar suatu bagan alir, pemrogram dapat mengikuti pedoman-pedoman sebagai berikut :

- a. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
- b. Kegiatan di dalam bagan alir ditunjukkan dengan jelas.
- c. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhirnya.
- d. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
- e. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
- f. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
- g. Gunakanlah simbol-simbol bagan alir yang standar.

Ada lima macam bagan alir yaitu :

1) Bagan Alir Sistem (*System Flowchart*)

Adalah bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem, menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem dan menunjukkan apa yang dikerjakan di sistem.

2) Bagan Alir Dokumen (*Document Flowchart*)

Disebut juga bagan alir formulir (*document flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

3) Bagan Alir Skematik (*Schematic Flowchart*)

Adalah bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Perbedaannya adalah bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan.

4) Bagan Alir Program (*Program Flowchart*)

Adalah bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program terdiri dari dua macam yaitu :

a) Bagan Alir Logika Program (*Program Logic Flowchart*)

Digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Bagan alir ini dipersiapkan oleh analisis sistem.

b) Bagan Alir Program Komputer Terinci (*Detailed Computer Program Flowchart*)

Digunakan untuk menggambarkan instruksi-instruksi program komputer secara terinci. Bagan alir ini dipersiapkan oleh pemrogram.

5) Bagan Alir Proses (*Process Flowchart*)

Adalah bagan alir yang banyak di teknik industri.

Dalam penulisan tugas akhir ini, penulis menggunakan program *flowchart* sebagai alat pendukung. Adapun teknik pembuatan program *flowchart*, sebagai berikut :

a. *General Way*

Merupakan teknik pembuatan *flowchart*, dengan cara ini lazim digunakan dalam menyusun logika suatu program yang menggunakan pengulangan proses secara tidak langsung (*Non Direct Loop*).

b. *Iteration Way*

Merupakan teknik pembuatan *flowchart*, dengan cara ini biasanya dipakai untuk logika program yang cepat serta bentuk permasalahan yang kompleks.

4. Pengkodean

Kode menurut Jogiyanto (2005d:384) digunakan : “untuk tujuan mengklasifikasikan data, memasukan data ke dalam komputer dan untuk mengambil bermacam-macam informasi yang berhubungan dengannya”. Adapun teknik pengkodean, sebagai berikut :

a. Petunjuk Pembuatan Kode

1) Harus mudah diingat

Supaya kode mudah diingat, maka dapat dilakukan dengan cara menghubungkan kode tersebut dengan objek yang diwakili dengan kodenya. Misalnya QGZX328 untuk kuliah system informatika akan sulit untuk diingat. Kode yang terlalu panjang sebaiknya dipecah menjadi bagian-bagian

yang lebih pendek. Misalnya kode 033656231 akan mudah diingat bila ditulis 033-656-231.

2) Harus Unik

Kode harus unik untuk masing-masing item yang diwakilinya. Unik berarti tidak ada kode yang kembar.

3) Harus fleksibel

Kode harus fleksibel sehingga memungkinkan perubahan-perubahan atau penambahan item baru dapat tetap diwakili oleh kode.

4) Harus Efisiensi

Kode harus sependek mungkin, selain mudah diingat juga akan efisien bila direkam disimpanan luar computer. Misalnya panjang dari kode cukup sepanjang 4 digit saja dan tidak akan efisien bila dipergunakan kode yang lebih dari 4 digit.

5) Harus Konsisten

Bilamana mungkin, kode harus konsisten dengan kode yang telah dipergunakan. Misalnya perusahaan hanya membeli barang dagangan dari seorang pemasok saja, maka dapat dipergunakan kode-kode barang yang sudah dipergunakan oleh pemasok.

6) Harus distandarisasi

Kode harus distandarisasi untuk seluruh tingkatan dan departemen dalam organisasi. Kode yang tidak standar akan melibatkan kebingungan, salah pengertian dan dapat cenderung terjadi kesalahan pemakaian bagi yang

menggunakan kode tersebut. Misalnya kode untuk tanggal harus standar, yaitu tanggal, bulan dan tahun atau bulan, tanggal dan tahun.

7) Spasi dihindari

Spasi didalam kode sebaiknya dihindari, karena dapat menyebabkan kesalahan didalam menggunakannya.

8) Hindari karakter yang mirip

Karakter-karakter yang hampir serupa bentuk dan bunyi pengucapannya sebaiknya tidak digunakan dalam kode. Misalnya O, I, Z, S dan V dapat membingungkan dengan angka 0, 1, 2, 5 dan huruf U.

9) Panjang kode harus sama

Masing-masing kode yang sejenis harus mempunyai panjang yang sama. Misalnya panjang dari kode adalah 6 digit, maka kode 8210E sebaiknya ditulis 08210E.

b. Tipe dari Kode

1) Kode Mnemonik

Kode mnemonik digunakan untuk tujuan supaya mudah diingat. Kode mnemonik dibuat dengan dasar singkatan atau mengambil sebagian karakter dari item yang akan diwakili dengan kode ini. Misalnya kode “P” untuk mewakili pria dan kode “W” untuk wanita akan mudah diingat.

2) Kode Urut

Kode urut disebut juga dengan kode seri merupakan kode yang nilainya urut antara satu kode dengan kode berikutnya.

Contoh:

001 Kas

002 Piutang dagang

Kelebihannya:

- a) Sangat sederhana
- b) Mudah diterapkan
- c) Kode dapat pendek tetapi unik
- d) Mudah dicari bila kodenya diketahui
- e) Cocok untuk rekaman di file yang menggunakan nomer record relatif, sehingga nomer record dapat sama dengan kodenya, dengan demikian *file* tidak perlu diindeks.
- f) Baik untuk pengendalian, karena kode yang hilang dapat mudah diketahui.

Kelemahannya:

- a) Penambahan kode hanya dapat ditambahkan pada akhir urutan dan tidak dapat disisipkan.
- b) Tidak mempunyai dasar logika tentang informasi item yang diwakilinya, kecuali hanya berdasarkan urutannya saja.
- c) Tidak fleksibel bila terjadi perubahan kode.

3) Kode Blok

Kode blok mengklasifikasi item kedalam kelompok blok tertentu yang mencerminkan satu klasifikasi tertentu atas dasar pemakaian maksimum yang diharapkan.

Contoh rekening-rekening dalam buku besar dapat diberi kode dengan mengklasifikasikan kedalam kelompok rekening utama sebagai berikut:

Blok	Kelompok
1000-1999	Aktiva Lancar
2000-2999	Aktiva Tetap
3000-3999	Hutang Lancar
4000-4999	Modal

Dari blok-blok kode untuk masing-masing kelompok rekening utama, maka rekening-rekening Aktiva Lancar dapat mempunyai kode diantara 1000-1999 sebagai berikut:

1000	Kas
1100	Piutang Dagang
1200	Persediaan Produk Selesai
1210	persedian Produk Dalam Proses

Kelebihannya:

- Nilai dari kode mempunyai arti, yaitu masuk dalam blok yang sudah tertentu.
- Mudah diperluas.
- Kode dapat ditambah atau dibuang sebagian.

- d) Proses pembuatan laporan keuangan dapat dilakukan dengan lebih mudah, karena tiap-tiap kelompok rekening dapat diketahui dari blok kodenya.

Kekurangannya:

- a) Panjang kode tergantung dari jumlah bloknya, akibatnya kode menjadi cukup panjang.
- b) Kurang mudah diingat.

4) Kode Group

Kode group merupakan kode berdasarkan field-field dan tiap-tiap *field* kode mempunyai arti. Kalau anda mengamati buku-buku teks, maka akan terlihat suatu kode yang disebut dengan ISBN(International Standart Book Number) yang terdiri dari 10 digit terbagi 4 *field*. ISBN merupakan kode group yang masing-masing *field* mempunyai arti tertentu.

Kelebihannya:

- a) Nilai dari kode mempunyai arti.
- b) Mudah diperluas.
- c) Dapat ditambah atau dibuang sebagian.
- d) Dapat menunjukkan jenjang dari data.

Kekurangannya:

- a) Kode dapat menjadi panjang.

5) Kode Desimal

Kode desimal mengklasifikasi kode atas dasar 10 unit angka desimal dimulai dari angka 0 sampai dengan angka 9 atau dari 00 sampai dengan 99 tergantung dari banyaknya kelompok.

Contoh:

0. Aktiva Lancar

00100 Kas

00200 Piutang Dagang

1. Aktiva Tetap

01100 Tanah

01200 Bangunan Kantor

Beberapa macam kode-kode komputer yang digunakan dari komputer generasi pertama hingga generasi sekarang, yaitu:

a. Kode Hollerith

Diciptakan oleh DR. Herman Hollerith, memiliki susunan dan aturan tertentu.

b. BCD Code

Kode ini disusun kombinasi 4 buah digit biner, hanya memuat simbol angka saja dan maksimal terbentuk 2^4 , tapi hanya 10 digunakan.

c. SBCDIC (*Standard Binary Coded Decimal Interchange Code*)

Kode ini disusun dari kombinasi 6 buah digit binary di tambah satu digit untuk parity check, mulai digunakan pada komputer generasi kedua.

d. EBCDIC (*Extented Binary Coded Desimal Interchange Code*)

Kode ini disusun dari kombinasi 8 buah digit binary di tambah satu digit untuk parity check, mulai digunakan pada komputer generasi ketiga.embuat

e. ASCII (*American Standard Code for Information Interchange*)

Kode ASCII bertujuan untuk membuat kode binar standar yang dikembangkan oleh ANSI (*Amerika Nasional Standard Information*).

2.2. Analisa Perancangan Program

Dalam pembuatan program penjualan *hardware* komputer ini diawali dengan pembuatan database yang meliputi tabel kasir, tabel barang dan tabel penjualan, selanjutnya perancangan *form* untuk masing-masing data tersebut, setelah perancangan *form* selesai, kemudian menuliskan *listing program* sesuai dengan proses-proses yang dijalankan. Dan disini juga dijelaskan tentang uraian rancangan pembuatan program yang terinci dalam sub-sub bab yang dimulai dari spesifikasi bentuk masukan dan keluaran, normalisasi file, spesifikasi file, spesifikasi program dan program *flowchart*.

