BAB II

LANDASAN TEORI

2.1 Ruang Lingkup Perusahaan

Sesuai dengan perkembangan, dimana peran aparatur pemerintah semakin dirasakan, pemerintah menganggap perlu menetapkan kembali kedudukan, fungsi, tugas, dan organisasi KUP. Pandangan ini sebagaimana diatur dalam Peraturan Pemerintah Nomor 32 Tahun 1950 beserta peraturan pelaksanaannya yang dimaksud dalam Keputusan Perdana Menteri RI Nomor 30/PM/1951 tanggal 7 April 1951.

Untuk maksud tersebut, maka KUP yang merupakan institusi yang bertugas melakukan pembinaan kepegawaian diubah menjadi Badan Administrasi Kepegawaian Negara (BAKN) dengan Peraturan Pemerintah Nomor 32 Tahun 1972. Penetapan Peraturan Pemerintah ini adalah juga sebagai pelaksanaan dari Undang-undang Nomor 18 Tahun 1961. Dengan Peraturan Pemerintah Nomor 32 Tahun 1972, maka kedudukan, fungsi, tugas, susunan dan tata kerja institusi yang mengelola kepegawaian, semakin dikembangkan.

Berdasarkan Peraturan Pemerintah tersebut, BAKN ditetapkan sebagai sebuah lembaga pemerintah non departemen yang berkedudukan langsung dibawah dan bertanggung jawab kepada Presiden, mempunyai fungsi untuk menyempurnakan,

memelihara dan mengembangkan administrasi negara di bidang kepegawaian sehingga tercapai kelancaran jalannya pemerintahan.

Untuk dapat menyelenggarakan fungsinya, BAKN mempunyai tugas sebagai berikut :

1. Merencanakan pembinaan kepegawaian sesuai dengan kebijaksanaan Presiden;
2. Merencanakan peraturan perundang-undangan di bidang kepegawaian;
3. Menyelenggarakan tata usaha kepegawaian dan tata usaha pensiun;
4. Menyelenggarakan pengawasan, koordinasi dan bimbingan terhadap pelaksanaan peraturan perundang-undangan di bidang kepegawaian dan pensiun pada departemen-departemen dan lembaga-lembaga negara/Lembaga-lembaga Pemerintah Nondepartemen.

Sedangkan susunan organisasi BAKN, terdiri dari :

1. Kepala
2. Sekretariat
3. Biro-biro
4. Staf Ahli.

BAKN dipimpin oleh seorang Kepala yang dalam melaksanakan tugasnya bertanggung jawab langsung kepada Presiden.

Sekretariat BAKN merupakan unsur pembantu pimpinan dan mempunyai tugas untuk menyelenggarakan administrasi umum meliputi :

1. tata usaha kantor dan urusan dalam;
2. tata kepegawaian (personalia);
3. tata peralatan; dan
4. tata keuangan

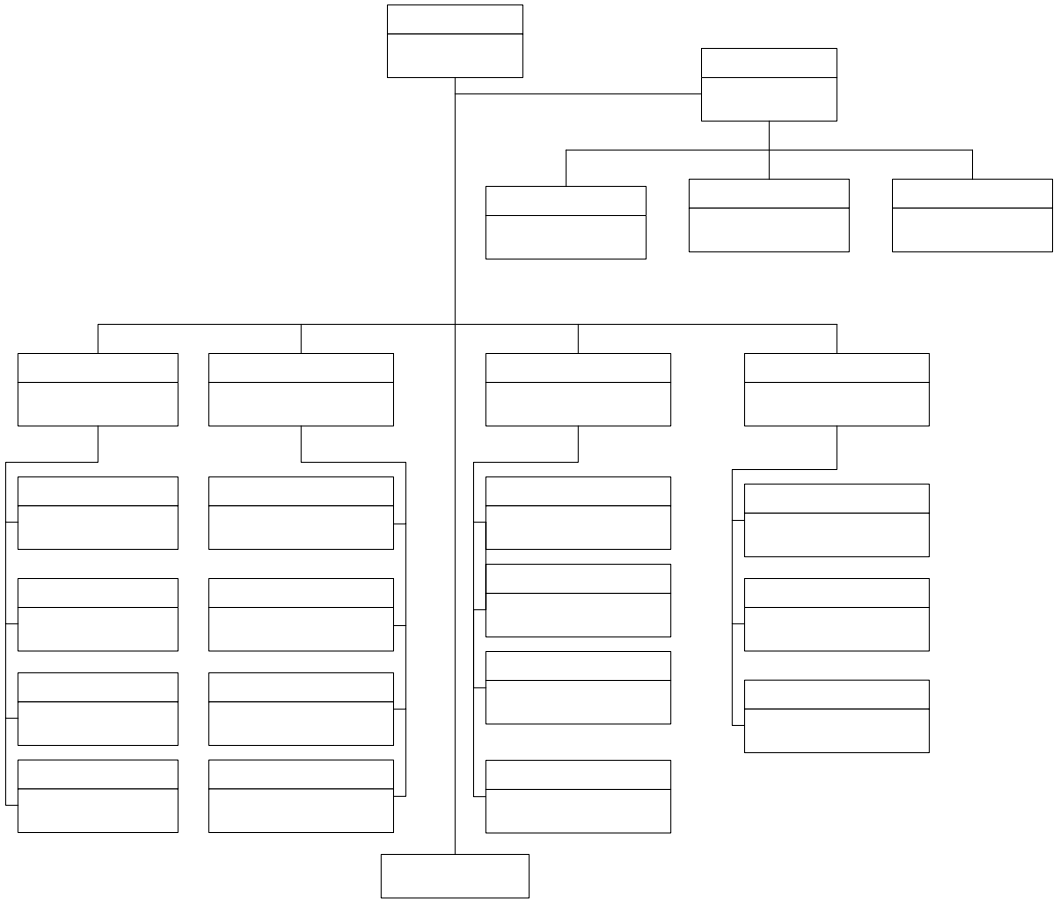
Sekretariat BAKN terdiri dari bagian-bagian sebanyak-banyaknya 5(lima) bagian, dan bagian-bagian terdiri dari sub-sub bagian, masing-masing sebanyak-banyaknya 5(lima) Sub Bagian.

Biro-biro merupakan unsur pelaksana yang terdiri dari :

1. Biro Perencanaan;
2. Biro Kepegawaian Umum;
3. Biro Kepangkatan dan Penggajian;
4. Biro Tata Usaha Kepegawaian;
5. Biro Pensiun dan Tunjangan;
6. Biro pengawasan.

Tiap-tiap Biro terdiri atas bagian-bagian, masing-masing bagian sebanyak-banyaknya 5(lima) bagian dan bagian-bagian terdiri atas sub-sub bagian, masing-masing sebanyak-banyaknya 5 (lima) sub bagian.

