BAB II

TINJAUAN PUSTAKA

2.1 Ruang Lingkup Instansi

SDN Langensari Subang merupakan salah satu sekolah tingkat dasar yang terletak di Jalan Ki Hajar Dewantara No. 26 kelurahan Dangdeur kecamatan Subang kabupaten Subang. Berdiri pada tahun 1973 dengan Bapak Muslimin sebagai kepala sekolah untuk pertama kalinya. Berikut adalah periode pergantian kepala sekolah :

a. Bapak Muslimin menjabat kepala sekolah tahun 1973 – 1979
b. Bapak Adang Sadeli menjabat kepala sekolah tahun 1979 – 1984
c. Ibu Unengsih Menjabat kepala sekolah tahun 1984 – 1992
d. Bapak Didi Juardi menjabat kepala sekolah tahun 1992 – 1993
e. Bapak Eman Sulaeman menjabat kepala sekolah tahun 1993 – 1998
f. Ibu Hj. Juwitasari menjabat kepala sekolah tahun 1998 sampai sekarang.

SDN Langensari menempati tanah milik negara dengan luas tanah 2600 m^2 yang penggunaanya dikhususkan untuk pendidikan, SDN Langensari mempunyai jumlah bangunan 3 unit, jumlah bangunan RDK 1 unit, jumlah bangunan RDG 1 unit, jumlah bangunan RDP 1 unit, jumlah WC 5 ruang, jumlah kantor 1 ruang, jumlah kelas 7 ruang, jumlah ruang UKS 1 ruang, jumlah ruang moshala 1 ruang, jumlah ruang multimedia 1 ruang, warung sekolah 1 unit.

Adapun visi, misi dan strategi SDN Langensari adalah sebagai berikut :

