BAB II

LANDASAN TEORI

Dalam bab ini akan membahas tentang teori – teori yang berhubungan dengan sistem informasi yang akan dibuat.

2.1 Konsep Dasar Sistem

Agar lebih mudah memahami apa dan bagaimana sistem itu akan digunakan dua pendekatan, yaitu pendekatan prosedur dan pendekatan komponen. Pemahaman sistem dengan pendekatan prosedur, yaitu suatu urutan kegiatan yang saling berhubungan, berkumpul bersama – sama untuk mencapai tujuan tertentu. Pemahaman sistem dengan pendekatan komponen / elemen, yaitu kumpulan komponen yang saling berkaitan dan bekerja sama untuk mencapai suatu tujuan tertentu.

2.1.1 Pengertian Sistem

Pendekatan sistem yang lebih menekankan pada elemen – elemen atau komponennya, mendefenisikan sistem sebagai sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Dengan demikian di dalam suatu sistem, komponen – komponen ini tidak dapat berdiri sendiri – sendiri,

tetapi sebaliknya, saling berhubungan hingga membentuk satu kesatuan sehingga tujuan sistem itu dapat tercapai.

2.1.2 Karakter Sistem

Suatu sistem mempunyai karakteristik atau sifat – sifat tertentu, yaitu mempunyai komponen – komponen, batas sistem, lingkungan luar sistem, masukan, keluaran, pengolah dan sasaran atau tujuan.

1. Komponen sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk suatu kesatuan. Komponen – komponen sistem atau elemen – elemen sistem dapat berupa suatu subsistem atau bagia – bagian dari sistem.

2. Batasan sistem

   Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem

   Lingkungan luar dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan. Lingkungan luar yang menguntungkan

merupakan energi dari sistem dan dengan demikian harus dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, jika tidak maka akan mengganggu kelangsungan hidup sistem.

4. Penghubung sistem

   Penghubung sistem merupakan media yang menghubungkan antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini kemungkinan sumber – sumber daya mengalir dari satu subsistem yang lainnya. Keluaran dari satu subsistem akan menjadi masukan untuk subsistem lainnya melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan sistem

   Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal maintanance input adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan. Sinyal input adalah energi yang diproses untuk mendapatkan keluaran dari sistem.

6. Keluaran sistem

   Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.

7. Pengolahan sistem

   Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran sistem

   Suatu sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

2.1.3 Klasifikasi Sistem

Sistem dapat diklasifikasikan kedalam beberapa sudut pandang, sebagai berikut :

a. Sistem diklasifikasikan sebagai sistem abstrak dan sistem fisik.

   Sistem abstrak adalah sistem yang berupa pemikiran atau ide – ide yang tidak tampak secara fisik. Misalnya sistem teologi. Sistem fisik merupakan sistem yang ada secara fisik. Misalnya sistem komputer, sistem operasi, sistem penjualan, dan lain sebagainya.

b. Sistem diklasifikasikan sebagai sistem alamiah dan sistem buatan manusia.

   Sistem alamiah adalah sistem yang terjadi karena proses alam tidak dibuat oleh manusia. Misalnya sistem perputaran bumi, sistem pergantian siang dan malam, dan lain sebagainya. Sistem buatan manusia adalah sistem yang dirancang atau dibuat oleh manusia sendiri. Sistem buatan manusia yang melibatkan interaksi manusia dengan mesin disebut dengan *human – machine* atau biasa disebut *machine system.* Contohnya adalah sistem informasi.

c. Sistem diklasifikasikan sebagai sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*).

   Sistem tertentu beropasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi diantara bagian – bagiannya dapat dideteksi dengan pasti, sehingga keluaran dari sistem dapat diramalkan. Contohnya adalah sistem komputer. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas. Sistem sosial, sistem politik, sistem demokrasi, dan lain sebagainya merupakan contoh dari sistem tak tentu.

d. Sistem diklasifikasikan sebagai sistem tertutup dan sistem terbuka

   Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lain.

2.2 Konsep Dasar Data dan Informasi

2.2.1 Pengertian data

Data adalah deskripsi dari sesuatu dan kejadian yang kita hadapi atau data juga dapat didefenisikan sebagai suatu kenyataan yang menggambarkan suatu kejadian – kejadan.

2.2.2 Pengertian informasi

Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi.

Sumber informasi adalah data. Informasi diperoleh setelah data – data mentah diproses atau diolah. Data belum memiliki nilai sedangkan informasi sudah memiliki nilai. Nilai sebuah informasi ditentukan dari dua hal yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkan informasi tersebut.

2.2.3 Kualitas informasi

Kualitas informasi (*quality of information*) sangat dipengaruhi oleh beberapa hal sebagai berikut :

a. Relevan (r*elevancy*)

   Seberapa jauh tingkat relevansi informasi tersebut terhadap kenyataan kejadian masa lalu, kejadian hari ini dan kejadian yang akan datang. Informasi yang berkualitas akan mampu menunjukkan benang merah relevansi kejadian masa lalu, hari ini, dan masa depan sebagai sebuah bentuk aktivitas yang kongkrit dan mampu dilaksanakan, dan dibuktikan oleh siapa saja.

b. Akurat (*accuracy*)

   Suatu informasi dikatakan berkualitas jika seluruh kebutuhan informasi tersebut telah tersampaikan (*completeness*), seluruh pesan telah benar atau sesuai (*correctness*), serta pesan yang disampaikan sudah lengkap.

c. Tepat Waktu (t*imeliness*)

   Berbagai proses dapat diselesaikan dengan tepat waktu, laporan – laporan yang dibutuhkan dapat disampaikan tepat waktu.

d. Ekonomis (*economy*)

   Informasi yang dihasilkan mempunyai daya jual yang tinggi serta biaya operasional untuk menghasilkan informasi tersebut minimal, informasi tersebut juga mampu memberikan dampak yang luas terhadap laju pertumbuhan ekonomi dan teknologi informasi.

e. Efisien (*efficiency*)

   Informasi yang berkualitas memiliki sintaks ataupun kalimat yang sederhana

f. Dapat dipercaya (*reliability*)

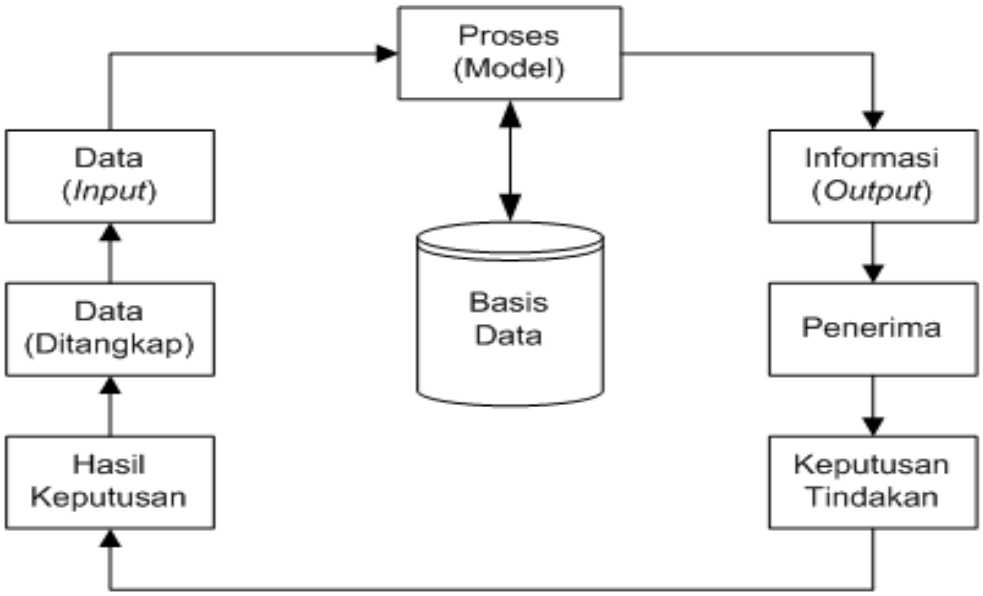
Informasi tersebut berasal dari sumber yang dapat dipercaya. Sumber tersebut juga telah teruji tingkat kejujurannya.

