BAB II
TINJAUAN PUSTAKA

2.1.Tinjauan Perusahaan

2.1.1 Sejarah SMA N 1 Pabuaran Subang

Pendirian SMAN 1 Pabuaran Subang diawali dengan penugasan Kepala Dinas Pendidikan Jawa Barat agar merintis pembukaaan sekolah di desa tertingal. Kemudian pada tanggal 8 April 1982, Drs. Tarma Sutha dan Oji Setiaji diberi tugas untuk menghubungi kepala Kakandep Dikbudcam Subang dan sekaligus minta izin kepada Kepala SMP Negeri 1 Pabuaran untuk meminjam tempat pendaftaran calon siswa baru.

2.1.2 Tugas Pokok SMA N 1 Pabuaran Subang

1. Mendidik siswa agar mampu mengembangkan diri secara optimal dengan memanfaatkan kelebihan diri serta memperbaiki kekurangannya.
2. Meningkatkan kecerdasan, pengetahuan, kepribadian, keterampilan serta akhlak mulia siswa agar dapat hidup mandiri dan mengikuti pendidikan lebih lanjut.

2.1.3 Visi dan Misi SMA N 1 Pabuaran Subang

2.1.3.1 Visi

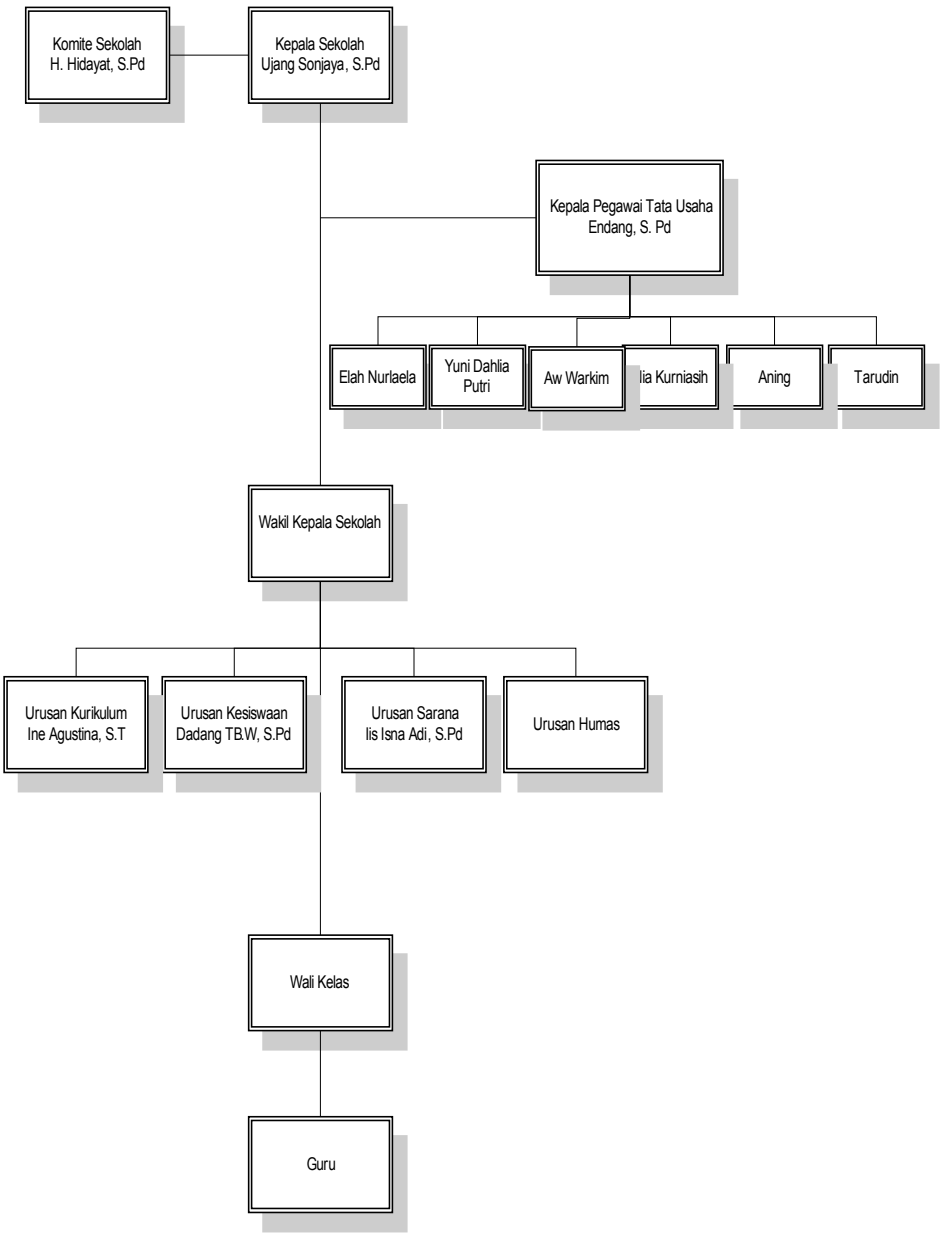
Menjadi salah satu SMA yang unggul dalam prestasi, beriman, bertakwa, bertanggung jawab, berkepribadian dan perduli terhadap lingkungan.

