BAB II

TINJAUAN PUATAKA

2.1 Profil Tempat Kerja Praktek

PT KAHAR DUTA SARANA adalah perusahaan multinasional yang bergerak di bidang *Retail IT Solution* yang melayani pra jual dan purna jual *point of sale* dan perangkat pendukung lainnya.

2.1.1 Sejarah Instansi

Kelompok usaha PT KAHAR DUTA SARANA (KDS) berdiri pada tanggal 16 Februari 1979 dengan badan usaha yang memasok alat tulis dan mesin-mesin kantor. Dengan memiliki 6 karyawan dan sebuah ruang kantor sederhana di daerah Senen Jakarta Pusat. PT Kahar Duta Sarana meniti tahun-tahun pertamanya dengan ketekunan dan kerja keras.

Dengan mutu dan layanan yang baik membuat PT Kahar Duta Sarana dipercaya memasok keperluan mesin kasir sebuah instansi pemerintah. Momentum ini tidak disia-siakan begitu saja. PT Kahar Duta Sarana telah berhasil menyakinkan sebuah Perusahaan Jepang yaitu Tokyo Electrik Corporation (TEC), untuk bermitra dan untuk pertama kalinya PT Kahar Duta Sarana menjadi *Sole Distributor* Mesin Kasir TEC. Dalam perkembangan selanjutnya, PT Kahar Duta Sarana dipercaya untuk memasarkan produk-produk TEC yang lain seperti: *Digital Scale, Point of Sale System,* dan lain-lain. Pada pertengahan tahun 1980 PT Kahar Duta Sarana lebih menegaskan kiprahnya yang berfokus pada penyelenggaraan *retail system solution* dengan memasuki bidang *Marketing System,* yaitu dengan memasarkan Barcode Printer dan memasuki era produksi

Marking Labels pada tahun 1990. Saat ini perkembangan pasarnya meliputi industrial Labels yang diantara pelanggannya termasuk pula *Multinational Corporation.* Produksi PT Kahar Duta Sarana telah memenuhi syarat kualitas dan mendapat sertifikasi dari UL (*Underwriting Laboratory*) USA. Dalam bidang pemasaran makin lengkap dengan ditunjuknya PT Kahar Duta Sarana sebagai distributor oleh PSC Inc. untuk produk *Laser Scanner* dan oleh Love shaw (UK) untuk produk *Large Character* Printer. Untuk saat ini, perkembangan yang paling akhir di kiprah PT Kahar Duta Sarana adalah telah berdirinya Divisi Niaga pada awal dekade 1990 yang mengayomi sejumlah Outlet Swalayan Fotografi di beberapa tempat di Jakarta, kemudian disusul dengan berdirinya Divisi-divisi lain seperti: Divisi CASIO yang memegang keagenan tunggal Digital Camera, Star *Mini Printer* dan *Personal Data Assitant* (PDA) Dalam memasuki dekade pertama abad 21, Indonesia dengan pasti akanberkembang menjadi pemain yang penting dalam ekonomi global. Kahar Duta Sarana akan terus mengambil peran aktif dan dinamis serta berkembang terus memenuhi tuntutan zaman.

2.1.2 Logo Instansi



Gambar 2.1 Logo Instansi

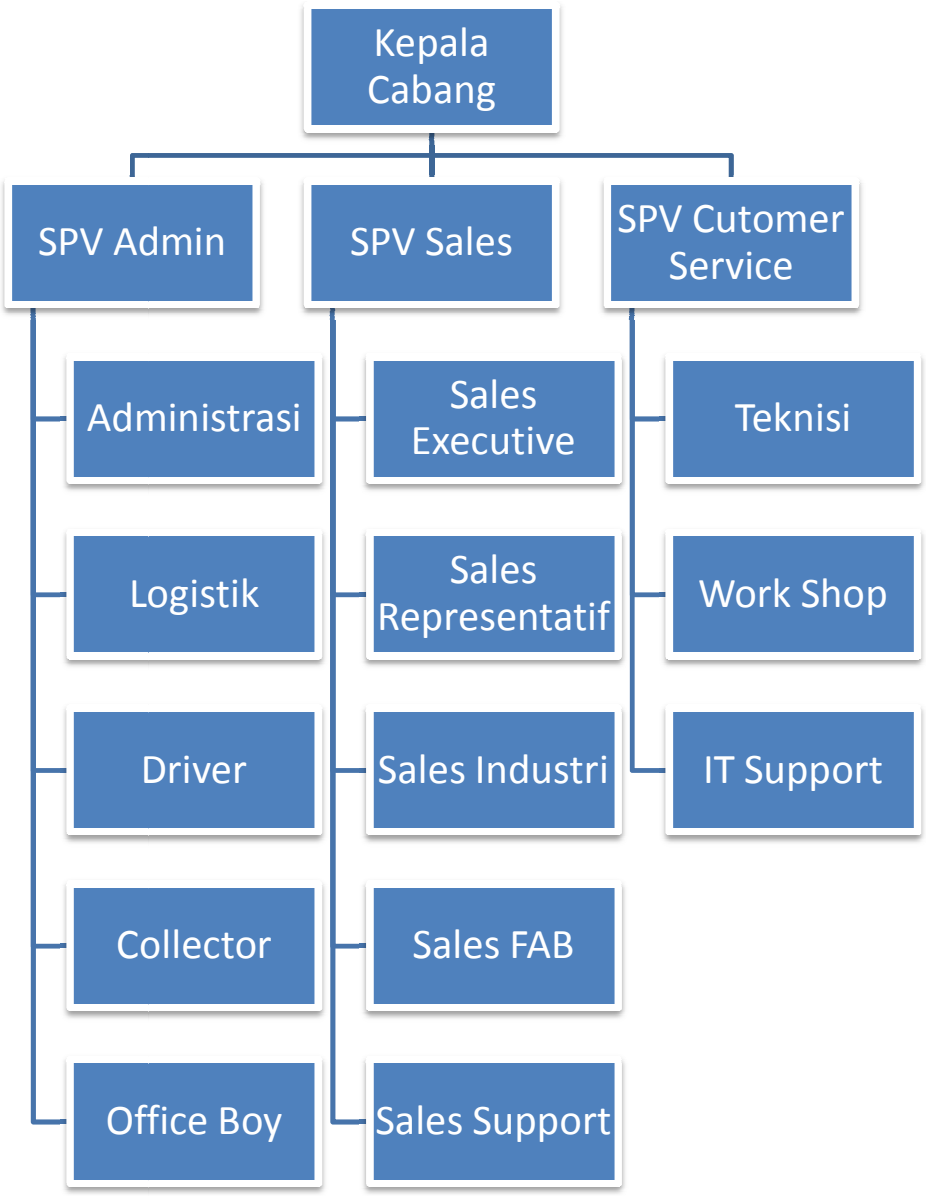
2.1.3 Badan Hukum Instansi

Berdasarkan akta notaris tanggal 16 Februari 1979 perusahaan ini disebut perseroan terbatas dengan nama PT. Kahar Duta Sarana.

Dengan kepengurusan sebagai berikut :

Chairman / CEO	: H. Koendjono Harjadi
President Director	: Frans B. Sungkono
Director	: Djoko Wirjono
Director	: Wayan Wandi
Director	: Kemal Toha
Director	: Jeffrey Ramsay

2.1.4 Struktur Organisasi



Gambar 2.2 Bagan Struktur Organisasi PT Kahar Duta Sarana Cabang Bandung

2.1.5 Visi, Misi dan Nilai Perusahaan

VISI

Menjadi perusahaan terkemuka di bidang "*Provider Retail Solution*" yang handal dan terpercaya dalam memberikan layanan kepada pelanggan baik layanan pra jual maupun purna jual dan mengutamakan kepuasan pelanggan.

MISI

- Memberikan pelayanan yang terbaik demi kepuasan pelanggan dengan kualitas pekerjaan dan integerasi sistem yang baik dan handal serta mengutamakan kecepatan, ketepatan, dan layanan purna jual yang handal dan tepat waktu
- Meningkatkan kemampuan dan mengoptimalkan sumber daya manusia yang baik, unggul dan dapat dipercaya
- Mengedepankan penggunaan teknologi yang tepat guna untuk mendapatkan hasil yang terbaik dan berkualitas
- Membangun kemitraan strategis dan bersinergi dengan klien maupun partner dengan prinsip saling menguntung

NILAI

- Kepuasan Pelanggan
- Profesional
- Komitmen
- Produk dan Jasa yang bernilai tinggi
- Integritas
- Kesejahteraan Karyawan

2.1.6 Portofolio

TAHUN	HISTORY
1979	KDS berawal dari bisnis penyuplai peralatan alat tulis dan perlangkapan perkantoran, dengan mutu dan layanan yang baik membuat PT KDS dipercaya memasok keperluan mesin kasir sebuah instansi pemerintah dan bekerjasama dengan Toshiba TEC.
1980	KDS mengembangkan tiga divisi lainya yaitu Divisi Pengembangan IT, Divisi Label dan Percetakan, dan Divisi identifikasi otomatis dan pengumpulan data (AIDC)
1990	Setelah memiliki citra yang baik di setiap divisi, KDS membangun kerjasama dengan Wincor Nixdorf, IBM, PSC, dan Casio.
2005	KDS bekerjasama dengan Bizerba, Metrologic, dan Microsoft.
2006	KDS bersertifikasi ISO 9001
2009	KDS bekerjasama dengan Hawlett Packard sebagai distributor tunggal mesin kasir HP di Indonesia.
2010	KDS mendirikan *Enterprise Solution Division* (ESD) untuk fokus pada *software enterprise* solution dan membangun kerjasama dengan Aldata.
2011	Membangun kerjasama dengan Checkpoint EAS dan Motorola Solution. KDS terus memimpin sebagai *provider* penyedia *Retail IT Solution* di Indonesia.