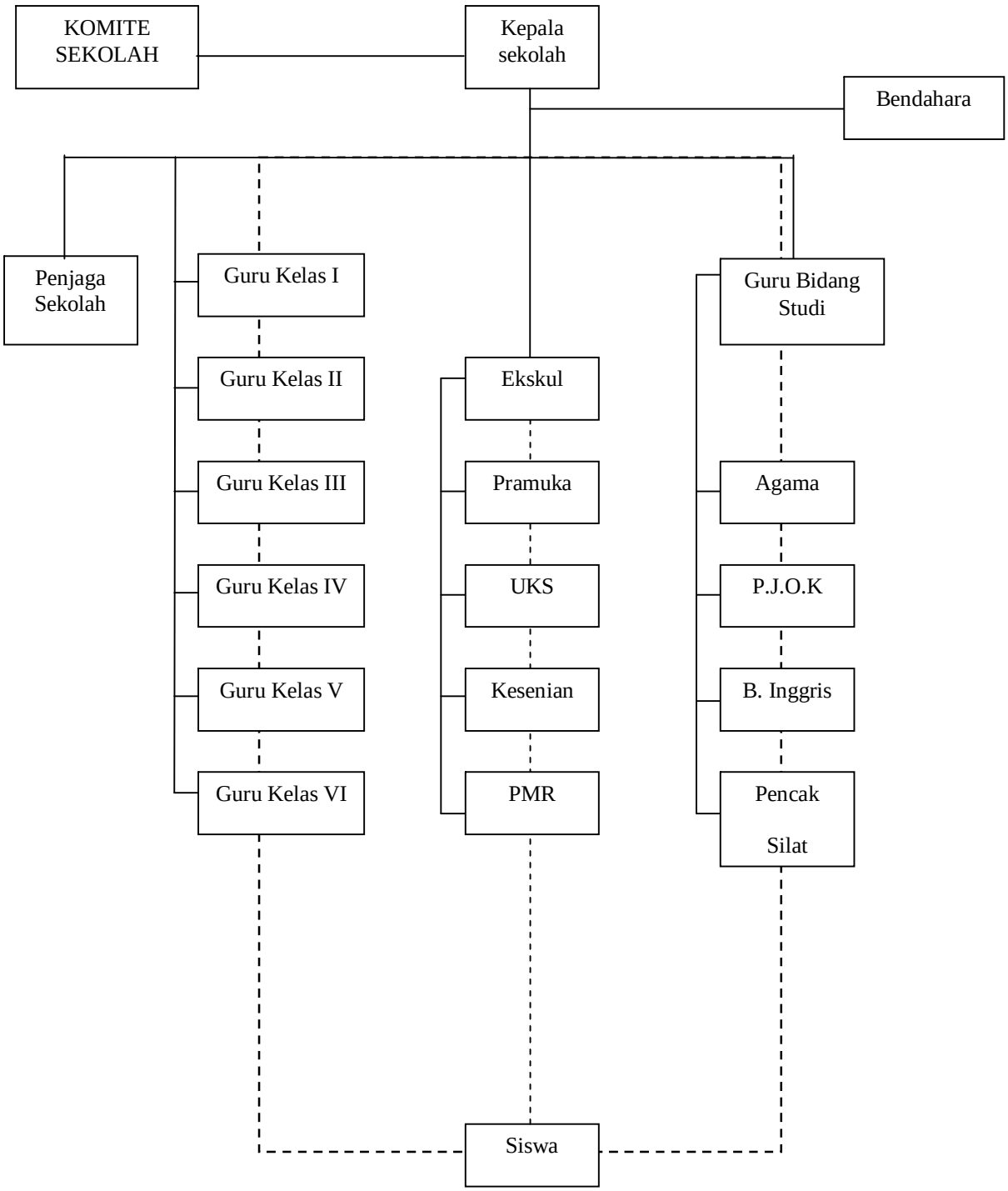
1. Visi sekolah

   Menjadi sekolah yang beriman, cerdas dan berprestasi di kabupaten Subang pada tahun 2011

2. Misi sekolah
   a. Meningkatkan keimanan dan ketaqwaan peserta didik kepada tuhan yang maha esa
   b. Mengembangkan kecerdasan dan keterampilan peserta didik
   c. Menanamkan rasa kebanggan terhadap prestasi dalam kehidupan sehari-hari
   d. Menumbuhkan sikap cinta kebersihan.
3. Strategi sekolah
   a. Melakukan Bersih lingkungan kelas, 5 menit sebelum masuk (K3)
   b. Cuci tangan sebelum belajar
   c. Melakukan sikat gigi bersama setiap hari jum’at
   d. Memelihara keindahan lingkungan.

2.1.1 Struktur Organisaasi

Untuk lebih jelasnya akan digambarkan skema struktur organisasi di SDN Langensari Subang, dapat dilihat pada Gambar 2.1



Gambar 2.1 Struktur Organisasi SDN Langensari Subang

1. Tugas komite sekolah adalah memantau dan menilai kemajuan sekolah
2. Tugas Kepala Sekolah
    a. Sebagai penanggung jawab pelaksanaan pendidikan di sekolah termasuk dalam pelaksanaan administrasi sekolah.
    b. Mengatur kegiatan sekolah dengan mengeluarkan atau memberikan intruksi terhadap kemajuan sekolah
    c. Penyusunan program kegiatan sesuai dengan kalender pendidikan
3. Tugas Bendahara adalah mengatur pemasukan pengeluaran kas sekolah dan semua yang berhubungan dengan masalah keuangan sekolah.
4. Tugas Guru bertanggung jawab kepada kepala sekolah dan mempunyai tugas melaksanakan proses belajar mengajar secara efektif dan efisien.
5. Tugas Guru Bidang Studi (Mulok) bertugas bertanggung jawab kepada kepala sekolah dan mempunyai tugas melaksanakan proses belajar mengajar sesuai dengan bidang studinya masing-masing.
6. Tugas penjaga sekolah adalah menjaga keamanan sekolah.

2.2 Landasan Teori

Dalam sub bab ini akan dibahas tentang teori-teori yang berhubungan dengan pembangunan sistem informasi akademik.