2.1.3.2 Misi

Adapun misi dari instansi adalah :

1. Membina siswa menjadi manusia yang berilmu, berakhlak terpuji, berdisiplin tinggi dan berkepribadian kuat.
2. Menciptakan suasana sekolah yang kondusif sehingga siswa dapat mengembangkan potensi dirinya secara optimal.
3. Menumbuhkan semangat keunggulan secara intensif kepada seluruh warga sekolah.
4. Meningkatkan kualitas proses pembelajaran yang optimal.
5. Menerapkan manajemen partisipatif dengan melibatkan seluruh warga sekolah dan komite sekolah.
6. Membina hubungan lintas sektoral dengan berbagai institusi untuk lebih meningkatkan kinerja dan program sekolah.

2.1.4 Struktur Organisasi



Gambar 2.1 Struktur Organisasi di SMAN 1 Pabuaran Subang

2.2.Landasan Teori

Dalam sub bab ini akan dibahas mengenai teori-teori yang berhubungan dengan pembangunan sistem absensi pada SMA N 1 Pabuaran Subang.

2.2.1.Pengertian Sistem Informasi

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

Dari pengertian diatas dapat diambil suatu kesimpulan bahwa suatu sistem merupakan elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

2.2.2.Karakteristik Sistem

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu memiliki komponen-komponen *(components)*, batas sistem *(boundary)*, lingkungan luar sistem *(environment)*, penghubung *(interprest)*, masukan *(input)*, keluaran *(output)*, pengolah *(process)* dan sasaran *(objective)* dan tujuan *(goal)*.

1. Komponen Sistem (*System Components*)

   Komponen sistem atau elemen-elemen sistem dapat berupa suatu sub sistem atau bagian-bagian dari sistem. Setiap sistem baik besar

maupun kecil, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu yang lebih besar yang disebut *supra system*.

2. Batas Sistem (*System Boundary*)

Batas sistem merupakan daerah-daerah yang membatasi antara satu sistem dengan sistem lainnya dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menujukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*System Environment*)

Lingkungan luar sistem dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem yang dapat bersifat menguntungkan dan dapat pula merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus dijaga dan dipelihara. Sedangkan yang merugikan harus ditahan dan dikendalikan, karena akan mengganggu kelangsungan hidup sistem.

4. Penghubung Sistem *(System Interprest)*

Penghubung sistem merupakan media penghubung antara satu sub sistem dengan subsistem yang lainnya. Melalui penghubung ini

memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung akan terjadi interaksi antar subsistem, sehingga membentuk satu kesatuan.

5. Masukan Sistem *(System Input)*

Masukan adalah suatu energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Contoh *maintenance input* di dalam sistem komputer adalah program, yang digunakan untuk mengoperasikan komputer. Sedangkan *signal input* adalah energi yang diproses untuk mendapatkan keluaran. Contoh *signal input* di dalam sistem komputer adalah data, yang dapat diolah menjadi Informasi.

6. Keluaran Sistem *(System Output)*

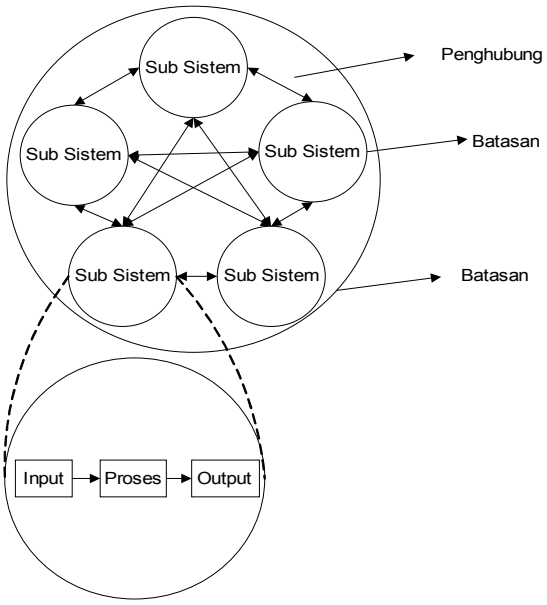
Keluaran (*Output*) merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

7. Pengolah Sistem (*System Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya, yang bertugas untuk merubah masukan menjadi keluaran.