Anggaran Belanja BAKN menurut Peraturan Pemerintah Nomor 32 Tahun 1972 ini dibebankan kepada Anggaran Belanja Sekretariat Negara/Sekretaris Kabinet. Dengan berlakunya Peraturan Pemerintah Nomor 32 Tahun 1972 maka penataan, pembinaan dan pengembangan administrasi kepegawaian sebagai bagian dari pembinaan aparatur pemerintah, dapat dilaksanakan secara lebih berdayaguna dan berhasil guna.



KEPALA KANRE

Drs DEDE DJUNA
NIP 26000336

Gambar 2.1. Struktur Organisasi Badan Kepegawaian Negara Regional III

2.2 Landasan Teori

Pada bagian ini akan dibahas teori sistem informasi yang berhubungan dengan sistem yang akan dibuat. Penjelasan secara rinci akan diuraikan pada sub-sub bab berikut ini.

2.2.1 Pengertian Sistem

Terdapat dua kelompok pendekatan dalam mendefinisikan sistem, yaitu :

a. Menekankan pada prosedurnya

Pendekatan sistem yang lebih menekankan pada prosedur menurut Jogiyanto HM (Analisis dan Desain Sistem Informasi, 2001, hal 1) mendefinisikan sistem sebagai berikut :

> "*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*".

b. Menekankan pada komponen atau elemen.

Menurut Jogiyanto HM (Analisis dan Desain Sistem Informasi, 2001, hal 2) mendefinisikan sebagai berikut :

> "*Sistem adalah kumpulan dari elemen-elemen yang berinterkasi untuk mencapai suatu tujuan tertentu*".

Pengertian kedua sistem tersebut benar dan tidak bertentangan, yang berbeda hanya cara pendekatannya saja.

Karakteristik sistem terdiri dari :

1. Komponen Sistem

Komponen sistem terdiri dari sejumlah komponen sistem yang berinteraksi dan bekerja sama membentuk satu kesatuan komponen-komponen yang dapat terdiri dari beberapa subsistem, dimana setiap subsistem tersbut memiliki fungsi khusus dan akan mempengaruhi proses secara keseluruhan.

2. Batasan Sistem

Batasa sistem merupakan daerah yang membatasi antara satu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan Luar Sistem

Lingkungan luar sistem adalah apaun yang diluar batas sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan. Lingkungan luar sistem yang meguntungkan merupakan energi dari sistem, dengan demikian harus tetap dijaga dan dipelihara tetapi lingkungan luar yang merugikan harus ditahan dan dikendalikan, jika tidak akan mengganggu kelangsungan hidup sistem.

4. Penghubung

Penghubung sistem merupakan media penghubung antara subsistem dengan subsistem lainnya, melalui penghubung ini memungkinkan sumberdaya mengalir dari satu subsistem kesubsistem lainnya.

5. Masukan Sistem

Masukan adalah energi yang dimasukan kedalam subsistem. Masukan dapat berupa perawatan yaitu energi yang dimasukan agar sistem tersebut dapat beroperasi dan masukan sinyal yaitu energi yang diproses untuk didapatkan keluaran.

6. Keluaran Sistem

Keluaran adalah hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsisten yang lain atau kepada supra sistem.

7. Pengolahan Sistem

Sistem dapat mempunyai suatu bagian pengolahan atau sistem itu sendiri sebagai pengolahnya. Pengolah akan merubah masukan menjadi keluaran.

8. Sasaran Sistem

Sistem pasti memiliki sasaran atau tujuan, jika tidak maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang

dibutuhkan sistem dan keluaran yang akan dihasilkan sistem dikatakan berhasil jika mengenai sasaran atau tujuannya.

2.2.2. Sekilas tentang Sistem Pengolahan Data

Data menurut Azhar Susanto (Sistem Informasi Manajemen, 2000, hal 37) adalah sebagai berikut :

"Data adalah fakta atau apapun yang dapat digunakan sebagai *input* untuk menghasilkan informasi"

Sistem pengolahan data adalah satu kesatuan yang saling terhubung yang terdiri dari peralatan, tenaga, prosedur dan data. Setiap sistem informasi mencakup sebuah sistem yang mengerjakan data adalah untuk mengumpulkan data, mengklasifikasi data, menyimpan data, mengkonfirmasikan data, mentransformasikan data, menyeleksi data, menyimpan data, mengkonfirmasi data, mentransformasi data, menyeleksi data dan membandingkan data.

2.2.3 Daur Hidup Perangkat Lunak (Sofware Life Cycle)

Proses sebagai *problem solving* pengembangan perangkat lunak dapat dianggap sebagai lingkaran pemecahan masalah. Untuk menyelesaikan masalah besar, dipecah menjadi kecil terus-menerus sampai paling kecil, kemudian diselesaikan (*recursive*). Model proses atau daur hidup Perangkat Lunak adalah

model proses untuk Rekayasa Perangkat Lunak (RPL) dipilih berdasarkan sifat aplikasi dan proyeknya, metode dan *tool* yang digunakan, serta kontrol *dan deliverable* yang diinginkan.

1. *Linear sequential model (classic life cycle / model cycle)*

Model yang paling luas dipakai dan tertua. Mengusulkan suatu pendekatan sekuensial dan sistematis pada pengembangan Perangkat Lunak. Meliputi aktivitas-aktivitas :

a. *System / information engineering* (rekayasa dan pemodelan sistem) adalah menyangkut pengumpulan kebutuhan (*requirement gathering*) pada level sistem dengan sejumlah kecil analisis serta top desain.
b. Analisis adalah menyangkut kebutuhan Perangkat Lunak, proses *requirement gathering* diintesifkan dan difokuskan, khususnya pada Perangkat Lunak. Untuk memahami sifat program yang dibangun, analis harus memahami domain informasi, tingkah laku, unjuk kerja, dan *interface* yang diperlukan. Kebutuhan sistem maupun Perangkat Lunak didokumentasikan dan di*review* bersama user.
c. Desain adalah menyangkut dengan desain *database*, arsitektur Perangkat Lunak, *interface*, dan algoritma prosedural. Proses desain menerjemahkan kebutuhan ke dalam representasi Perangkat Lunak sebelum dimulai *coding*.
d. *Coding* adalah menerjemahkan desain ke dalam bahasa yang dimengerti mesin.

e. *Testing,* fokus pada :
    1. Logika internal Perangkat Lunak : memastikan bahwa semua *statement* telah diuji
    2. Fungsi eksternal : mengarahkan *testing* untuk menemukan kesalahan-kesalahan dan memastikan bahwa input yang diberikan akan menghasilkan output sesuai yang diinginkan.
f. *Maintenance* adalah menyangkut perawatan sistem yang telah dibangun.

