# BAB II

# TINJAUAN PUSTAKA

## 2.1 Profil Tempat Kerja Praktek

### 2.1.1 Sejarah Infolahtadam III/Siliwangi

#### 2.1.1.1 Lahirnya Infolahta TNI AD

Organisai Paban V Suad / Renlitbang, sebagai embrio dari badan pengumpulan dan pengolahan data TNI-AD, yang berdasarkan keputusan Kasad Nomor : Skep/275/III/976 diresmikan menjadi dinas pengumpulan dan pengolahan data TNI- AD, disingkat DISINFOLAHTAD.

Sejalan dengan diresmikannya Disingolahtad ditingkat Mabesad berdasarkan Surat Keputusan Kasad Nomor : Skep/609/1976 tanggal 5 Mei 1976 ditingkat Kotama/ Lakpus dibentuk pula Badan Pengumpulan dan pengolahan data ( Pullahta Kotama/Lakpus).

#### 2.1.1.2 Perkembangan Disinfolahtad

1. Terhitung tanggal 2 Nopember 1978 Organisasi Dispullahtad dikembangkan dengan kekuatan personel 212 orang.
2. Terhitung tanggal 25 April 1985 ditingkatkan kembali dengan kekuatan personel menjadi 261 orang.
3. Berdasarkan perhitungan Kasad Nomor : Kep/1/II/1993 tanggal 26 Februari 1993, Dispullahtad mengalami validasi organisasi dari Dinas Pengumpulan dan Pengolahan Data TNI Angkatan Darat menjadi Dinas Informasi dan Pengolahan Data TNI Angkatan Darat disingkat Disinfolahtad dengan kekuatan personil 344 orang.
4. Berdasarkan Keputusan Kasad Nomor : Kep / 1 / II / 2003 tanggal 4 Februari 2003, Disinfolahtad mengalami validasi organisasi dengan kekuatan personil menjadi 243 orang.

### 2.1.1.3 Perkembangan Pullahta Kotama / Lakpus

Berdasarkan Surat Telegram Kasad Nomor : ST / 199 / 1993 tanggal 9 Maret 1993, sebutan Pullahta Kotama / Lakpus menjadi Infollahta Kotama / Lakpus.

### 2.1.1.4 Perkembangan Infolahta Kodam III//Siliwangi

a. Tahun 1976

1) Peresmian Organisasi Pullahtadam VI/Slw, dilaksanakan pada tanggal 23 Juni 1976 oleh Pangdam VI/Slw di Aula Makodam VI/Slw, berdasarkan pembentukan organisasi Pullahtadam sesuai dengan Surat Keputusan Pangdam VI/Slw Nomor : SKEP/73-2/VI/1976.
2) Kedudukan organisasi Pullahtadam VI/Slw adalah sebagai Badan Staf Khusus Kodam VI./Slw dengan susunan organisasi terdiri dari Kepala, Wakil Kepala, Karo Prog, Karo Dalta, Karo Ops, Karo Tuud dengan kekuatan personil sebanyak 45 orang.

b. Tahun 1985

Berdasarkan Surat Keputusan Kasad Nomor : SKEP/64/IX/1985 tanggal 18 September 1985, dilaksanakan reorganisasi Pullahtadam VI/Slw menjadi Pullahtadam III/Slw dengan perubahan-perubahan sebagai berikut :

1) Kedudukan organisasi sebagai Badan Pelayanan Staf Kodam III/Slw.
2) Susunan organisasi terdiri dari : Kepala, Kasi Tuud, Kasi Dalsiapta, Kasi Duklahta, Kasi Sisfomin, Kasi Sisfobinkuat.
3) Kekuatan personil sebanyak 73 orang.

c. Tahun 1991

Berdasarkan Surat Telegram Kasad Nomor : ST/12/1991 tanggal 4 Januari 1991 dan Surat Telegram Kadispullahtad Nomor : ST/02/1992 tanggal 2 Pebruari 1991 ditetapkan bahwa tanggal 8 Maret sebagai hari jadi Badan Pullahta TNI-AD

d. Tahun 1993

1) Berdasarkan Surat Keputusan Kasad Nomor : SKEP/I/III/1993 tanggal 26 Pebruari 1993 dan Surat Telegram Pangdam III/Slw Nomor : ST/148/1993 tanggal 18 Maret 1993, Sebutan organisasi Pullahtadam III/Slw, menyesuaikan sebutan organisasi Dispullahtad yang mengalami validasi organisasi menjadi Disinfolahtad.