A. Tinjauan Kasus

Setiap terjadi perubahan data barang seperti ada penambahan nama barang baru, maka bagian penjualan Toko *hardware* komputer “XYZ” menerima dokumen masukan dari bagian gudang berupa data barang yang berisi data nama barang baru

atau perubahan data barang, kemudian bagian penjualan *menginput* data tersebut ke dalam data barang.

Bagi kasir yang belum menginput kode, nama, dan *passwordnya*, masing-masing harus mengisi data-datanya secara pribadi ke dalam data kasir yang akan digunakan untuk login sebelum melakukan transaksi penjualan, kode kasir telah ditetapkan oleh petugas, sedangkan nama dan *password* dibuat oleh kasir itu sendiri. Apabila kasir tersebut ingin merubah nama dan *passwordnya* dapat di *edit* lalu disimpan kembali ke dalam data kasir.

Pada saat pembeli atau pelanggan melakukan transaksi pembayaran, maka akan mendapatkan bukti pembayaran barang berupa faktur penjualan. Dari hasil transaksi tersebut akan menghasilkan laporan penjualan, dan laporan data barang yang telah terjual.

Setelah melakukan transaksi penjualan stok barang akan berkurang, maka bagian gudang melakukan penambahan stok barang, kemudian bagian penjualan *menginput* data stok barang tersebut kedalam tambah stok barang.

B. Spesifikasi Rancangan Program

Disini dijelaskan tentang uraian rancangan pembuatan program yang terinci dalam sub-sub bab yang dimulai dari spesifikasi bentuk masukan dan keluaran, normalisasi file, spesifikasi file, spesifikasi program dan program *flowchart*.

1. Rancangan Bentuk Masukan

Bentuk dokumen masukan merupakan dokumen atau data yang dipakai sebagai acuan untuk pembuatan dan mengisi data pada program aplikasi yang dibuat, dokumen tersebut terdiri dari :

- a. Nama Dokumen : Data Barang
 - Fungsi : Menunjukkan data barang yang tersedia
 - Sumber : Bagian Gudang
 - Tujuan : Bagian Penjualan
 - Media : Kertas
 - Jumlah : \pm Satu Lembar
 - Frekuensi : Setiap terjadi penambahan atau perubahan data barang
 - Bentuk : Lampiran A.1
- b. Nama Dokumen : Data Pengguna
 - Fungsi : Untuk menginput data pengguna baru
 - Sumber : Gudang
 - Tujuan : Bagian Penjualan
 - Media : Kertas
 - Jumlah : \pm Satu Lembar
 - Frekuensi : Setiap kali ada pengguna baru
 - Bentuk : Lampiran A.2

Rancangan Bentuk Keluaran

Bentuk dokumen keluaran merupakan dokumen atau data yang dibuat berdasarkan hasil dari program aplikasi yang dibuat dan digunakan pada bagian penjualan barang.

- a. Nama Dokumen : Nota Penjualan
 - Fungsi : Sebagai tanda bukti penjualan barang
 - Sumber : Bagian Penjualan
 - Tujuan : Pembeli
 - Media : Kertas
 - Jumlah : Satu Lembar
 - Frekuensi : Setiap terjadi transaksi
 - Bentuk : Lampiran B.1
- b. Nama Dokumen : Laporan Data Barang
 - Fungsi : Mengetahui data barang yang ada
 - Sumber : Bagian Penjualan
 - Tujuan : Pemilik
 - Media : Cetakan Komputer
 - Jumlah : \pm Satu Lembar
 - Frekuensi : Setiap hari
 - Bentuk : Lampiran B.2
- c. Nama Dokumen : Laporan Penjualan
 - Fungsi : Mengetahui jumlah transaksi penjualan
 - Sumber : Bagian Penjualan

Tujuan	: Pemilik
Media	: Kertas
Jumlah	: ± Satu Lembar
Frekuensi	: Setiap Hari, Setiap Minggu, Setiap Bulan
Bentuk	: Lampiran B.3

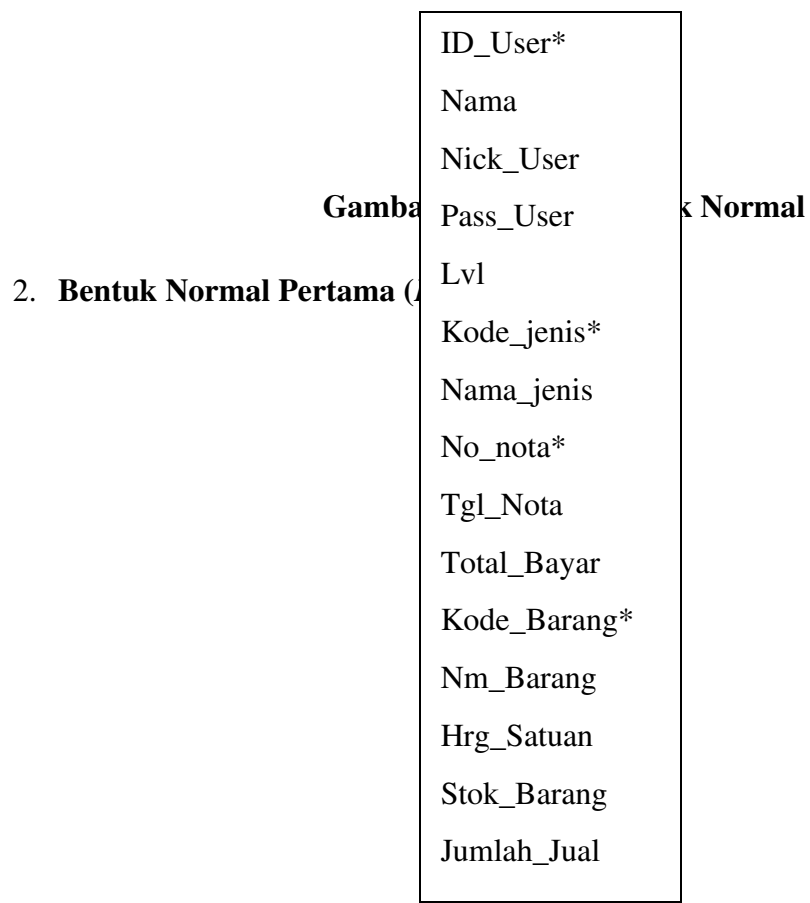
C. Normalisasi File

Normalisasi file merupakan suatu proses pengelompokan elemen data kedalam tabel yang menyatakan hubungan antara entitas sehingga terwujud suatu bentuk yang memudahkan adanya perubahan dengan dampak terkecil. Proses pada pengolahan data yang mengakibatkan efek samping terjadi akibat *inserting*, *updating* atau *deleting*. Oleh karena itu diperlukan normalisasi file.

Adapun bentuk normalisasi yang ada pada program penjualan *hardware* komputer dapat diuraikan sebagai berikut :

1. Bentuk Tidak Normal (Unnormalized)

ID_User
Nama
Nick_User
Pass_User
Lvl
Kode_jenis
Nama_jenis
No_nota
Tgl_Nota
Total_Bayar
Kode_Barang
Nm_Barang
Hrg_Satuan
Stok_Barang
Jumlah_Jual

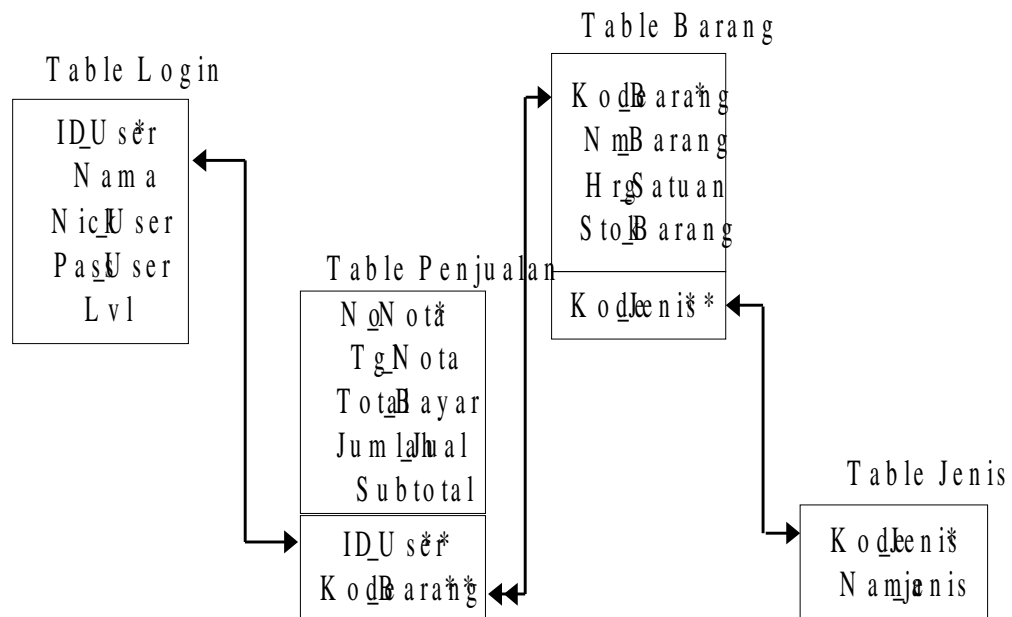


Keterangan :

* : *Candidate Key*

Gambar II.2 Bentuk Normal Kesatu

3. Bentuk Normal Kedua (*Second Normal Form*)



Keterangan :