2.2 Landasan Teori

Landasan teori berisi teori-teori tentang ilmu yang mendasari tentang bidang yang sedang diteliti. Penyajian teori dalam landasan teori bersumber dari buku, artikel dan modul.

2.2.1 Sistem Informasi

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

2.2.1.1 Konsep Dasar Sistem

Suatu sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu. Dari definisi ini dapat dirinci lebih lanjut pengertian sistem secara umum, yaitu:

1. Setiap sistem terdiri dari unsur-unsur.
2. Unsur-unsur tersebut merupakan bagian terpadu sistem yang bersangkutan.
3. Unsur sistem tersebut bekerja sama untuk mencapai tujuan sistem.
4. Suatu sistem merupakan bagian dari sistem lain yang lebih besar.

2.2.1.2 Konsep Dasar Informasi

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan. Informasi merupakan data yang telah diklasifikasikan atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.

2.2.1.3 Konsep Dasar Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan.

Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

2.2.1.4 Komponen dan Elemen Sistem Informasi

A. Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari komponen *input*, komponen model, komponen output, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut

saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran.

1. Komponen *input*.

Input mewakili data yang masuk kedalam sistem informasi. *Input* disini termasuk metode dan media untuk menangkap data yang akan dimasukan, yang dapat berupa dokumen-dokumen dasar.

2. Komponen *model*

Komponen ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Komponen *outputt*

Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.

4. Komponen teknologi

Teknologi merupakan "*tool box*" dalam sistem informasi, teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

5. Komponen *hardware*

Hardware berperan penting sebagai suatu penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung *database* atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

6. Komponen *software*

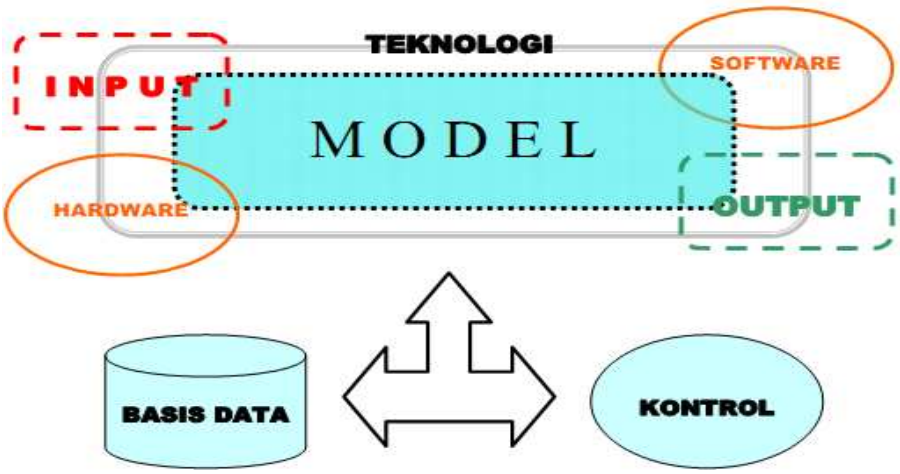
Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari *hardware* untuk menciptakan suatu informasi.

7. Komponen basis data

Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

8. Komponen kontrol

Banyak hal yang dapat merusak informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.



Gambar 2.3 Interaksi Sistem Informasi

B. Elemen sistem informasi

Sistem informasi terdiri dari elemen-elemen yang terdiri dari orang, prosedur, perangkat keras, perangkat lunak, basis data, jaringan komputer dan komunikasi data. Semua elemen ini merupakan komponen fisik.

1. Orang

   Orang atau personil yang dimaksudkan yaitu operator komputer, analis sistem, programmer, personil data entry, dan manajer sistem informasi/EDP

2. Prosedur

   Prosedur merupakan elemen fisik. Hal ini disebabkan karena prosedur disediakan dalam bentuk fisik seperti buku panduan dan instruksi. Ada 3 jenis prosedur yang dibutuhkan, yaitu instruksi untuk pemakai, instruksi untuk penyiapan masukan, instruksi pengoperasian untuk karyawan pusat komputer.

3. Perangkat keras

Perangkat keras bagi suatu sistem informasi terdiri atas computer (pusat pengolah, unit masukan/keluaran), peralatan penyiapan data dan terminal masukan/keluaran.

4. Perangkat lunak

   Perangkat lunak dapat dibagi dalam 3 jenis utama:

   a. Sistem perangkat lunak umum, seperti sistem pengoperasian dan sistem manajemen data yang memungkinkan pengoperasian sistem komputer.
   b. Aplikasi perangkat lunak umum, seperti model analisis dan keputusan.
   c. Aplikasi perangkat lunak yang terdiri atas program yang secara spesifik dibuat untuk setiap aplikasi.

5. Basis data

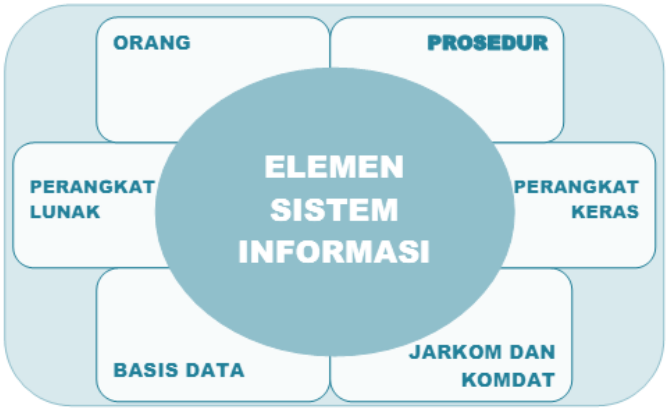
   File yang berisi program dan data dibuktikan dengan adanya media penyimpanan secara fisik seperti diskette, hard disk, magnetic tape dan sebagainya. *File* juga meliputi keluaran tercetak dan catatan lain diatas kertas, mikro film dan lain sebagainya.

6. Jaringan komputer

   Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan lainnya yang terhubung dalam satu kesatuan. Informasi dan data bergerak melalui kabel-kabel atau tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data.

7. Komunikasi data

Komunikasi data adalah merupakan bagian dari telekomunikasi yang secara khusus berkenaan dengan transmisi atau pemindahan data dan informasi diantara komputer-komputer dan piranti-piranti yang lain dalam bentuk digital yang dikirimkan melalui media komunikasi data. Data berarti informasi yang disajikan oleh isyarat digital. Komunikasi data merupakan bagian vital dari suatu sistem informasi karena sistem ini menyediakan infrastruktur yang memungkinkan komputer-komputer dapat berkomunikasi satu sama lain



Gambar 2.4 Hubungan Elemen Sistem Informasi

2.2.1.5 Arsitektur dan Klasifikasi Sistem Informasi

A. Arsitektur Sistem Informasi

Sistem informasi dapat dibentuk sesuai kebutuhan organisasi masing-masing. Oleh karena itu, untuk dapat menerapkan sistem yang efektif dan efisien diperlukan perencanaan, pelaksanaan, pengaturan dan evaluasi sesuai keinginan masing-masing organisasi. Guna dari sistem yang efektif dan efisien tidak lain

untuk mendapatkan keunggulan dalam berkompetisi. Semua orang dapat menggunakan sistem informasi dalam organisasi, tetapi faktor efisiensi setiap sistem adalah berbeda.

Perlu diketahui, perubahan sistem, baik besar maupun kecil, selalu akan melalui tingkatan-tingkatan sebagai berikut:

Tingkat I : Ide, mengetahui perlu adanya perubahan.

Tingkat II : *Design*, merancang cara pemecahannya.

Tingkat III : Pelaksanaan, menerapkan design kedalam sistem.

Tingkat IV : Kontrol, memeriksa tingkat pelaksanaan dijalankan sesuai dengan *design*

Tingkat V : Evaluasi, memeriksa apakah perubahan yang terjadi sesuai dengan tujuan semuala.

Tingkat VI : Tindak lanjut, melaksanakan perubahan sesuai dengan hasil evaluasi yang ada.

Adapun tingkatan yang menjadi kunci yang digunakan untuk memecahkan bagian masalah baik itu secara menyeluruh maupun per bagian, yaitu:



Gambar 2.5 Tingkatan kunci untuk memecahkan masalah

B. Klasifikasi Sistem Informasi

Sistem informasi merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap

kasus yang terjadi yang ada di dalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya :

a. Sistem abstrak atau sistem fisik

   Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu sistem yang berupa pemikiran hubungan antara manusia dengan tuhan.

b. Sistem alamiah dan sistem buatan manusia

   Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi.

   Sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin, yang disebut *human machine system*. Sistem informasi berbasis *internet* merupakan contoh *human machine system* karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

c. Sistem *determinist*ik dan sistem *probabilistik*

   Sistem *deteministik* adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi.

   Sistem *probabilistik* adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur *probabilistik*.

d. Sistem terbuka dan sistem tertutup

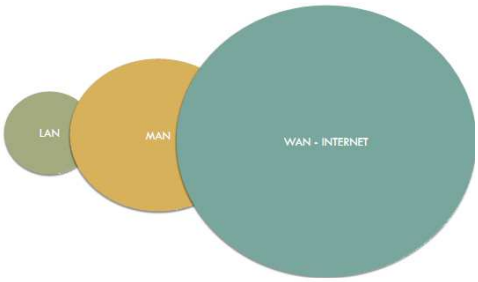
   Sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya.

   Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

Sistem tertutup adalah sistem yang tidak terhubung dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan pihak luar.

