BAB II

LANDASAN TEORI

2.1 Gambaran Umum Perusahaan

2.1.1 Sejarah Perusahaan

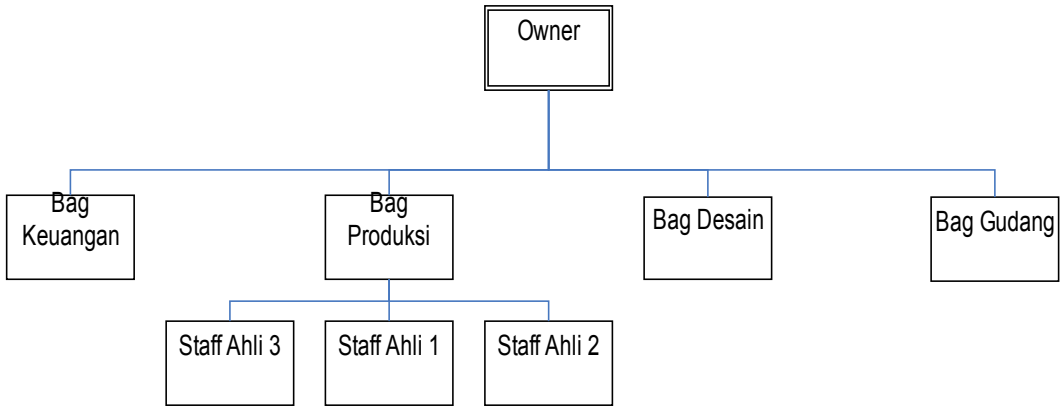
INDOGRAFIKA merupakan sebuah industri rumahan (*home industry*) yang bergerak di bidang konveksi pakaian. Berdiri pada bulan Desember tahun 2006 di Bandung oleh beberapa anak muda yang memiliki jiwa kreatifitas tinggi. Bertempat di Jl. Awiligar Ria II no 1 inilah INDOGRAFIKA mengerjakan seluruh pesanannya.

Usaha utama yang dilakukan INDOGRAFIKA saat ini adalah menerima pesanan pembuatan pakaian jadi baik dari distro besar maupun distro kecil, instansi atau lembaga, bahkan individual. Walaupun INDOGRAFIKA bertempat di Bandung, namun distribusi penjualan INDOGRAFIKA saat ini sudah mencapai luar kota, seperti Jabodetabek dan sekitarnya, bahkan sudah pula mencapai luar pulau Jawa.

INDOGRAFIKA memiliki visi bahwa pakaian merupakan cerminan kepribadian seseorang dan dengan pakaian yang sesuai, maka seseorang akan lebih dihargai. Sedangkan misinya adalah meningkatkan mutu dan kualitas pakaian produksi dalam negeri agar tidak kalah bersaing dengan produksi dari mancanegara.

2.1.2 Struktur Organisasi Perusahaan

Struktur organisasi perusahaan merupakan bangunan fungsi bagian-bagian manajemen yang tersusun dari suatu kesatuan hubungan yang menunjukan tingkatan fungsi, tugas, wewenang dan tanggung jawab dalam manajemen perusahaan.



Gambar 2.1. Struktur Organisasi

2.2 Pengertian Sistem

Untuk mempermudah dalam mempelajari suatu sistem ada baiknya kita mengetahui terlebih dahulu pengertian dari sistem itu sendiri. Sistem dapat didefinisikan dengan dua pendekatan yaitu yang menekankan pada prosedur dan

yang menekankan pada komponen atau elemennya. Pendekatan yang menekankan pada prosedur, mendefinisikan sistem sebagai berikut :

"Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu". [4]

Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponennya, mendefinisikan sistem sebagai berikut :

"Sistem adalah kumpulan dari elemen-elemen yang saling berinteraksi untuk mencapai suatu tujuan tertentu". [4]

2.2.1 Karakteristik Sistem

Menurut Jogiyanto HM dalam bukunya "*Analisis dan Desain*" sistem mempunyai karakteristik atau sifat-sifat tertentu yaitu mempunyai komponen-komponen (*components*), batas sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolahan (*process*), dan sasaran (*objectives*) atau tujuan (*goal*).

1. Komponen-komponen (*components*), suatu sistem terdiri dari sejumlah komponen yang saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa sub sistem atau bagian-bagian dari sistem.
2. Batas sistem (*boundary*), merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya.

3. Lingkungan luar sistem (*environments)*, dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem.
4. Penghubung (*interface)*, merupakan media penghubung antara suatu sub sistem dengan sub sistem lainnya.
5. Masukan (*input*), adalah energi yang dimasukan ke dalam sistem.
6. Keluaran (*output)*, adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukkan untuk sub sistem yag lain atau kepada supra sistem.
7. Pengolahan (*process*), suatu sistem dapat mempunyai suatu bagian pengolahan yang akan merubah masukan menjadi keluaran.
8. Sasaran (*objectives*) atau tujuan (*goal*), suatu sistem pasti mempunyai sasaran atau tujuan, jika sistem tidak mempunyai sasaran dan tujuan maka opersai tidak akan ada gunanya. Sasaran sistem sangat menentukan sekali masukan sistem dan keluaran yang dihasilkan sistem.

2.2.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari berbagai sudut pandang, menurut Jogiyanto HM klasifikasi tersebut diantaranya adalah sebagai berikut :

1. Sistem sebagai sistem abstrak (*abstrac system*) dan sistem fisik (*Physical system*). Sistem abstrak adalah sistem yang berupa pemikiran-pemikiran atau ide-ide yang tidak tampak secara fisik. Sedangkan sistem fisik adalah sistem yang ada secara fisik.

2. Sistem sebagai sistem ilmiah (*natural system*) dan sistem buatan manusia (*human made system*). Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Sedangkan sistem buatan manusia adalah sistem yang dirancang dan diciptakan oleh manusia.
3. Sistem sebagai sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*). Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi diantara bagian-bagiannya sehingga bentuk keluaran dapat diramalkan. Sedangkan sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
4. Sistem sebagai sistem tertutup (*closed system*) dan sistem terbuka (*open system*). Sistem tertutup merupakan sistem yang tidak terpengaruh dan tidak berhubungan dengan lingkungan luarnya. Sistem ini secara otomatis berjalan tanpa ada campur tangan dari pihak luar.
5. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh oleh lingkungan luarnya. Sistem ni menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem-subsistem lainnya.