8. Sasaran Sistem (*System Objective*)

Suatu sistem pasti memiliki tujuan (*goal*) atau sasaran (*objective*). Suatu operasi sistem akan berguna dan berhasil apabila mencapai sasaran atau tujuannya. Sasaran sistem sangat menentukan masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.



Gambar 2.2 Karakteristik sistem

2.2.3.Konsep Dasar Data dan Informasi

2.2.3.1Pengertian Data

Data adalah kumpulan dari fakta-fakta, kejadian-kejadian yang dapat berupa simbol, angka, huruf, dan lain-lain yang berguna bagi suatu pengolahan data (process) atau sebagai masukan (*input*) bagi suatu proses.

2.2.3.2Pengertian Informasi

Informasi merupakan hasil pengolahan dari sebuah model, formasi, organisasi, ataupun suatu perubahan bentuk dari data yang memiliki nilai tertentu, dan bisa digunakan untuk menambah pengetahuan bagi yang menerimanya. Dalam hal ini, data bisa dianggap sebagai obyek dan informasi adalah suatu subyek yang bermanfaat bagi penerimanya. Informasi juga bisa disebut sebagai hasil pengolahan atau pemrosesan data.

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata, tetapi data merupakan bentuk yang masih mentah dan belum dapat memberikan arti banyak bagi pemakai, sehingga perlu diolah lebih lanjut untuk menghasilkan informasi yang dibutuhkan. Dengan kata lain informasi adalah hasil dari pengolahan data.

2.2.3.3Kualitas Informasi

Kualitas dari suatu informasi tergantung pada tiga hal pokok yaitu:

a. Akurat (*accurate*)

   Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan, dalam hal ini informasi harus jelas mencerminkan maksudnya.

b. Tepat waktu (*time lines*)

Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi karena informasi merupakan suatu landasan dalam mengambil sebuah keputusan dimana bila pengambilan keputusan terlambat maka akan berakibat fatal untuk organisasi.

c. Relevan (*relevance*)

Informasi harus mempunyai manfaat untuk pemakainya, dimana relevansi informasi untuk tiap-tiap individu berbeda tergantung pada yang menerima dan yang membutuhkan.

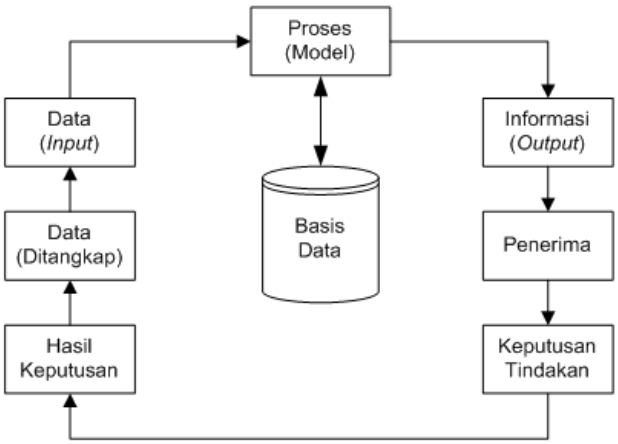
2.2.3.4Nilai Informasi

Nilai Informasi (*value of information*) ditentukan oleh dua hal yaitu manfaat dan biaya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efekif dibandingkan dengan biaya mendapatkannya.

2.2.3.5Siklus Informasi

Data agar menjadi lebih berarti dan berguna dalam bentuk Informasi, maka perlu diolah melalui suatu model tertentu. Data yang telah diolah tersebut kemudian diterima oleh penerima, lalu penerima membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai

input, dan diproses kembali lewat suatu model dan seterusnya sehingga membentuk suatu siklus. Siklus ini disebut dengan siklus Informasi (*information cycle*) atau disebut pula siklus pengolahan data (*processing cycles*).



Gambar 2.3 Siklus informasi

2.2.4. 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.

2.2.4.1Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building blok*) yaitu [1]:

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.

2.2.4.2Tujuan 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.