2.2.2 Jaringan Komputer

Saat ini, *internet* dan *world wide web* (www) sangat populer di seluruh dunia. Banyak masyarakat yang membutuhkan aplikasi yang berbasis *internet*, seperti *e-mail* dan akses *web* melalui *internet*. Sehingga makin banyak aplikasi bisnis yang berkembang berjalan di atas *internet*.



Gambar 2.6 Jaringan Komputer

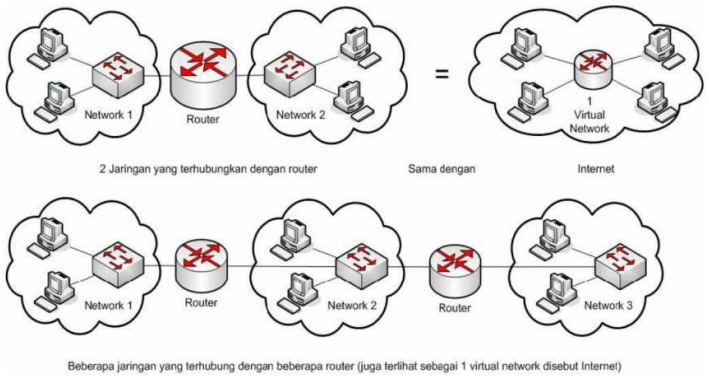
2.2.2.1 Model Arsitektur TCP/IP

TCP/IP (*Transmission Control Protocol/Internet Protocol*) adalah standar komunikasi data yang digunakan oleh komunitas *internet* dalam proses tukar-menukar data dari satu komputer ke komputer lain di dalam jaringan *Internet*. Protokol ini tidaklah dapat berdiri sendiri, karena memang protokol ini berupa kumpulan protokol. Protokol ini juga merupakan protokol yang paling banyak digunakan saat ini. Data tersebut diimplementasikan dalam bentuk perangkat lunak (*software*) di sistem operasi.

TCP/IP merngimplemenasikan arsitektur berlapis yang terdiri atas empat lapis. Empat lapis ini, dapat dipetakan (meski tidak secara langsung) terhadap model referensi OSI. Empat lapis ini, kadang-kadang disebut sebagai DARPA Model, *Internet* Model, atau DoD Model, mengingat TCP/IP merupakan protokol yang awalnya dikembangkan dari proyek ARPANET yang dimulai oleh Departemen Pertahanan Amerika Serikat.

2.2.2.2 *Internetwork*ing

Tujuan dari TCP/IP adalah untuk membangun suatu koneksi antar jaringan (*network*), dimana biasa disebut *internetwork*, atau *internet*, yang menyediakan pelayanan komunikasi antar jaringan yang memiliki bentuk fisik yang beragam. Tujuan yang jelas adalah menghubungkan komputer (*hosts*) pada jaringan yang berbeda, atau mungkin terpisahkan secara geografis pada area yang luas.



Gambar 2.7 *Internetworking*

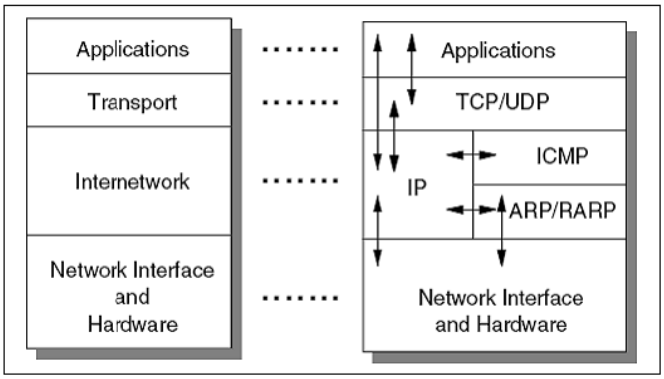
Internet dapat digolongkan menjadi beberapa group jaringan, antara lain:

1. *Backbone*: Jaringan besar yang menghubungkan antar jaringan lainnya. Contoh: NSFNET yang merupakan jaringan backbone dunia di Amerika, EBONE yang merupakan jaringan *backbone* di Eropa, dan lainnya.
2. Jaringan regional, contoh: jaringan antar kampus.
3. Jaringan yang bersifat komersial dimana menyediakan koneksi menuju *backbone* kepada pelanggannya.
4. Jaringan lokal, contoh: jaringan dalam sebuah kampus.

Aspek lain yang penting dari TCP/IP adalah membentuk suatu standarisasi dalam komunikasi. Tiap-tiap bentuk fisik suatu jaringan memiliki teknologi yang berbeda-beda, sehingga diperlukan pemrograman atau fungsi khusus untuk digunakan dalam komunikasi. TCP/IP memberikan fasilitas khusus yang bekerja diatas pemrograman atau fungsi khusus tersebut dari masing-masing fisik jaringan. Sehingga bentuk arsitektur dari fisik jaringan akan tersamarkan dari pengguna dan pembuat aplikasi jaringan. Dengan TCP/IP, pengguna tidak perlu lagi memikirkan bentuk fisik jaringan untuk melakukan sebuah komunikasi.Untuk dapat berkomunikasi antar 2 jaringan, diperlukan komputer yang terhubung dalam suatu perangkat yang dapat meneruskan suatu paket data dari jaringan yang satu ke jaringan yang lain. Perangkat tersebut disebut Router. Selain itu router juga digunakan sebagai pengarah jalur (*routing*). Untuk dapat mengidentifikasikan komputer(*host*) diperlukan sebuah alamat, disebut alamat IP(*IP address*). Apabila sebuah *host* memiliki beberapa perangkat jaringan (*interface*), seperti router, maka setiap *interface* harus memiliki sebuah *IP address* yang unik. *IP address* terdiri dari 2 bagian, yaitu : ***IP address* = <nomer jaringan><nomer *host*>**

2.2.2.3 Lapisan(*layer*) pada Protokol TCP/IP

Seperti pada perangkat lunak, TCP/IP dibentuk dalam beberapa lapisan (*layer*). Dengan dibentuk dalam *layer*, akan mempermudah untuk pengembangan dan pengimplementasian. Antar *layer* dapat berkomunikasi ke atas maupun ke bawah dengan suatu penghubung interface. Tiap-tiap *layer* memiliki fungsi dan kegunaan yang berbeda dan saling mendukung *layer* diatasnya. Pada protokol TCP/IP dibagi menjadi 4 *layer*, tampak pada Gambarberikut.



Gambar 2.8 Protokol TCP/IP

Keterangan:

1. *Layer Applications*

   Layer aplikasi digunakan pada program untuk berkomunikasi menggunakan TCP/IP. Contoh aplikasi antara lain Telnet dan *File Transfer Protocol* (FTP). *Interface* yang digunakan untuk saling berkomunikasi adalah nomor *port* dan *socket*.

2. *Layer Transport*

   Layer transport memberikan fungsi pengiriman data secara *end-to-end* ke sisi *remote*. Aplikasi yang beragam dapat melakukan komunikasi secara serentak (*simulaneously*). Protokol pada *layer transport* yang paling sering digunakan adalah *Transmission Control Protocol* (TCP), dimana memberikan fungsi pengiriman data secara *connection-oriented*, pencegahan duplikasi data, *congestion control* dan *flow control*. Protokol lainnya adalah *User Datagram Protocol* (UDP), dimana memberikan fungsi pengiriman *connectionless*, jalur yang tidak *reliabel*. UDP banyak digunakan pada aplikasi yang membutuhkan kecepatan tinggi dan dapat mentoleransi terhadap kerusakan data.

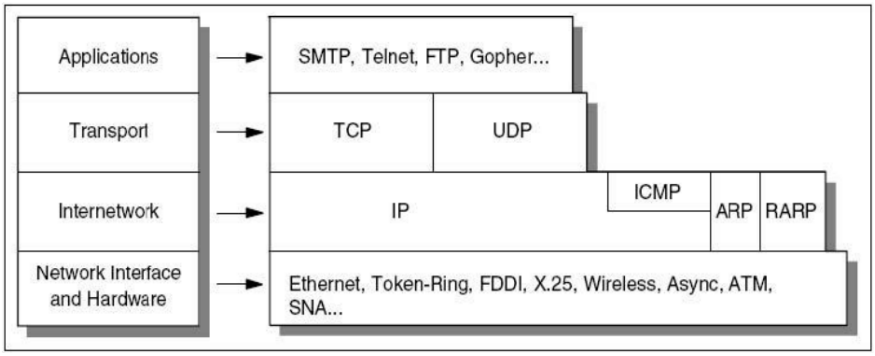
3. *Layer Internetwork*

   Layer Internetwork biasa disebut juga *layer internet* atau *layer network*, dimana memberikan "*vitual network*" pada *internet*. *Internet Protocol* (IP) adalah *protokol* yang paling penting. IP memberikan fungsi *routing* pada jaringan dalam pengiriman data. Protokol lainnya antaralain: IP, ICMP, IGMP, ARP, RARP *Layer*

4. *Network Interface*

   Layer network interface disebut juga *layer* link atau *layer* data link, yang merupakan perangkat keras pada jaringan. Contoh : IEEE802.2,X.25, ATM, FDDI, dan SNA.

   Secara detail dapat digambarkan pada Gambar berikut.



Gambar 2.9 Detail dari model Arsitektur Protokol TCP/IP

Level tertinggi pada *layer* TCP/IP adalah aplikasi. Dimana *layer* ini melakukan komunikasi sehingga dapat berinteraksi dengan pengguna.