* : *Primary Key*

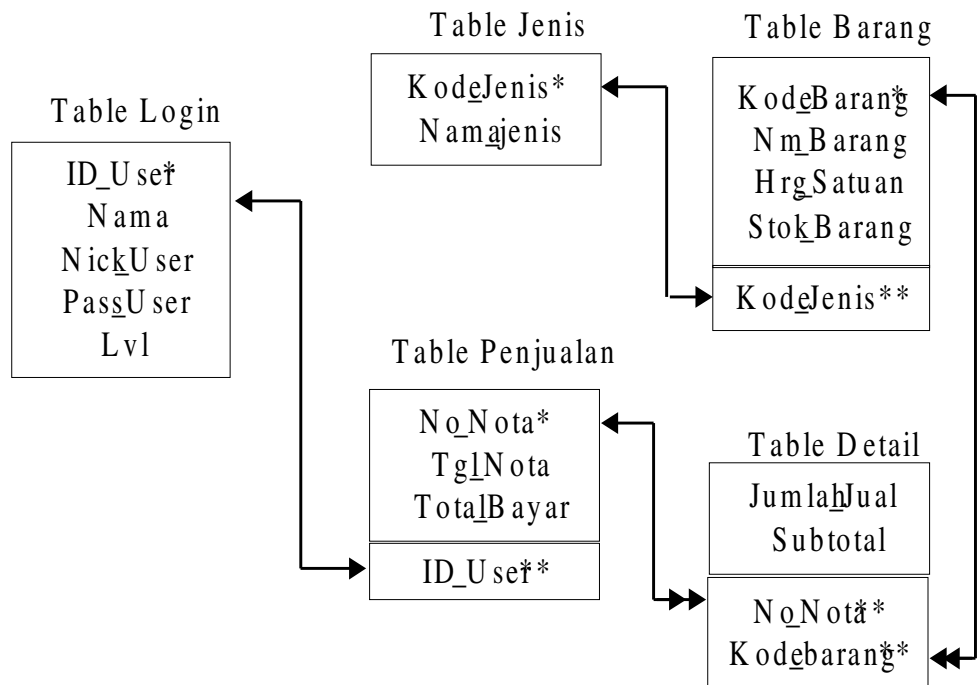
* * : *Foreign Key*

◀▶ : *One to many*

◀▶ : *One to one*

Gambar II.3 Bentuk Normal Kedua

4. Bentuk Normal Ketiga (*Third Normal Form*)



Keterangan :

* : *Primary Key*

** : *Foreign Key*

◄==► : *One to many*

◄==► : *One to one*

Gambar II.4 Bentuk Normal Ketiga

D. Spesifikasi File

Dalam pembuatan program ini penulis menggunakan sebuah file database yaitu Penjualan.mdb. Database Penjualan.mdb terdiri atas beberapa tabel yang penjelasannya sebagai berikut :

1. Tabel Barang

Nama Tabel	: Barang
Fungsi	: Untuk menyimpan data barang
Tipe Tabel	: Tabel Master
Organisasi Tabel	: <i>Index Sequential</i>
Akses Tabel	: <i>Random</i>
Media Tabel	: <i>Harddisk</i>
<i>Record Size</i>	: 60 Karakter
<i>Field Key</i>	: Kode_Barang
Paket Program	: <i>Microsoft Access</i>

Tabel II.1
Struktur File Data Barang

No	Elemen Data	Nama Field	Type	Panjang	Keterangan
1	Kode Barang	Kode_Barang	Text	5	<i>Primary Key</i>
2	Nama Barang	Nm_Barang	Text	50	
3	Harga Satuan	Hrg_Satuan	Long Integer		
4	Stok Barang	Stok_Barang	Long Integer		
5	Kode Jenis	Kode_Jenis	Text	5	<i>Foreign Key</i>

2. Tabel Login

Nama Tabel	: Login
------------	---------

Fungsi : Untuk menyimpan data pengguna
 Tipe Tabel : File Master
 Organisasi Tabel : *Index Sequential*
 Akses Tabel : *Random*
 Media Tabel : *Harddisk*
Record Size : 70 Karakter
Field Key : ID_User
 Paket Program : *Microsoft Access*

Tabel II.2
Struktur File Data Login

No	Elemen Data	Nama Field	Type	Panjang	Karakter
1	ID_User	ID_User	Text	5	<i>Primary Key</i>
2	Nama	Nama	Text	30	
3	Nick User	Nick_User	Text	15	
4	Password User	Pass_User	Text	15	
5	Level	Lvl	Text	5	

3. Tabel Penjualan

Nama Tabel : Penjualan
 Fungsi : Untuk menyimpan data transaksi
 Tipe Tabel : Tabel Transaksi
 Organisasi Tabel : *Index Sequential*
 Akses Tabel : *Random*
 Media Tabel : *Hard disk*

Record Size : 13 Karakter

Field Key : No_Nota

Paket Program : *Microsoft Access*

Tabel II.3
Struktur File Data Penjualan

No	Elemen Data	Nama Field	Type	Panjang	Karakter
1	Nomor Nota	No_Nota	Text	8	<i>Primary</i>
2	Tanggal Nota	Tgl_Nota	Date/Time		
3	Total_Bayar	Total_Bayar	Long Integer		
4	ID_User	ID_User	Text	5	<i>Foreign</i>

4. Tabel Detail

Nama Tabel : Detail

Fungsi : Untuk menyimpan data barang yang telah dijual

Tipe Tabel : Tabel Transaksi

Organisasi Tabel : *Sequential*

Akses Tabel : *Random*

Media Tabel : *Hard disk*

Record Size : 13 Karakter

Field Key : -

Paket Program : *Microsoft Access*

Tabel II.4
Struktur File Detail

No	Elemen Data	Nama Field	Type	Panjang	Keterangan
1	Jumlah Jual	Jumlah_Jual	Long Integer		
2	Subtotal	Subtotal	Long Integer		
3	Nomer Nota	No_Nota	Text	8	<i>Foreign Key</i>
4	Kode Barang	Kode_Barang	Text	5	<i>Foreign Key</i>

5. Tabel Jenis

Nama Tabel	: Jenis
Fungsi	: Untuk menyimpan data barang
Tipe Tabel	: Tabel Master
Organisasi Tabel	: <i>Index Sequential</i>
Akses Tabel	: <i>Random</i>
Media Tabel	: <i>Harddisk</i>
<i>Record Size</i>	: 35 Karakter
<i>Field Key</i>	: Kode_Jenis
Paket Program	: <i>Microsoft Access</i>

Tabel II.5

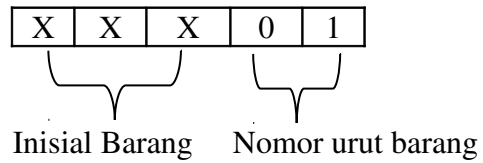
Struktur File Jenis Barang

No.	Elemen Data	Nama Field	Type	Panjang	Keterangan
1	Kode Jenis	Kode_Jenis	Text	5	<i>Primary Key</i>
2	Nama Jenis	Nama_jenis	Text	30	

Struktur Kode

Struktur kode yang digunakan dalam pembuatan program penjualan *hardware* komputer secara tunai pada toko “XYZ” sebagai berikut:

1. Kode Barang



Contoh :

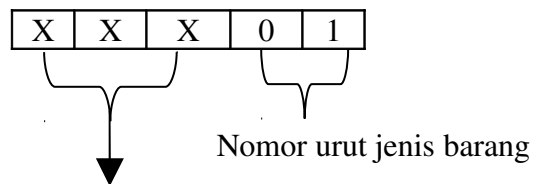
O	P	T	0	1
---	---	---	---	---

Keterangan:

OPT : Barang Elektronik adalah Optical

01 : No. urut barang = 1

2. Jenis Barang



Singkatan dari “Jenis”

Contoh :

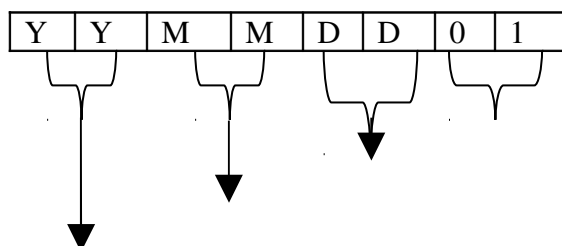
J	N	S	0	1
---	---	---	---	---

Keterangan :

JNS : Singkatan dari “Jenis”

01 : No. urut jenis barang = 1

3. Nomor Transaksi



Nomor Penjualan
 Hari Penjualan
 Bulan Penjualan
 Tahun Penjualan

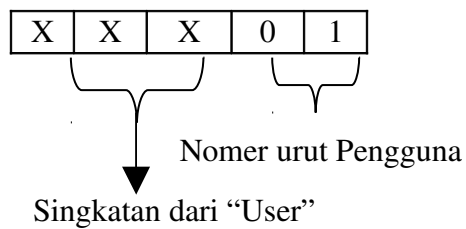
Contoh :

1	1	1	0	3	0	0	1
---	---	---	---	---	---	---	---

Keterangan :

11 : Tahun 2011
 10 : Bulan Oktober
 30 : Tanggal 20
 01 : No. urut Penjualan = 1

4. Kode Pengguna



Contoh :