2.2.4.3Manfaat 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

2.2.5. Konsep Rekayasa Perangkat Lunak

2.2.5.1Pengertian Rekayasa 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 :

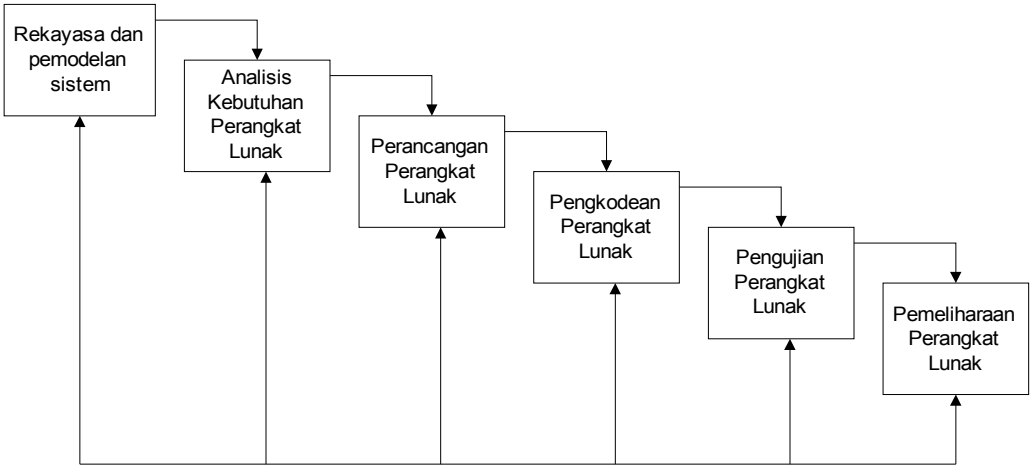
1. Model Sekuensial Linear

2. Model Prototipe
3. Model RAD
4. Model Proses Perangkat Lunak Evolusioner

2.2.5.2Model 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 2.4 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.

2.2.5.3Model Prototipe (Prototype Models)

Merupakan suatu teknik untuk mengumpulkan informasi tertentu mengenai kebutuhan-kebutuhan informasi pengguna secara cepat. Tahapan yang dilakukan :

1. Reaksi awal dari pengguna, diawali dengan menampilkan sebuah protipe sistem informasi, kemudian melihat reaksi dari pengguna saat bekerja dengan prototipe apakah fitur-fitur sistem pada prototipe tersebut sudah sesuai dengan kebutuhannya. Reaksi tersebut dikumpulkan dalam lembar observasi, wawancara dan kuesioner.
2. Saran-saran pengguna, saran-saran merupakan hasil interaksi pengguna dengan prototipe yang ditampilkan (evaluasi pengguna) yang merupakan masukan untuk perbaikan, pengubahan atau ‘menghentikan’ prototipe sehingga dapat memenuhi kebutuhan pengguan dengan lebih baik.

3. Inovasi, adalah kemampuan-kemampuan sistem baru yang sebelumnya tidak ada pada saat pengguna berinteraksi dengan prototipe. Inovasi prototipe jika berhasil akan menjadi bagian dari sistem hasil jadi.
4. Rencana revisi, prototipe menggambarkan sistem di masa datang. Rencana revisi membantu mengidentifikasikan prioritas-prioritas apa saja yang akan diprototipekan selanjutnya.

2.2.5.4Model RAD (Rapid Application Development)

Rapid Application 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 dimana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen. RAD menekankan perkembangan komponen program yang bisa digunakan kembali (reusabilitas).

1. Pemodelan Bisnis (Business Modelling)

Aliran informasi di antara fungsi-fungsi bisnis dimodelkan untuk mengetahui informasi yang mengendalikan proses bisnis, informasi yang dimunculkan, pelaku yang memunculkan informasi, tujuan informasi, dan siapa saja yang memproses informasi tersebut.