Kelemahan *Linear sequential model* :

1. Meskipun mengakomodasi iterasi, model linier melakukannya secara tidak langsung sehingga perubahan-perubahan dapat menyebabkan keraguan saat tim proyek berjalan
2. Jika user sulit menyatakan semua kebutuhannya secara explisit, model ini sulit mengakomodasi ketidakpastian itu
3. User harus bersabar karena hasil baru bisa dinikmati setelah *testing* (akhir waktu). Jika ada kesalahan besar yang tidak terdeteksi sampai program yang bekerja tersebut dikaji ulang, bisa menjadi petaka.
4. Pengembang sering melakukan penundaan yang tidak perlu. Anggota tim harus menunggu anggota tim lain selesai mengerjakan tugasnya yang mempunyai keterkaitan dan ketergantungan tinggi.

Meski memiliki kelemahan, model ini masih lebih baik daripada pendekatan yang sembarangan.

Contoh : model Waterfall terdiri dari fase Investigasi, Analisis, Desain, Implementasi (*coding*), *Testing, Maintenance*.

2. *Prototyping Model*

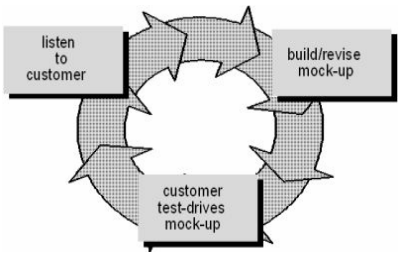
Berfungsi sebagai mekanisme pendefinisian kebutuhan. Awalnya, *Developer* menggali semua kebutuhan user secara cepat kemudian membangun *prototipe* yang sesuai dengan yang diinginkan dengan cepat pula dan ditunjukkan ke user, baru dibuat Perangkat Lunak yang sesungguhnya berdasarkan komentar user terhadap *prototipe*.

Kelebihan model adalah user dapat langsung melihat wujud Perangkat Lunak yang akan dibangun meskipun sederhana dan dari sana dapat digali kebutuhan yang lebih dalam sebagai bahan penyusunan Perangkat Lunak berikutnya.

Masalah yang muncul adalah :

- User merasa *prototipe* merupakan Perangkat Lunak yang sesungguhnya, padahal ketika membuat *prototipe* belum disertakan kualitas Perangkat Lunak (PL) secara keseluruhan atau kemampuan pemeliharaan untuk jangka panjang.
- *Developer* sering membuat kompromi-kompromi implementasi untuk membuat *prototipe* bekerja dengan cepat sehingga akan ditemui ketidakcocokan pada *prototipe* ketika *prototipe* dibangun dengan bahasa yang sederhana.

- Program dibuat ulang atau *prototipe* selalu baru



Gambar 2.12. *Prototyping Model*

3. *Rapid Application Development (RAD) Model*

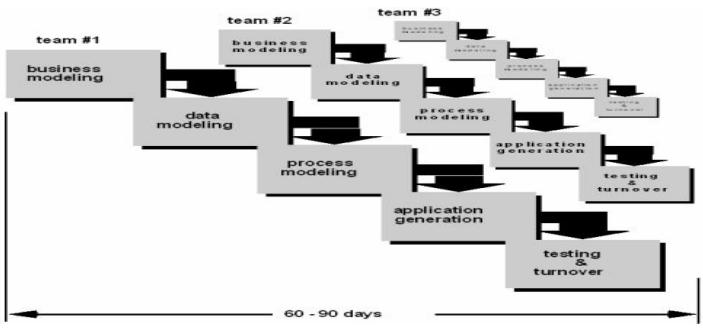
Model proses perkembangan Perangkat Lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Menekankan perkembangan komponen program yang bisa dipakai lagi sehingga mendasari konsep *Object-Oriented*.

Fase-Fase pendekatan RAD adalah :

a. *Bussines modeling* adalah fase analisis dan pemodelan aturan atau sistem yang sedang berjalan atau yan akan diterapkan.
b. *Data modeling* adalah kegiatan pemodelan data yang terlibat dalam tiap proses.
c. *Proses modeling* adalah kegiatan pemodelan proses yang ada dalam sistem tersebut.
d. *Application generation* adalah RAD mengasumsikan pemakaian teknik 4G (generasi keempat). Selain menciptakan Perangkat Lunak dengan

bahasa pemrograman generasi ketiga yang konvensional, RAD lebih banyak memproses kerja untuk memakai lagi komponen program atau menciptakan komponen yang bias dipakai lagi.

e. *Testing and Turn Over* adalah menekankan pada *reusability*, banyak komponen program yang telah diuji sehingga mengurangi keseluruhan waktu pengujian. Tapi komponen baru harus diuji dan semua *interface* harus dilatih secara penuh.



Gambar 2.13. RAD Model

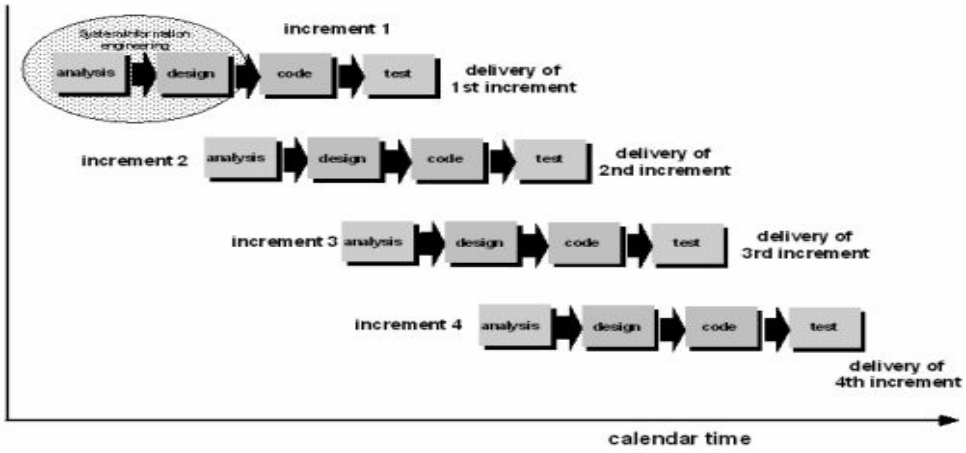
4. *The Incremental Model*

Mengadopsi model sekuensial linier dan model *prototipe*. Fungsi dasar sama, tapi ada tambahan asesoris, Fungsi tambahan ditambahkan terus untuk membuat sistem menjadi lebih baik. Pada *increment* pertama Perangkat Lunak yang jadi, mengakomodasi kebutuhan inti. Baru pada tahap berikutnya ditambahkan kemampuan baru.

Contoh : pengembangan *microsoft word*.

a. Increment 1 adalah hanya memberi fungsi inti yaitu hanya bisa mengetik saja
b. Increment 2 adalah bisa *word art*, *spelling*, dan lain-lain.

Kelebihan model ini cocok untuk produksi masal.