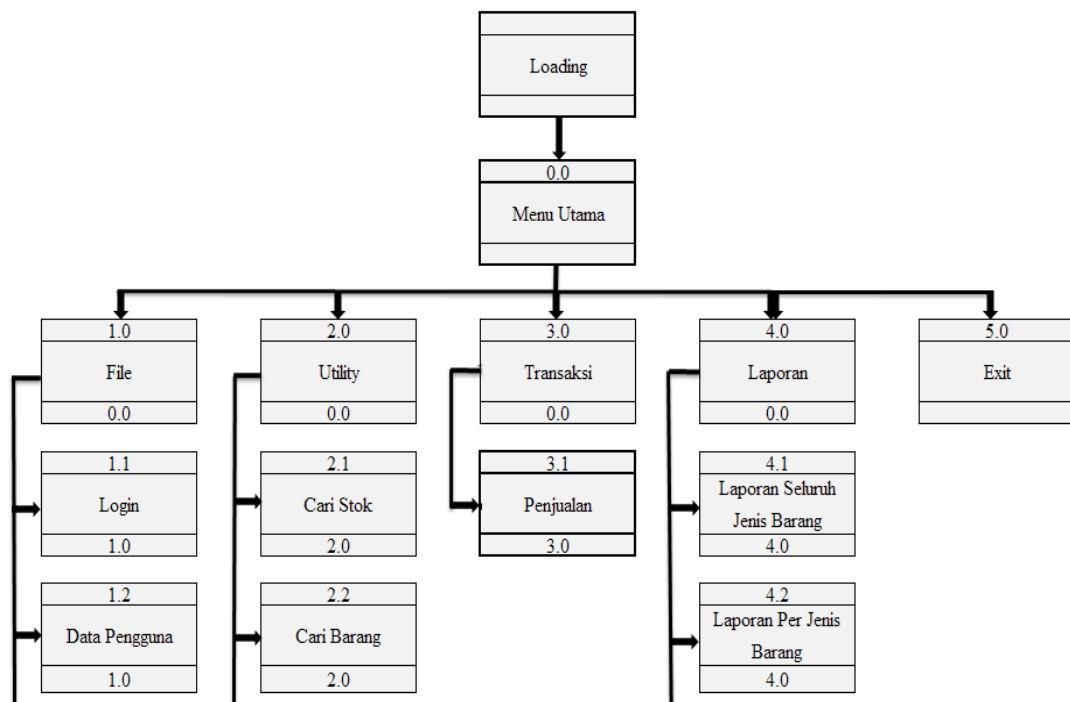
U	S	R	0	1
---	---	---	---	---

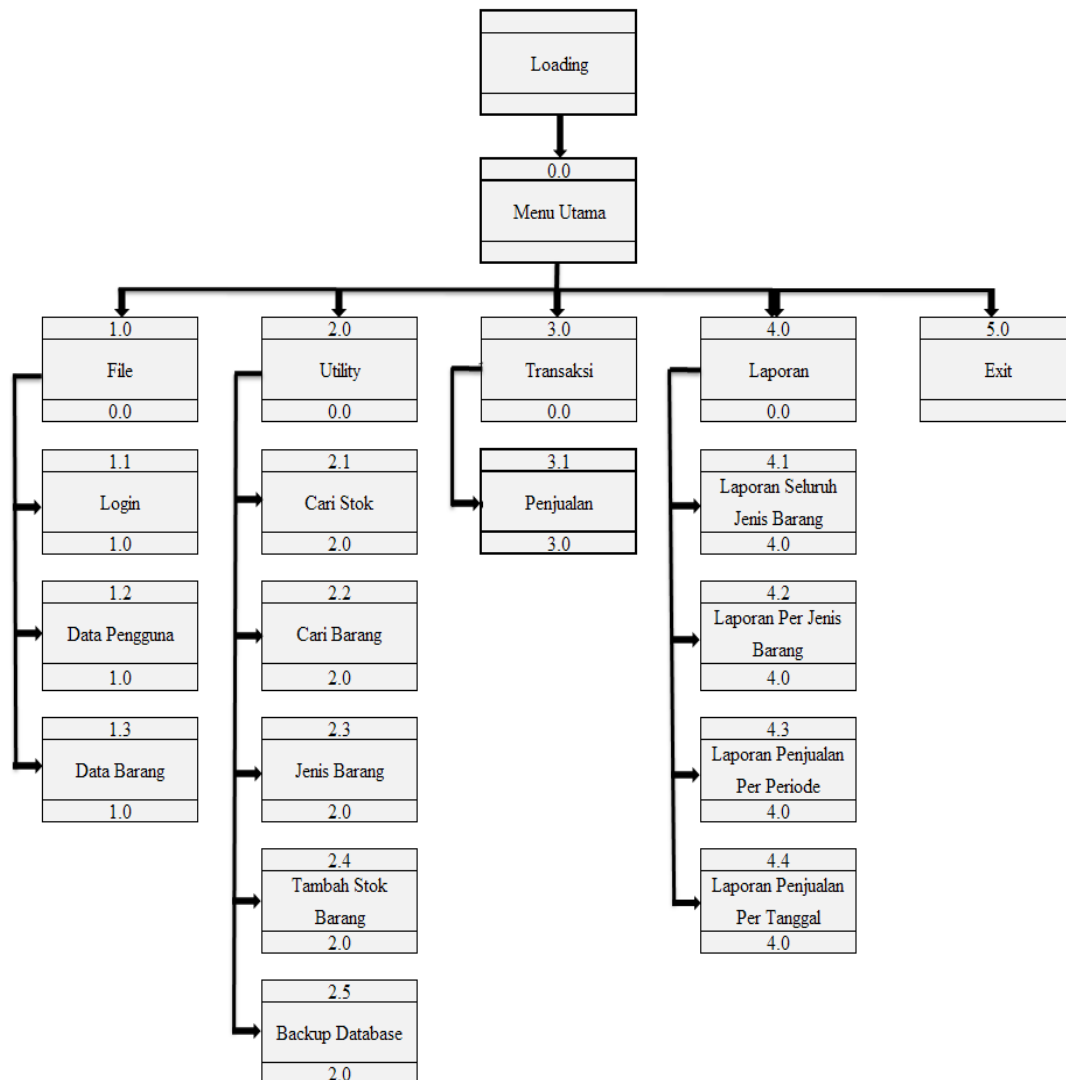
Keterangan :

USR : Singkatan dari "User"
 01 : Nomer urut Pengguna = 1

Spesifikasi Program

Spesifikasi program merupakan pembahasan mengenai cara penggunaan program aplikasi, yang penulis sediakan guna pemrosesan data. Spesifikasi program pada perancangan program dalam pembuatan tugas akhir ini dapat dilihat pada diagram HIPO sebagai berikut :





Gambar II.5 Diagram HIPO

Dengan adanya spesifikasi program ini diharapkan pemakai dapat mengetahui cara operasi program aplikasi ini.

1. Spesifikasi Program *Loading*

Nama Program : *Loading*

Akronim : *Loading.frm*

Fungsi : Membuka *loading* untuk masuk ke menu utama.

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.1

Proses : Form *loading* akan tampil setelah *user* mengklik program penjualan.

2. Spesifikasi Program Menu Utama

Nama Program : Menu

Akronim : Menu.frm

Fungsi : Menampilkan menu pilihan program

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.2

Proses :

Pada menu utama terdapat lima buah pilihan, yaitu :

a. File

Apabila *user* memilih menu file, maka akan berisi sub menu Login, data pengguna dan data barang.

b. Utility

Apabila *user* memilih menu utility, maka akan berisi sub menu cari stok, cari barang, jenis barang, tambah stok barang dan backup database.

c. Transaksi

Apabila *user* memilih menu transaksi, maka akan berisi sub menu Penjualan.

d. Laporan

Apabila *user* memilih menu Laporan, maka akan berisi sub menu Laporan Seluruh Jenis Barang, Laporan Per Jenis Barang, Laporan Penjualan Per Periode dan Laporan Penjualan Per Tanggal Transaksi.

e. Keluar

Apabila user memilih menu keluar maka akan keluar dari program.

3. Spesifikasi Program *Login Password*

Nama Program : *Login*

Akronim : *Login.frm*

Fungsi : Pengamanan bagi pemakai

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.3

Proses :

- a. Pilih menu file maka akan tampil sub menu login.
- b. Masukkan nama user, jika nama *user* benar maka *password* aktif lalu masukkan *password*, jika *password* aktif maka tombol login aktif dan klik tombol login, maka menu utama akan aktif dan siap digunakan.
- c. Klik *Close* untuk mengakhiri program.

4. Spesifikasi Program Data Pengguna

Nama Program : Pengguna

Akronim : *Pengguna.frm*

Fungsi : Untuk memanipulasi data pengguna

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.4

Proses :

- a. Pilih Menu file maka akan tampil sub menu data pengguna.
- b. Klik Input untuk menambah data pengguna baru.
- c. Klik Edit jika ingin merubah data barang pengguna.
- d. Klik Delete jika ingin menghapus data yang tidak diinginkan.
- e. Klik Close untuk keluar dari program data pengguna dan kembali ke menu utama.

5. Spesifikasi Program Data Barang

Nama Program : Barang

Akronim : Barang.frm

Fungsi : Untuk memanipulasi data barang

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.5

Proses :

- a. Pilih menu file yang berisi sub menu data barang.
- b. Klik Input untuk menambah data barang yang baru.
- c. klik save, digunakan untuk menyimpan data barang yang telah diinput apabila sudah yakin untuk disimpan.
- d. Klik Edit untuk mengedit data barang yang ada, jika selesai klik save dan jika tidak jadi mengedit tekan cancel.

- e. Klik Delete untuk menghapus data barang yang tidak diinginkan.
- f. Klik Close untuk keluar dari program data barang dan kembali ke menu utama.

6. Spesifikasi Program Cari Stok

Nama Program : Caristok

Akronim : Caristok.frm

Fungsi : Untuk mengetahui stok yang ada

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.6

Proses :

- a. Pilih menu utility yang berisi sub menu cari stok.
- b. Klik View untuk mengkosongkan table.
- c. Klik Refresh untuk melihat stok yang tersedia.
- d. Klik Close untuk keluar dari program cari stok dan kembali ke menu utama.

7. Spesifikasi Program Cari Barang

Nama Program : Caribarang

Akronim : Caribarang.frm

Fungsi : Untuk mencari barang yang tersedia

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.7

Proses :

- a. Pilih menu utility yang berisi sub menu cari barang.

- b. Klik kode barang lalu ketik barang yang dicari atau klik nama barang lalu ketik nama barang yang dicari .
- c. Klik Close untuk keluar dari program cari barang dan kembali ke menu utama.

8. Spesifikasi Program Jenis Barang

Nama Program : Jenis

Akronim : Jenis.frm

Fungsi : Untuk menambah stok barang.

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.8

Proses :

- a. Pilih Menu Utility maka akan tampil sub menu jenis barang.
- b. Klik Input untuk menambah jenis barang baru.
- c. Klik Edit jika ingin merubah jenis barang.
- d. Klik Delete jika ingin menghapus jenis barang yang tidak diinginkan.
- e. Klik Close untuk keluar dari program jenis barang dan kembali ke menu utama.

9. Spesifikasi Program Tambah Stok Barang

Nama Program : Tambahstok

Akronim : Tambahstok.frm

Fungsi : Untuk menambah stok barang.

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.9

Proses :

- a. Pilih menu utility yang berisi sub menu tambah stok barang.
- b. Klik Input untuk menambah barang yang baru datang dan klik save jika sudah yakin akan disimpan.
- c. Klik Close untuk keluar dari program tambah stok dan kembali ke menu utama.

10. Spesifikasi Program Backup Database

Nama Program : Backup

Akronim : Backup.frm

Fungsi : Untuk mengamankan database

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.10

Proses :

- a. Pilih menu utility yang berisi sub menu backup database.
- b. Pilih file sumber yang berisi database yang akan di backup, dan pilih direktori tujuan untuk membackup database lalu klik copy file.