2. Pemodelan Data (Data Modelling)

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

3. Pemodelan Proses (Process Modelling)

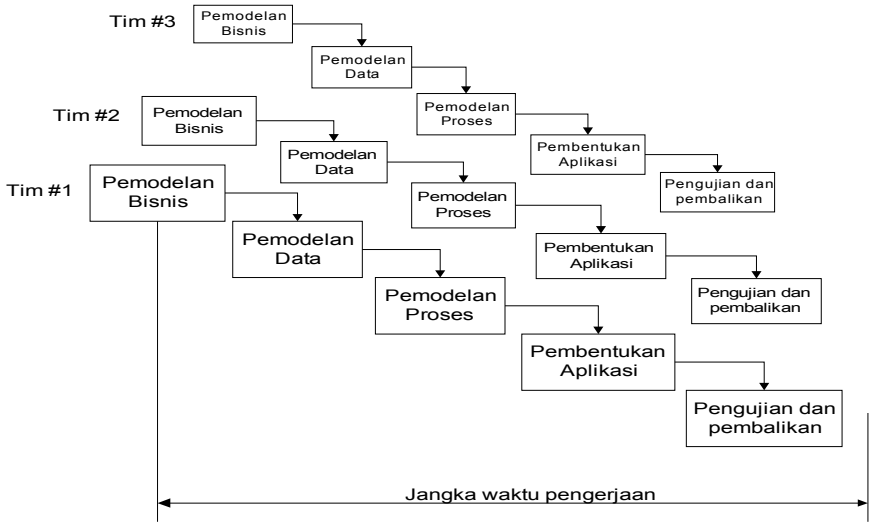
Aliran informasi yang didefinisikan di dalam fase pemodelan data ditransformasikan untuk mencapai aliran informasi yang perlu bagi implementasi sebuah fungsi bisnis. Gambaran pemrosesan diciptakan untuk menambah, memodifikasi, menghapus, atau mendapatkan kembali sebuah objek data.

4. Penggunaan Generasi (Application Generation)

RAD mengasumsikan pemakaian teknik generasi keempat. Selain menciptakan perangkat lunak dengan menggunakan bahasa pemrograman generasi ketiga yang konvensional, RAD lebih banyak memproses kerja untuk memakai lagi komponen program yang ada (pada saat memungkinkan) atau menciptakan komponen yang bisa digunakan lagi (bila perlu).

5. Pengujian dan Pembalikan (Testing and Turnover)

Karena proses RAD menekankan pada pemakaian kembali, banyak komponen program telah diuji. Hal ini mengurangi keseluruhan waktu pengujian. Tetapi komponen baru harus diuji dan semua antarmuka harus dilatih secara penuh. RAD digambarkan sebagai berikut :



Gambar 2.5 Model RAD

2.2.5.5Model Proses Perangkat Lunak Evolusioner

Merupakan model iteratif, yang ditandai dengan tingkah laku yang memungkinkan perekayasa perangkat lunak mengembangkan versi perangkat lunak yang lebih lengkap sedikit demi sedikit.

2.2.6. 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).

2.2.6.1Diagram 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.

2.2.6.2Diagram Alir 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 [5]:

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 (*proccess*)

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*.

2.2.7. 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.

2.2.7.1Konsep 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).

2.2.7.2Basis Data 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).

2.2.7.3DDL (Data Definition 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	Untuk menghapus tabel dan database

2.2.7.4DML (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

2.2.7.5Perancangan Basis Data

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

2.2.7.6Entity Relationship Diagram (ERD)

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

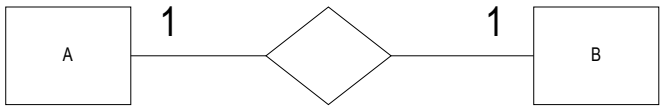
   Seacara umum atribut adalah sifat atau karakteristik dari tiap entitas maupun tiap relationship. Maksudnya adalah sesutau yang menjelaskan apa sebenarnya yang dimaksud entitas maupun relationship, sehingga sering dikatakan bahwa atribut adalah elemen dari setiap entitas dan relationship.

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

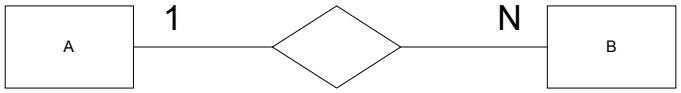
Tingkat hubungann satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.



Gambar 2.6 *One to One Relationship*

b. One to many Relationship

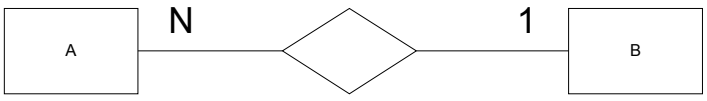
Tingkat hubungan satu ke banyak adalah untuk satu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua.



Gambar 2.7 *One to Many Relationship*

c. Many To One Relationship

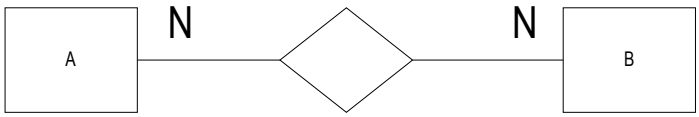
Untuk banyak kejadian pada entitas yang pertama hanya dapat mempunyai satu hubungan dengan kejadian pada entitas yang kedua.