Gambar 2.14. *Incremental Model*

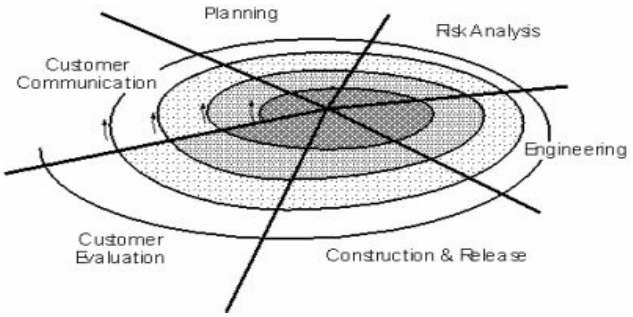
5. *An Evolutionary (Spiral) Model*

Penomoran dimulai dari arsiran paling dalam. Proses yang terjadi :

a. Proyek pengembangan konsep
b. Proyek pengembangan produk baru
c. Proyek perbaikan produk
d. Proyek pemeliharaan produk

Semua proyek tersebut dibangun dengan mengikuti model spiral melalui tahap :

a. *Customer communication* adalah tahap melakukan Tanya jawab atau wawancara kepada konsumen.
b. *Planning* adalah tahap pembuatan rencana dari data yan diperoleh hasil wawancara.
c. *Risk analysis* adalah kegiatan analisis terhadap resiko yang mungkin terjadi dalam proyek
d. *Engineering* adalah melakukan perekayasaan system yang akan dibangun.
e. *Construction and release* adalah pembangunan sistem menjadi sebuah sistem yang utuh.
f. *Customer evaluation* adalah pengujian sistem yang dilakukan oleh konsumen



Gambar 2.15. *Evolutionary (Spiral) Model*

6. *Fourth Generation (4GT) Techniques*

Menggunakan alat Perangkat Lunak untuk meng*generate source code* untuk sistem Perangkat Lunak dari representasi perincian level tinggi.

2.2.4 Konsep Dasar Informasi

Keberadaan informasi dalam suatu instansi atau organisasi sangat penting dalam setiap rencana yang akan disusun. Informasi berasal dari data yang diolah. Menurut Jogiyanto HM (Analisis dan Desain Sistem Informasi, 2000, hal 8) adalah :

> "Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya".

2.2.4.1. Kualitas Informasi

Kualitas dari informasi tergantung dari tiga hal berikut :

a. Akurat adalah Informasi harus bebas dari kesalahan – kesalahan dan tidak menyesatkan.

b. Tepat Pada Waktunya yaitu informasi yang diperoleh oleh pihak yang berkepentingan tidak boleh terlambat.

c. Relevan adalah informasi harus memiliki manfaat untuk pemakai.

2.2.4.2. Hubungan Data dengan Informasi

Dalam menghasilkan informasi kita terlebih dahulu harus tahu bagaimana mengolah suatu data menjadi informasi setelah itu kita harus tahu informasi yang

diperlukan, karena dalam pengembangan system informasi banyak orang terjebak pada situasi dimana mereka mengumpulkan data terlebih dahulu tanpa tahu informasi yang diperlukan.

2.2.5 Sistem Informasi

Seperti yang telah dibahas sebelumnya, Informasi merupakan hal yang sangat penting bagi suatu instansi atau organisasi dalam proses penyusunan rencana ataupun pengambilan keputusan yang cerdik, Informasi tesebut diperoleh dari suatu sistem informasi. Menurut Robert A Leitch dan K.Roscoe Davis dalam Jogiyanto HM (Analisis dan Desain Sistem Informasi, 2001, hal 11) sebagai berikut :

> "Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan".

2.2.5.1. Kegiatan Sistem Informasi

a. *Input* adalah menggambarkan suatu kegiatan untuk menyediakan data untuk diproses.

b. Proses adalah menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi yang bernilai tambah.

c. *Output* adalah suatu kegiatan untuk menghasilkan laporan dari proses diatas tersebut.

d. Penyimpanan adalah suatu kegiatan untuk memelihara dan menyimpan data.

e. Control adalah suatu aktifitas untuk menjamin bahwa sistem informasi tersebut berjalan sesuai dengan yang diharapkan.

2.2.6 Analisis Sistem

Tahap analisis sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap perancangan sistem. Menurut HM Jogiyanto HM (Analisis dan Desain Sistem Informasi, 2001 hal 21) analisis sistem adalah

> "Penguaraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevalusi permasalahan-permasalahan , kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya" .

Langkah-langkah dasar dalam menganalisis sistem adalah sebagai berikut :

a. *Identify* adalah mengindetifikasi masalah.

b. *Understand* adalah memahami kerja dari sistem yang ada.

c. *Analyze* yaitu menganalisis sistem.

d. *Report* adalah membuat laporan hasil analisis.

2.2.7 Perancangan Sistem

Perancangan sistem dilakukan setelah analisis sistem dilakukan, analisis sistem telah menggambarkan dengan jelas apa yang harus dikerjakan dan bagi analisis untuk memikirkan bagaimana mambentuk suatu sistem tertentu.

Tujuan dari perancangan sistem adalah sebagai berikut :

1. untuk memenuhi kebutuhan pada pemakai sistem.
2. untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemogram komputer dan ahli-ahli teknik lainnya yang terlibat.

2.2.7.1. Diagram Arus Data (DFD)

Diagram arus data atau *Data Flow diagram* (DFD) adalah diagram yang menggunakan notasi - notasi untuk menggambarkan arus dari data sistem, (Jogiyanto HM, Analisis dan Sistem, 2001, hal 700).

Berdasarkan pengertian diatas , Diagram konteks merupakan arus data yang berfungsi untuk menggambarkan keterkaitan aliran-aliran data antara sistem dengan bagan-bagan luar (entitas luar). Entitas luar ini merupakan sumber arus data atau tujuan data yang berhubungan dengan sistem informasi.

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan. DFD merupakan alat yang sangat popular pada saat ini, karena dapatmenggambarkan arus data didalam sistem dengan terstruktur dan jelas.

Simbol yang digunakan dalam DFD adalah (Jogiyanto HM, Analisis dan Desain, 2001, hal 700) :

1. *External Entity*

Entitas luar merupakan kesatuan dilingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada dilingkungan luarnya yang akan memberikan *input* atau menerima *Output* dari sistem. Entitas luar dapat disimbolkan dengan notasi kotak atau suatu kotak dengan sisi kiridan atasnya berbentuk garis tebal.



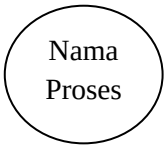
Gambar 2.1 Entitas

2. *Data Flow* (Arus Data), di DFD diberikan simbol panah. Arus data ini mengalir diantara proses (*process*), simpanan data (*data store*) dan entitas luar (*external Entity*). Arus data ini menunjukkan arus data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.