2.3 **Pengertian Informasi**

Informasi didefinisikan oleh Jogiyanto Hartono adalah sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*event*) yang nyata (*fact*) yang digunakan untuk pengambilan keputusan.

Informasi yang baik haruslah mencakup hal-hal berikut :

a. *Relevancy*

Berarti informasi harus memberikan manfaat bagi pemakainya. Relevansi informasi untuk tiap-tiap orang yang satu dengan yang lainnya berbeda, misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan kepada ahli teknik perusahaan.

b. Accurasi

Informasi harus bebas dari kesalahan-kesalahan, tidak menyesatkan, dan harus jelas mencerminkan maksudnya. Ketidak akuratan dapat terjadi karena sumber informasi (data) mengalami gangguan atau kesengajaan sehingga merusak/merubah data-data asli tersebut.

c. Timeliness

Informasi yang dihasilkan/dibutuhkan haruslah tepat waktu, tidak boleh terlambat. Informasi yang tidak tepat waktu akan berakibat fatal dalam keputusan dan tindakan apabila digunakan sebagai dasar untuk mengambil keputusan.

d. Efficiency

Informasi yang diberikan haruslah efisien.

e. Reliability

Informasi tersebut harus dapat dipercaya.

2.4 **Pengertian Sistem Informasi**

Semua organisasi pasti memiliki sistem informasi. Sistem informasi adalah *entity* (kesatuan) formal yang terdiri dari berbagai sumberdaya fisik maupun logik. Dari organisasi ke organisasi, sumberdaya-sumberdaya ini disusun atau distrukturkan dengan beberapa cara yang berlainan karena organisasi dan sistem informasi merupakan sumberdaya-sumberdaya yang bersifat dinamis. Menurut sumber yang mendefinisikan sistem informasi.

> *"Sistem Informasi adalah suatu sistem manusia-mesin yang terpadu untuk menyajikan informasi guna mendukung fungsi operasi, manajemen dan pengambilan keputusan dalam organisasi".* [10]

Menurut Jeffrey L. Whitten, Lonnie D. Bentley, Kevin C. Dittman, sistem informasi adalah :

> *"Pengaturan orang, data, proses dan teknologi informasi yang berinteraksi untuk mengumpulkan, memproses, menyimpan, dan menyediakan sebagai output informasi yang diperlukan untuk mendukung sebuah organisasi ".*

Sistem Informasi dapat didefinisikan sebagai suatu sistem di dalam organisasi yang merupakan kombinasi orang-orang, fasilitas, teknologi, media, prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi yang penting, memproses tipe transaksi dan memberi sinyal kepada manajemen serta yang lainnya yang di dalamnya terdapat kejadian-kejadian eksternal dan

internal yang penting serta menyediakan suatu dasar informasi untuk pengambilan keputusan.

Sistem informasi dapat terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*), sebagai suatu sistem blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarannya.

1. Blok Masukan

   Input mewakili data yang masuk ke dalam sistem informasi. Input diisi termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

   Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di dasar data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran

   Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

   Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem keseluruhan. Teknologi terdiri dari 3 bagian utama

yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*).

5. Blok Dasar Data

Dasar data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali

Supaya sistem informasi dapat berjalan sesuai dengan yang diinginkan, maka perlu diterapkan pengendalian-pengendalian di dalamnya.

Kegiatan yang terdapat di dalam sistem informasi mencakup antara lain :

1. Input, menggambarkan suatu kegiatan untuk menyediakan data untuk diproses.
2. Proses, menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah.
3. Output, suatu kegiatan untuk menghasilkan laporan dari proses di atas tersebut.
4. Penyimpanan, suatu kegiatan untuk memelihara dan menyimpan data.
5. Kontrol, suatu aktivitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

2.5 Pengertian Data

Data adalah suatu objek dari perilaku kejadian yang berupa keterangan atau *file* yang menerangkan sesuatu. Dapat dibagi atas beberapa, sebagai berikut:

a. Pengolahan Data.

Pengolahan data dapat diartikan sebagai suatu proses untuk menghasilkan bentuk yang lebih berguna dan berarti dari suatu kejadian berupa informasi yang diperlukan.

b. Siklus Pengolahan Data

Suatu proses pengolahan data terdiri dari 3 tahapan dasar, yang disebut dengan siklus pengolahan data (*Data Processing Cycle*). Yaitu *input*, *Processing* dan *output* .[3]



Gambar 2.2 Siklus pengolahan data

c. Sistem Pengolahan Data

Sistem pengolahan data adalah suatu kesatuan yang saling berhubungan yang terdiri dari peralatan, tenaga pelaksana, dan prosedur-prosedur yang akan menghasilkan suatu informasi.

Proses pengolahan data biasanya terdiri dari sejumlah operasi pengolahan data yang dilaksanakan dalam beberapa langkah sebagai berikut :

1. Pencatatan

Prosedur pengolahan data dari suatu kejadian dituliskan ke dalam bentuk formulir atau dokumen yang berguna untuk masukkan dalam pengolahan data.

2. Pemeriksaan Data

Proses pemeriksaan terhadap data masukkan yang bertujuan agar data yang di dalam dokumen dapat di pindahkan ke dalam komputer secara tepat.

3. Pengelompokkan Data

   Proses pengelompokkan data berdasarkan kategori tertentu dan bertujuan untuk memudahkan atau memberi arti tertentu bagi pengolahan data

4. Penyusunan atau pemilahan

   Proses penyusunan data berdasarkan peringkat tertentu dengan memberikan item penyortirannya yang disebut *key* (kunci).

5. *Merging*

   Dalam proses merging dilakukan pencampuran dua atau lebih kumpulan data, semua kumpulan tersebut diurutkan berdasarkan *key* yang sama, dan meletakkan kumpulan tersebut bersama-sama menjadi bentuk kumpulan data tunggal yang telah diurutkan.