11. Spesifikasi Program Penjualan

Nama Program : Jual

Akronim : Jual.frm

Fungsi : Untuk membuat data transaksi penjualan

Bahasa Program : *Visual Basic 6.0*

Bentuk Lampiran : Lampiran C.11

Proses :

- a. Pilih menu transaksi yang berisi sub menu penjualan.
- b. Klik kode barang jika kode barang ditemukan maka akan tampil nama barang dan harga.
- c. Input Jumlah, lalu tekan enter untuk menampilkan total dan jumlah bayar yang dihasilkan dan klik add untuk menambah pada table penjualan.
- d. Jika kasir memasukkan uang bayar lalu tekan enter maka uang kembali akan muncul secara otomatis Klik Batal untuk membatalkan transaksi penjualan
- e. Klik Simpan untuk menyimpan data transaksi dan melakukan proses pencetakan nota.
- f. Klik Tutup untuk menutup form transaksi dan kembali ke menu utama.

12. Spesifikasi Program Laporan Seluruh Jenis Barang

Nama Program : Barang

Akronim : Barang.rpt



Fungsi : Untuk melihat dan mencetak laporan seluruh jenis barang

Bahasa Program : *Crystal Reports 8.5*



Bentuk Lampiran : Lampiran C.12

Proses :

- a. Pilih menu laporan yang berisi sub menu laporan seluruh jenis barang.
- b. Tentukan kriteria pencetakan laporan yaitu semua data atau perhalaman.



- c. Klik  untuk mencetak laporan seluruh jenis barang ke printer.
- d. Klik  untuk kembali ke menu utama.

13. Spesifikasi Program Laporan Per Jenis Barang

- Nama Program : Lapjenis
- Akronim : Lapjenis.frm
- Fungsi : Untuk melihat dan mencetak laporan per jenis barang
- Bahasa Program : *Visual Basic 6.0*
- Bentuk Lampiran : Lampiran C.13
- Proses :
- a. Pilih menu laporan yang berisi sub menu laporan per jenis barang.
 - b. Klik jenis barang, lalu klik print.
 - c. Tentukan kriteria pencetakan laporan yaitu semua data atau perhalaman.
 - d. Klik  untuk mencetak laporan data barang ke printer.
 - e. Klik  untuk kembali ke laporan jenis barang.

14. Spesifikasi Program Laporan Penjualan Per Periode

- Nama Program : Lappenjualanperperiode
- Akronim : Lappenjualanperperiode.frm
- Fungsi : Untuk melihat dan mencetak penjualan per periode
- Bahasa Program : *Visual Basic 6.0*
- Bentuk Lampiran : Lampiran C.14
- Proses :

- a. Pilih menu laporan yang berisi sub menu laporan per periode.
- b. Pilih tahun, bulan dan hari, lalu klik print.
- c. Tentukan kriteria pencetakan laporan yaitu semua data atau perhalaman.
- d. Klik  untuk mencetak laporan data barang ke printer.
- e. Klik  untuk kembali ke laporan penjualan per periode.

15. Spesifikasi Program Laporan Penjualan Per Tanggal

Nama Program : Lappenjualan



Akronim : Lappenjualan.frm

Fungsi : Untuk melihat dan mencetak penjualan per tanggal

Bahasa Program : *Visual Basic 6.0*

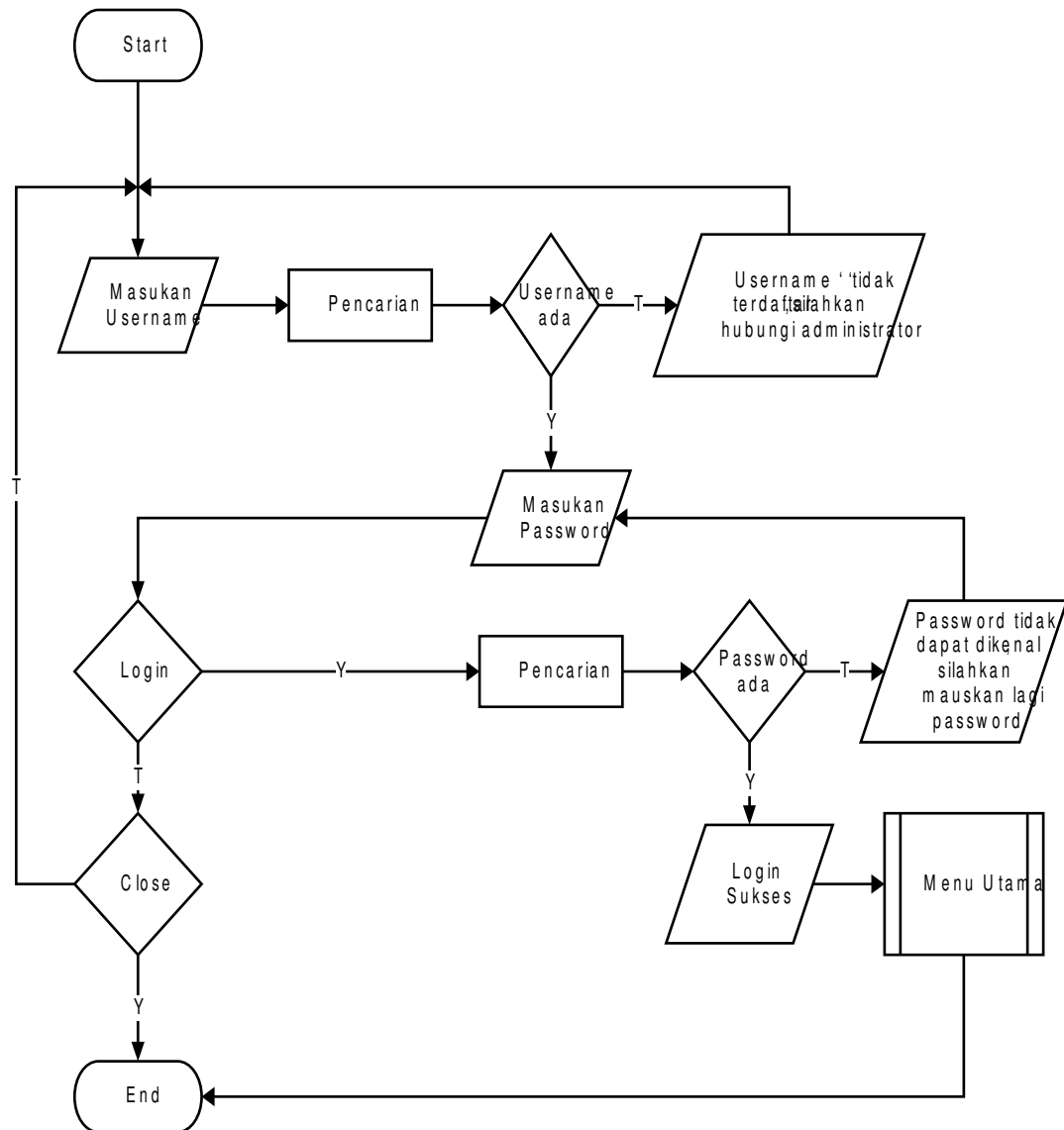
Bentuk Lampiran : Lampiran C.15

Proses :

- a. Pilih menu laporan yang berisi sub menu laporan per tanggal transaksi.
- b. Pilih tahun, bulan dan hari, lalu klik print.
- c. Tentukan kriteria pencetakan laporan yaitu semua data atau perhalaman.
- d. Klik  untuk mencetak laporan data barang ke printer.
- e. Klik  untuk kembali ke laporan penjualan per tanggal transaksi.