2.2.4 Nilai informasi

Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya dan sebagian besar informasi tidak dapat ditaksir keuntungannya dengan satuan nilai uang, tetapi dapat ditaksir nilai efektivitasnya.

2.2.5 Siklus informasi

Untuk memperoleh informasi yang bermanfaat bagi penerimanya, perlu untuk dijelaskan bagaimana siklus yang terjadi atau dibutuhkan dalam menghasilkan informasi. Siklus informasi atau siklus pengolahan data adalah sebagai berikut :



Gambar 2.1 Siklus informasi

2.3 Konsep Dasar Sistem Informasi

Sistem informasi dalam sebuah sistem meliputi pemasukan data (*input*) kemudian diolah melalui suatu model dalam pemrosesan data, dan hasil informasi akan ditangkap kembali sebagai suatu input dan seterusnya sehingga membentuk siklus informasi yang dapat diperoleh dari sistem informasi sebagai sistem khusus dalam organisasi untuk mengolah informasi tersebut

Sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi. Pengertian lain dari sistem informasi adalah sekumpulan

prosedur organisasi yang pada saat dilaksanakan dan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan informasi.

2.3.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building blok*) yaitu :

a. *Hardware* yaitu suatu perangkat keras dalam komputer yang kita bisa sentuh dan rasakan.
b. *Software* yaitu suatu perangkat lunak di dalam komputer yang berfungsi untuk mengoperasikan suatu aplikasi di dalam sistem komputer.
c. Data yaitu sekumpulan karakter yang diterima sebagai masukan (*input*) untuk sistem informasi dan disimpan serta diolah.
d. Prosedur yaitu suatu urutan pekerjaan tata usaha yang biasanya melibatkan beberapa orang dalam satu bagian atau lebih, dan disusun untuk menjamin adanya perlakuan yang seragam terhadap transaksi-transaksi perusahaan yang terjadi.
e. *User* yaitu orang yang terlibat dalam sistem informasi seperti operator, pemimpin sistem informasi, dan sebagainya.

2.3.2 Tujuan Sistem Informasi

Sistem Informasi memiliki beberapa tujuan, yaitu:

1. Integrasi sistem
    a. Menghubungkan sistem individu/kelompok
    b. Pengkolektifan data dan penyambungan secara otomatis
    c. Peningkatan koordinasi dan pencapaian sinergi.
2. Efisiensi pengelolaan
    a. Penggunaan basis data dalam upaya kesamaan pengadministrasian data
    b. Pengelolaan data berkaitan dengan karakteristik Informasi
    c. Penggunaan dan pengambilan Informasi.
3. Dukungan keputusan untuk manajemen
    a. Melengkapi Informasi guna kebutuhan proses pengambilan kebutuhan
    b. Akuisisi Informasi eksternal melalui jaringan komunikasi
    c. Ekstraksi dari Informasi internal yang terpadu.

2.3.3 Manfaat Sistem Informasi

Sistem Informasi memiliki beberapa manfaat, yaitu:

1. Menghemat tenaga kerja
2. Peningkatan efisiensi
3. Mempercepat proses
4. Perbaikan dokumentasi

5. Pencapaian standar
6. Perbaikan keputusan

2.4 Konsep Rekayasa Perangkat Lunak

2.4.1 Pengertian Rekayas Perangkat Lunak

Rekayasa Perangkat Lunak adalah aplikasi dari sebuah pendekatan kuantifiabel, disiplin, dan sistematis kepada pengembangan, operasi, dan pemeliharaan perangkat lunak. Usaha yang berhubungan dengan rekayasa perangkat lunak dapat dikategorikan ke dalam tiga fase umum dengan tanpa mempedulikan area aplikasi, ukuran proyek, atau kompleksitasnya, yaitu :

1. Fase Definisi (*Definition Phase*)

   Fase ini berfokus pada “apa” (what); dimana, pada definisi ini pengembang perangkat lunak harus mengidentifikasi informasi apa yang akan diproses, fungsi dan unjuk kerja apa yang dibutuhkan, tingkah laku sistem seperti apa yang diharapkan, antarmuka apa yang akan dibangun, batasan perancangan serta kriteria validasi untuk mendefinisikan sistem yang sukses. Tugas teknis yang harus selalu ada dalam fase ini yaitu rekayasa sistem atau informasi, perencanaan proyek perangkat lunak, serta analisis kebutuhan.

2. Fase Pengembangan (*Development Phase*)

   Fase ini berfokus pada "bagaimana" (how), yaitu dimana selama masa pengembangan perangkat lunak, teknisi harus mendefinisikan bagaimana data dikonstruksikan, bagaimana fungsi-fungsi diimplementasikan sebagai sebuah arsitektur perangkat lunak, bagaimana detail prosedur akan diimplementasikan, bagaimana antarmuka dikarakterisasi, bagaimana rancangan akan diterjemahkan ke dalam bahasa pemrograman serta bagaimana pengujian akan dilakukan. Tugas teknis yang harus selalu ada dalam fase ini yaitu rancangan perangkat lunak, pemunculan kode, dan pengujian perangkat lunak.

3. Fase Pemeliharaan (*Maintenance Phase*)

   Fase ini berfokus pada "perubahan" (*change*), yang dihubungkan dengan koreksi kesalahan, penyesuaian yang dibutuhkan ketika lingkungan perangkat lunak berkembang, serta perubahan kebutuhan pelanggan. Fase ini mengaplikasikan kembali langkah-langkah pada fase definisi dan pengembangan namun semuanya tetap bergantung pada konteks perangkat lunak yang ada.

Untuk menyelesaikan masalah aktual di dalam sebuah setting industri, rekayasa perangkat lunak atau tim perekayasa harus menggabungkan strategi pengembangan yang melingkupi lapisan proses, metode, dan alat-alat bantu serta fase-fase generik. Strategi ini sering diacukan sebagai model proses atau paradigma 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 penyampaian yang dibutuhkan.

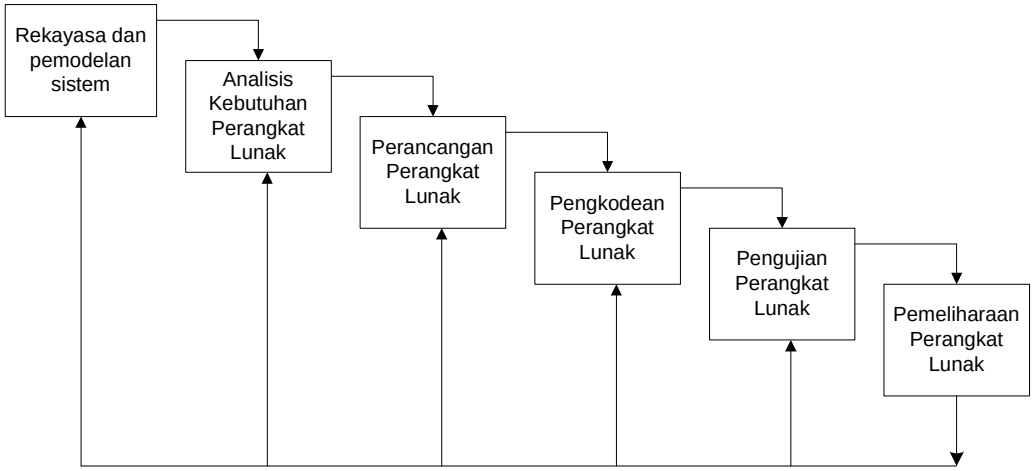
Model proses untuk rekayasa perangkat lunak dibagi menjadi empat model yaitu :

1. Model Sekuensial Linear
2. Model Prototipe
3. Model RAD
4. Model Proses Perangkat Lunak Evolusioner

2.4.2 Model Sekuensial Linear (*Waterfall*)

Model ini mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak yang sistematik dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian dan pemeliharaan.