Karakteristik dari protokol aplikasi antara lain:

1. Merupakan program aplikasi yang dibuat oleh pengguna, atau aplikasi yang merupakan standar dari produk TCP/IP. Contoh aplikasi yang merupakan produk dari TCP/IP antara lain :
   - TELNET, terminal interaktif untuk mengakses suatu *remote* pada *internet*.
   - HTTP (*Hyper Text Transfer Protocol*), pengiriman *file hypertext* ukuran besar.
   - FTP (*File Transfer Protocol*), transfer *file* berkecepatan tinggi antar komputer
   - SMTP (*Simple Mail Transfer Protocol*), sistem bersurat di *internet*
   - dll
2. Menggunakan mekanisme TCP atau UDP.
3. Menggunakan model interaksi *client/server*.

2.2.2.4 Topologi Jaringan Komputer

Topologi menggambarkan struktur dari suatu jaringan atau bagaimana sebuah jaringan didesain. Pola ini sangat erat kaitannya dengan metode *access* dan media pengiriman yang digunakan. Topologi yang ada sangatlah tergantung dengan letak geofrapis dari masing-masing terminal, kualitas kontrol yang dibutuhkan dalam komunikasi ataupun penyampaian pesan, serta kecepatan dari pengiriman data. Dalam definisi topologi terbagi menjadi dua, yaitu topologi fisik (*physical topology*) yang menunjukan posisi pemasangan kabel secara fisik dan topologi logik (*logical topology*) yang menunjukan bagaimana suatu media diakses oleh *host.* Adapun topologi fisik yang umum digunakan dalam membangun sebuah jaringan adalah :

1. *Point to Point* (Titik ke-Titik).

   Jaringan kerja titik ketitik merupakan jaringan kerja yang paling sederhana tetapi dapat digunakan secara luas. Begitu sederhananya jaringan ini, sehingga seringkali tidak dianggap sebagai suatu jaringan tetapi hanya merupakan komunikasi biasa. Dalam hal ini, kedua simpul mempunyai kedudukan yang setingkat, sehingga simpul manapun dapat memulai dan mengendalikan hubungan dalam jaringan tersebut. Data dikirim dari satu simpul langsung kesimpul lainnya sebagai penerima, misalnya antara terminal dengan CPU.

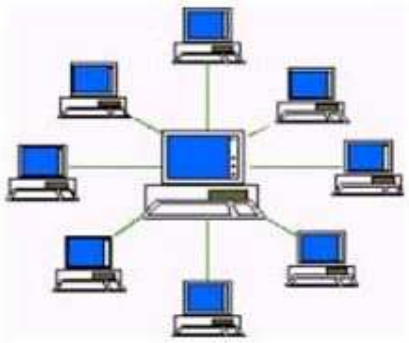
2. *Star Network* (Jaringan Bintang).

   Dalam konfigurasi bintang, beberapa peralatan yang ada akan dihubungkan kedalam satu pusat komputer. Kontrol yang ada akan dipusatkan pada satu titik, seperti misalnya mengatur beban kerja serta pengaturan sumber daya yang ada. Semua *link* harus berhubungan dengan pusat apabila ingin

menyalurkan data kesimpul lainnya yang dituju. Dalam hal ini, bila pusat mengalami gangguan, maka semua terminal juga akan terganggu. Model jaringan bintang ini relatif sangat sederhana, sehingga banyak digunakan oleh pihak per-bank-kan yang biasanya mempunyai banyak kantor cabang yang tersebar diberbagai lokasi. Dengan adanya konfigurasi bintang ini, maka segala macam kegiatan yang ada di-kantor cabang dapatlah dikontrol dan dikoordinasikan dengan baik. Disamping itu, dunia pendidikan juga banyak memanfaatkan jaringan bintang ini guna mengontrol kegiatan anak didik mereka.

Kelebihan :

i. Kerusakan pada satu saluran hanya akan mempengaruhi jaringan pada saluran tersebut dan station yang terpaut.
ii. Tingkat keamanan termasuk tinggi.
iii. Tahan terhadap lalu lintas jaringan yang sibuk.
iv. Penambahan dan pengurangan station dapat dilakukan dengan mudah.
v. Kekurangan
vi. Jika *node* tengah mengalami kerusakan, maka maka seluruh jaringan akan terhenti.
vii. Penanganan: Perlunya disiapkan *node* tengah cadangan.

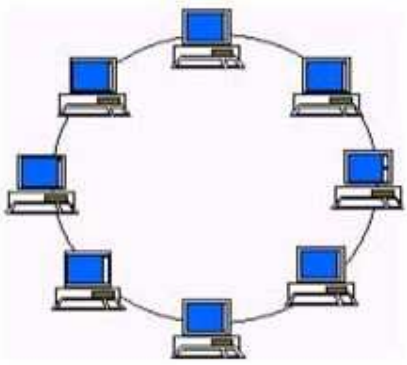


Gambar 2.10 *Topologi* jaringan bintang

3. *Ring Network*s (Jaringan Cincin)

   Pada jaringan ini terdapat beberapa peralatan saling dihubungkan satu dengan lainnya dan pada akhirnya akan membentuk bagan seperti halnya sebuah cincin. Jaringan cincin tidak memiliki suatu titik yang bertindak sebagai pusat ataupun pengatur lalu lintas data, semua simpul mempunyai tingkatan yang sama. Data yang dikirim akan berjalan melewati beberapa simpul sehingga sampai pada simpul yang dituju. Dalam menyampaikan data, jaringan bisa bergerak dalam satu ataupun dua arah. Walaupun demikian, data yang ada tetap bergerak satu arah dalam satu saat. Pertama, pesan yang ada akan disampaikan dari titik ketitik lainnya dalam satu arah. Apabila ditemui kegagalan, misalnya terdapat kerusakan pada peralatan yang ada, maka data yang ada akan dikirim dengan cara kedua, yaitu pesan kemudian ditransmisikan dalam arah yang berlawanan, dan pada akhirnya bisa berakhir pada tempat yang dituju. Konfigurasi semacam ini *relative* lebih mahal apabila dibanding dengan konfigurasi jaringan bintang. Hal ini disebabkan, setiap simpul yang ada akan bertindak sebagai komputer yang akan mengatasi setiap aplikasi yang dihadapinya, serta harus mampu membagi sumber daya yang

dimilikinya pada jaringan yang ada. Disamping itu, sistem ini lebih sesuai digunakan untuk sistem yang tidak terpusat (*decentralized-system*), dimana tidak diperlukan adanya suatu prioritas tertentu.

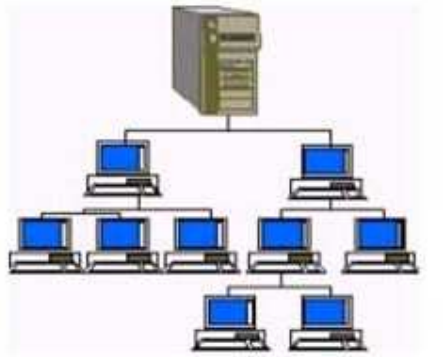


Gambar 2.11 Topologi jaringan cincin

4. *Tree Network* (Jaringan Pohon)

   Pada jaringan pohon, terdapat beberapa tingkatan simpul (*node*). Pusat atau simpul yang lebih tinggi tingkatannya, dapat mengatur simpul lain yang lebih rendah tingkatannya. Data yang dikirim perlu melalui simpul pusat terlebih dahulu. Misalnya untuk bergerak dari komputer dengan *node*-3 kekomputer *node*-7 seperti halnya pada gambar, data yang ada harus melewati *node*-3, 5 dan *node*-6 sebelum berakhir pada *node*-7. Keungguluan jaringan model pohon seperti ini adalah, dapat terbentuknya suatu kelompok yang dibutuhkan pada setiap saat. Sebagai contoh, perusahaan dapat membentuk kelompok yang terdiri atas terminal pembukuan, serta pada kelompok lain dibentuk untuk terminal penjualan. Adapun kelemahannya adalah, apabila simpul yang lebih tinggi kemudian tidak berfungsi, maka kelompok lainnya yang berada

dibawahnya akhirnya juga menjadi tidak efektif. Cara kerja jaringan pohon ini relatif menjadi lambat



Gambar 2.12 Topologi jaringan pohon

5. *Bus Network*

Konfigurasi lainnya dikenal dengan istilah bus-*network*, yang cocok digunakan untuk daerah yang tidak terlalu luas. Setiap komputer (setiap simpul) akan dihubungkan dengan sebuah kabel komunikasi melalui sebuah *interface*. Setiap komputer dapat berkomunikasi langsung dengan komputer ataupun peralatan lainnya yang terdapat didalam *network*, dengan kata lain, semua simpul mempunyai kedudukan yang sama. Dalam hal ini, jaringan tidak tergantung kepada komputer yang ada dipusat, sehingga bila salah satu peralatan atau salah satu simpul mengalami kerusakan, sistem tetap dapat beroperasi. Setiap simpul yang ada memiliki *address* atau alam sendiri. Sehingga untuk meng-*access* data dari salah satu simpul, user atau pemakai cukup menyebutkan

alamat dari simpul yang dimaksud. Keunggulan topologi *Bus* adalah pengembangan jaringan atau penambahan *workstation* baru dapat dilakukan dengan mudah tanpa mengganggu workstation lain. Kelemahan dari topologi ini adalah bila terdapat gangguan di sepanjang kabel pusat maka keseluruhan jaringan akan mengalami gangguan.



Gambar 2.13 Topologi jaringan bus

6. *Plex Network* (Jaringan Kombinasi)

   Merupakan jaringan yang benar-benar interaktif, dimana setiap simpul mempunyai kemampuan untuk meng-access secara langsung tidak hanya terhadap komputer, tetapi juga dengan peralatan ataupun simpul yang lain. Secara umum, jaringan ini mempunyai bentuk mirip dengan jaringan bintang. Organisasi data yang ada menggunakan de-sentralisasi, sedang untuk melakukan perawatan, digunakan fasilitas sentralisasi.