6. Perhitungan.

   Dalam pengolahan data sering diperlukan perhitungan tertentu terhadap data, agar dapat diperoleh hasil akhir yang diperlukan sebagai laporan.

7. Penyimpanan Data

   Proses ini dilakukan untuk menempatkan data ke dalam suatu media penyimpanan, misalnya kertas, *disk* dan sebagainya. Penyimpanan data ini berguna untuk memudahkan dalam pencarian data yang sebelumnya apabila diperlukan.

2.6 Pengertian Sistem Basis Data

Basis data dapat diartikan sebagai kumpulan data tentang suatu benda atau kejadian yang saling berhubungan satu sama lain. Sedangkan data merupakan fakta yang mewakili suatu obyek seperti manusia, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang dapat dicatat dan mempunyai arti yang implisit. Data dicatat atau direkam dalam bentuk angka, huruf, simbol, gambar, bunyi, atau kombinasinya. Sebagai contoh, terdapat daftar nama, nomor telepon, dan alamat orang-orang yang menjadi anggota suatu organisasi. [Waljiyanto00]

Konsep mengenai basis data dapat dipandang dari beberapa sudut. Dari sisi sistem, basis data merupakan kumpulan tabel-tabel atau files yang saling berelasi. Sementara dari sisi manajemen, basis data dapat dipandang sebagai kumpulan data yang memodelkan aktivitas-aktivitas yang terdapat di dalam *enterprise*-nya. Selain itu, basis data juga mengandung pengertian kumpulan data *non-redundant* yang dapat digunakan bersama (*share*) oleh sistem-sistem aplikasi yang berbeda.

Penggunaan istilah basis data menurut Elmasri R. (1994) lebih dibatasi pada arti implisit yang khusus, yaitu :

1. Basis data merupakan penyajian suatu aspek dari dunia nyata (*"real word"* atau *"miniword"*).
2. Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit. Sehingga data yang terkumpuk secara acak dan tanpa mempunyai arti, tidak dapat disebut basis data.

3. Basis data perlu dirancang, dibangun, dan data dikumpulkan dalam suatu tujuan. Basis data dapat digunakan oleh beberapa pemakai dan beberapa aplikasi sesuai dengan kepentingan pemakai.

Dari keterangan diatas dapat disimpulkan bahwa basis data mempunyai berbagai sumber data dalam pengumpulan data, bervariasi derajad interaksi kejadian dari dunia nyata, dirancang.

Berikut adalah beberapa pengertian atau definisi lain dari basis data yang dikembangkan atau dasar sudut pandang yang berbeda dan diambil dari pustaka [Fathan02]:

1. Himpunan kelompok data (file atau arsip) yang saling berhubungan dan saling berorganisasi sedemikian rupa agar dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan dan disimpan bersama sedemikian rupa tanpa pengulangan yang tidak perlu (*redudancy*) untuk memenuhi berbagai kebutuhan.
3. Kumpulan file atau arsip atau tabel yang saling berhubungan yang disimpan di dalam media penyimpanan elektronik.

Sistem basis data merupakan perpaduan antara basis data dan sistem manajemen basis data (*DBMS*). Komponen-komponen sistem basis data meliputi :

1. Perangkat Keras (*Hardware*) sebagai pendukung operasi pengolahan data.
2. Sistem Operasi (*Operating System*) atau perangkat lunak untuk mengelola basis data.

3. Basis data (*Database*) sebagai inti dari sistem basis data.
4. Sistem Manajemen Basis Data (*DBMS*).
5. Pemakai (*User*).
6. Aplikasi lain.

Sistem manajemen basis data (*DBMS*) adalah suatu program komputer yang memasukkan, mengubah, menghapus, memanipulasi, dan memperoleh data dan informasi dengan praktis dan efisien. *DBMS* merupakan suatu paket perangkat lunak (tanpa basisdata) *general purpose* yang digunakan untuk membangun sistem basisdata tertentu.

DBMS akan menentukan bagaimana data diorganisasikan, disimpan, diubah dan dipanggil. DBMS juga yang menerapkan mekanisme pengamanan data, penggunaan data bersama *(data sharing),* konsistensi data dan sebagainya. Perangkat lunak yang termasuk DBMS adalah keluarga DBASE (II, III+, IV, dll) atau XBase, FoxBase, FoxPro, Paradox, Clipper, Access, Oracle, Sybase, DB2, dan sebagainya.

Didalam DBMS terdapat beberapa model basisdata yang digunakan. Model ini menyatakan hubungan antara *record-record* yang ada dalam basisdatanya. Model basisdata tersebut adalah :

1. *Flat file* (tabular) -- Data terletak dalam tabel tunggal (tidak terdapat kaitan antara tabel yang satu dengan table-tabel lainnya).
2. *Hierarchical* -- Pada model ini data diorganisasikan kedalam struktur hubungan hirarki bapak-anak (*Parent-Child*) sebagai satu *tree*. Setiap *parent*

dapat memiliki *child* lebih dari satu, sementara setiap *child* hanya memiliki satu *parent*.

3. *Network* -- Model ini sangat mirip dengan model *hierarchical*, tetapi pada model ini setiap *child* dapat memiliki lebih dari satu *parent.*
4. *Relational* – Model ini terdiri dari tabel-tabel data (data direpresentasikan dalam tabel yang terdiri dari baris-baris dan kolom-kolom) ternormalisasi dengan *field-field* kunci sebagai penghubung relasional antar tabel.

2.7 Konsep Dasar Rekayasa Perangkat Lunak

Dalam pengembangan perangkat lunak terdapat proses di dalam pengembanganya, mungkin kita bertanya apakah sebenarnya *perangkat lunak* itu? *Dalam bukunya, Roger S. Pressman, Ph.D mendefinisikan sebagai berikut :*

> *"(1) perintah (proses computer) yang bila dieksekusi memberikan fungsi dan unjuk kerja yang diinginkan, (2) struktur data yang memungkinkan program memanipulasi informasi secara propesional, dan (3) Dokumen yang menggambarkan operasi dan keggunaan program. Proses perangkat lunak adalah sebuah kerangka kerja untuk tugas – tugas yang dibutuhkan untuk membangun perangkat lunak dengan kualitas yang tinggi"*

2.7.1 Rekayasa Perangkat lunak

Definisi perangkat lunak yang diusulkan oleh Fritz Bauer adalah :

"Rekayasa perangkat lunak adalah pengembangan dan penggunaan prinsip pengembangan untuk memperoleh perangkat lunak secara ekonomis yang realiabel dan bekerja secara efisien pada mesin nyata."