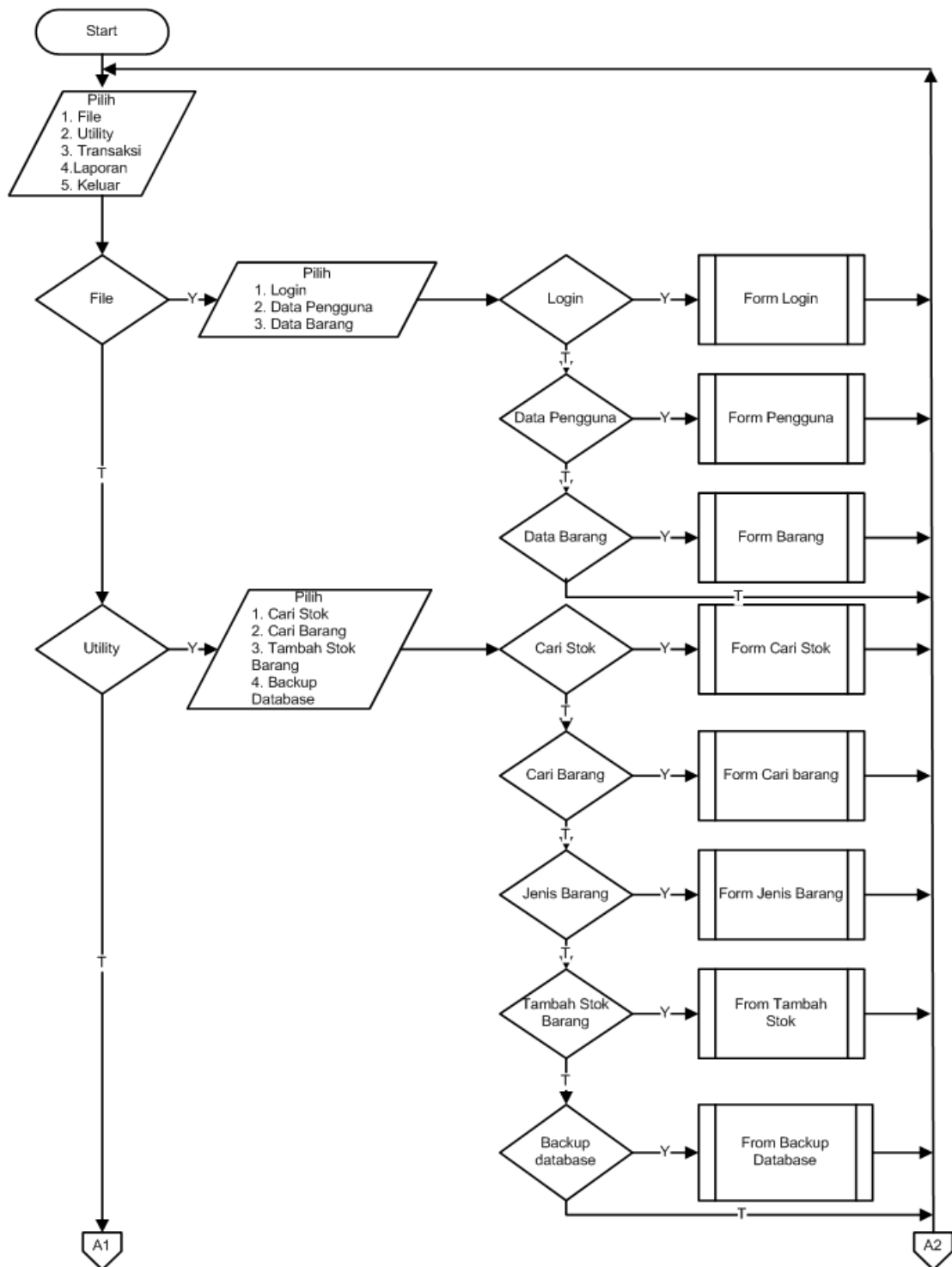
Flowchart Program

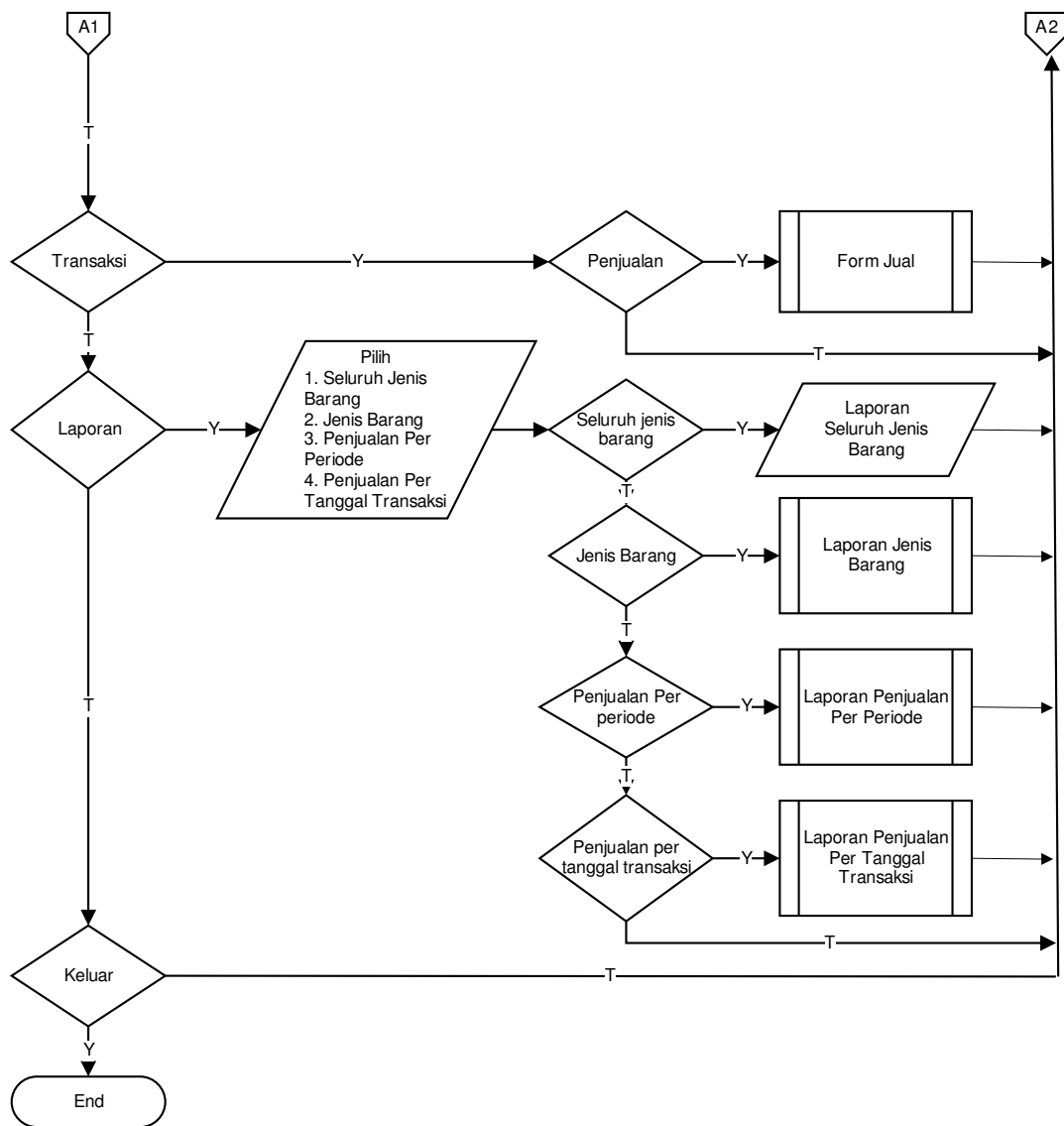
1. Flowchart Login Password



Gambar II.6 *Flowchart Login Password*

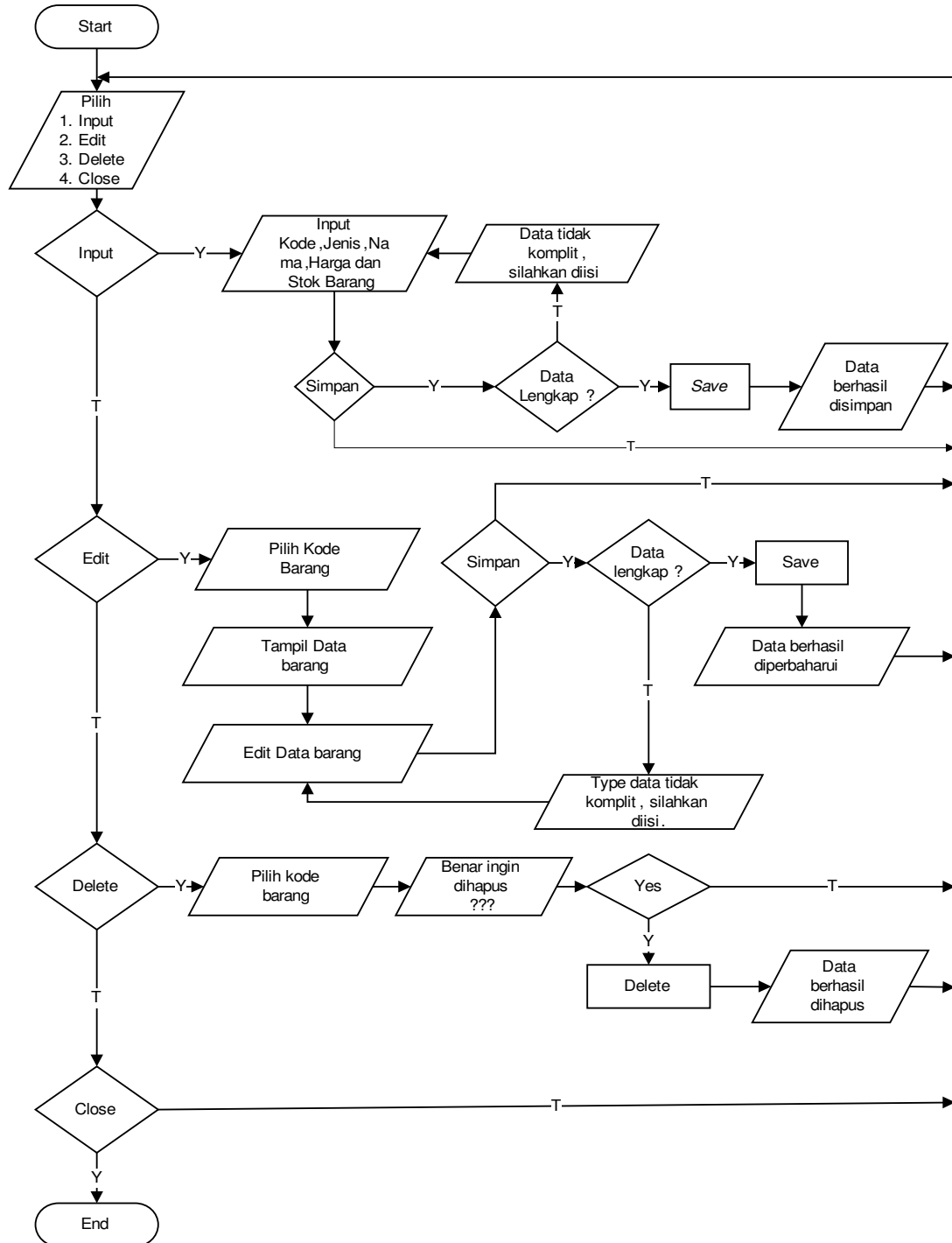
2. *Flowchart* Menu Utama





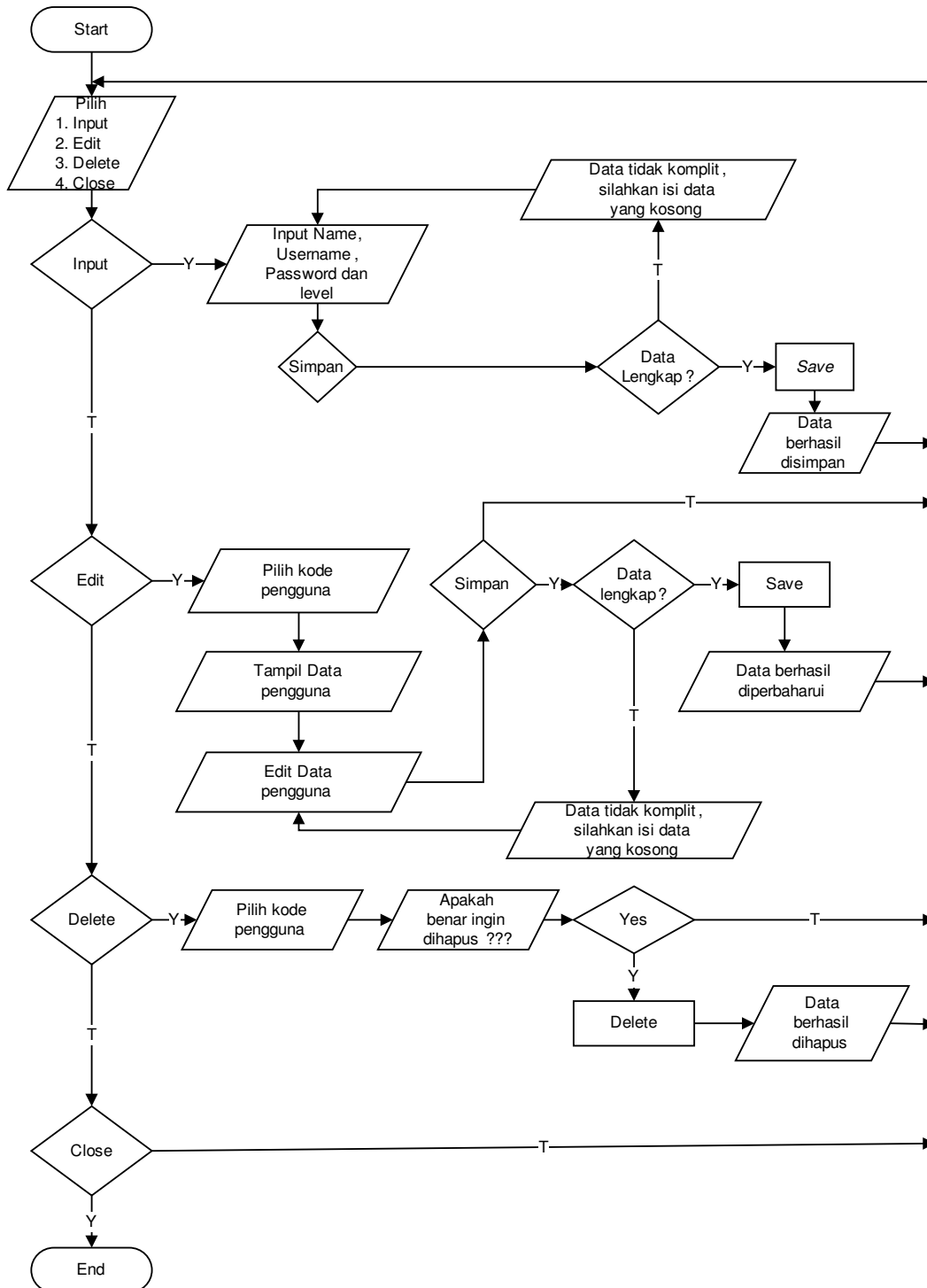
Gambar II.7 Flowchart Menu Utama

3. Flowchart Data Barang



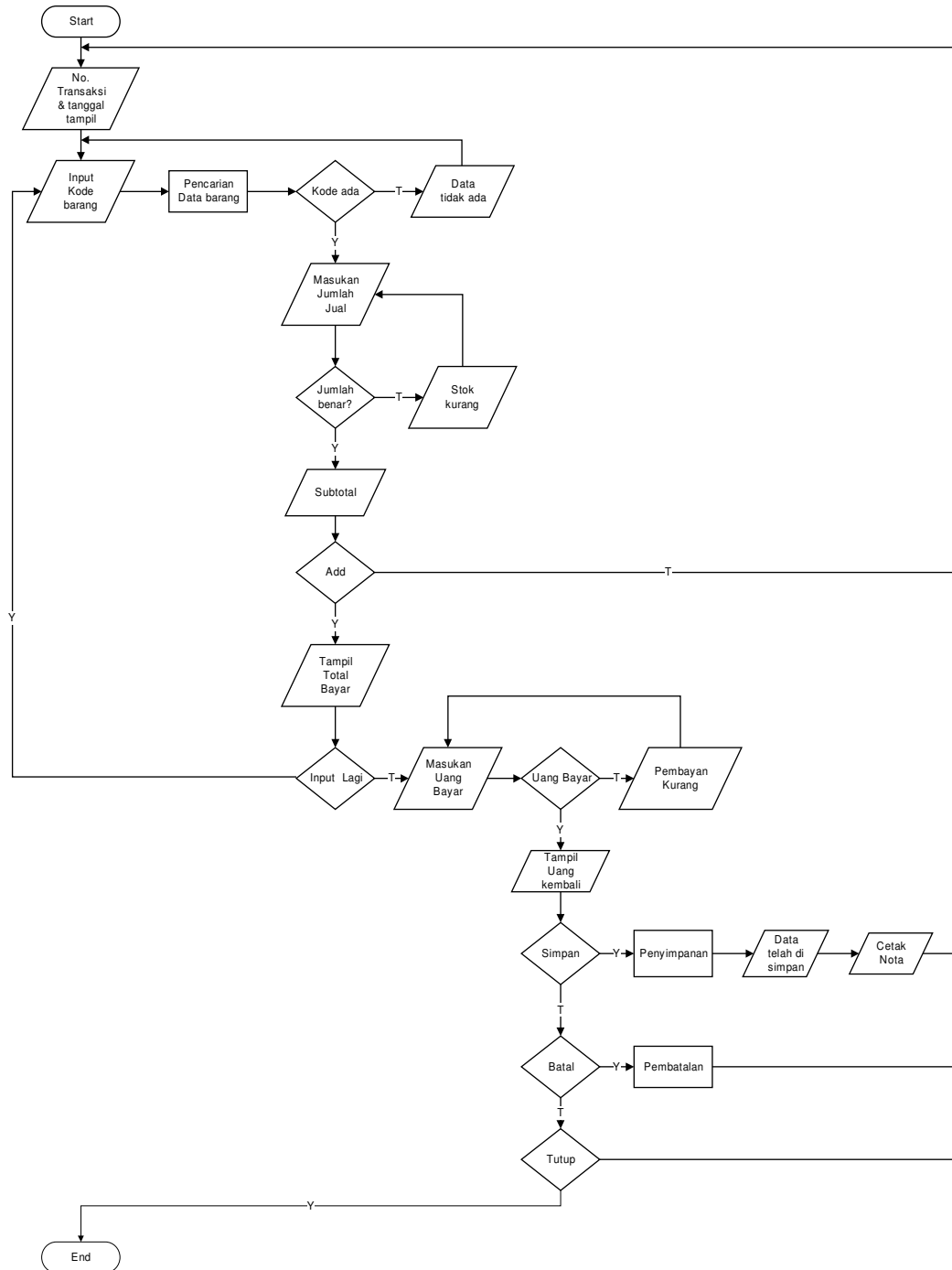
Gambar II.8 Flowchart Data Barang

4. Flowchart Data Pengguna



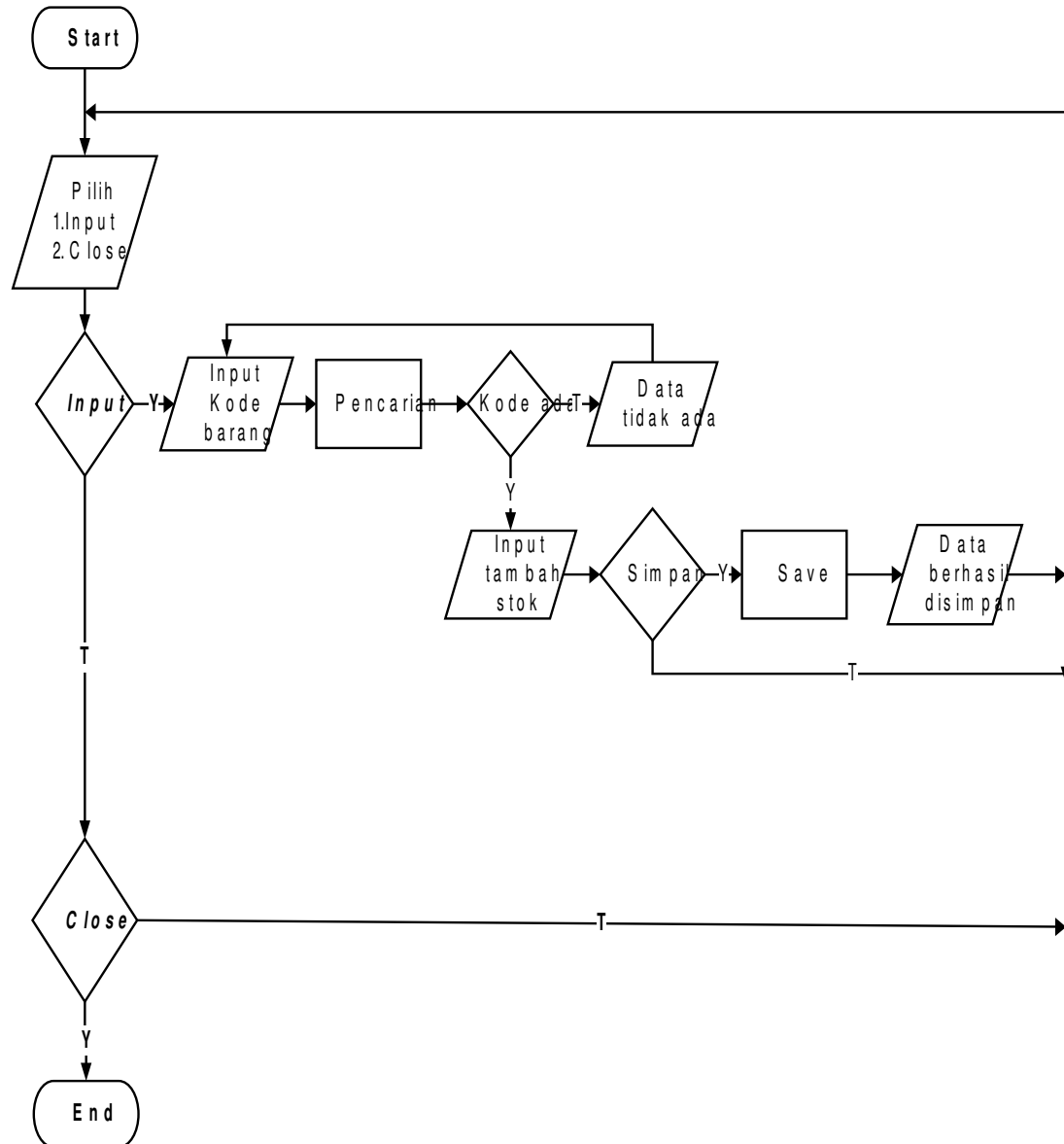
Gambar II.9 Flowchart Data Pengguna

5. Flowchart Transaksi Penjualan



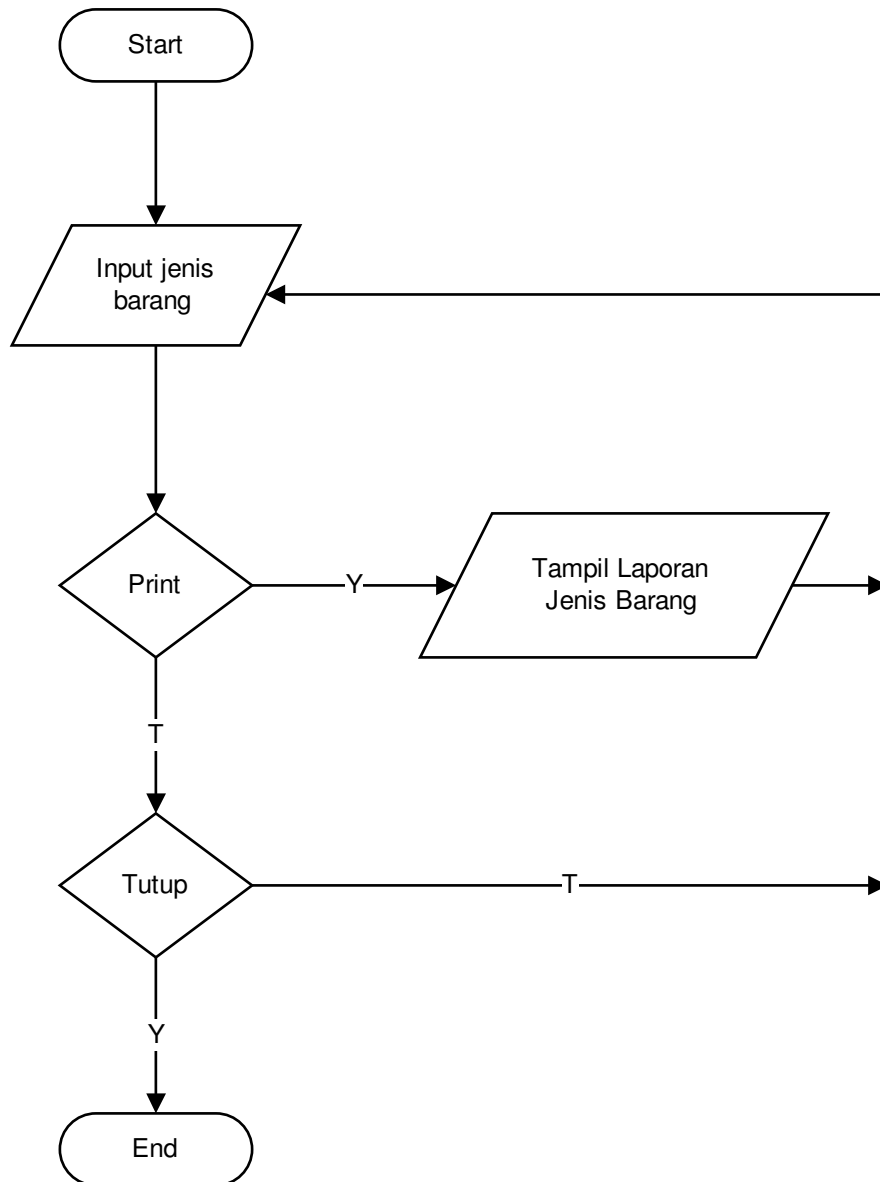
Gambar II.10 Flowchart Transaksi Penjualan

6. Flowchart Tambah Stok



Gambar II.11 Flowchart Tambah Stok

7. *Flowchart* Laporan Per Jenis Barang



Gambar II.12 *Flowchart* Laporan

H.

Sarana Pendukung Program

Untuk memaksimalkan daya komputer sesuai dengan kebutuhan, maka dibutuhkan pendukung yang baik tidak harus yang mahal atau jumlah yang banyak akan tetapi pendukung yang mampu membantu aplikasi program atau komputerisasi sistem yang efektif dan efisien. Dalam memilih peralatan pendukung perlu dipertimbangkan hal-hal sebagai berikut :