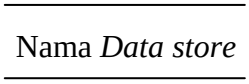
Gambar 2.2 notasi arus data di DFD

3. Proses (*Process*) , adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang keluar dari proses. Suatu proes dapat ditunjukkan dengan simbol lingkaran.



Gambar 2.3 Notasi proses di DFD

4. *Data store* (Simpana Data), Merupakan simpanan dari data yang dapat berupa suatu *File* atau data base disistem komputer, arsip atau catatan manual, kotak tempat data dimeja seseorang, table acuan manual dan agenda atau buku. Simpanan data di DFD dapat disimbolkan dengan sepasang garis horizontal parallel yang tertutup disalah satu ujungnya.



Gambar 2.4 Simbol dari simpanan data di DFD

2.2.7.2. Pengertian Kamus Data

Kamus data atau data dictionary atau disebut juga dengan istilah *Sistem Data Ditionary* adalah catalog kata fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. (Jogiyanto HM, Analisis dan Desain , 2001 hal 725).

Fungsi kamus data , dapat mengetahui data yang mengalir di sistem dengan lengkap. Kamus data dapat digunakan dengan dua tahap yaitu tahap analisis danperencanaan sistem. Pada tahap menganalisis suatu sistem, kmaus data dapat digunakan sebagai alat komunikasi antara analissis den pemakai sistem, mengenai data yang masuk kedalam sistem dan informasi yang dibutuhkan dalam sistem. Sedangkan pada tahap perencanaan sistem, kamus data digunakan untuk merancang *input*, merancang laporan-laporan dan *database*.

2.2.7.3 Bagan Alir Sistem

Bagan Alir Sistem (System Flowchart) merupakan bagan yang menunjukkan srus pekerjaan secara keseluruhan dari sistem. Bagan ini mejelaskan urut –urutan dari prosedur – prosedur yang ada di dalam sistem dan menunjukkan apa yang dikerjakan sistem.

2.2.7.4 Bagan Alir Dokumen

Bagan Alir Dokumen (*Document Flowchart*) atau Bagan Alir Formulir (Form *Flowchart*) atau Paperwork *Flowchart* atau *Flowma*p merupakan bagan alir yang

menuujukkan arus dari laporan dan formulir termasuk tembusan-tembusan. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan didalam bagan alir sistem.

2.2.8 Pengertian *Database*

Database adalah kumpulan *File-File* yang saling berelasi, relasi tersebut biasa ditunjukkan dengan kunci tiap *File* yang ada. (Harianto Kristanto, Konsep dan Perancangan *Database*, 1994, hal 1).

Ada bebrapa hal istilah yang terdapat atau merupakan bagian dari *database* adalah sebagai berikut :

1. *Entity*, adalah orang, tempat, kejadian, atau konsep yang informasinya direkam.
2. *Atribute*, disebut sebagai elemen, data *field*, data item untuk mewakili suatu *Entity*.
3. *Data Value* (Nilai atau Isi data), merupakan data aktual atau informasi yang disimpan pada tiap data elemen atau *attribute*.
4. *Record*, merupakan kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu *Entity* secara lengkap.

5. *File*, merupakan kumpulan *Record-Record* yang mempunyai panjang elemen yang sama, *attribute* yang sama, namun berbeda – beda data dalam *value*nya.

Merancang *database* merupakan suatu hal yang sangat penting, kesulitan utama dalam merancang *database* adalah bagaimana merancang sehingga *database* dapat memuaskan keperluan saat ini dan masa mendatang. Terdapat dua buah teknik untuk merancang *database*, yaitu dengan teknik normalisasi dan teknik *Entity relationship*.

2.2.8.1 *Atribute* Kunci atau *field*

Terdapat beberapa *attribute* kunci pada normalisasi (Harianto Kristanto, Konsep dan Perancangan *Database*, 1994,hal 19), yaitu :

1. *Candidate key* (Kunci Kandidat atau Kunci Calon), adalah satu *attribute* atau satu set minimal *attribute* yang mengidentifikasikan secara unik suatu kejadian spesifik dari *Entity*.

2. *Primary key* (kunci Primer), adalah satu *attribute* atau set minimal *attribute* yang tidak hanya mengidentifikasikan secara unik suatu kejadian spesifik, tatapi juga dapat mewakili setiap kejadian dari suatu *Entity*.

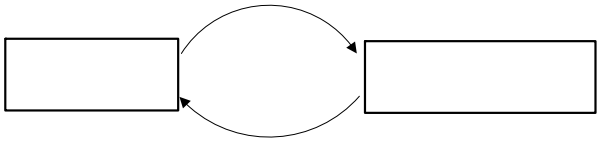
3. *Alternate Key* (Kunci Alternatif), adalah kunci kandidat yabg tidak dapat dipakai sebagai primary key.

4. *Foreign Key* (Kunci tamu), adalah satu *attribute* yang melengkapi satu *relationship* (hubungan) yang menunjukkan ke induknya.

2.2.8.2 Teknik *Entity Relationship*

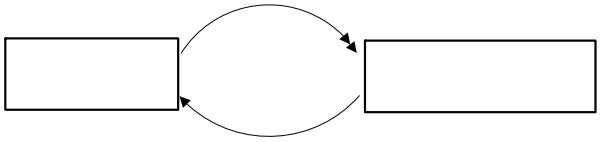
Database adalah kumpulan *file* yang saling berkaitan. Pada model data relational hubungan antar *file* direlasikan dengan kunci relasi, yang merupakan kunci utama dari masing-masing *File*. Relasi antara dua *file* atau table dapat dikategorikan menjadi tiga macam (Harianto kristanto, Konsep dan Perancangan *Database*, 1994, hal 35), yaitu :

1. *One to one relationship* 2 *File*, merupakan hubungan antara *Entity* satu dengan *Entity* yang kedua dengan satu berbanding satu. Misal pada mahasiswa mempunyai atau mengambil satu mata kuliah.



Gambar 2.5 Relasi one to one

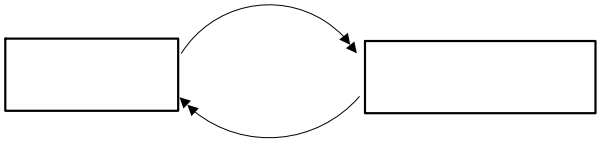
2. *One to many relationship* 2 *File*, merupakan hubungan antara *Entity* satu dengan *Entity* kedua adalah satu berbanding banyak atau dapat pula dibalik, banyak lawan satu. Missal satu dosen mengajar banyak mahasiswa.



Gambar 2.6 Relasi one to many

3. *Many to many relationship* 2 *File*, adalah *Entity* pertama dengan *Entity* yang kedua dengan banyak berbanding banyak. Misal pada sistem pengajaran diperguruan tinggi dimana satu dosen mengajar banyak mahasiswa dan mahasiswa diajar oleh banyak dosen pula.