Gambar 2.8 *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 2.9 *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).

2.2.7.7Kamus 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.

2.2.7.8Database Management System (DBMS)

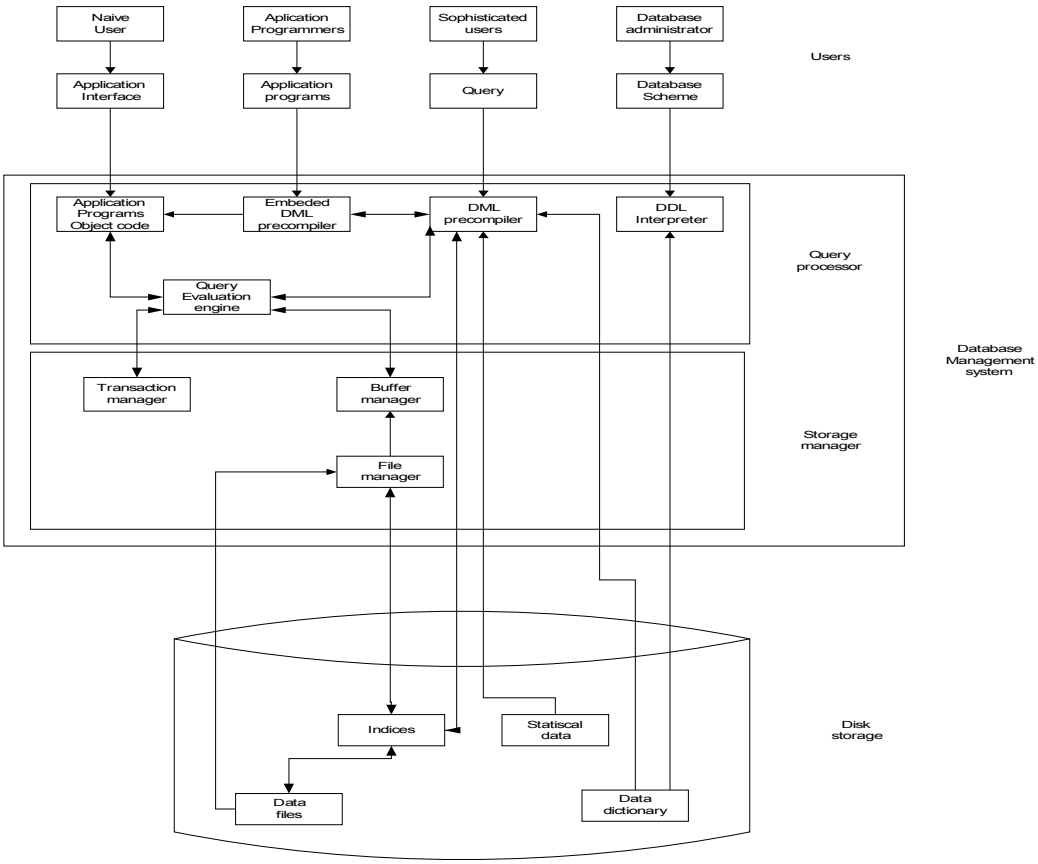
Database adalah kumpulan data yang saling berkaitan, berhubungan yang disimpan secara bersama-sama sedemikian rupa tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan. Data-data ini harus mengandung semua Informasi untuk mendukung semua kebutuhan sistem.

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 2.10 Struktur Sistem

2.2.8. 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”.

2.2.8.1Jenis-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.*

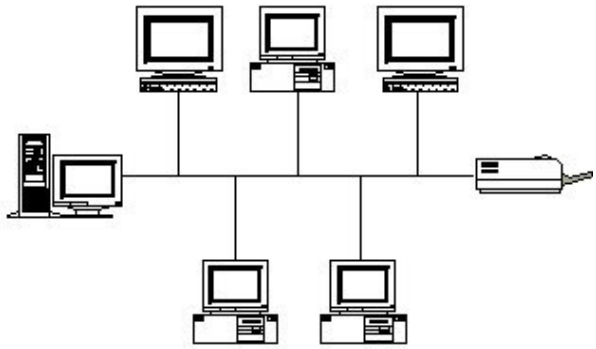
2.2.8.2Topologi Jaringan Komputer

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.