2.2.1 Pengertian Sistem

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)*

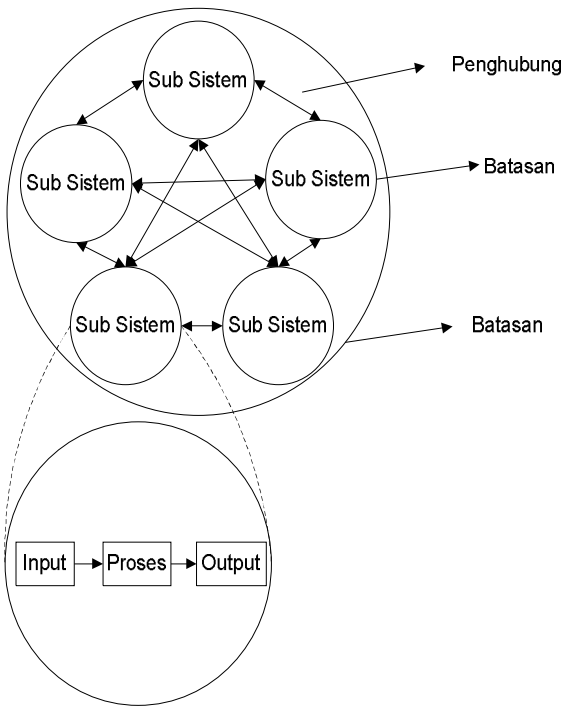
Keluaran (*Output*) merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

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, seperti terlihat pada Gambar 2.2.



Gambar 2.2 Karakteristik sistem

2.2.3 Konsep Dasar Data dan Informasi

2.2.3.1 Pengertian 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.

2.2.3.2 Pengertian 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.

2.2.3.3 Kualitas Informasi

Kualitas dari suatu informasi tergantung pada lima 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*)

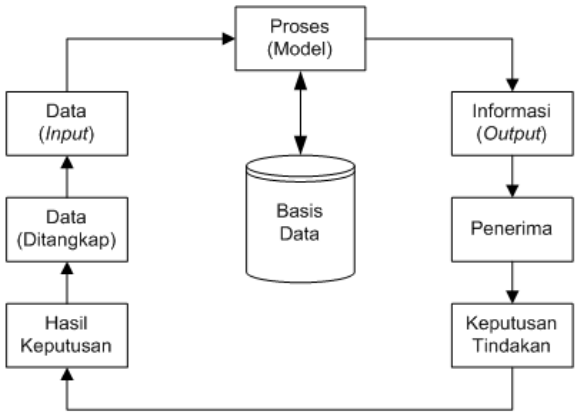
   Informasi harus mempunyai manfaat untuk pemakainya, dimana relevansi informasi untuk tiap-tiap individu berbeda tergantung pada yang menerima dan yang membutuhkan.

2.2.3.4 Nilai 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.

2.2.3.5 Siklus 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*), seperti terlihat pada Gambar 2.3.



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.

2.2.4.1 Komponen 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.

2.2.4.2 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.

2.2.4.3 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

2.2.5 Konsep Rekayasa Perangkat Lunak

2.2.5.1 Pengertian 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.

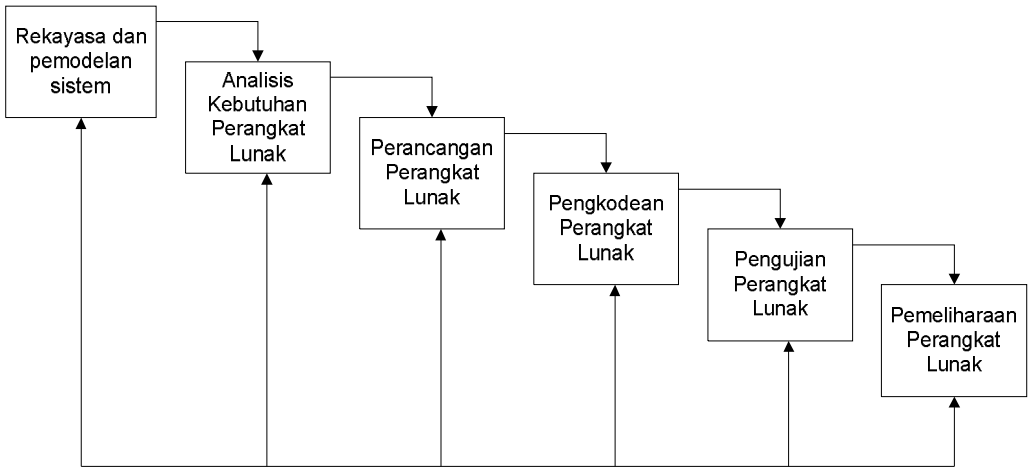
Untuk menyelesaikan masalah aktual di dalam sebuah setting industri, rekayasa perangkat lunak atau tim perekayasa harus menggabungkan strategi pengembangan yang melingkupi lapisan proses, metode, dan alat-alat bantu serta fase-fase generik. Strategi ini sering diacukan sebagai model proses atau paradigma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

2.2.5.2 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 seperti terlihat pada Gambar 2.4 **[4]**.