International (IEEE) telah mengembangkan definisi yang lebih komperehensif sebagai berikut :

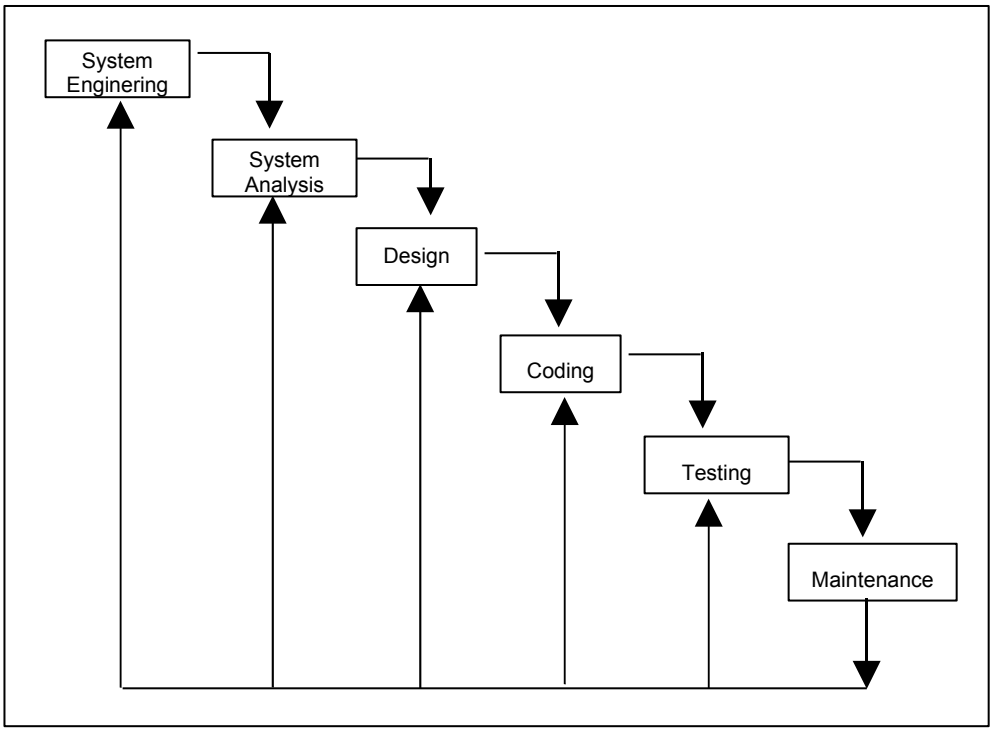
"Rekayasa perangkat lunak : (1) Aplikasi dari sebuah Pendekatan kuantifiabel, disiplin, dan sistematis kepada pengembangan, operasi, dan pemeliharaan perangkat lunak; yaitu aplikasi dari rekayasa perangkat lunak.(2) Studi tentang Pendekatan-pendekatan seperti pada butir "

2.7.2 Model Proses Rekayasa Perangkat Lunak

Model proses untuk rekayasa perangkat lunak dipilih berdasarkan sifat aplikasi dan proyeknya, metode dan alat-alat bantu yang akan dipakai, dan kontrol serta penyampaian yang dibutuhkan. Model-model proses tersebut adalah :

1. Model Sekuensial Linier atau model air terjun .
2. Model *prototype*.
3. Model RAD (*Rapid Application Development*).
4. Model Pertambahan.
5. Model Spiral.
6. Model Rakitan Komponen.
7. Model Perkembangan Konkuren.
8. Model Formal.
9. Teknik Generasi Keempat.

Dalam menyelesaikan tugas akhir ini digunakan metode paradigma *waterfall* (*classic life cycle*). Metode ini mudah digunakan dan jika terjadi kesalahan tidak perlu diperbaiki dari awal, adapun diagram dari paradigma *waterfall* adalah sebagai berikut :



Gambar 2.3 Proses Waterfall (classic life cycle)

Model sekuensial linier yang sering disebut juga dengan *classic life cycle* atau *waterfall* mempunyai beberapa tahap, sebagai berikut :

1. Rekayasa sistem (*System Engineering*)

   Sebelum sistem dikembangkan, terlebih dahulu melakukan pengumpulan data yang berhubungan dengan pembuatan sistem. Tahapan pertama ini ditujukan untuk memperoleh data mengenai permasalahan yang dihadapi. Teknik pengumpulan data yang digunakan adalah : Studi pustaka, wawancara dan studi dokumentasi.

2. Analisa Sistem (*System Analysis*)

   Merinci lebih jauh mengenai sistem yang dikembangkan dengan menggunakan DAD (Diagram Alir Dokumen) beserta *flowmap*. Sehingga diperoleh spesifikasi dari sistem yang akan dibangun.

3. Perancangan (*Design*)

   Mempelajari terlebih dahulu sistem aplikasi yang akan dipakai, kemudian menentukan kerangka dasar serta komponen penunjang lainnya dengan pembuatan DFD (*Data Flow Diagram*) dan juga diagram hubungan antar entitas (*Entity Relationship-Diagram*) yang memudahkan desain implementasi sistem.

4. Pengkodean (*Coding*)

   Adalah kegiatan penulisan kode program yang akan dieksekusi oleh computer berdasarkan hasil dari desain rinci yang akan menghasilkan program yang sesuai dengan kebutuhan pemakai. Adapun *software developer* yang dipakai untuk pembuatan kode adalah Borland Delphi 7 *Enterprise Edition*.

5. Pengujian (*Testing*)

   Adalah tahapan-tahapan pengujian antar modul dan pengujian integrasi, untuk mengantisipasi kesalahan yang mungkin terjadi ketika aplikasi siap dilepas ke pasar, agar mendapatkan hasil yang diinginkan. Metoda yang digunakan dalam testing ini adalah *alpha dan beta-test*, artinya pengujian dilakukan oleh pembuat program dan user yang bersangkutan.