- a. Ruang lingkup kerja.
- b. Target yang harus dicapai.

1. Perangkat Keras (*Hardware*)

Kata *hardware* sebenarnya berarti barang-barang yang tersedia dari logam termasuk peralatan mesin, kunci-kunci dan lain-lain. Dalam bidang komputer dapat dijelaskan secara singkat bahwa semua peralatan yang dapat dilihat dengan mata atau secara fisik. Adapun spesifikasi perangkat keras yang digunakan dalam tugas akhir yang penulis buat adalah sebagai berikut :

- a. *Processor* : *Pentium IV 2.4 GHz*
- b. *Memori* : 1 GB
- c. *Monitor* : *SVGA 1024 x 768 pixel*
- d. *Hard Disk* : 40 GB
- e. *Keyboard* : 108 Keys

- f. *Mouse* : Standard
- g. *Printer* : Dot Matrik
- h. *CD Rw* : 12x

2. Perangkat Lunak (*Software*)

Kata *software* dimaksud adalah serangkaian instruksi-instruksi atau perintah sehingga membuat komputer melakukan tugas tertentu. Oleh karena itu yang lebih lazim disebut *software* adalah sama dengan program aplikasi. Aplikasi atau program aplikasi atau terkadang disebut program saja, merupakan sederatan kode yang digunakan untuk mengatur komputer untuk melakukan sesuatu sesuai dengan keinginan yang membuatnya.