Gambar 2.14 Topologi jaringan kombinasi

Topologi Logik pada umumnya terbagi mejadi dua tipe, yaitu :

a. Topologi *Broadcast*

   Secara sederhana dapat digambarkan yaitu suatu *host* yang mengirimkan data kepada seluruh *host* lain pada media jaringan.

b. Topologi *Token Passing*

   Mengatur pengiriman data pada *host* melalui media dengan menggunakan token yang secara teratur berputar pada seluruh *host*. *Host* hanya dapat mengirimkan data hanya jika *host* tersebut memiliki token. Dengan *token* ini, *collision* dapat dicegah.

Faktor – faktor yang perlu mendapat pertimbangan untuk pemilihan topologi adalah sebagai berikut :

i. Biaya

   Sistem apa yang paling efisien yang dibutuhkan dalam organisasi.

ii. Kecepatan

   Sampai sejauh mana kecepatan yang dibutuhkan dalam sistem.

iii. Lingkungan

Misalnya listrik atau faktor – faktor lingkungan yang lain, yang berpengaruh pada jenis perangkat keras yang digunakan.

iv. Ukuran

Sampai seberapa besar ukuran jaringan. Apakah jaringan memerlukan *file server* atau sejumlah *server* khusus.

v. Konektivitas

Apakahpemakaiyanglainyangmenggunakankomputerlaptopperlu mengakses jaringan dari berbagai lokasi.

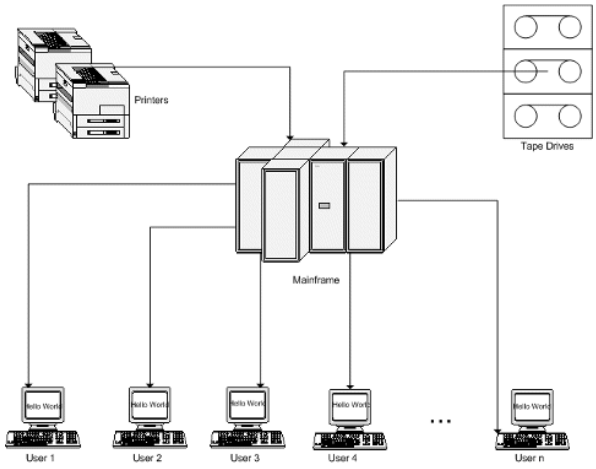
2.2.3 Arsitektur *Client Server*

Istilah arsitektur mengacu pada desain sebuah aplikasi, atau dimana komponen yang membentuk suatu sistem ditempatkan dan bagaimana mereka berkomunikasi. Arsitektur terdistribusi – sebuah istilah yang relatif baru untuk menjelaskan arsitektur aplikasi – berarti bahwa pemrosesan dari suatu aplikasi terjadi pada lebih dari satu mesin.

Terdapat beberapa macam arsitektur aplikasi, yaitu :

1. *Standalone* (one-tier)

   Pada arsitektur ini semua pemrosesan dilakukan pada *mainframe*. Kode aplikasi, data dan semua komponen sistem ditempatkan dan dijalankan pada *host*. Seperti terlihat pada gambar 2.13.



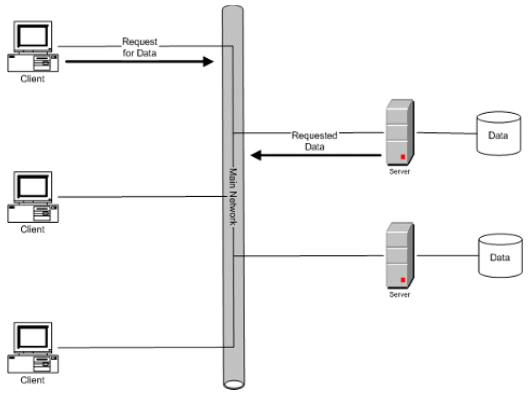
Gambar 2.15 Arsitektur *Standalone* (*one-tier*)

Walaupun komputer *client* dipakai untuk mengakses *mainframe*, tidak ada pemrosesan yang terjadi pada mesin ini, dan karena mereka "*dump-client*" atau "*dump-terminal*". Tipe model ini, dimana semua pemrosesan terjadi secara terpusat, dikenal sebagai berbasis-*host*. Sekilas dapat dilihat kesalahan pada model ini. Ada dua masalah pada komputasi berbasis *host*: Pertama, semua pemrosesan terjadi pada sebuah mesin tunggal, sehingga semakin banyak user yang mengakses *host*, semakin kewalahan jadinya. Jika sebuah perusahaan memiliki beberapa kantor pusat, user yang dapat mengakses mainframe adalah yang berlokasi pada tempat itu, membiarkan kantor lain tanpa akses ke aplikasi yang ada.

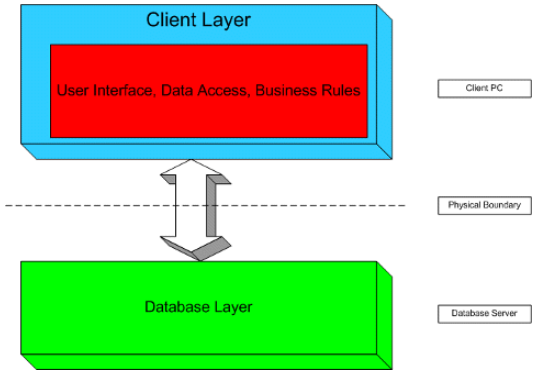
2. *Client*/*Server* (*two-tier*)

   Dalam model *client*/*server*, pemrosesan pada sebuah aplikasi terjadi pada *client* dan *server*. *Client*/*server* adalah tipikal sebuah aplikasi two-tier dengan banyak *client* dan sebuah *server* yang dihubungkan melalui sebuah jaringan,

seperti terlihat dalam gambar 2.14. Aplikasi ditempatkan pada komputer *client* dan mesin *database* dijalankan pada *server* jarak-jauh. Aplikasi *client* mengeluarkan permintaan ke *database* yang mengirimkan kembali data ke *client*-nya.



Gambar 2.16 *Client Server Physical Model*



Gambar 2.17 Arsitektur *Client*/*Server* (*two-tier*) - *Logical View*

Dalam *client*/*server*, *client-client* yang cerdas bertanggung jawab untuk bagian dari aplikasi yang berinteraksi dengan user, termasuk logika bisnis dan komunikasi dengan *server database*. Tipe-tipe tugas yang terjadi pada *client* adalah :

i. Antarmuka pengguna

ii. Interaksi *database*

iii. Pengambilan dan modifikasi data

iv. Sejumlah aturan bisnis

v. Penanganan kesalahan

Server database berisi mesin *database*, termasuk tabel, prosedur tersimpan, dan trigger (yang juga berisi aturan bisnis). Dalam sistem *client*/*server*, sebagian besar logika bisnis biasanya diterapkan dalam *database*. *Server database* manangani :

- Manajemen data
- Keamanan
- Query, trigger, prosedur tersimpan
- Penangan kesalahan

Arsitektur *client*/*server* merupakan sebuah langkah maju karena mengurangi beban pemrosesan dari komputer sentral ke komputer *client*. Ini berarti semakin banyak user bertambah pada aplikasi *client*/*server*, kinerja *server file* tidak akan menurun dengan cepat. Dengan *client*/*server* user dair berbagai lokasi dapat mengakses data yang sama dengan sedikit beban pada sebuah mesin tunggal. Namun masih terdapat kelemahan pada model ini. Selain menjalankan tugas-tugas tertentu, kinerja dan skalabilitas merupakan tujuan nyata dari sebagian besar aplikasi. Model *client*/*server* memiliki sejumlah keterbatasan :

i. Kurangnya skalabilitas

ii. Koneksi *database* dijaga

iii. Tidak ada keterbaharuan kode

iv. Tidak ada tingkat menengah untuk menangani keamanan dan transaksi

Aplikasi-aplikasi berbasis *client*/*server* memiliki kekurangan pada skalabilitas. Skalabilitas adalah seberapa besar aplikasi bisa menangani suatu kebutuhan yang meningkat – misalnya, 50 user tambahan yang mengakses aplikasi tersebut. Walaupun model *client*/*server* lebih terukur daripada model berbasis *host*, masih banyak pemrosesan yang terjadi pada *server*. Dalam model *client*/*server* semakin banyak *client* yang menggunakan suatu aplikasi, semakin banyak beban pada *server*.

Koneksi *database* harus dijaga untuk masing-masing *client*. Koneksi menghabiskan sumber daya *server* yang berharga dan masing-masing *client* tambahan diterjemahkan ke dalam satu atau beberapa koneksi. Logika kode tidak bisa didaur ulang karena kode aplikasi ada dalam sebuah pelaksanaan execuTabel monolitik pada *client*. Ini juga menjadikan modifikasi pada kode sumber sulit. Penyusunan ulang perubahan itu ke semua komputer *client* juga membuat sakit kepala.

Keamanan dan transaksi juga harus dikodekan sebagai pengganti penanganan oleh COM+/MTS. Bukan berarti model *client*/*server* bukanlah merupakan model yang layak bagi aplikasi-aplikasi. Banyak aplikasi yang lebih kecil dengan jumlah user terbatas bekerja sempurna dengan model ini. Kemudahan pengembangan aplikasi *client*/*server* turut menjadikannya sebuah solusi menarik bagi perusahaan.