6. Implementasi dan Pemeliharaan (*Implementation and Maintenance*)

Adalah tahapan terakhir dimana perangkat lunak yang sudah selesai dan dapat mengalami perubahan-perubahan dan penambahan sesuai dengan kebutuhan.

2.7.3 Pendekatan Rekayasa Perangkat Lunak

Terdapat beberapa Pendekatan yang dapat digunakan untuk menganalisa dan memodelkan suatu permasalahan. Misalnya berorientasi objek (coad P&E. yourdan : OOA), berorientasi struktur data, misalnya *data structure system development* (DSSD/warnier-Orr dan *Jackson sistem development*), dan berorientasi analisis terstruktur (*Structured Analysis*).

Dalam skripsi ini, untuk membuat pemodelan dan menganalisis permasalahan yang ada, penulis menggunakan pendekatan analisis terstruktur (*structured analysist*) pertimbangannya adalah karena metode ini menurut Tom Demarco (1979) telah berkembang dengan baik dan memperoleh dukungan substansi oleh komunitas rekayasa perangkat lunak.

a. Analisis terstruktur

Pokok keberhasilan dari usaha pengembangan dan pengejaan perangkat lunak adalah lengkapnya spesipikasi keperluan yang fungsinya adalah menemukan permasalahan dan alternatif pemecahan yang relevan. Langkah pertama yaitu menentukan batasan (*scope*) perangkat yang akan dikerjakan lalu diperbaiki dan dianalisis, sehingga batasan secara rinci diketahui dengan jelas dan pasti. Kemudian, alternatif pemecahaan masalah yang ada dianalisis dan dialokasikan pada elemen-elemen perangkat lunak.

Analisis keperluan perangkat lunak dapat dibagi menjadi empat, yaitu :

1. Pengenalan masalah.
2. Evaluasi dan sintesis.
3. Spesifikasi.
4. *Review* (Tinjau ulang).

Selama melakukan tahapan analisis keperluan terhadap perangkat lunak, sistem atau permasalahan yang dihadapi dibuatkan model-model analisisnya. Model yang dibuat tersebut menekankan terhadap apa yang akan dilakukan oleh sistem saja, dan tidak menggambarkan bagaimana sistem tersebut melakukan prosesnya.

Model analisis harus dapat mencapai tiga sasaran utama :

1. Untuk menggambarkan apa yang dibutuhkan oleh pelanggan.
2. Untuk membangun dasar bagi pembuatan desain perangkat lunak.
3. Untuk membatai serangkaian persyaratan yang dapat divalidasi pada saat perangkat lunak dibangun.

Pada inti model terdapat kamus data (*Data Dictionary*), yaitu penyimpanan yang berisi deskripsi dari semua objek data yang dikonsumsi atau diproduksi oleh perangkat lunak. *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM) yang menggambarkan hubungkan antara objek data.

Data Flow Diagram (DFD) mempunyai dua tujuan :

a. Untuk memberikan indikasi mengenai bagaimana data ditransformasi pada saat data bergerak melalui sistem,
b. Untuk mentransformasikan aliran data DFD memberikan informasi tambahan yang digunakan selama analisis domain informasi dan berfungsi

sebagai dasar bagi Pemodelan fungsi. Deskripsi setiap fungsi yang disajikan pada DFD diisikan dalam sebuah Spesifikasi Proses (*Process Specification* atau PSPEC).

Informasi tambahan mengenai aspek kontrol dari perangkat lunak diisikan dalam spesifikasi kontrol (*Control Specifidation* atau CSPEC).

b. **Desain Perangkat Lunak**

Desain perangkat lunak berada pada inti teknik dari proses rekayasa perangkat lunak dan aplikasikan tanpa memperhatikan model proses perangkat lunak yang digunakan . ketika persyaratan perangkat lunak telah mulai dianalisis dan ditentukan, maka aktivitas selanjutnya adalah desain perangkat lunak.

Elemen-elemen pada model analisis terstruktur memberikan informasi yang diperlukan untuk menciptakan suatu model persyaratan perangkat lunak yang ditunjukkan oleh data, fungsional, dan model-model prilaku mengisi langkah desain. Tahap desain menghasilkan desain data, desain arsitektur, desain *interface*, dan desain prosedural.

Desain data mentransformasi model domain informasi yang dibuat selama analisis ke dalam terstruktur utama dari program. Representasi desain tersebut dapat diperoleh dari model-model analisis dan interaksi subsistem yang ditentukan dalam model analisis.

Desain *interface* menggambarkan bagaimana perangkat lunak berkomunikasi dalam dirinya sendiri, dengan sistem yang beroperasi denganya, dan dengan manusia yang menggunakannya. *Interface* mengimplementasikan aliran informasi.

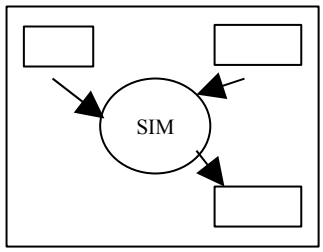
Desain prosedural mentransformasi elemen-elemen struktural dari arsitektur program kedalam suatu deskripsi prosedural dari komponen-komponen perangkat lunak. Informasi yang diperoleh dari PSPEC, CSPEC, dan STD berfungsi sebagai dasar bagi desain prosedural:

1. **Diagram Konteks *(Context Diagram)***

Diagram konteks adalah arus data yang berfungsi untuk menggambarkan keterkaitan aliran-aliran data sistem dengan bagian-bagian luar (kesatuan luar). Kesatuan luar ini merupakan sumber arus data atau tujuan data yang berhubugan dengan sistem informasi tersebut.