Model waterfall dapat digambarkan sebagai berikut :



Gambar 2.2 Diagram *Waterfall*

1. Rekayasa dan Pemodelan Sistem

   Merupakan langkah awal dari rekayasa perangkat lunak, yang dimulai dari membangun syarat dari semua elemen sistem dan mengalokasikan beberapa bagian dari kebutuhan ke perangkat lunak tersebut.

2. Analisis Kebutuhan Perangkat Lunak

   Tahap ini digunakan untuk menjelaskan permasalahan yang ada dan menganalisa proses-proses yang akan dilakukan dalam perangkat lunak yang akan dibuat. Dalam tahap ini dilakukan pengumpulan kebutuhan perangkat lunak secara intensif.

3. Perancangan Perangkat Lunak

   Merupakan proses penerjemahan kebutuhan sistem ke dalam representasi perangkat lunak yang dapat diperkirakan sebelum dimulainya pemunculan

kode perangkat lunak (pengkodean). Proses ini berfokus perancangan pada struktur data, arsitektur program, representasi *interface* dan detail (algoritma) prosedural.

4. Pengkodean Perangkat Lunak

   Pengkodean merupakan suatu kegiatan mengolah hasil rancangan ke dalam bentuk bahasa pemrograman (kode-kode pemrograman) agar dapat dibaca mesin. Pengkodean dapat diselesaikan secara mekanis jika perancangan dilakukan dengan lengkap.

5. Pengujian Perangkat Lunak

   Pengujian perangkat lunak atau program merupakan proses pengujian suatu program yang dilakukan secara terintegrasi maupun secara per-unit, untuk mengetahui kesalahan yang terjadi pada program yang sedang berjalan, dan memastikan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan yang dibutuhkan.

6. Pemeliharaan Perangkat Lunak

   Perangkat lunak yang telah diuji dan berhasil, maka akan digunakan dan dipelihara. Pemeliharaan terjadi jika terdapat perubahan kebutuhan perangkat lunak. Pemeliharaan perangkat lunak mengaplikasikan lagi setiap fase program sebelumnya dan tidak membuat aplikasi baru lagi.

2.4.3 Model Prototipe

Prototyping paradigm dimulai dengan pengumpulan kebutuhan. Pengembang dan pelanggan bertemu dan mendefenisikan obyektif keseluruhan dari perangkat lunak, mengidentifikasikan segala kebutuhan yang diketahui, dan area garis besar dimana defenisi lebih jauh merupakan keharusan kemudian dilakukan perancangan kilat. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan Nampak bagi pelanggan/pemakai (contohnya pendekatan input dan format output). Perancangan kilat membawa kepada konstruksi sebuah prototype. Prototype tersebut dievaluasi oleh pelanggan/pemakai dan dipakai untuk menyaring kebutuhan pengembangan untuk secara lebih baik memahami apa yang harus dilakukannya.

Secara ideal prototipe berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Bila prototype yang sedang bekerja dibangun, pengembang harus mempergunakan fragmen-fragmen program yang ada atau mengaplikasikan alat-alat bantu (contohnya report generator, window manager, dll) yang memungkinkan program yang bekerja untuk dimunculkan secara cepat.

Pada sebagian proyek, sistem pertama yang dibangun baru saja bisa dipergunakan. Sistem mungkin terlalu pelan, terlalu besar, janggal di dalam pemakaian. Tidak ada alternatif lain selain mulai lagi, tidak dengan halus tetapi dengan lebih halus lagi, dan membangun sebuah versi yang dirancang kembali di mana masalah-masalah tersebut bisa diselesaikan. Ketika sebuah konsep sistem baru

atau teknologi baru dipergunakan, seseorang harus membangun sebuah sistem sebagai pembuangan, bahkan untuk perencanaan terbaik sekalipun, tidak akan mudah untuk membuatnya benar pada pertama kalinya. Dengan demikian, pertanyaan manajemen tidaklah untuk membangun sebuah sistem contoh dan untuk membuangnya.

2.4.4 Model RAD

Rapid Aplication Development (RAD) adalah sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Model RAD ini merupakan sebuah adaptasi kecepatan tinggi dari model sekuensial linier di mana perkembangan cepat dicapai dengan menggunakanpendekatan konstruksi berbasis komponen. Jika kebutuhan dipahami dengan baik, proses RAD memungkinkan tim pengembangan menciptakan sistem fungsional yang utuh dalam periode waktu yang sangat pendek (kira-kira 60 sampai 90 hari). Karena dipakai terutama pada aplikasi sistem konstruksi, pendekatan RAD melingkupi fase-fase sebagai berikut :

1. Bussiness modeling

   Aliran informasi diantara fungsi-fungsi bisnis dimodelkan dengan suatu cara untuk menjawab pertanyaan-pertanyaan berikut : Informasi apa yang mengendalikan proses bisnis? Informasi apa yang dimunculkan? Siapa yang memunculkannya? Kemana informasi itu pergi? Siapa yang memprosesnya?

2. Data modeling

   Aliran informasi yang didefinisikan sebagai bagian dari fase bussiness modelling disaring ke dalam serangkaian objek data yang dibutuhkan untuk menopang bisnis tersebut. Karakteristik masing-masing objek diidentifikasi dan hubungan antara objek-objek tersebut didefinisikan.

3. Prosess modeling

   Aliran inforamsi yang didefinisikan di dalam fase data modeling ditransformasikan untuk mencapai aliran inforamsi yang perlu bagi implementasi sebuah fungsi bisnis. Gambaran pemrosesan diciptakan untuk menambah, momodifikasi, menghapus, atau mendapatkan kembali sebuah objek data.

4. Aplication generation

   RAD mengasumsikan pemakaian teknik generasi keempat. Selain menciptakan perangkat lunak dengan menggunakan bahasa pemrograman generasi ketiga yang konvensional, RAD lebih banyuak memproses kerja untuk memakai lagi komponen program yang ada atau menciptakan komponen yang bisa dipakai lagi. Pada semua kasus, alat-alat bantu otomatis dipakai untuk memfasilitasi konstruksi perangkat lunak.

5. Testing and turnover

   Karena proses RASD menekankan pada pemakaian kembali, banyak komponen program telah diuji. Hal ini mengurangi keseluruhan waktu pengujian. Tetapi komponen baru harus diuji dan semua interface harus dilatih secara penuh.

Secara jelas batasan waktu yang dibebankan pada sebuah proyek RAD memerlukan ruang lingkup yang bisa diskala. Jika sebuah aplikasi bisnis dapat dimodulkan dengan cara tertentu sehingga memungkinkan setiap fungsi mayor untuk dilengkapi dalam waktukurang dari 3 bulan, maka aplikasi itu merupakan kandidat bagi RAD. Masing-masing fungsi mayor bisa dibicarakan oleh suatu tim RAD yang terpisah, dan kemudian diintegrasikan untuk membentuk suatu kesatuan.

Seperti semua proses model yang lain, pendekatan RAD memiliki kekurangan yaitu :

1. Bagi proyek yang besar tetapi berskala, RAD memerlukan sumber daya manusia yang memadai untuk menciptakan jumlah tim RAD yang baik.
2. RAD menuntut pengembang dan pelanggan memiliki komitmen di dalam aktivitas *rapid-fire* yang diperlukan untuk melengkapi sebuah sistem, di dalam kerangka waktu yang sangat diperpendek. Jika komitmen tersebut tidak ada dari tiap konstituen, proyek RAD akan gagal.