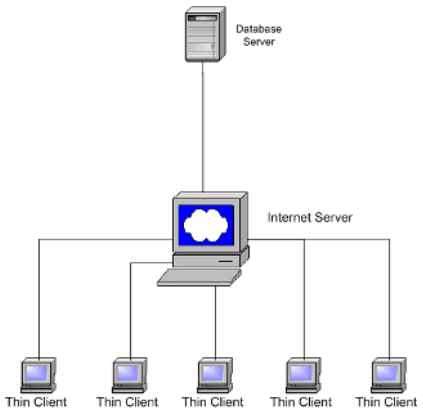
Pengembangan umumnya jauh lebih cepat dengan tipe sistem ini. Siklus pengembangan yang lebih cepat ini tidak hanya menjadikan aplikasi meningkat dan berjalan dengan cepat namun juga lebih hemat biaya.

3. *Three-Tier* / *Multi-Tier*

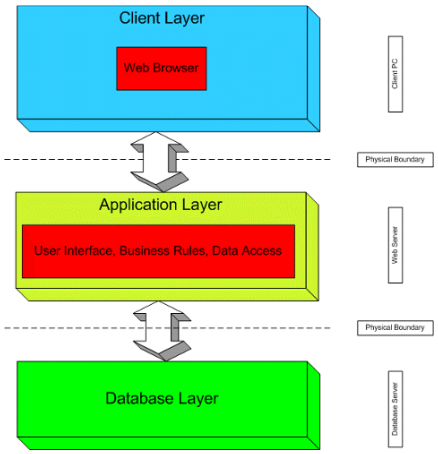
Model three-tier atau multi-tier dikembangkan untuk menjawab keterbatasan pada arsitektur *client*/*server*. Dalam model ini, pemrosesan disebarkan di dalam tiga lapisan (atau lebih jika diterapkan arsitektur multitier). Lapisan ketiga dalam arsitektur ini masing-masing menjumlahkan fungsionalitas khusus. Yaitu :

- Layanan presentasi (tingkat *client*)
- Layanan bisnis (tingkat menengah)
- Layanan data (tingkat sumber data)

Layanan presentasi atau logika antarmuka pengguna ditempatkan pada mesin *client*. Logika bisnis dikeluarkan dari kode *client* dan ditempatkan dalam tingkat menengah. Lapisan layanan data berisi *server database*. Setiap tingkatan dalam model three-tier berada pada komputer tersendiri, seperti pada gambar 2.16.



Gambar 2.18 *Three-Tier* / *Multi-Tier*



Gambar 2.19 *Three-Tier/Multi-Tier - Logical View*

Konsep model *three-tier* adalah model yang membagi fungsionalitas ke dalam lapisan-lapisan, aplikasi mendapatkan skalabilitas, keterbaharuan, dan keamanan.

2.2.4 Basis Data

Basis data adalah sekumpulan data yang terintegrasi yang diorganisasi untuk memenuhi kebutuhan pemakai untuk keperluan organisasi.

DBMS (*Data Base Management System*) adalah perangkat lunak yang menangani semua pengaksesan ke data base.

Sistem Basis Data = DBMS + Basis data

Struktur *File Database*

1. Data adalah satu satuan informasi yang akan diolah, dimana sebelum diolah dikumpulkan di dalam suatu *file database*. Pengumpulan data dilakukan secara sistematis menurut struktur *file database* tersebut.

2. *RECORD* adalah data yang isinya merupakan satu kesatuan seperti Nama, Alamat, Nomor Telepon. Setiap keterangan yang mencakup Nama, Alamat dan Nomor Telepon dinamakan satu *record*. Dan setiap record diberi nomor urut yang disebut nomor *record* (*Record Number*). Ukuran suatu *file database* ditentukan oleh jumlah record yang tersimpan di dalamnya.
3. *FIELD* adalah sub bagian dari Record. Dari contoh isi record diatas maka terdiri dari 3 field, yaitu field Nama, field Alamat dan field Nomor Telepon.

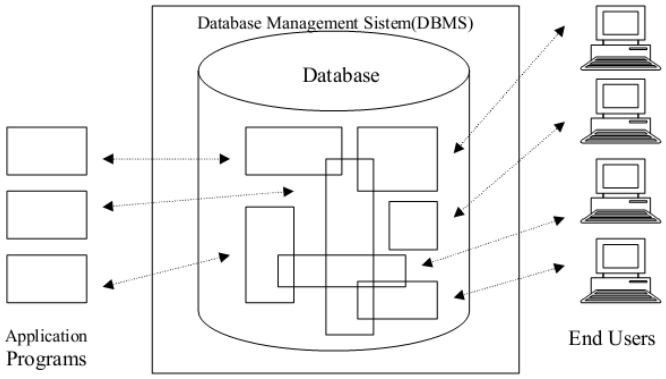
Tabel 2.2 Perbedaan dan Kelemahan *File* Manajemen

Perbedaan	
File manajemen tradisional	*File* manajemen data base
1. Program Oriented 2. Kaku 3. Kerangkapan data	1. Data Oriented 2. Luwes 3. Tidak terjadi kerangkapan data
Kelemahan	
File manajemen tradisional	*File* manajemen data base

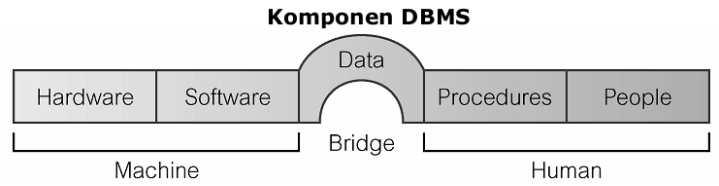
a. Timbulnya data rangkap dan ketidak konsistenan data b. Data tidak dapat digunakan secara bersama-sama c. Kesukaran dalam mengakses data d. Tidak fleksibel e. Data tidak standar	a. Storage yang dibutuhkan besar b. Dibutuhkan tenaga spesialis c. *Software* mahal d. Kerusakan pada data base dapat mempengaruhi departemen lain yang terkait kerangkapan data

Keuntungan *file* manajemen data base :

1. Tidak terjadi kerangkapan data
2. Data lebih konsisten
3. Data dapat digunakan bersama-sama
4. Data dapat distandarisasi
5. Keamanan data dapat terjamin
6. Integritas data terpelihara
7. Data *independen*



Gambar 2.20 Sistem basis data



Gambar 2.21 Komponen DBMS

Hardware

- Meliputi PC sampai dengan jaringan komputer.
- Tempat penyimpanan *secondary* (*manegtic disk*), I/O *device* ex : *disk drives*), device Controller, I/O Channels, dan lainnya.
- *Hardware* processor dan main memory, digunakan untuk mendukung saat eksekusi system *software database*.

Software

- DBMS, operating system, *network software* (jika diperlukan) dan program aplikasi pendukung lainnya.

Data

Data pada sebuah system *database* baik itu single-user system maupun multi-user system harus terintegrasi dan dapat digunakan bersama (*Integrated and Shared*).

- Digunakan oleh organisasi dan deskripsi dari data disebut schema.

Procedures

- Instrukti dan aturan yang harus disertakan dalam mendesain dan menggunakan *database* dan DBMS.

People

- DA (*Data Administrator*), seseorang yang berwenang untuk membuat keputusan stategis dan kebijakan mengenai data yang ada
- DBA (*Database* Administrator), menyediakan dukungan teknis untuk implementasi keputusan tersebut, dan bertanggungjawab atas keseluruhan kontrol system pada level teknis
- *Database* Designer (Logical and Physical)
- Application Programmers, bertanggungjawab untuk membuat aplikasi *database* dengan menggunakan bahasa pemrograman yang ada, seperti : C++, Java, dan lainnya.
- End Users, Siapapun yang berinteraksi dengan sistem secara
- online melalui workstation/terminal.

Kerugian DBMS

- Rumit (Complexity)

  Karena penetapan fungsi dari DBMS yang baik, menyebabkan DBMS menjadi *software* yang cukup rumit. Seluruh user harus mengetahui fungsi-fungsi yang ada dengan baik, sehingga dapat memperoleh manfaatnya.
- Ukuran (*Size*)

Kerumitan dan banyaknya fungsi yang ada menyebabkan DBMS memerlukan banyak *software* pendukung yang mengakibatkan penambahan tempat penyimpanan dan memory.

- Biaya DBMS (*Cost of DBMS*)
- Biaya Tambahan *Hardware* (*Additional hardware costs*)
- Biaya Konversi (*Cost of conversion*)
- Performance

  Pada dasarnya DBMS dibuat untuk menyediakan banyak aplikasi, akibatnya mungkin beberapa aplikasi akan berjalan tidak seperti biasanya.
- Higher impact of a failure

  Karena system yang terpusat, jika seluruh user dan aplikasi terakses dari DBMS maka kerusakan pada bagian manapun dari sistem, akan menyebabkan operasi terhenti.