Diagram konteks bisa artikan sebagai :

> "*Model sistem pokok (fundamental system model) mewakili elemen software dengan masukan (input) dan keluaran (output) yang diindikasikan dengan anak panah masuk dan keluar memperlihatkan suatu hubungan antara sistem dengan lingkungannya yang menjadi sumber data". [9]*



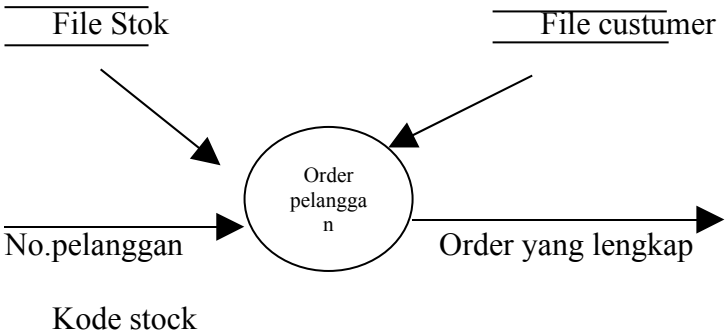
Gambar 2.4 Contoh diagram Kontek

2. ***Data Flow Diagram (DFD)***

Data flow diagram adalah :

"Teknik grafis yang mengambarkan aliran informasi dan perubahan yang dipergunakan sebagai perpindahan data dari masukan (input) ke keluaran (output)". [9]

Terbagi atas beberapa level yang mengambarkan peningkatan aliran informasi dan detail fungsional. Arus data yang ditunjukkan pada suatu level harus sama dengan level sebelumnya.



Gambar. 2.5 Contoh diagram level n

Untuk melambangkan fungsi di atas maka dipergunakanlah simbol-simbol yang bisa dilihat pada daftar simbol. Sedangkan arti dari masing-masing simbol adalah sebagai berikut :

a. Data Entitas

   Entitas merupakan individu yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Data entitas adalah segala sesuatu baik yang nyata maupun abstrak yang datanya akan direkam.

b. Atribut

   Atribut mendefinisikan karakteristik *(property)* dari entitas tersebut.

c. Relasi Data

Relasi menunjukkan adanya hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

d. Kardinalitas

Kardinalitas adalah penentuan jumlah kejadian dari satu entitas yang dapat terhubung dengan sejumlah kejadian pada entitas lain, juga menujukkan jumlah maksimum entitas yang dapat berelasi terhadap himpunan entitas yang lain. Terdapat empat kemungkinan kombinasi kardinalitas :

1. Satu ke Satu *(One to One)*

   Setiap entitas yang ada pada himpunan entitas A hanya boleh berhubungan satu entitas yang ada di himpunan entitas B, begitu pula sebaliknya.

2. Satu ke Banyak *(One to Many)*

   Setiap entitas yang ada pada himpunan entitas A boleh berhubungan ke banyak entitas yang ada di himpunan entitas B. Sedangkan untuk setiap entitas yang ada pada himpunan entitas B hanya boleh berhubungan satu entitas yang ada di himpunan entitas A.

3. Banyak ke satu *(Many to One)*

   Setiap entitas yang ada pada himpunan entitas A hanya boleh berhubungan ke satu entitas yang ada di himpunan entitas B. Sedangkan untuk setiap entitas yang ada pada himpunan entitas B boleh berhubungan ke banyak entitas yang ada di himpunan entitas A.

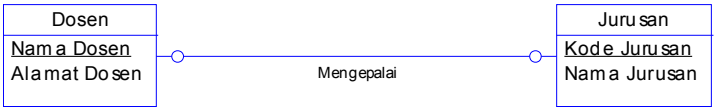
4. Banyak ke Banyak *(Many to Many)*

   Setiap entitas yang ada pada himpunan entitas A boleh berhubungan

ke banyak entitas yang ada pada himpunan entitas B, begitu pula sebaliknya.

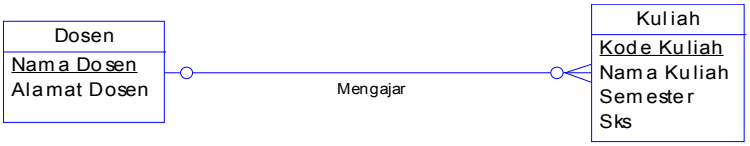
Berikut adalah contoh penggambaran relasi antara himpunan entitas lengkap dengan kardinalitas relasi dan atribut-atributnya:

1. Relasi **satu-ke-satu**



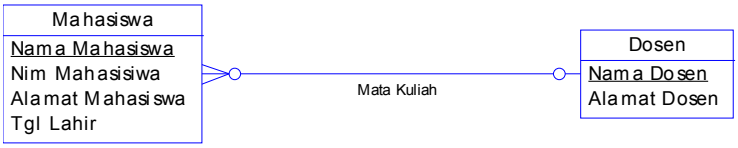
Gambar 2.6 Contoh relasi satu ke Satu

2. Relasi **satu-ke-banyak**



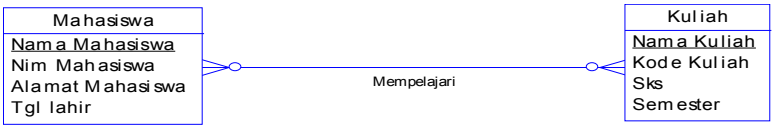
Gambar 2.7 Contoh relasi satu ke Banyak

3. Relasi **banyak-ke-satu**



Gambar 2.8 Contoh relasi Banyak ke-Satu

4. Relasi **banyak-ke-banyak**



Gambar 2.9 Contoh relasi Banyak ke-Banyak

3. **Kamus Data**

Kamus data yang digunakan dalam struktur analis dan desain merupakan suatu katalog yang menjelaskan lebih detail Diagram Arus Data mencakup proses, arus data, tempat penyimpanan data dan terminologi bisnis. Kamus data dapat digunakan di metodologi berorientasi data dengan menjelaskan lebih detail Diagram Hubungan Entitas seperti atribut-atribut suatu entitas dan di metodologi objek, kamus data dapat menjelaskan lebih detail atribut maupun *method atau service* suatu objek. Kamus data dapat digunakan saat analisa sistem maupun pada saat desain analisa sistem, pada saat analisa kamus data dapat digunakan untuk mencatat terminologi bisnis (seperti memo kredit, memo debet retur ,batas kredit dan sebagainya), selain itu juga dapat digunakan untuk mencatat busines rule standar (seperti panjang karakter, batasan nilai suatu *field*), *bussiness rule custom* (syarat-syarat penetapan diskon). Kamus data dan komponen-konponen lainnya yang dikumpulkan saat analisa sistem sangat dibutuhkan dalam perancangan sistem dan dapat juga digunakan oleh *programmer* yang membutuhkan penjelasan tambahan), sedangkan kamus data saat desain sistem merupakan petunjuk penting untuk pembuatan sistem.