Dosen



Gambar 2.7 Relasi many to many

2.2.9 Sekilas tentang SQL

SQL merupakan singkatan dari *Structured Query Language*. SQL sering dibaca sebagai "*Sequel*" adalah bahasa terstruktur yang digunakan untuk query, mengupdate dan mengolah relasi antar database. User dapat menggunakan bahasa ini untuk memuat, mengurutkan, dan menyaring suatu data sehingga dihasilkan suatu data yang spesifik dari suatu database.

Dosen

2.2.10 Sekilas Tentang Tata Naskah

Sistem digitalisasi saat ini sudah banyak dilakukan oleh para pengembang IT (Information Technology) hampir di seluruh belahan dunia.Begitu juga pada bagian Tata Naskah, digitalisasi kumpulan data pegawai yang ada sudah dilakukan secara berangsur-angsur sampai semua dokumen yang ada menjadi koleksi dokumen digital. Untuk membangun sistem Tata Naskah kepegawaian diperlukan data Pegawai Negeri Sipil (PNS) yang akurat dan lengkap yang dapat digunakan dalam setiap proses yang ingin dilakukan.

Dalam teknologi informasi modern, kebutuhan manajeman tata naskah dirasakan makin perlu, karena sebagai dokumen yang berfungsi menunjang tugas pokok instansi atau organisasi yang menggunakannya, dan Manfaat pencatatan data Kepegawaian adalah sebagai sarana untuk menyimpan data kepegawaian secara sistimatis, sehingga memudahkan penemuan kembali jika diperlukan. Data kepegawaian diolah menjadi informasi kepegawaian, sangat diperlukan untuk proses mutasi maupun pensiun. Arsip di bagian Tata Naskah Badan Kepegawaian Negara bukan hanya dikaitkan dengan data masa lampau, namun juga bermanfaat bagi perencanaan masa depan sebagai bahan dasar untuk perencanaan pembangunan selanjutnya.

2.2.11 Jaringan Komputer

Jaringan komputer adalah sekumpulan peralatan komputer yang dihubungkan agar dapat saling berkomunikasi dengan tujuan membagi sumber daya (seperti file dan printer).

2.2.11.1. Jenis-Jenis Jaringan

Jaringan komputer dapat dibagi dalam beberapa jenis :

A. Jaringan berdasarkan koneksi (keterhubungan) :

1. Broadcast Links, dimana setiap node menerima setiap transmisi data yang dikirim oleh sebuah node.
2. Point-to-point links, yaitu transmisi data yang mengalir di *network* dalam satu arah dan setiap station di-*poll* satu persatu.

B. Jaringan berdasarkan skala :

1. LAN (Local Area Network)

   LAN digunakan untuk menghubungkan komputer yang berada didalam suatu area kecil, misalnya dalam suatu gedung perkantoran atau kampus. Jarak antar komputer yang dihubungkannya bias mencapai 5 sampai 10 km dan berkerja pada kecepatan 10 Mbps sampai 100 Mbps.

2. MAN (MetropolitanArea Network)

MAN merupakan suatu jaringan yang cakupannya meliputi suatu kota. MAN menghubungkan LAN-LAN yang lokasinya berjauhan. Jangkauan MAN biasanya mencapai 10 km sampai beberapa ratus km dan berkerja pada kecepatan 1,5 Mbps sampai 150 Mbps.

3. WAN (Wide Area Network)

WAN dirancang untuk menghubungkan komputer-komputer yang terletak pada suatu cakupan geograpis yang luas, seperti hubungan dari satu kota ke kota lainnya dalam suatu Negara. Cakupan WAN bias meliputi 100 km sampai 1000 km dengan kecepatan antara 1,5 Mbps sampai 2,4 Gps.

4. GAN (Global Area Network)

GAN merupakan suatu jaringan yang menghubungkan Negara-negara diseluruh dunia. Kecepatan GAN berpariasi mulai dari 1,5 Mbps sampai 100Gps yang cakupannya meliputi ribuan km contohnya adalah internet.

C. Berdasarkan Topologi

1. Topologi Bus

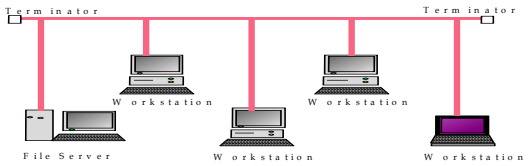
Topologi ini memiliki karakteristik sebagai berikut**:**

a. Merupakan satu kabel yang kedua ujung nya ditutup, dimana sepanjang kabel
   terdapat node-node.
b. Paling umum karena sederhana dalam instalasi

c. Signal melewati kabel dalam dua arah dan mungkin terjadi *collision*

d. Kekurangannya adalah kabel putus, jika salah satu segmen kabel putus, maka seluruh

jaringan akan terhenti.



Gambar 2.8. Topologi Bus

3. Topologi Star

Topologi ini mempunyai karakteristik sebagai berikut:

a. Setiap node berkomunikasi langsung dengan *central node*, traffic data mengalir dari

node ke *central node* dan kembali lagi.

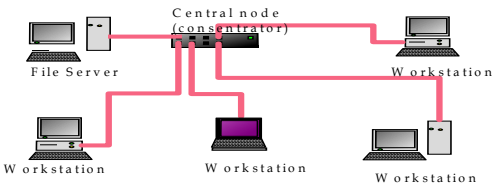
b. Mudah dikembangkan, karena setiap node hanya memiliki kabel yang langsung

terhubung ke *central node.*

c. Keunggulan : jika satu kabel node terputus yang lainnya tidak terganggu.

d. Dapat digunakan kabel yang "*lower grade*" karena hanya menghandel satu traffic

node, biasanya digunakan kabel UTP.



Gambar 2.9. Topologi Star

4. Topologi Ring

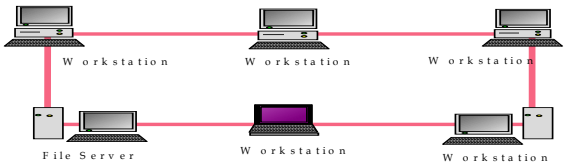
Topologi ini mempuyai karakteristik sebagai berikut:

a. lingkaran tertutup yang berisi node-node
b. sederhana dalam layout
c. signal mengalir dalam satu arah, sehingga dapat menghindarkan terjadinya *collision*

   (dua paket data bercampur), sehingga memungkinkan pergerakan data yang cepat

   dan *collision detection* yang lebih sederhana
d. problem: sama dengan topologi bus
e. biasanya topologi ring tidak dibuat secara fisik melainkan direalisasikan dengan

   sebuah *consentrator* dan kelihatan seperti topologi star



Gambar 2.10. Topologi Ring

D. Jaringan berdasarkan protocol :

1. Ethernet

a. Dikembangkan oleh Xerox Corp. pada tahun 70-an, dan menjadi populer pada

   tahun 80-an karena diterima sebagai standard IEEE 802.3
b. Ethernet bekerja berdasarkan *broadcast network*, dimana setiap node menerima

setiap transmisi data yang dikirim oleh sebuah node.

c. Menggunakan metoda CSMA/CD (*carrier sense multiple access/collision detection*)

baseband

d. Cara kerja ethernet secara ringkas adalah sebagai berikut: sebelum mengirimkan

paket data, setiap node melihat apakah network juga sedang mengirimkan paket

data. Jika network *busy*, node itu mengunggu sampai tidak ada sinyal lagi yang

dikirim oleh network. Jika network sepi, barulah itu node mengirimkan paketnya.