Tidak semua aplikasi sesuai untuk RAD. Bila sistem tidak dapat dimodulkan dengan teratur, pembangunan komponen penting pada RAD akan menjadi sangat problematis. RAD menjadi tidak sesuai jika risiko teknisnya tinggi. Hal ini terjadi bila sebuah aplikasi baru memforsir teknologi baru atau bila perangkat lunak baru membutuhkan tingkat interoperabilitas yang tinggi dengan program komputer yang ada. RAD menekankan perkembangan komponen program yang bisa dipakai kembali.

2.4.5 Model Proses Perangkat Lunak Evolusioner

Ada pengakuan yang tumbuh bahwa perangkat lunak, seperti semua sistem kompleks yang lain, mencukupi lebih dari satu periode waktu. Kebutuhan bisnis dan produk kadang-kadang berubah seiring dengan laju perkembangannya, membentuk sebuah jejak garis lurus menuju hasil akhir yang tidak realistis; batas waktu pasar yang ketat menyebabkan pelengkapan produk perangkat lunak yang komprehensif menjadi tidak mungkin, tetapi sebuah versi yang terbatas harus dikenalkan untuk memenuhi tekanan yang kompetitif; sejumlah produk inti atau ekstensi sistem harus dibatasi. Di dalam situasi ini maupun yang lainnya, perekayasa perangkat lunak membutuhkan sebuah model proses yang sudah dirancang secara eksplisit untuk mengakomodasi produk perkembangan sepanjang waktu.

Model sekuensial linier dirancang bagi perkembangan garis lurus. Pada dasarnya pendekatan air terjun ini mengandaikan bahwa sebuah sistem lengkap akan disampaikan setelah urutan linier tersebut dilengkapi. Model prototipe dirancang untuk mendorong konsumen agar memahami kebutuhan. Secara umum model itu tidak dirancang untuk menyampaikan sistem produksi. Sifat evolusioner perangkat lunak tidak dimasukkan di dalam salah satu dari paradigma rekayasa perangkat lunak klasik tersebut.Model evolusioner adalah model iteratif. Model itu ditandai dengan tingkah laku yang memungkinka perekayasa perangkat lunak mengembangkan versi perangkat lunak yang lebih lengkap sedikit demi sedikit.

2.5 Konsep Perancangan Sistem

Perancangan sistem secara umum adalah suatu tahap dimana di dalamnya terdapat identifikasi komponen-komponen sistem Informasi yang akan dirancang secara rinci yang bertujuan untuk memberikan gambaran kepada pengguna atau *user* mengenai sistem yang baru. Sedangkan desain sistem secara terinci dimaksudkan untuk pembuat program komputer dan ahli teknik lainnya yang akan mengimplementasikan sistem.

Penggambaran dan rancangan model sistem Informasi secara logika dapat dibuat dalam bentuk Diagram Konteks dan Diagram Alir Data (DAD) atau Data *Flow Diagram* (DFD).

2.5.1 Diagram Konteks

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

2.5.2 Diagram Aliran Data

Diagram Alir Data atau *Data Flow Diagram* (DFD) adalah suatu model yang menjelaskan arus data mulai dari pemasukan sampai dengan keluaran data. Tingkatan

DFD dimulai dari diagram konteks yang menjelaskan secara umum suatu sistem atau batasan sistem aplikasi yang akan dikembangkan. Kemudian DFD dikembangkan menjadi DFD tingkat 0 atau level 0 dan kemudian DFD level 0 dikembangkan lagi menjadi level 1 dan selanjutnya sampai sistem tersebut tergambarkan secara rinci menjadi tingkatan-tingkatan lebih rendah lagi.

DFD merupakan penurunan atau penjabaran dari diagram konteks. Dalam pembuatan DFD harus mengacu pada ketentuan sebagai berikut :

1. Setiap penurunan level yang lebih rendah harus mempresentasikan proses tersebut dalam spesifikasi proses yang jelas.
2. Penurunan dilakukan apabila memang diperlukan.
3. Tidak semua bagian dari sistem harus ditunjukkan dengan jumlah level yang sama.

Simbol-simbol yang digunakan pada diagram alir data atau *data flow diagram* antara lain :

a. Entitas luar (*external entity*)

   Sesuatu yang berada di luar sistem, tetapi memberikan data ke dalam sistem atau memberikan data dari sistem, disimbolkan dengan suatu kotak notasi. External entity tidak termasuk bagian dari sistem bila sistem informasi dirancang untuk satu bagian maka bagian lain yang masih terkait menjadi external entity.

b. Arus data (*data flow*)

Arus data merupakan tempat mengalirnya informasi dan digambarkan dengan garis yang menghubungkan komponen dari sistem. Arus data ditunjukan dengan arah panah dan garis diberi nama atau arus data yang mengalir. Arus data ini mengalir diantara proses, data store dan menunjukan arus data dari data yang berupa masukan untuk sistem atau hasil proses sistem.

c. Proses (*procces*)

Proses merupakan apa yang dikerjakan oleh sistem. Proses dapat mengolah data atau aliran data masuk menjadi aliran data keluar. Proses berfungsi mentransformasikan satu atau beberapa data keluaran sesuai dengan spesifikasi yang diinginkan. Setiap proses memiliki satu atau beberapa masukan serta menghasilkan satu atau beberapa data keluaran. Proses sering juga disebut *bubble*.

d. Simpanan data (*data store*)

Simpanan data merupakan tempat penyimpanan data yang ada dalam sistem. Data *store* dapat disimbolkan dengan dua garis sejajar atau dua garis dengan salah satu sisi samping terbuka. Proses dapat mengambli data dari atau memberikan data ke *database*.

2.6 Konsep Basis Data

2.6.1 Pengertian Basis Data

Basis data terdiri dari kata basis dan data. Basis dapat diartikan gudang atau tempat bersarang dan data yang berarti representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, hewan, peristiwa, konsep dan sebaginya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya.

Jadi dapat disimpulkan bahwa basis data merupakan kumpulan data (arsip) yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan. Atau bisa diartikan sebagai kumpulan file, tabel, arsip yang saling berhubunngan yang disimpan dalam media penyimpanan elektronik.

2.6.2 Konsep Dasar Basis Data

Basis data (*database*) dapat dibayangkan sebagai sebuah lemari arsip yang ditempatkan secara berurutan untuk memudahkan dalam pengambilan kembali data tersebut. Basis data menunjukan suatu kumpulan data yang dipakai dalam sistem informasi disebut sistem basis data (*database system*).

2.6.3 Database Relasional

Konsep sebuah database adalah terdiri atas tabel-tabel yang terorganisasi. Tabel-tabel tersebut dapat saling berelasi untuk menghasilkan suatu informasi, untuk mengakses data yang ada dalam tabel-tabel tersebut digunakan sebuah perintah SQL (*Structured Query Language*).

2.6.4 SQL (*Structured Query Language*)

Structured Query Language atau SQL merupakan sebuah bahasa *query* standar yang digunakan untuk mengakses database relasional yang meliputi perintah-perintah untuk menyimpan, menerima, memelihara dan mengatur akses-akses ke database serta digunakan untuk memanipulasi dan menampilkan data dari *database* tersebut.

Dimana *query* itu sendiri adalah suatu proses melakukan pengambilan atau pencarian informasi yang ada dalam *database* relasional berdasarkan kriteria tertentu.

Berdasarkan fungsinya, deklarasi atau perintah SQL dibagi ke dalam dua kategori yaitu DDL dan DML.