Fungsi-fungsi DBMS

1. Data definition

   DBMS harus dapat mengolah pendefinisian data
2. Data manipulation

   DBMS harus dapat menangani permintaan-permintaan dari pemakai untuk mengakses data
3. Data security and integrity

   DBMS harus dapat memeriksa keamanan dan integriti data yang didefinisikan oleh DBA
4. Data recovery and concurrency

DBMS harus dapat menangani kegagalan pengaksesan *database* yang disebabkan oleh kesalahan system, kerusakan disk dan sebagainya. DBMS harus dapat memantau pengaksesan data yang konkuren yaitu bila satu data diakses secara bersama-sama oleh lebih dari satu pemakai pada saat bersamaan

5. Data dictionary

   Tempat penyimpanan informasi yang menggambarkan data dalam *database*. Data dictionary disebut juga metadata (data mengenai data) Berisi tentang:

   - Nama-nama user yang mempunyai wewenang untuk penggunaan DBMS
   - Nama-nama item data
   - Jenis-jenis dan ukuran item data
   - Batasan untuk masing-masing item data

6. Performance

   DBMS harus dapat menangani unjuk kerja dari semua fungsi se-efisien mungkin

Model Data

Kumpulan konsep-konsep yang terintegrasi untuk menggambarkan data, relationships antar data, dan batasan-batasan data dalam organisasi.

a. Data Model terdiri dari :

   - Bagian struktural, berisikan sekumpulan aturan berdasarkan *database* yang dapat dibuat.
   - Bagian manipulasi, mendefinisikan tipe operasi yang boleh dilakukan.
   - Aturan-aturan Integritas.

b. Kegunaan untuk

- Merepresentasikan data kedalam bentuk yang lebih mudah untuk dipahami.
- Untuk menetapkan konsistensi dalam memandang, mengorganisir, menginterpretasikan dan memperlakukan *database*.

c. Jenis-jenis Model data :

- Object-Based Data Models Entity-Relationship, Semantic, Functional, Object-Oriented.
- Record-Based Data Models Relational Data Model, *Network* Data Model, Hierarchical Data Model.
- Physical Data Models: Menerangkan bagaimana data disimpan dalam komputer, merepresentasikan informasi seperti ; struktur record, permintaan record, dan jalur akses
- Conceptual Modelling
- Skema konseptual merupakan bagian utama dari system yang menampilkan view seluruh user.
- Merupakan representasi yang akurat dan lengkap dari kebutuhan data pada organisasi.
- Merupakan proses pembentukan suatu model informasi yang digunakan dalam organisasi yang terlepas dari detail implementasi.
- Hasilnya merupakan model data konseptual.

Basis data relasional (relational *database*) ialah model data yang berbentuk tabel dua dimensi yang terdiri atas baris dan kolom.

Keuntungan basis data relasional

1. bentuknya sederhana

2. mudah melakukan berbagai operasi data

Istilah dalam basis data relasional

1. relasi : sebuah tabel yang terdiri dari beberapa kolom dan beberapa baris
2. atribut/field : kolom pada sebuah relasi
3. tupel/record : baris pada sebuah relasi
4. domain : kumpulan nilai yang valid untuk satu atau lebih atribut
5. degree (derajat) : jumlah atribut dalam sebuah relasi
6. cardinality : jumlah tupelo dalam sebuah relasi

Relational key :

a. Super key

   Satu atau kumpulan atribut yang secara unik mengidetifikasi sebuah tupel didalam relasi.

b. Candidate key

   Atribut didalam relasi yang biasanya mempunyai nilai unik.

c. Primary key

   Candidate key yang dipilih untuk mengidentifikasikan tupel secara unik dalam relasi.

d. Alternate key

   Candidate key yang tidak terpilih sebagai primary key.

e. Foreign key

   Atribut dengan domain yang sama yang menjadi kunci utama pada sebuah relasi tetapi pada relasi lain atribut tersebut hanya menjadi atribut biasa.

Relational integrity role :

1. Null

   Nilai suatu atribt yang tidak diketahui dan tidak cocok untuk tupel tersebut.

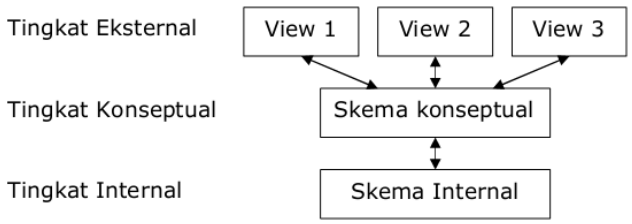
2. Entity Integrity

   Tidak ada satu komponen primary key yang bernilai null.

3. Referential Integrity

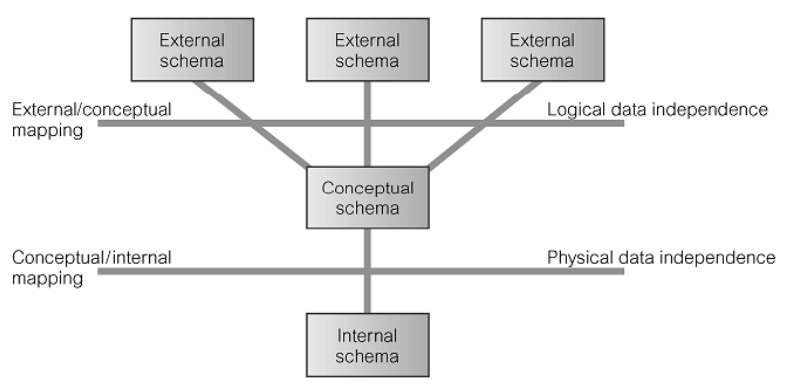
   Suatu domain dapat dipakai sebagai kunci utama bila merupakan atribut tunggal pada domain yang bersangkutan.

Lingkungan basis data



Gambar 2.22 Lingkungan basis data1

Tingkat eksternal yaitu menerangkan view data base dari sekelompok pemakai. Tingkat konseptual yaitu menerangkan informasi *database* secara menyeluruh dengan menyembunyikan data secara fisik. Tingkat internal yaitu menerangkan struktur penyimpanan *database* secara fisik.



Gambar 2.23 Lingkungan basis data2

Data independen : kapasitas untuk mengubah skema pada satu level sistem basis data tanpa mengubah skema pada level lain.

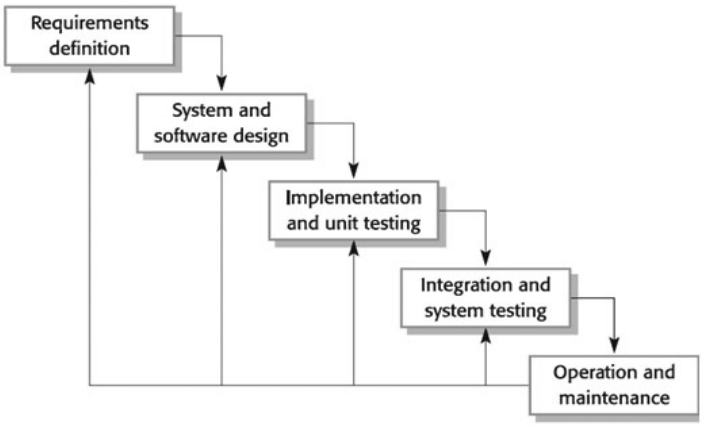
1. Physical Data Independent, perubahan pada skema internal tidak mempengaruhi skema lain.
2. Logical Data Independent, perubahan pada skema konseptual tidak mempengaruhi skema lain.

Mapping (Transformasi), proses pendefinisian informasi dari satu level ke level lainnya.

- Konseptual /internal mapping, pendefinisian hubungan antara view konseptual dengan *database* dilevel internal.
- Eksternal /konseptual mapping, pendefinisian hubungan antara view konseptual dengan *database* dilevel eksternal.

2.2.5 Konsep Pemodelan *Waterfall*

Waterfall model merupakan model lama namun sangat beralasan digunakan ketika kebutuhan dari sistem telah dipahami dengan baik. Gambaran dari konsep pemodelan *waterfall* dapat dilihat pada gambar 2.5.



Gambar 2.24 Pemodelan *Waterfall*

1. ***Requirements analysis and definition:*** Mengumpulkan kebutuhan secara lengkap kemudian kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun. Fase ini harus dikerjakan secara lengkap untuk bisa menghasilkan desain yang lengkap.
2. ***System and software design:*** Desain dikerjakan setelah kebutuhan selesai dikumpulkan secara lengkap.
3. ***Implementation and unit testing:*** desain program diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan. Program yang dibangun langsung diuji baik secara unit.
4. ***Integration and system testing:*** Penyatuan unit-unit program kemudian diuji secara keseluruhan (*system testing*).

5. ***Operation and maintenance:*** mengoperasikan program dilingkungannya dan melakukan pemeliharaan, seperti penyesuaian atau perubahan karena adaptasi dengan situasi sebenarnya.

Kekurangan yang utama dari model ini adalah kesulitan dalam mengakomodasi perubahan setelah proses dijalani. Fase sebelumnya harus lengkap dan selesaisebelum mengerjakan fase berikutnya.

Masalah dengan *waterfall* :

1. Perubahan sulit dilakukan karena sifatnya yang kaku.
2. Karena sifat kakunya, model ini cocok ketika kebutuhan dikumpulkan secaralengkap sehingga perubahan bisa ditekan sekecil mungkin. Tapi pada kenyataannya jarang sekali konsumen/pengguna yang bisa memberikan kebutuhan secara lengkap, perubahan kebutuhan adalah sesuatu yang wajar terjadi

2.2.3 UML

UML (Unified Modeling Language) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, danpendokumentasian dari sebuah sistempengembangan *software* berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software* (http://www.omg.org).

Obyek dalam‘*software* analysis & design’ adalah sesuatu berupa konsep (concept), benda (thing), dan sesuatu yang membedakannya dengan

lingkungannya. Secara sederhana obyek adalah mobil, manusia, alarm dan lain-lainnya. Tapi obyek dapat pula merupakan sesuatu yang abstrak yang hidup didalam sistem sepertitabel, *database*, event, system messages.

Obyek dikenali dari keadaannya dan juga operasinya. Sebagai contoh sebuah mobil dikenali dari warnanya, bentuknya, sedangkan manusia dari suaranya. Ciri-ciri ini yang akan membedakan obyek tersebut dari obyek lainnya.

Alasan mengapa saat ini pendekatandalam pengembangan *software* dengan object-oriented, pertama adalahscalability dimana obyek lebih mudah dipakai untuk menggambarkan sistem yang besar dan komplek. Keduadynamic modeling, adalah dapat dipakai untuk permodelan sistem dinamis dan real time.