Jika pada saat yang sama ada dua node yang mengirimkan data, maka terjadi

collision. Jika terjadi *collision*, kedua node mengirimkan sinyal *jam* ke network dan

semua node berhenti mengirimkan paket data dan kembali menunggu. Kemudian

secara random, node-node itu kembali menunggu atau mengirimkan data. Paket

yang mengalami *collision* akan dikirimkan kembali saat ada kesempatan.

e. Kecepatannya 10 mbps, dan menurun dengan semakin banyaknya node yang terpasang

2. Token ring

   a. Berdasarkan standard IEEE 802.5 yang dikembangkan oleh IBM.

   b. Untuk menghindari *colision* tidak menggunakan *collision detection* melainkan *token*

      passing scheme

   c. *Token passing scheme* dapat dijelaskan secara sederhana sebagai berikut: Sebuah

      token yang bebas mengalir pada setiap node melalui network. Saat sebuah node

      ingin mengirikam paket, node itu meraih dan melekatkan frame atau paket-nya ke

      token. Sekarang token tidak dapat digunakan lagi oleh node yang lain sampai data

      mencapai tujuannya. Jika telah sampai token dilepaskan lagi oleh originating

      station. Token mengalir di network dalam satu arah dan setiap station di-poll satu

      persatu.

   d. Kecepatannya 4 mbps dan 16 mbps

E. Jaringan berdasarkan arsitektur :

   1. peer to peer

Model hubungan *peer to peer* memungkinkan user membagi sumber dayanya yang ada di komputernya baik itu berupa *file*, layanan *printer*, dan lain-lain serta mengakses sumber daya yang terdapat pada komputer lainnya. Model ini tidak mempunyai sebuah *file server* atau sumber daya yang terpusat.

2. Client-Server

Model hubungan *Client/server* memungkinkan jaringan untuk mensentralisasi fungsi dan aplikasi kepada satu atau dua *dedicated file server*. Sebuah *file server* menjadi jantung dari keseluruhan sistem, memungkinkan untuk mengakses sumber daya, dan menyediakan keamanan.

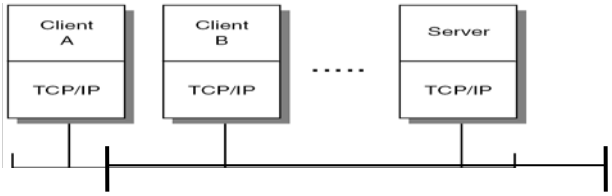
2.2.11.2 Jaringan Client-Server

Sistem Client-Server biasanya berjalan pada setidaknya dua sistem komputer yang berbeda. Satu komputer bertindak sebagai client dan yang lainnya sebagai server. Sistem client-server ditujukan untuk mengatasi kelemahan yang terdapat pada sistem tersentralisasi.

Pada sistem client-server, client akan mengirimkan message (perintah), selanjutnya server yang akan menerima dan mengolah message tersebut, kemudian hasilnya akan dikirim ke client, dengan begitu proses transfer data jauh lebih efisien, dari proses pengiriman data tersebut sehingga sebuah *server* menjadi

jantung dari keseluruhan sistem, memungkinkan untuk mengakses sumber daya, dan menyediakan keamanan

Adanya pemilahan fungsi client dan server, disamping meningkatkan kompleksistas tersendiri dalam pembangunan aplikasi (sistem informasi) secara keseluruhan juga menimbulkan kelemahan lain yaitu aktivitas pemasangan aplikasi (deployment) yang tidak praktis jika terjadi perubahan atau perbaikan aplikasi basis data, maka kita harus mengulang pekerjaan instalasi disemua client yang digunakan.



Gambar 2.11. Client-Server

2.2.11.3 Tujuan Jaringan Client-Server

Ada 3 tujuan jaringan Client Server

1. Resource sharing , bertujuan agar seluruh program, peralatan, khususnya data dapat digunakan oleh stiap orang yang ada pada jaringan tanpa terpengaruh oleh lokasi resource dan pemakai. Jadi source sharing adalah usaha untuk menghilangkan kendala jarak.
2. Skalabilitas , yaitu kemampuan untuk eningkatkatkan kenerja sistem secara berangsur angsur sesuai dengan beban pekerjaan dengan hanya menambahkan sejumlah prosesor. Pada komputer mainframe yang tersentralisasi, jika sistem sudah jenuh, maka komputer harus diganti

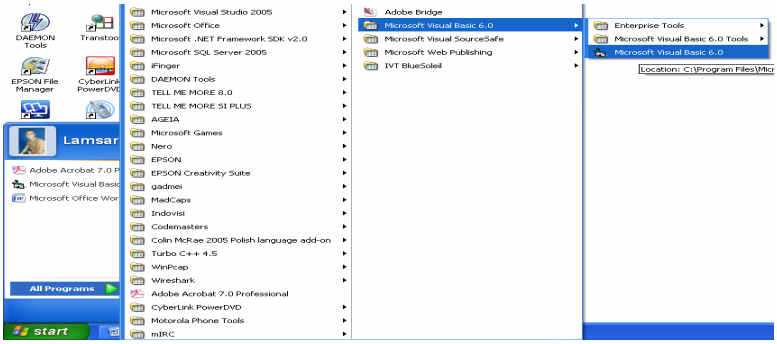
dengan kemampuan lebih besar. Hal ini membutuhkan biaya yang sangat besar dan dapat menyebabkan gamngguan terhadap kontinuitas kerja para pemakai.

3. Media komunikasi, sebuah jaringan komputer mampu bertindak sebagai media komunikasi yang baik bagi para pegawai yang terpisah jauh, dengan menggunakan jaringan, dua orang atau lebih yang tinggal berjauhan akan lebih mudah bekerja sama dalam mengerjakan laporan.