Selain dapat menjelaskan suatu model, kamus data juga berfungsi untuk menghindari penggunaan kata yang sama, karena kamus data disusun menurut abjad.

a. Notasi Kamus Data

Notasi kamus data lebih mudah menjelaskan data dibandingkan dengan narasi.

b. Notasi Struktur Data

Digunakan untuk membuat spesifikasi elemen data. Notasi yang umum digunakan antara lain:

Tabel 2.1 Notasi Struktur Data

Notasi	Keterangan
=	Terdiri dari
+	And (dan)
( )	Pilihan (boleh ada atau tidak)
{ }	Iterasi/pengulangan
[ ]	Pilih salah datu pilihan
l	Pemisah pilihan didalam tanda []
*	Keterangan/catatan
@	Petunjuk (Key Field)

c. Notasi Tipe Data

Digunakan untuk membuat spesifikasi format input maupun output suatu data. Notasi yang umum digunakan antara lain :

Tabel 2.2 Notasi Tipe Data

Notasi	Keterangan
X	Setiap karakter
9	Angka Numerik
A	Karakter Alfabet
Z	Angka nol ditampilkan sebagai spasi kosong
.	Titik, sebagai pemisah ribuan
,	Koma, sebagai pemisah pecahan
-	Tanda penghubung (contoh: 021-6625825)
/	Tanda pembagi (Contoh 31/01/2005)

2.8 Tinjauan Umum Pengelolaan Gudang

Tinjauan ini untuk mengetahui konsep dasar dan definisi yang berkaitan dengan pengelolaan gudang.

2.8.1 Arti dan Peranan Pengelolaan Gudang

Setiap perusahaan selalu memerlukan pengelolaan gudang. Tanpa adanya pengelolaan gudang, para pengusaha akan dihadapkan pada resiko bahwa

perusahaannya pada suatu waktu tidak dapat menjalankan perusahaan. Hal ini bisa terjadi, karena tidak selamanya barang-barang tersedia pada setiap saat, yang berarti pula bahwa pengusaha akan kehilangan kesempatan yang seharusnya ia dapatkan.

Manajemen pengelolaan gudang dapat didefinisikan sebagai proses pemindahan dan penyimpanan barang, suku cadang dan barang jadi dari para pemasok atau barang produksi, dan bertujuan menyampaikan bermacam-macam barang dan jumlah yang tepat pada waktu yang dibutuhkan, dalam keadaan yang dapat dipakai, ke lokasi di mana ia dibutuhkan dan dengan total biaya yang terendah.

Pengelolaan gudang yang diadakan dimulai dari pemesanan sampai penerimaan, pendistribusian dan pemeliharaan barang yang tersimpan, antara lain:

1. Menghilangkan resiko keterlambatan datangnya barang.
2. Menghilangkan resiko barang yang rusak.
3. Mempertahankan stabilitas operasi perusahaan.
4. Mencapai penggunaan mesin yang optimal.

Catatan penting dalam sistem pengawasan pengelolaan gudang :

1. Permintaan untuk di beli.
2. Laporan penerimaan.
3. Catatan persediaan
4. Daftar permintaan barang.

Pengelolaan gudang ditujukan untuk mengantisipasi kebutuhan permintaan

barang. Permintaan ini meliputi, persediaan barang mentah, barang dalam proses, barang jadi atau produk akhir, bahan-bahan pembantu atau pelengkap, dan komponen-komponen lain yang menjadi bagian keluaran produk perusahaan atau produk yang dijual.

Fungsi-fungsi pengelolaan gudang dalam hal penanganan stok barang :

1. Fungsi *Decoupling*

Adalah persediaan yang memungkinkan perusahaan dapat memenuhi permintaan langganan tanpa tergantung pada supplier. Persediaan barang mentah diadakan agar perusahaan tidak akan sepenuhnya tergantung pada pengadaannya dalam hal kuantitas dan waktu pengiriman. Persediaan barang dalam proses diadakan agar departemen- departemen dan proses- proses individual perusahaan terjaga "kebebasannya". Persedian barang jadi diperlukan untuk memenuhi permintaan produk yang tidak pasti dari para pelanggan. Persediaan yang diadakan menghadapi fluktuasi permintaan konsumen yang tidak dapat diperlukan atau diramalkan disebut *fluctuation stock.*

2. Fungsi *Economic Lot Sizing*.

Persediaan *lot size* ini perlu mempertimbangkan penghematan-penghematan atau potongan pembelian, biaya pengangkutan per unit menjadi lebih murah dan sebagainya. Hal ini disebabkan karena perusahaan melakukan pembelian dalam kuatitas yang lebih besar, dibandingkan dengan biaya- biaya yang timbul karena besarnya persediaan (biaya sewa gudang, investasi, resiko, dan sebagainya).

3. Fungsi Antisipasi.

Apabila perusahaan menghadapi fluktuasi permintaan yang dapat diperkirakan dan diramalkan berdasarkan pengalaman atau data-data masa lalu, yaitu permintaan musiman. Dalam hal ini perusahaan dapat mengadakan persediaan musiman (*Seasional Inventories*).