Teknik Dasar OOA/D (Object-Oriented Analysis/Design)

Dalam dunia pemodelan, metodologi implementasi obyek walaupun terikat kaidah-kaidah standar, namun teknikpemilihan obyek tidak terlepas pada subyektifitas *software* analyst & designer. Beberapa obyek akan diabaikan dan beberapa obyek menjadi perhatian untuk diimplementasikan di dalam sistem. Hal ini sah-sah saja karena kenyataan bahwa suatu permasalahan sudah tentu memiliki lebih dari satu solusi. Ada 3(tiga) teknik/konsep dasar dalam OOA/D, yaitu pemodulan (encapsulation), penurunan (inheritance) dan polymorphism.

a. Pemodulan (Encapsulation)

   Pada dunia nyata, seorang ibu rumah tangga menanak nasi dengan menggunakan rice cooker, ibu tersebut menggunakannya hanya dengan menekan tombol. Tanpa harus tahu bagaimana proses itu sebenarnya terjadi. Disini terdapat penyembunyian informasi milikrice cooker, sehingga tidak

perlu diketahui seorang ibu. Dengan demikian menanak nasi oleh si ibu menjadi sesuatu yang menjadi dasar bagi konsep information hiding.

b. Penurunan (Inheritance)

Obyek-obyek memiliki banyak persamaan, namun ada sedikit perbedan. Contoh dengan beberapa buah mobil yang mempunyai kegunaan yang berbeda-beda. Ada mobil bak terbuka seperti truk, baktertutup seperti sedan dan minibus. Walaupun demikian obyek-obyek ini memiliki kesamaan yaitu teridentifikasi sebagai obyek mobil, obyek ini dapat dikatakan sebagai obyek induk (parent). Sedangkan minibus dikatakan sebagai obyek anak (child), hal ini juga berarti semua operasi yang berlaku pada mobil berlaku juga pada minibus.

c. Polymorphism

Pada obyek mobil, walaupun minibus dan truk merupakan jenis obyek mobil yang sama, namun memiliki juga perbedaan. Misalnya suara truk lebih keras dari pada minibus, hal ini juga berlaku pada obyek anak (child) melakukan metoda yang sama dengan algoritma berbeda dari obyek induknya. Hal ini yang disebut polymorphism, teknik atau konsep dasar lainnya adalah ruang lingkup / pembatasan. Artinya setiap obyek mempunyai ruang lingkup kelas, atribut, dan metoda yang dibatasi.

2.2.6 PHP

PHP adalah bahasa pemograman yang memungkinkan para web developer untuk membuat aplikasi web yang dinamis dengan cepat. PHP merupakan singkatan dari “PHP: Hypertext Preprocessor”. PHP ditulis dan diperkenalkan pertama kali sekitar tahun 1994 oleh Rasmus Lerdorf melalui situsnya untuk mengetahui siapa saja yang telah mengakses ringkasan online-nya. PHP merupakan salah satu bahasa script yang terbilang baru dan tersedia secara bebas dan masih memungkinkan untuk dikembangkan lebih lanjut. PHP dapat diintegrasikan (embedded) ke dalam web *server*, atau dapat berperan sebagai program CGI yang terpisah. Karakteristik yang paling unggul dan paling kuat dalam PHP adalah lapisan integrasi *database* (*database* integration *layer*). *Database* yang didukung PHP adalah: Oracle, Adabas-D, Sybase, *File*Pro, mSQL, Velocis, MySQL, Informix,Solid, dBase, ODBC, Unix dbm, dan PostgreSQL.

PHP mempunyai banyak kelebihan antara lain:

1. Mudah dibuat dan kecepatan akses sangat tinggi.
2. Dapat berjalan dalam *web server* yang berbeda dan dalam sistem operasi yang berbeda pula. PHP dapat berjalan di sistem operasi UNIX, Windows98, Windows NT, dll.
3. PHP juga dapat berjalan pada *web server* Microsoft Personal Web *Server*, Apache, IIS, Xitami dan sebagainya.
4. PHP termasuk bahasa yang *embedded* (biasa ditempat/diletakan dalam tag HTML).
5. PHP termasuk *server-side programming*.

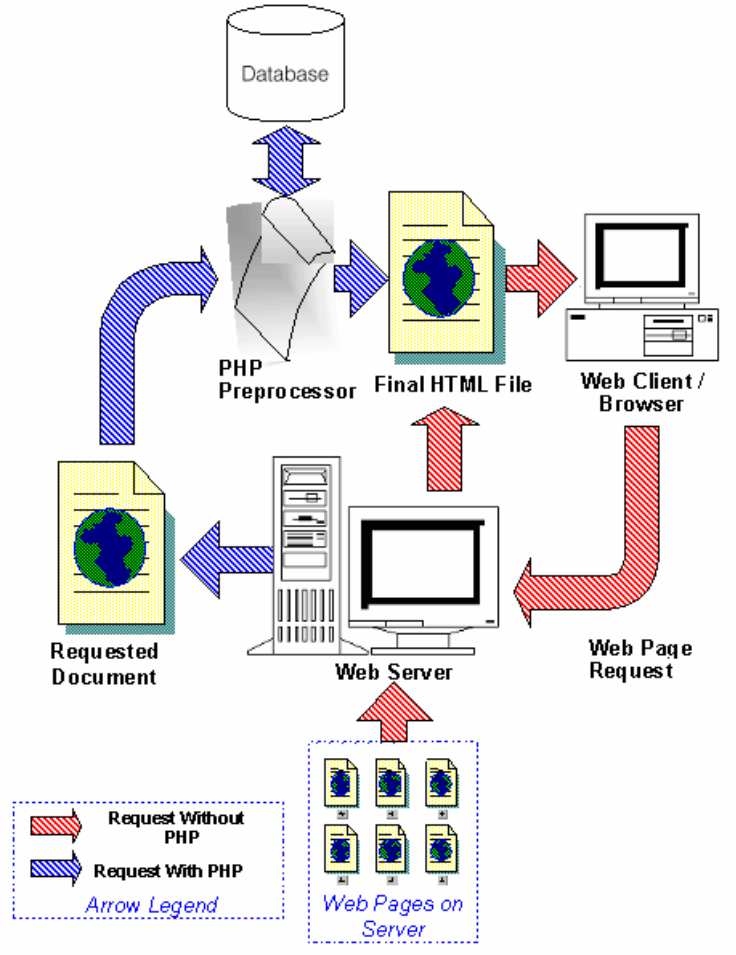
6. Selain itu PHP dapat diperoleh dari *internet* sebagai *open source software* yang terbukti mempunyai keandalan dalam menjalankan fungsinya dan banyak dipakai aplikasi *Server Internet* dalam dunia informatika.

Untuk menjalankan sistem PHP dibutuhkan tiga komponen yaitu:

a. Web *Server*, karena PHP termasuk bahasa pemrograman *server-side.*

b. Program PHP(PHP interpreter), program yang memproses script PHP.

c. *Database Server*, yang berfungsi untuk mengelola *database*



Gambar 2.25 Alur Kerja PHP

Proses-proses yang terjadi adalah:

1. *Client* me-request halaman web yang berisi script PHP
2. Jika *file* yang di-request ditemukan, maka *server* akan meneruskannya ke PHP interpreter (Penterjemah PHP) yang akan bekerja menghasilkan dokumen/halaman HTML berdasarkan script PHP. Jika dalam script tersebut terdapat permintaan terhadap *database*, maka akan terjadi proses query data ke *database server*.
3. Dokumen HTML hasil interpretasi oleh PHP interpreter dikembalikan ke web *server*.
4. Web *server* mengirim dokumen HTML(Response) ke *Client*.
5. Web *browser* akan menterjemahkan dokumen HTML ke Display/Monitor.

2.2.7 Mysql

MySQL ialah sebuah perangkat lunak sistem untuk memanajemen basis data SQL atau sering juga disebut DBMS (*Database*s Management System). MySQL adalah DBMS yang multithread, multiuser dengan perkiraan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL sebagai perangkat lunak gratis, dibawah lisensi GNU GPL.

Tidak sama hal nya dengan Apache, dimana perangkat lunak dikembangkan oleh komunitas umum. MySQL dimiliki dan d siponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya.

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang di distribusikan secara gratis dibawah lisensi GPL. SQL adalah sebuah konsep pengoperasian basisdata untuk pemilihan atau seleksi dan pemasukan data, yang pengoperasiannya dapat dengan mudah secara otomatis.

Sebagai basis data, MySQL mendukung operasi basis data transaksional maupun non-transaksional, pada operasi transaksional MySQL lebih unggul dari kompetitor lainnya. Namun demikian pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus non-transaksional.

MySQL memiliki beberapa keistimewaan, antara lain :

a. Portabilitas, yaitu dapat berjalan melintasi multiplatform dan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X *Server*, Solaris, Amiga, dan masih banyak lagi.
b. Perangkat lunak sumber terbuka atau *open source*, dibawah lisensi GPL sehingga dapat digunakan secara gratis.
c. *Multi-user*, yoitu dapat digunakan oleh beberapa pengguna atau user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

d. ‘*Performance tuning’*, memiliki kecepatan yang cepat dan stabil dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
e. Ragam tipe data, memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
f. Perintah dan Fungsi, memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (query).
g. Keamanan. MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama *host*, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.
h. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
i. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix soket* (UNIX), atau *Named Pipes* (NT).
j. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
k. Antar Muka. MySQL memiliki antar muka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

l. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.

m. Struktur tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABEL, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.