2.2.8.3Topologi 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 2.11 Topologi bus

Keuntungan

a. Hemat kabel
b. Layout kabel sederhana
c. Mudah dikembangkan

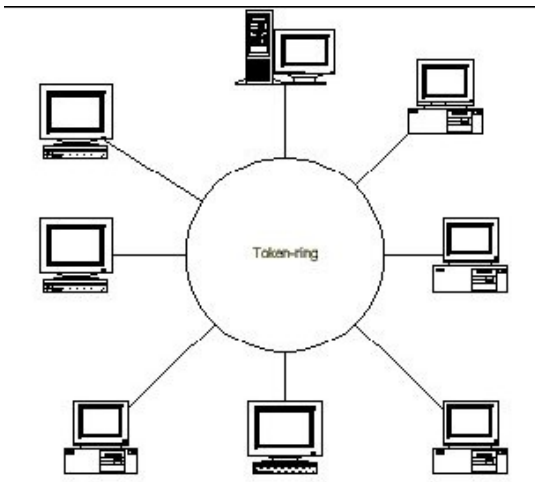
Kerugian

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

2.2.8.4Topologi 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 2.12 Topologi token ring

Keuntungan

a. Hemat Kabel

Kerugian

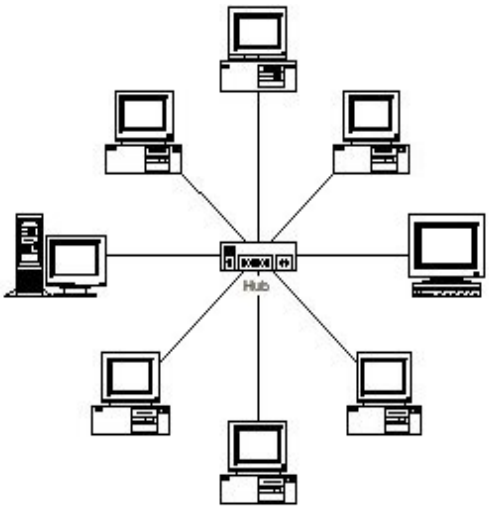
a. Peka kesalahan

b. Pengembangan jaringan lebih kaku

2.2.8.5Topologi 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 2.13 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 • Kemudahaan pengelolaan jaringan

Kerugian

a. Boros kabel
b. Perlu penanganan khusus

c. Kontrol terpusat (HUB) jadi elemen kritis

2.2.8.6Client-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.

2.2.8.7Manfaat Jaringan Komputer

1. 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.

2. 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.

3. Menghemat uang.

4. Komputer berukuran 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.

2.2.8.8Media Transmisi

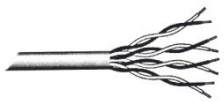
Kabel adalah penghubung untuk mengirim informasi dari satu computer ke computer yang lain. Ada beberapa macam tipe kabel yang umumnya digunakan pada LAN. Dalam beberapa kasus, sebuah jaringan hanya menggunakan satu macam tipe kabel, sedangkan di jaringan yang lain menggunakan kabel yang berbeda. Kabel 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

2.2.8.9Kabel Unshielded Twisted Pair (UTP)

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 2.14 Kabel *Unshielded Twisted Pair*

Jenis konektor untuk kabel jenis ini adalah RJ-45.



Gambar 2.15 *Konektor RJ-45*

2.2.8.10Kabel Shielded Twisted Pair (STP)

Kekurangan kabel jenis ii 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.

2.2.8.11Kabel Coaxsial

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.

2.2.8.12Kabel Fiber Optik

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



Gambar 2.16 Kabel fiber optik

2.2.9. Perangkat Lunak Pendukung

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

2.2.9.1MySQL

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 dipergunakna 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

2.2.9.2Borland Delphi 7

Borland delphi merupakan suatu bahasa pemrograman yang memberikan berbagai fasilitas pembuatan aplikasi visual. Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik serta diperkuat dengan pemrograman yang terstruktur. Keunggulan lain delphi adalah dapat dipergunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis windows.

1. Komponen Delphi

Component palette terdiri dari beberapa komponen yang dapat dipilih yang digunakan untuk menangani beberapa tugas pemrograman. Komponen-komponen yang terletak pada bagian *component palette* sudah ditata dalam beberapa tab yang masing-masing menunjukan maksud dan fungsi. Masing-masing tab ditampilkan dalam konfigurasi default yang semua juga tergantung pada versi delphi yang digunakan.

Tabel berikut menunjukan daftar tab default dan beberapa komponen yang terdapat di dalamnya.