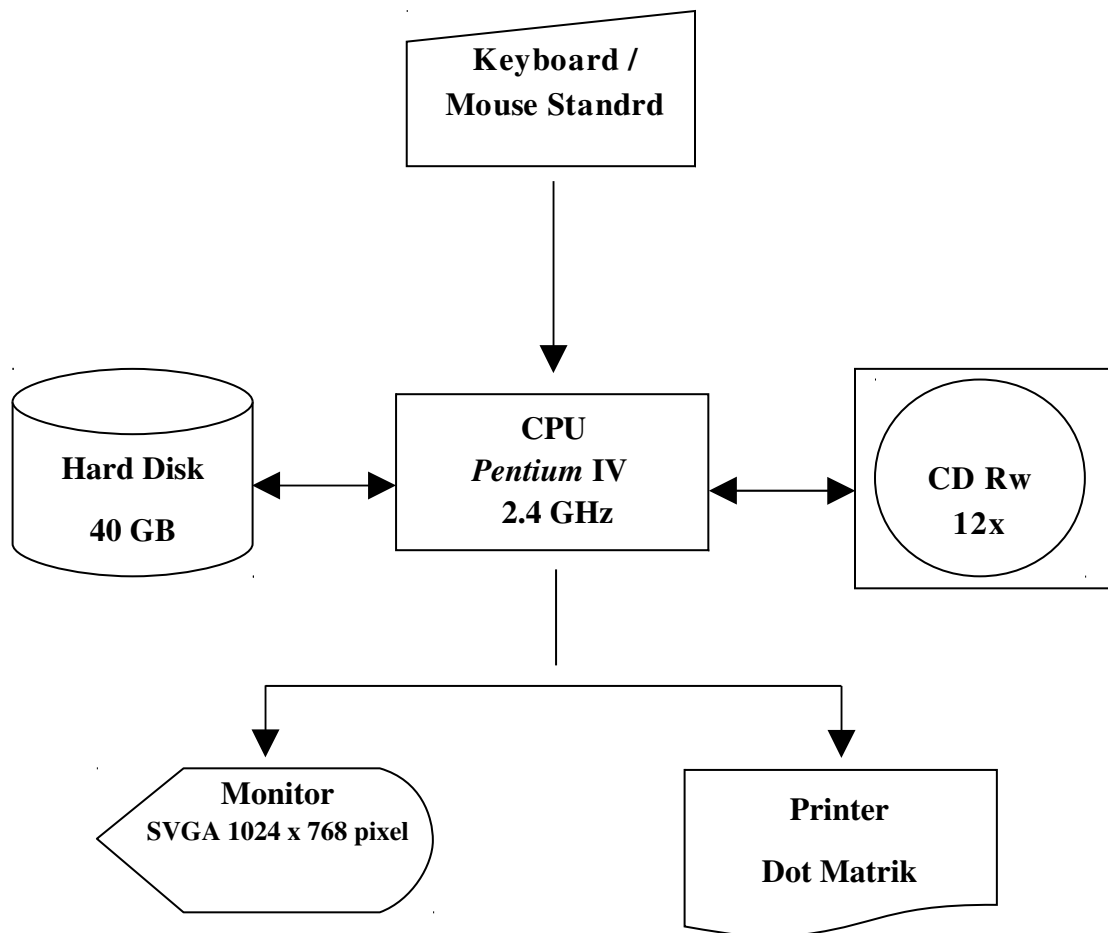
Aplikasi dapat dibedakan menjadi aplikasi *windows* dan aplikasi terminal. Aplikasi *windows* adalah aplikasi berjalan pada *windows* sedangkan aplikasi terminal merupakan aplikasi non *windows* atau aplikasi yang berjalan pada sistem DOS.

Perangkat lunak yang dibutuhkan untuk aplikasi penyimpanan data ini adalah :

- a. Sistem Operasi : *WindowsXP Professional*
- b. Bahasa Pemograman : *Microsoft Visual Basic 6.0*
- c. Laporan : *Crystal Reports 8.5*
- d. Skin Program : *Active Skin*
- e. Database : *Microsoft Acces 2007*

3. Konfigurasi Sistem Komputer

Perangkat keras komputer yang digunakan dalam perancangan program sangat menentukan jalannya aplikasi yang digunakan. Bila perangkat keras yang digunakan kurang memadai, maka hasil yang akan diperoleh menjadi kurang baik, demikian sebaliknya. Oleh karena itu penulis akan menampilkan konfigurasi sistem komputer yang dapat digunakan. Adapun konfigurasi komputer dari perangkat keras yang digunakan dalam perancangan program pada tugas akhir ini, dapat dilihat pada gambar berikut :



Gambar II.13 Konfigurasi Sistem Komputer