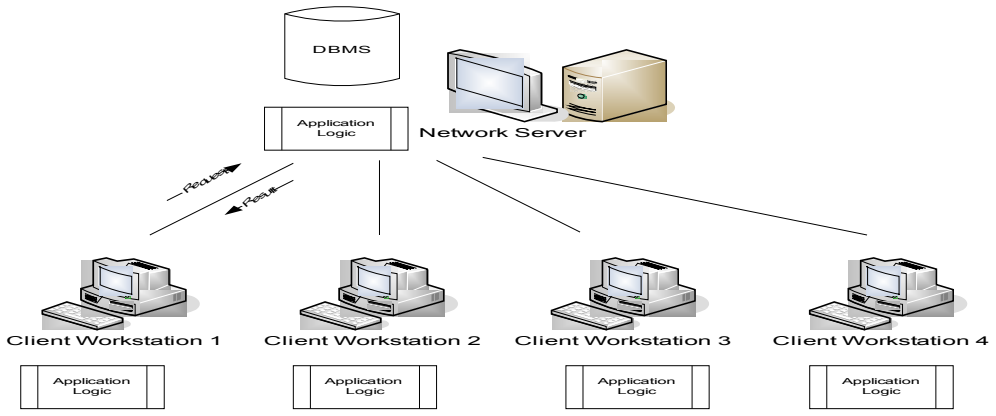
Disamping itu, perusahaan sering menghadapi ketidakpastian jangka waktu pengiriman dan permintaan barang-barang selama periode tertentu. Dalam hal ini perusahaan memerlukan persediaan ekstra yang disebut persediaan pengamanan (*Safety Stock atau Inventories*).

2.9 ***Client- Server***

Definisi yang banyak dipergunakan untuk menjelaskan sistem berbasis *client*/*server* adalah "sistem yang memisahkan antara tugas-tugas komputasi antara proses-proses *client* dan *server*." Dengan sistem *client/server*, kekuatan pemrosesan bisa disebarkan (didistribusikan) ke banyak mesin *client* dan mesin *server* yang terpisah secara fisik (itu sebabnya disebut *distributed system*).

Client/server mengacu kepada proses yang melibatkan setidaknya dua entitas yang independen, satu sebagai *client*, dan yang lainnya sebagai server. *Client* me-*request* ke *server* dan *server* melayani *request* dari *client*. Sebuah *request* dapat merupakan bentuk dari sebuah *query* SQL yang dikirimkan melalui *database engine* SQL. *Database engine* merubah proses permintaan (*request*) dan mengembalikkan ke client dalam bentuk hasil yang diinginkan. Dalam contoh ini, dua proses yang independen bekerja secara bersamaan untuk menyelesaikan tugas yang sama. *Windows printing* dan *print manager* adalah salah satu contoh

sederhana hubungan sistem *client/server*. Aplikasi *Windows*, seperti *Ms Excel* dan *Ms Word*, menyajikan dokumen dan disaat dokumen tersebut akan dicetak, maka dikirimkan perintah kepada *print manager*. *Print Manager* menyediakan servis untuk mengantrikan permintaan dan mengirimkan ke permintaan tersebut ke *printer*, memonitor pekerjaan yang sedang dilakukan, dan kemudian memberi tahu kepada aplikasi bila pekerjaan sudah selesai. Dalam contoh kasus ini *print manager* dianggap sebagai *server*, menyediakan servis untuk mengantrikan dan memproses pekerjaan pencetakan dokumen. Aplikasi *Ms Word* dan *Ms Excel* disebut sebagai *client*. Contoh ini menggambarkan bagaimana hubungan *client/server* dapat terjadi antar aplikasi yang mungkin tidak berhubungan dengan database.



Gambar 2.10 Arsitektur *Client/Server*

2.10 Borland Delphi

Delphi 7.0 adalah perangkat pengembang untuk membangun program aplikasi berbasis *windows*. Bahasa pemrogram yang digunakan adalah Object

Pascal, nama lengkapnya adalah Borland Delphi 7.0 dan diproduksi oleh Borland International, Inc sebuah perusahaan *software* dari Amerika.

Borland telah berpengalaman memproduksi perangkat pengembang yang andal seperti Turbo Pascal, Turbo Basic, Turbo C, Turbo Prolog, Turbo Assembler, Visual dBase, Borland C++, Borland Pascal maupun Borland C++ Builder. Delphi adalah perangkat lunak untuk menyusun program aplikasi yang berdasarkan bahasa pemrograman pascal dan bekerja dalam lingkungan operasi *Windows*.

Dengan menggunakan delphi pemrogram akan merasa begitu mudah menyusun program aplikasi, karena delphi menggunakan komponen-komponen penulisan program. Dengan delphi pemrogram tidak perlu lagi membuat *windows*, kotak dialog maupun perangkat kontrol lainnya seperti tombol perintah (*Command Box*), *menu pull down*, kotak combo (*Combo box*) dan sebagainya. Karena semuanya telah disediakan oleh delphi, pemrogram hanya tinggal mengklik saja dengan mouse. Dengan delphi pemrogram dapat membangun suatu program aplikasi dalam tatanan GUI (*Graphical User Interface*). Program aplikasi yang telah menerapkan GUI, banyak menggunakan perantara untuk erinteraksi dengan pemakai. Sarana yang sering disebut kontrol ini antara lain kotak dialog, tombol, menu, kotak cel, panel dan lain-lain.

2.11 Database MySQL

MySQL termasuk dalam kategori *database management system*,yaitu database yang terstruktur dalam pengolahan dan penampilan data. Sejak computer

dapat menangani data yang besar, database management system memegang peranan yang sangat penting dalam pengolahan data. Hal ini sangat diperlukan, karena data tersebut dapat diatur sesuai dengan kebutuhan pemakainya. MySQL merupakan *Relation Database Managemen System* yang dikembangkan dari bahasa SQL (*Structured Query Language*). SQL adalah bahasa yang terstuktur yang digunakan untuk interaksi antara script program dengan database server dalam hal pengolahan data dengan SQL, kita dapat membuat tabel yang nantinya akan diisi dengan data, manipulasi data (misalnya menambah, menghapus dan meng-update data), serta membuat suatu perhitungan dengan berdasarkan data yang ditemukan oleh karena itu SQL tidak hanya terbatas digunakan untuk mendapat suatu tampilan dari database yang statis.

2.12 Microsoft Visio

Merupakan aplikasi Bantu yang digunakan oleh penulis dalam pembuatan flowmap, diagram kontek, DFD, ER dan relasi. Aplikasi bantu ini terdiri dari *tool* yang mudah untuk digunakan sehingga sangat membantu penulis dalam pengerjaan tugas akhir ini.