2.6.4.1 DDL (*Data Defenition Language*)

Merupakan kelompok perintah yang digunakan untuk melakukan pendefinisian *database* dan pendefinisian tabel. Dengan kelompok perintah dalam DDL ini maka kita dapat membuat tabel, mengubah srukturnya, menghapus tabel, membuat indeks untuk tabel, dan lain-lain yang bermuara pada pembentukan struktur *database*. DDL atau *Data Definition Language* adalah bagian dari SQL yang digunakan untuk mendefinisikan data dan objek *database*, dimana terdapat perintah-perintah sebagai berikut :

Tabel 2.1 Perintah DDL

Perintah	Keterangan
CREATE	Untuk mendefinisikan database, maupun tabel sebagai data yang akan disimpan maupun diakses
ALTER	Untuk memodifikasi tabel, baik itu menambah, menghapus, maupun mengganti kolom/field pada tabel
DROP	menghapus tabel dan database

2.6.4.2 Data Manipulation Language

Perintah (*statement*) SQL digunakan untuk melakukan manipulasi data dalam *database*, menambahkan (*insert*), Mengubah (*update*), menghapus (*delete*), mengambil dan mencari data (*query*). DML atau *Data Manipulation Language* adalah bagian dari SQL yang digunakan untuk memulihkan dan memanipulasi data. terdapat perintah-perintah yang digunakan dalam DML adalah sebagai berikut:

Tabel 2.2 Perintah DML

Perintah	Keterangan
SELECT	Untuk mengambil atau menampilkan data dari tabel pada database
INSERT	Untuk menyisipkan data pada tabel
UPDATE	Untuk memperbaharui nilai suatu data dalam database
DELETE	Untuk menghapus record pada tabel

2.6.5 Perancangan Basis Data

Perancangan basis data dibutuhkan agar diperoleh suatu sistem yang sesuai dengan apa yang diinginkan, dimana dapat melalui tahapan berikut :

2.6.5.1 Entity Relationship Diagram

Entity Relationship Diagram atau biasa dikenal dengan diagram E-R secara grafis menggambarkan isi sebuah *database*. Diagram ini memiliki dua komponen utama yaitu *entity* dan relasi. Untuk melambangkan fungsi diatas maka digunakan simbol-simbol yang bisa dilihat pada daftar simbol.

Elemen-elemen *Entity Relationship Diagram* adalah sebagai berikut:

1. *Entity* (Entitas)

   Pada E-R diagram, *entity* digambarkan dengan sebuah bentuk persegi panjang. *Entity* adalah sesuatu apa saja yang ada didalam sistem, nyata maupun abstrak dimana data tersimpan. Entitas diberi nama dengan kata benda dan dapat dikelompokan dalam empat jenis nama, yaitu : orang, benda, lokasi kejadian (terdapat unsur waktu didalamnya).

2. *Relationship* (Relasi)

   Pada E-R diagram, *relationship* dapat digambarkan dengan sebuah bentuk belah ketupat. *Realationship* adalah hubungan alamiah yang terjadi antara entitas. Pada umumnya *relationship* diberi nama dengan kata kerja dasar, sehinga memudahkan untuk melakukan pembacaan relasinya.

3. Atribut

   Seacara umum atribut adalah sifat atau karakteristik dari tiap entitas maupun tiap *relationship*. Maksudnya adalah sesutau yang menjelaskan apa sebenarnya yang

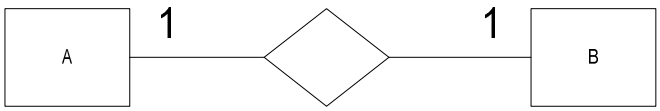
dimaksud entitas maupun *relationship*, sehingga sering dikatakan bahwa atribut adalah elemen dari setiap entitas dan *relationship*.

4. Kardinalitas

Kardinalitas relasi menunjukan jumlah maksimum tupel yang dapat berelasi dengan entitas yang lainnya. Dari sejumlah kemungkinan banyaknya hubungan yang terjadi dari entitas, kardinalitas relasi merujuk kepada hubungan maksimum yang terjadi dari entitas yang satu ke entitas yang lainnya dan begitu juga sebaliknya. Terdapat tiga macam kardinalitas relasi, yaitu:

a. *One to one Relationship*

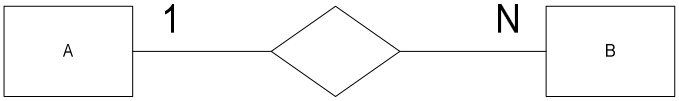
Tingkat hubungann satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.



Gambar 2.3 *One to One Relationship*

b. *One to many Relationship*

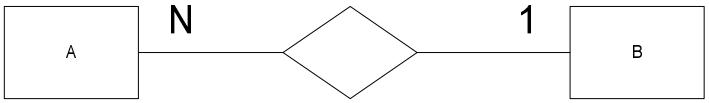
Tingkat hubungan satu ke banyak adalah untuk satu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua.



Gambar 2.4 *One to Many Relationship*

c. *Many To One Relationship*

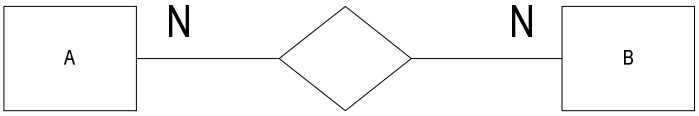
Untuk banyak kejadian pada entitas yang pertama hanya dapat mempunyai satu hubungan dengan kejadian pada entitas yang kedua.



Gambar 2.5 *Many to One Relationship*

d. *Many to many Relationship*

Tingkat hubungan banyak ke banyak terjadi jika tiap kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya, baik dilihat dari sisi entitas yang pertama maupun dilihat dari sisi yang kedua.



Gambar 2.6 *Many to Many Relationship*

5. *Key* (Kunci)

   Sebuah atribut atau set atribut yang nilainya mengidentifikasikan entitas secara unik dalam suatu entitas. *Key* memiliki beberapa jenis sesuai dengan kegunaannya masing-masing, yaitu *primary key* (kunci utama), *foreign key* (kunci tamu).

2.6.5.2 Kamus Data

Kamus data merupakan kumpulan data-data. Kamus data adalah katalog fakta tentang data dan kebutuhan informasi dari suatu sistem informasi. Kamus data merupakan tempat penyimpanan definisi dari aliran-aliran data, file-file dan proses-proses dalam sebuah sistem informasi. Dengan menggunakan kamus data, analisis sistem dapat memberikan informasi mengenai definisi struktur pemakaian masing-masing elemen, dapat mendefinisikan data yang mengalir di sistem dengan lengkap, dapat menghindari duplikasi elemen-elemen dan menghindari konflik antara elemen-elemen. Kamus data berfungsi untuk membantu pelaku sistem untuk mengartikan alokasi secara detail dan mengorganisasikan semua elemen data yang digunakan dalam sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses.

2.7 *Database Management System* (DBMS)