Tabel 2.3 Tabel komponen delphi

Nama Tab	Isi
Standart	Kontrol-kontrol standar program windows dan menu
Additional	Kontrol-kontrol tambahan
Win32	Kontrol-kontrol umum windows 9x/NT 4.0
System	Komponen dan kontrol-kontrol dari sistem komputer termasuk *timer*, multimedia dan DDE
Data Access	Komponen-komponen non-visual yang digunakan untuk mengakses tabel-tabel *database*, *query*, dan *report*
Data Controls	Komponen-komponen visual, dan kontrol-kontrol data-*aware*
dbExpress	Komponen-komponen non-visual yang digunakan aplikasi untuk berhubungan dengan *database* dengan menggunakan *dbExpress*
DataSnap	Komponen dan kontrol-kontrol non-visual yang digunakan untuk membuat aplikasi *database* bertingkat (*multi-tiered*)
BDE	Komponen dan kontrol-kontrol non-visual yang digunakan untuk menghubungkan Informasi *database* dengan menggunakan *Borland Database Engine* (BDE)
ADO	Komponen dan kontrol-kontrol non-visual yang digunakan untuk menghubungkan Informasi *database* dengan menggunakan *ActiveX Data Object* (ADO)
InterBase	Komponen dan kontrol-kontrol non-visual yang digunakan untuk menghubungkan secara langsung *database interbase* tanpa menggunakan BDE ataupun ADO

Nama Tab	Isi
InternetExpress	Komponen yang digunakan untuk membangun aplikasi *InternetExspress* yang simultan dengan *Web Server* dan klien dari suatu aplikasi *database* bertingkat

2. Fitur Pada Delphi 7

Fitur baru dan perbaikan yang ada pada Borland Delphi 7 ini adalah :

a. IDE (Interface Development Environtment)

Lingkungan pengembangan aplikasi (IDE) Borland Delphi 7 telah mengalami perubahan dari versi sebelumnya. Diantaranya, terdapat Compiler Message, perubahan pada Component Pallete, Code Insight dan Debugger

b. Web

Borland Delphi 7 menyediakan Intraweb buatan AtoZed Software, yang dapat digunakan untuk membuat aplikasi *web server* dengan sarana standar. Borland Delphi 7 juga mendukung pada Apache 2. Borland juga menghilangkan Win-CGI sebagai target aplikasi Web server dan web service. Fasilitas untuk *Web server* juga mengalami perbaikan - perbaikan

c. COM

Sekarang ini dengan Delphi 7.0, dapat membuat CoClass wrapper bagi pengembangan – pengembangan .NET dengan cara menggunakan kotak dialog *Import Type Library*. Dengan adanya *resulting wrapper*, maka fitur interoperabilitas dari Microsoft's .NET Framework dapat digunakan.

d. Database

Pada Delphi 7.0 driver dbExpress telah diupdate bagi Informix SE, Oracle 9i, DB2 7.2, InterBase 6.5, dan MySQL 3.23.49. Driver baru bagi MSSQL 2000 juga tersedia. Disamping itu, beberapa hal baru dan perubahan juga dilakukan pada komponen *database*. Borland juga telah membuang SQL Links. Borland merekomendasikan pemakaian dbExpress bagi database SQL Server yang diakses di Delphi.

e. Component Library

Jika ditelusuri komponen librari Delphi 7.0, maka akan ditemukan komponen baru, unit baru, komponen yang berubah, komponen yang hilang dan komponen yang mendukung bagi tema Windows XP.

f. Runtime Library

Beberapa perubahan di Runtime Library antara lain ialah perubahan pada unit Classes, Math, StdConv,StrUtils, SysUtils, VarCmplx, dan Variants.

g. Compiler

Kompiler Delphi dcc32 sekarang ini support terhadap tiga warning kompiler tambahan, yaitu *Unsafe_Type*, *Unsafe_Code*, dan *Unsafe_Cast*. Warnings tersebut defaultnya adalah *disabled*, tetapi dapat di-*enabled*. Fitur ini sangat membantu kita ketika akan mem-port kode ke lingkungan eksekusi terkendali di *platform* Microsoft's .NET.

h. Model Maker

Sarana baru yang disebut ModelMaker dapat membantu memudahkan proses desain, konstruksi, dan pengelolaan *class* dan *interface*. *Model Maker* juga memiliki sarana untuk pembuatan diagram UML-style, yang dapat dipakai untuk membuat dan memodifikasi *source code project*.