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.

2.2.6 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).

2.2.6.1 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.

2.2.6.2 Diagram 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 **[4]:**

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.

2.2.7.1 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*) **[3]**.

2.2.7.2 Basis 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*).

2.2.7.3 DDL (*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 seperti dijelaskan pada Tabel 2.1

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*

2.2.7.4 DML (*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 seperti dijelaskan pada Tabel 2.2.

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

2.2.7.5 Perancangan Basis Data

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

2.2.7.6 *Entity 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

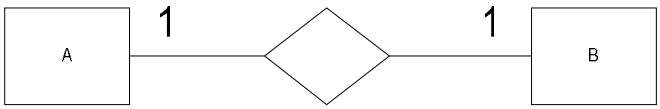
Seacara umum atribut adalah sifat atau karakteristik dari tiap entitas maupun tiap relationship. Maksudnya adalah sesutau yang menjelaskan apa sebenarnya yang dimaksud entitas maupun relationship, sehingga sering dikatakan bahwa atribut adalah elemen dari setiap entitas dan relationship.

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

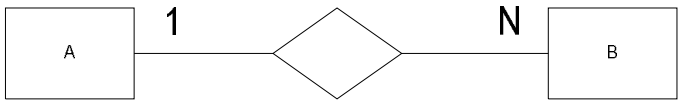
Tingkat hubungann satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya, seperti digambarkan pada Gambar 2.5.



Gambar 2.5 *One to One Relationship*

b. One to many Relationship

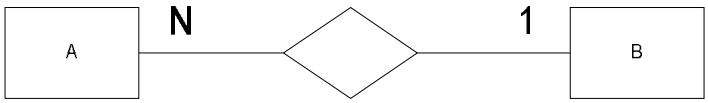
Tingkat hubungan satu ke banyak adalah untuk satu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua, seperti digambarkan pada Gambar 2.6.



Gambar 2.6 *One to Many Relationship*

c. Many To One Relationship

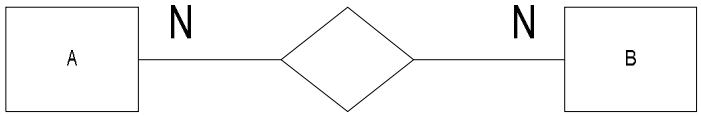
Untuk banyak kejadian pada entitas yang pertama hanya dapat mempunyai satu hubungan dengan kejadian pada entitas yang kedua seperti digambarkan pada Gambar 2.7.



Gambar 2.7 *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 seperti digambarkan pada Gambar 2.8



Gambar 2.8 *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).

2.2.7.7 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.

2.2.7.8 *Database Management System* (DBMS)