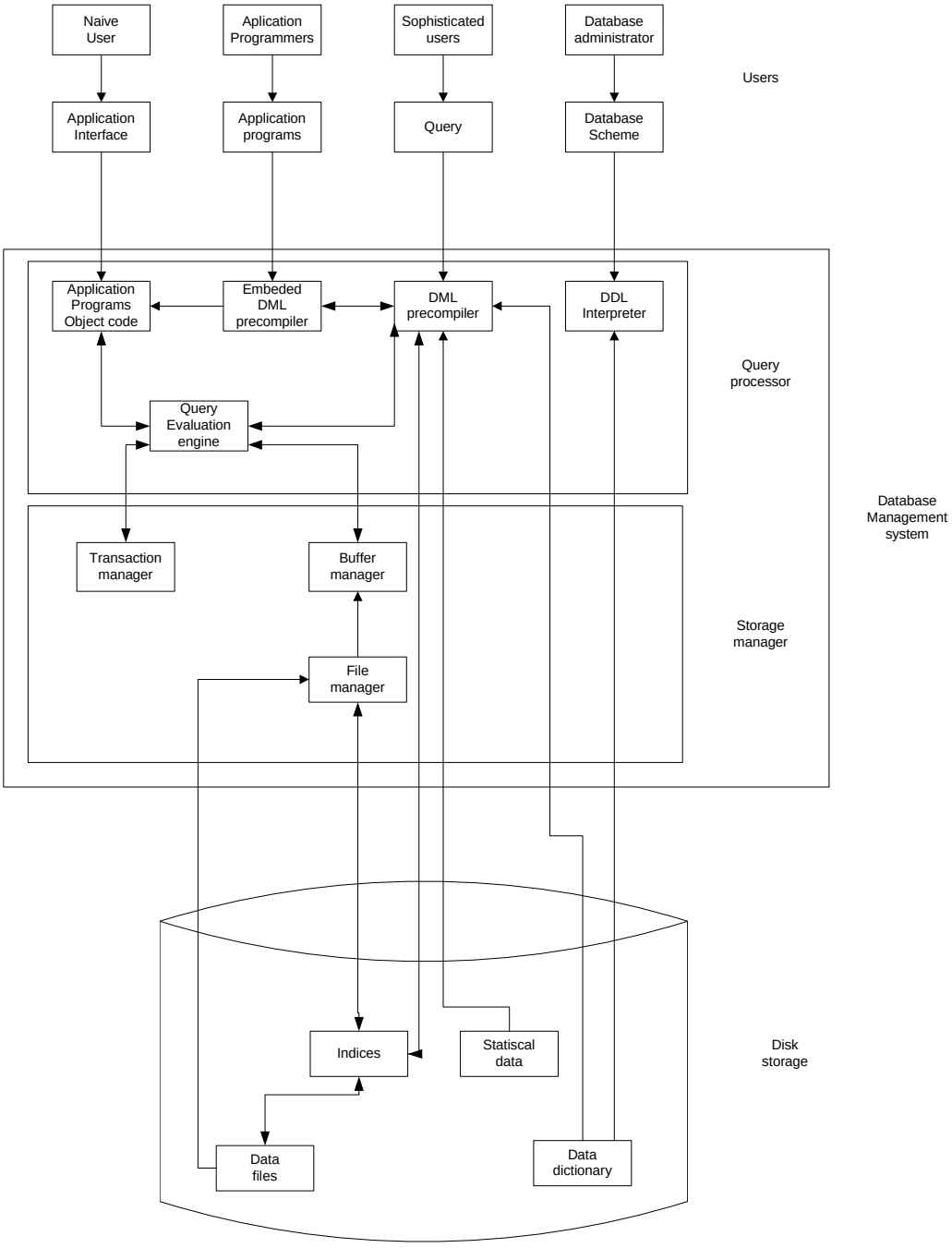
Database adalah kumpulan data yang saling berkaitan, berhubungan yang disimpan secara bersama-sama sedemikian rupa tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan. Data-data ini harus mengandung semua Informasi untuk mendukung semua kebutuhan sistem.

Proses dasar yang dimiliki oleh *database* ada 4, yaitu :

1. Pembuatan data-data baru (*create database*)
2. Penambahan data (*insert*)
3. Mengubah data (*edit*)
4. Menghapus data (*delete*)

Sistem manajemen *database* (*Database Management System*) merupakan sistem pengoperasian dan sejumlah data pada komputer. Dengan sistem ini dapat merubah data, memperbaiki data yang salah dan menghapus data yang tidak dapat dipakai. Sistem manajemen database merupakan suatu perluasan *software* sebelumnya mengenai *software* pada generasi komputer yang pertama.

Salah satu tujuan DBMS adalah untuk menyediakan fasilitas atau antarmuka (*interface*) dalam melihat atau menikmati data kepada pemakai. Untuk itu, sistem tersebut seringkali akan menyembunyikan detail tentang bagaimana data disimpan, dipakai atau dipelihara. Karena itu, seringkali data yang dilihat oleh pemakai sebelumnya berbeda dengan yang tersimpan secara fisik.



Gambar 2.7 Struktur Sistem

2.8 Jaringan (*Network*)

2.8.1 Pengertian Jaringan

Pengertian jaringan dapat didefinisikan sebagai berikut [8]:

“ komunikasi data yang dapat diartikan sebagai perpindahan data dari satu tempat ke tempat yang lainnya melalui media tertentu, sedangkan jaringan akan muncul ketika dua atau lebih peralatan komunikasi data digunakan untuk menghubungkan data. Peralatan komunikasi ini dapat berupa apapun yang bersifat maya (virtual) yang dapat mengkomunikasikan data”.

2.8.2 Jenis - jenis Jaringan Komputer

Jaringan komputer dapat dibedakan berdasarkan cakupan geografisnya ada 4 kategori utama dalam jaringan komputer [9], yaitu

a. LAN

   LAN digunakan untuk menghubungkan komputer yang berada di dalam suatu area yang kecil, misalnya di dalam suatu gedung perkantoran atau kampus. Jarak antara komputer yang dihubungkannya bisa mncapai 5 sampai 10 Km. Suatu LAN biasanya bekerja pada kecepatan mulai 10 Mbps sampai 100 Mbps. LAN menadi populer karena memungkinkan banyak pengguna untuk memakai sumber daya yang dapat digunakan itu misalnya suatu *mainframe, file server,* printer dan sebagainya.

b. MAN

MAN merupakan suatu jaringan yang cangkupannya meliputi suatu kota. MAN menhubungkan LAN-LAN yang lokasinya berjauhan. Jangkauan MAN bisa mencapai 10 Km sampai beberapa ratus Km. Suatu MAN biasanya bekerja pada kecepatan 1,5 sampai 150 Mbps.

c. WAN

WAN dirancang untuk menghubungkan komputer-komputer yang terletak pada suatu cakupan geografis yang luas, seperti hubungan dari satu kota ke kota lain di dalam suatu negara. Cakupan WAN bisa meliputi 100 Km sampai 1.000 Km, dan kecepatan antar kota bisa bervariasi antara 1,5 Mbps samapai 2,4 Gbps. Biasanya jaringan WAN dimiliki dan dioperasikan sebagai suatu jaringan publik.

d. GAN

GAN merupakan suatu jaringan yang menghubungkan negara-negara di seluruh dunia. Kecepatan GAN bervariasi mulai dari 1,5 Mbps samapai dengan 100 Gbps dan cakupannya mencapai ribuan kilometer. Contoh yang sangat baik dari GAN ini adalah *internet.*

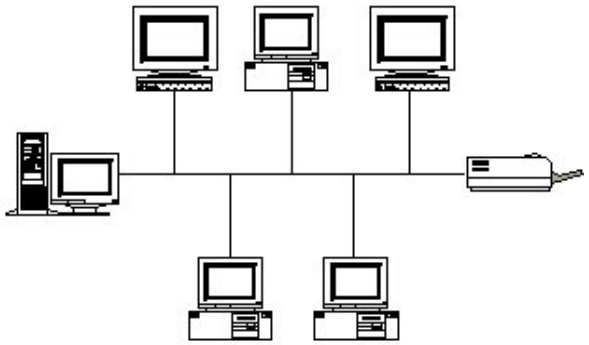
2.8.3 Topologi Jaringan

Topologi adalah suatu cara menghubungkan komputer yang satu dengan komputer lainnya sehingga membentuk jaringan. Cara yang saat ini banyak

digunakan adalah bus, token-ring, star dan peer-to-peer *network*. Masing-masing topologi ini mempunyai ciri khas, dengan kelebihan dan kekurangannya sendiri.