2.2.12 Sekilas Tentang Microsoft Visual Basic 6.0

Microsoft Visual Basic 6.0 merupakan program aplikasi database berbasis bahasa pemograman Basic dari Microsoft. Selain itu Visual basic juga memberikan fasilitas pembuatan aplikasi visual, seperti halnya Borlan Delphi. Untuk mulai menjalankan program Visual basic 6.0, ikuti langkah-langkah berikut ini :

a. Nyalakan komputer anda, tunggu sampai komputer menampilkan area kerja (*desktop*) *Windows*
b. Klik tombol Start yang ada di taskbar
c. Pilih menu Program
d. Pilih dan klik Microsoft Visual Basic 6.0. lihat pada gambar kemudian tunggu sampai jendela aplikasi Microsoft Visual Basic 6.0 ditampilkan



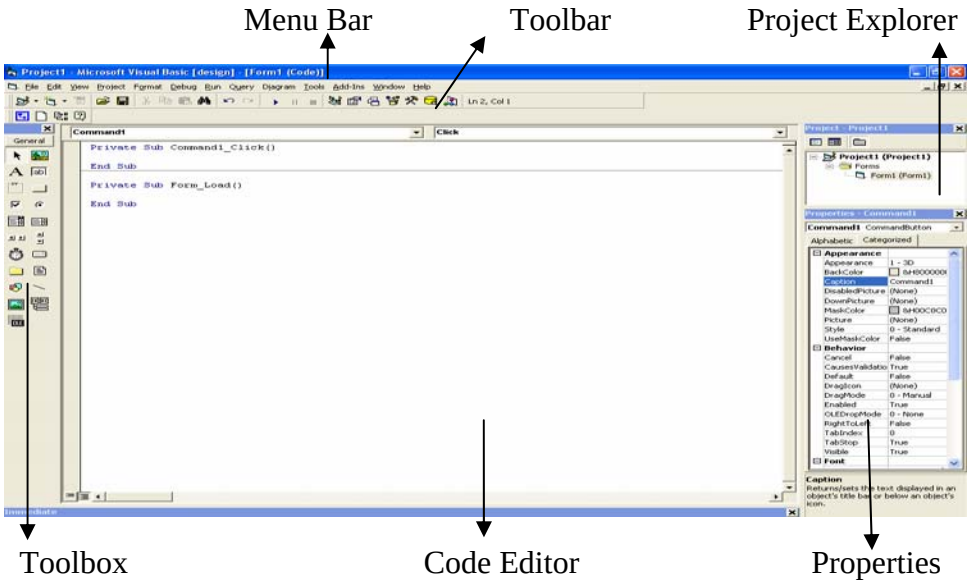
Gambar 2.16 Menjalankan Microsoft Visual Basic

e. Pada *window* ***New Project***, misalnya pilih ***Standart Exe***



Gambar 2.17 New Project Visual Basic

f. Kemudian akan tampil IDE (Integrated Development Environment) dari Microsoft Visual Studio 6.0



Gambar 2.18 IDE dari Microsoft Visual Basic 6.0

Integrated Development Environment atau IDE adalah sebuah lingkungan yang berisi tool-tool yang diperlukan untuk mendesain, menjalankan dan mengetes sebuah aplikasi, disajikan dan terhubung dengan baik sehingga memudahkan pengembangan program. Pada Visual Basic 6.0 terdiri dari Main WindowMenu Bar, Toolbar, Object Tree View, Properties, Code Editor dan Komponen. Integrasi ini memberikan kita kemudahan dalam mengembangkan aplikasi yang kompleks.

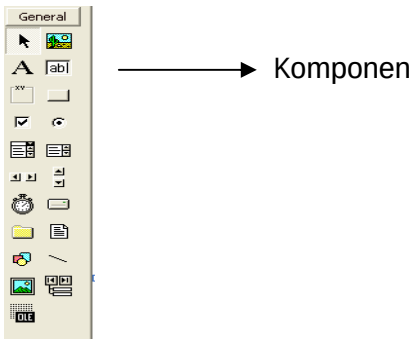
Memakai Komponen

1. Menempatkan Komponen pada Form

Untuk membuat antarmuka aplikasi, kita harus menempatkan komponen-komponen pada sebuah form, mengubah propertinya. Komponen-komponen tersedia di Toolbox Pallete.

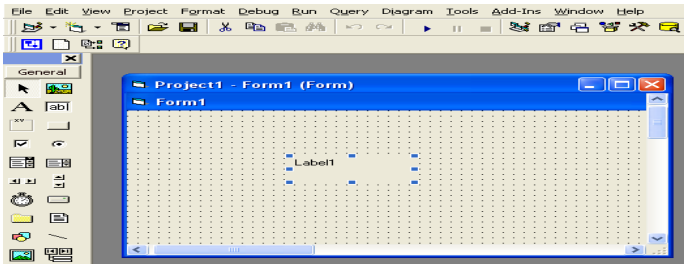
Langkah-langkah untuk menempatkan komponen pada sebuah form :

a. Klik sebuah komponen pada Toolbox Pallete, misalnya komponen **Label**

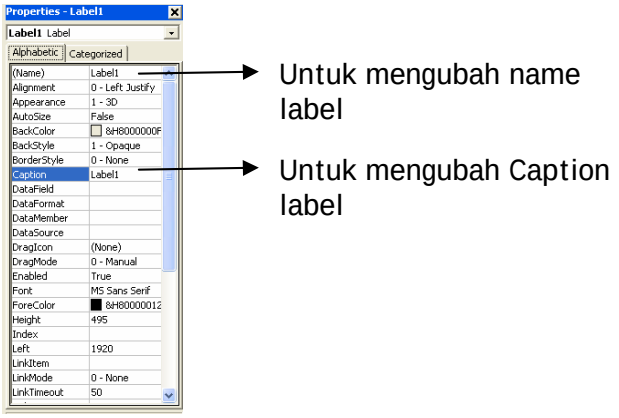


Gambar 2.19 Komponen Toolbox pada Visual Basic

b. Klik pada form dimana kita ingin menempatkan komponen tersebut



Gambar 2.20 memakai komponen Label Visual Basic



Gambar 2.21 Properties Visual Basic

2.2.13 Sekilas Tentang SQL Server 2000

SQL Server 2000 adalah sebuah database relational yang dirancang untuk mendukung aplikasi dengan arsitekture client/server dengan bantuan akses menggunakan SQL. Database terdapat pada komputer pusat yang disebut server, dan informasi digunakan bersama-sama oleh beberapa user yang menjalankan aplikasi didalam komputer lokalnya yang disebut client. Arsitekture semasam ini meemberikan integritas data yang tinggi, karena semua user bekerja dengan informasi yang sama. Melalui aturan bisnis, kendali diterapkan kepada semua user mengenai informasi yang detambahkan kedalam database. SQL Server 2000 bisa mengandung beberapa database yang digunakan oleh beberapa user. Sebuah

instansi bisa memiliki satu database yang hanya digunakan khusus beberapa user tertentu dalam masing-masing departemen. Apabila database dibuat, user bias memiliki ijin akses yang telah diberikan kepadanya.