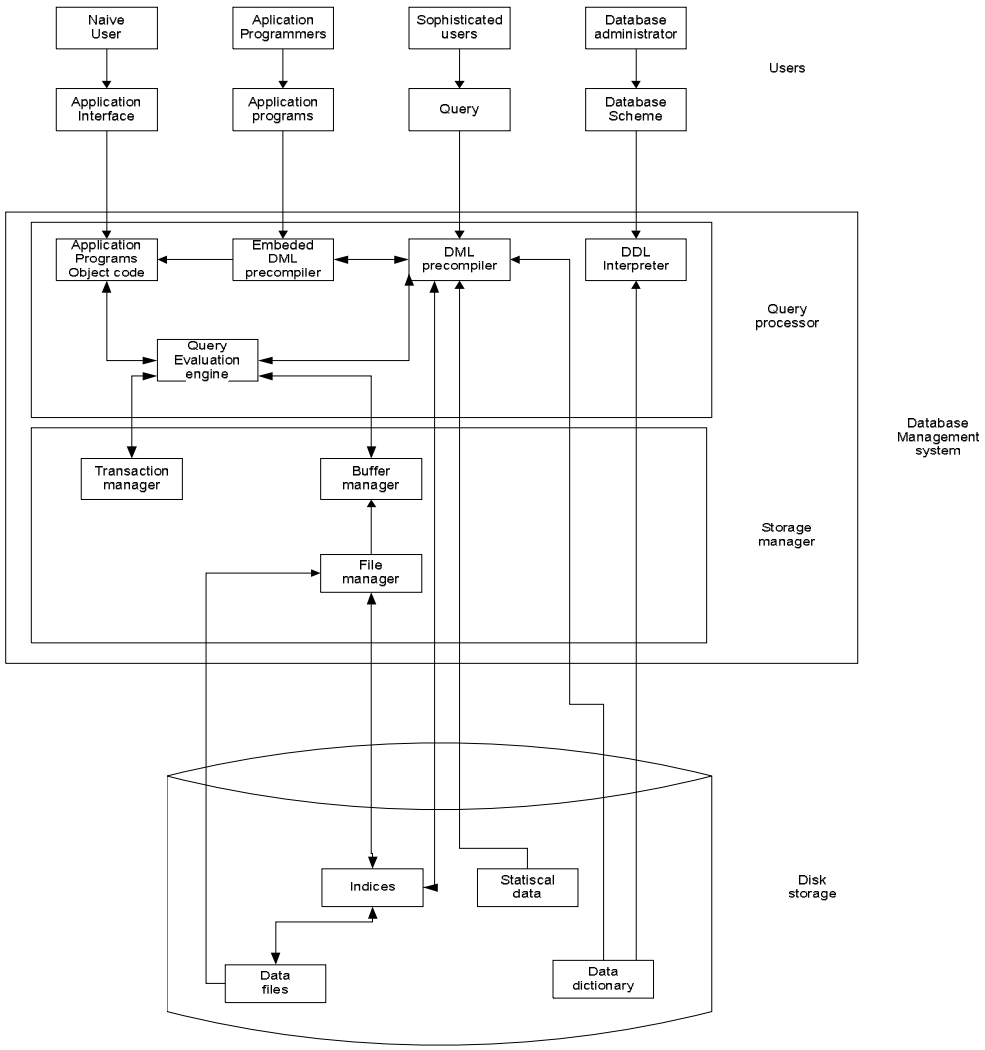
Database adalah kumpulan data yang saling berkaitan, berhubungan yang disimpan secara bersama-sama sedemikian rupa tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan. Data-data ini harus mengandung semua Informasi untuk mendukung semua kebutuhan sistem.

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, seperti terlihat pada Gambar 2.9



Gambar 2.9 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".

2.2.8.1 *Local Area Network* (LAN)

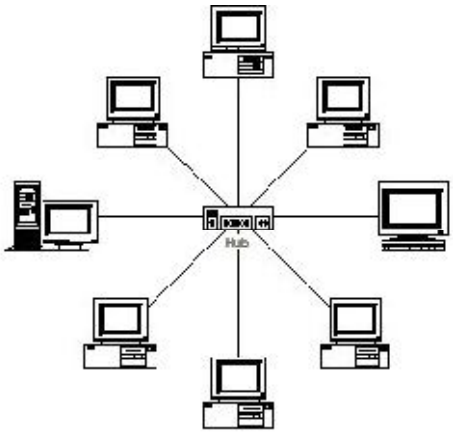
LAN (*Local Area Network*) digunakan untuk menghubungkan komputer yang berada di dalam suatu area yang kecil, misalnya di dalam suatu gedung perkantoran atau kampus. Jarak antara komputer yang dihubungkannya bisa mncapai 5 sampai 10 Km. Suatu LAN biasanya bekerja pada kecepatan mulai 10 Mbps sampai 100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

2.2.8.2 Topologi 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.

2.2.8.2.1 *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, seperti dijelaskan pada Gambar 2.10



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 • Kemudahaan pengelolaan jaringan

Kerugian

a. Boros kabel

b. Perlu penanganan khusus

c. Kontrol terpusat (HUB) jadi elemen kritis

2.2.8.3 *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.

2.2.8.4 Manfaat 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 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.

2.2.9 Perangkat Lunak Pendukung

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

2.2.9.1 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 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

2.2.9.2 Borland 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 **[2]**.

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

Nama Tab	Isi
	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
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. 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*.