2.8.3.1 Topologi Linear Bus

Topologi *linear bus* terdiri dari satu jalur kabel utama dimana pada masing-masing ujungnya diberikan sebuah *terminator*. Semua *nodes* pada jaringan (*file server*, *workstation*, dan perangkat lainnya) terkoneksi sebuah kabel utama (*backbone*), jaringan-jaringan *ethernet* dan *local talk* mengunakan topologi liner bus ini



Gambar 2.8 Topologi BUS

Keuntungan

a. Hemat kabel
b. Layout kabel sederhana

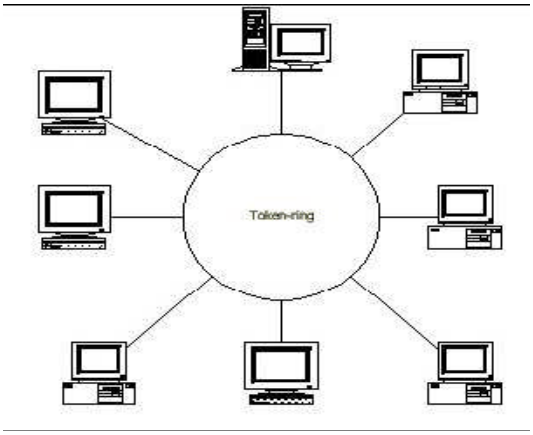
c. Mudah dikembangkan

Kerugian

a. Deteksi dan isolasi kesalahan sangat kecil
b. Kepadatan lalu lintas
c. Bila salah satu client rusak, maka jaringan tidak bisa berfungsi.
d. Diperlukan repeater untuk jarak jauh.

2.8.3.2 Topologi Ring

Metode token-ring (sering disebut ring saja) adalah cara menghubungkan komputer sehingga berbentuk ring (lingkaran). Setiap simpul mempunyai tingkatan yang sama. Jaringan akan disebut sebagai loop, data dikirimkan kesetiap simpul dan setiap informasi yang diterima simpul diperiksa alamatnya apakah data itu untuknya atau bukan



Gambar 2.9 Topologi TokenRING

Keuntungan

a. Hemat Kabel

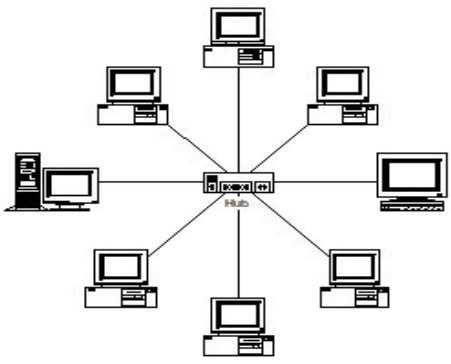
Kerugian

a. Peka kesalahan

b. Pengembangan jaringan lebih kaku

2.8.3.3 Topologi Star

Kontrol terpusat, semua link harus melewati pusat yang menyalurkan data tersebut kesemua simpul atau client yang dipilihnya. Simpul pusat dinamakan stasium primer atau server dan lainnya dinamakan stasiun sekunder atau client server. Setelah hubungan jaringan dimulai oleh server maka setiap client server sewaktu-waktu dapat menggunakan hubungan jaringan tersebut tanpa menunggu perintah dari server.



Gambar 2.10 Topologi STAR

Keuntungan :

a. Paling fleksibel
b. Pemasangan/perubahan stasiun sangat mudah dan tidak mengganggu bagian jaringan lain
c. Kontrol terpusat
d. Kemudahan deteksi dan isolasi kesalahan/kerusakan

Kerugian :

a. Boros kabel
b. Perlu penanganan khusus
c. Kontrol terpusat (HUB) jadi elemen kritis

2.8.4 Client – Server

Konsep dari *client server* adalah sebagai Sebuah aplikasi yang dapat dianggap sebagai *requestor* (*client*) atau dapat juga dianggap sebagai *provider* (*server*).

Biasanya jumlah *client* jauh lebih banyak daripada jumlah *server* dan mampu memberikan layanan kepada banyak *client* dengan kemampuan yang sama sebagaimana ketika hanya melayani sebuah *client* dari sisi suatu arsitektur *client server*, bahwa *client* adalah sebuah aplikasi yang berjalan pada komputer pribadi dan bergantung pada *server* untuk mengerjakan oprerasi. Sedangkan *server* adalah node

yang memungkinkan node lain pada LAN untuk mengakses sumbernya. *Server* ini bersifat terdedikasi yang artinya node tersebut dapat dipakai dengan cara lain.

2.8.5 Media Transmisi

Kabel adalah penghubung untuk mengirim informasi dari satu komputer ke komputer yang lain. Ada beberapa macam tipe kabel yang umumnya digunakan pada LAN. Dalam beberapa kasus, sebuah jaringan hanya menggunakan satu macam tipe kabel, sedangkan di jaringan yang lain menggunakan kabel yang berbeda. Kabel yang dipilih adalah berdasarkan dengan topologi jaringan, protocol jaringan , dan ukurannya. Hal ini sangat penting untuk diketahui karena kesuksesan jaringan bergantung dari semua perihal tersebut.

Pada bagian ini dibahas mengenai tipe-tipe dan penggunaan kabel yang digunakan di dalam jaringan local diantaranya:

2.8.5.1 Kabel UTP (*Unshielded Twisted Pair*)

Kabel Twisted Pair terdiri dari dua tipe yaitu sheilded and unshielded. *Unsheilded twisted pair* (UTP) adalah yang paling populer dan umumnya merupakan pilihan yang terbaik untuk jaringan sederhana.



Gambar 2.11 Kabel *Unshielded Twisted Pair*

Jenis konektor untuk kabel jenis ini adalah RJ-45.



Gambar 2.12 *Konektor RJ-45*

2.8.5.2 Kabel STP (*Shielded Twisted Pair*)

Kekurangan kabel jenis ini adalah, sangat sensitif terhadap signal radio dan listrik. Kabel seperti ini sangat baik digunakan dimana lingkungan pengaruh listrik kurang, serta biasanya digunakan pada jaringan yang menggunakan topologi token ring.

2.8.5.3 Kabel Coaxisal

Coaxial Cabel merupakan kabel yang dibungkus dengan metal yang lembek. Coaxial Cabel mempunyai tingkat transmisi data yang lebih tinggi dibandingkan dengan kabel biasa, tetapi lebih mahal.

2.8.5.4 Kabel Fiber Optik

Fiber Optik cable dibuat dari serabut-serabut kaca (Optikal Fibers) yang tipis dengan diameter sebesar diameter rambut manusia. Fiber Optik cable mempunyai kecepatan pengiriman data sampai sepuluh kali lebih besar dari coaxial kabel.



Gambar 2.13 Kabel *Fiber Optik*

2.9 Manfaat Jaringan Komputer

a. Resource Sharing

   Dapat menggunakan sumber daya yang ada secara bersama-sama. Misal seorang pengguna yang berada 100 km jauhnya dari suatu data, tidak mendapatkan kesulitan dalam menggunakan data tersebut, seolah-olah data tersebut berada didekatnya. Hal ini sering diartikan bahwa jaringan komputer mangatasi masalah jarak.

b. Reliabilitas tinggi

   Dengan jaringan komputer kita akan mendapatkan reliabilitas yang tinggi dengan memiliki sumber-sumber alternatif persediaan. Misalnya, semua file dapat disimpan atau dicopy ke dua, tiga atu lebih komputer yang terkoneksi

kejaringan. Sehingga bila salah satu mesin rusak, maka salinan di mesin yang lain bisa digunakan.

c. Menghemat uang.

Komputer berukutan kecil mempunyai rasio harga/kinerja yang lebih baik dibandingkan dengan komputer yang besar. Komputer besar seperti mainframe memiliki kecapatan kira-kira sepuluh kali lipat kecepatan komputer kecil/pribadi. Akan tetap, harga mainframe seribu kali lebih mahal dari komputer pribadi. Ketidakseimbangan rasio harga/kinerja dan kecepatan inilah membuat para perancang sistem untuk membangun sistem yang terdiri dari komputer-komputer pribadi.

2.10 Perangkat Lunak Pendukung

Perangkat lunak merupakan suatu kebutuhan yang harus dipenuhi oleh suatu sistem. Kebutuhan ini diperlukan untuk mencapai suatu tujuan.

2.10.1 MySQL

MySQL adalah suatu sistem manajemen *database*. Suatu *database* adalah sebuah kumpulan data yang terstruktur. Untuk menambahkan, mengakses, dan memproses data yang tersimpan pada suatu *database* komputer anda memerlukan sistem manajemen *database* seperti MySQL. Karena komputer sangat unggul dalam menangani sejumlah besar data, sistem manajemen *database* memainkan suatu

peranan yang penting dalam komputasi, baik sebagai *utility stand-alone* maupun bagian dari aplikasi lainnya.

Suatu *database* relasional menyimpan data dalam tabel-tabel terpisah. Hal ini memungkinkan kecepatan dan fleksibilitas. Tabel-tabel yang dihubungkan dengan relasi yang ditentukan membuatnya bisa mengkombinasikan data dari beberapa tabel pada suatu permintaan. Bagian SQL dari kata MYSQL berasal dari "Structured Query Language" bahasa paling umum yang dipergunakan untuk mengakses database.

Beberapa perintah dasar SQL yang sering dipergunakan pada MySQL adalah sebagai berikut :

a. Create Database

   Yaitu perintah yang digunakan untuk membuat database baru.

   Sintaks : Create database database_nama database

b. Drop Database

   Yaitu perintah yang digunakan untuk menghapus database.

   Sintaks : Drop Tabel Tabel_name

c. Create Tabel

   Yaitu perintah yang digunakan untuk membuat tabel baru.

   Sintaks Create Tabel tabel_name (create_definition)

d. Describe

   Yaitu perintah yang digunakan untuk mendeskripsikan tabel atau logam

   Sintaks Describe (Desc) tabel [colum]

e. Alter Tabel

Yaitu perintah yang digunakan untuk memodifikasi tabel

Sintaks Alter [Ignor] Tabel table_name

f. Drop Tabel

Yaitu perintah yang digunakan untuk menghapus tabel

Sintaks Drop Tabel tabel_name [tabel_name..]

g. Delete

Yaitu perintah yang digunakan untuk menghapus record dri tabel

Sintaks Delete From tabel_name Where Where_definiition

h. Select

Yaitu perintah yang digunakan untuk query ke database

Sintaks select*from tabel_name

Select Field from tabel_name

2.10.2 Visual Basic

Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang bersifat *event driven* dan menawarkan *Integrated Development Environment* (IDE) visual untuk membuat program aplikasi berbasis sistem operasi Microsoft Windows dengan menggunakan model pemrograman Common Object Model (COM). Visual Basic merupakan turunan bahasa BASIC dan menawarkan pengembangan aplikasi komputer berbasis grafik dengan cepat, akses ke

basis data menggunakan Data Access Objects (DAO), Remote Data Objects (RDO), atau ActiveX Data Object (ADO), serta menawarkan pembuatan kontrol ActiveX dan objek ActiveX. Beberapa bahasa skrip seperti Visual Basic for Applications (VBA) dan Visual Basic Scripting Edition (VBScript), mirip seperti halnya Visual Basic, tetapi cara kerjanya yang berbeda.

Para programmer dapat membangun aplikasi dengan menggunakan komponen-komponen yang disediakan oleh Microsoft Visual Basic Program-program yang ditulis dengan Visual Basic juga dapat menggunakan Windows API, tapi membutuhkan deklarasi fungsi eksternal tambahan.

Dalam pemrograman untuk bisnis, Visual Basic memiliki pangsa pasar yang sangat luas. Dalam sebuah survey yang dilakukan pada tahun 2005, 62% pengembang perangkat lunak dilaporkan menggunakan berbagai bentuk Visual Basic, yang diikuti oleh C++, JavaScript, C#, dan Java.

Visual Basic merupakan bahasa yang mendukung OOP, namun tidak sepenuhnya. Beberapa karakteristik obyek tidak dapat dilakukan pada Visual Basic, seperti *Inheritance* tidak dapat dilakukan pada class module. *Polymorphism* secara terbatas bisa dilakukan dengan mendeklarasikan *class module* yang memiliki *Interface* tertentu. Visual Basic (VB) tidak bersifat case sensitif.

2.11 Pengertian Koperasi

Menurut Sagimun [6] dalam kongres ICA (International Coorperative Alliance) ke 100 tahun 1995 di Manchester, United Kingdom. Koperasi adalah perkumpulan otonom dari orang-orang yang bersatu secara sukarela untuk memenuhi kebutuhan-kebutuhan dan aspirasi-aspirasi ekonomi, sosial dan budaya bersama melalui perusahaan yang mereka miliki bersama dan mereka kendalikan secara demokratis.

2.11.1 Simpan Pinjam

Koperasi memberikan kesempatan kepada anggotanya untuk memperoleh pinjaman dengan mudah dan dengan bunga yang rendah. Akan tetapi untuk mendapatkan pinjaman atau kredit koperasi memerlukan modal-modal koperasi yang paling utama adalah dari simpanan anggota itu sendiri, dari uang yang dikumpulkan bersama dan usaha pinjaman diberikan kepada anggota yang memerlukan.

Adapun sumber-sumber dana kredit koperasi tersebut yaitu :

1. Simpanan sukarela anggota

   Yaitu simpanan yang menjadi keharusan, tetapi diterima oleh pihak koperasi dari anggota atas dasar kehendak anggota itu sendiri dengan waktu pembayaran simpanan tidak ditentukan oleh pihak koperasi.

2. Simpanan Wajib

Yaitu Simpanan wajib adalah jumlah simpanan tertentu yang harus dibayarkan oleh anggota kepada koperasi dalam waktu dan kesempatan tertentu, misalnya tiap bulan dengan jumlah simpanan yang sama untuk setiap bulannya.

2.11.2 Nilai – nilai Koperasi

Koperasi berlandaskan nilai-nilai menolong diri sendiri, tanggung jawab pribadi, demokratis, persamaan, keadilan dan tanggung jawab sosial serta kepedulian terhadap orang lain.

2.11.3 Prisip – prinsip Koperasi

Menurut Sinarhadi prinsip-prinsip koperasi adalah garis-garis penentu yang digunakan oleh koperasi untuk memasukan nilai-nilai tersebut dalam melaksanakan kegiatan koperasi. Prinsip-prinsip koperasi sebagai berikut :

a. Keanggotaan Sukarela Dan Terbuka

   Perkumpulan secara sukarela, terbuka bagi semua orang yang memerlukan jasa-jasa perkumpulan dan bersedia menerima tanggung jawab keanggotaan tanpa diskriminasi, sosial, rasio, politik dan agama.

b. Pengendalian Oleh Anggota Secara Demokratis

   Perkumpulan-perkumpulan secara demokratis yang dikendalikan oleh para anggota yang secara aktif berpartisipasi dalam menetapkan kebijaksanaan dan pengambilan keputusan.

c. Partisipasi Ekonomi Anggota

Para anggota secara adil memberikan andilnya ke dalam koperasi, secara demokratis mengawasi modal serta setidak-tidaknya sebagai dari modal adalah milik bersama.

d. Otonomi Dan Kebebasan

Otonomi merupakan perkumpulan yang menolong diri sendiri dan dikendalikan oleh anggota-anggotanya. Koperasi mengadakan kesepakatan dengan perkumpulan-perkumpulan lain termasuk pemerintah untuk memperoleh modal dari sumber-sumber lain.

e. Pendidikan, Pelatihan Dan Informasi

Koperasi memberikan pendidikan dan pelatihan bagi anggotanya, para wakil yang dipilih, karyawan dan sebagainya. Sehingga dapat memberikan sumbangan yang efektif bagi perkembangan koperasi, memberikan informasi kepada masyarakat umum mengenai sifat dan manfaat kerja sama.

f. Kerjasama Diantara Koperasi

Koperasi akan dapat memberikan pelayanan paling efektif kepada para anggota dan memperkuat gerakan koperasi dengan cara kerja sama melalui struktur-struktur lokal, nasional, regional dan internasional.