2) Perubahan sebutan Pullahtadam III/Slw menjadi Infolahtadam III/Slw belum merubah kedudukan, susunan organisasi dan kekuatan personil Infolahtadam III/Slw meskipun ditingkat Disinfolahtad sudah mengalami perubahan.

### 2.1.2 Motto Infolahtadam III/Siliwangi

WIDYATA AWASARA ANANDYAGMA, yang bermakna Cepat, Tepat dan AKurat dalam Penyajian Informasi dan Pengolahan Data

### 2.1.3 Struktur Organisasi dan Job Description

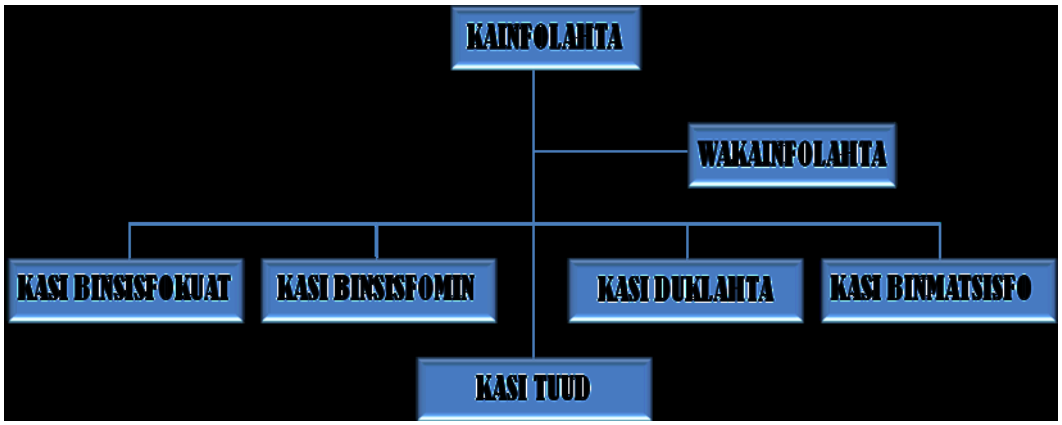
#### 2.1.3.1 Struktur Organisasi Infolahtadam III/Siliwangi

Susunan Organisasi Infolahtadam III/Siliwangi disususn dalam 3 (tiga) eselon sebagai berikut :

a. Eselon Pimpinan.
   1. Kepala Infolahta Kodam, disingkat Kainfolahtadam
   2. Wakil Kepala Infolahta Kodam, disingkat Wakainfolahtadam

b. Eselon Pembantu Pimpinan
   1. Kepala Seksi Pembinaan Sistem Informasi Administrasi, disingkat Kasibinsisfomin
   2. Kepala Seksi Pembinaan Sistem Informasi Kekuatan dan Likungan Operasi, disingkat Kasibinsisfokuat.
   3. Kepala Seksi Pembinaan Materiil Sistem Informasi, disingkat Kasibinmatsisfo.
   4. Kepala Seksi Dukungan Pengolahan Data, disingkat Kasiduklahta.

c. Eselon Pelayanan. Kepala Seksi Tatausaha dan Urusan Dalam, disingkat Kasituud.




Gambar 2.1 Struktur Organisasi Infolahtadam III/Siliwangi

### 2.1.3.2 Job Description

1. **Kainfolahtadam**

Dijabat oleh seorang Pamen Angkatan Darat berpangkat Letnan Kolonel, dengan tugas kewajiban sebagai berikut :

**Sebagai Pimpinan Infolahtadam**

a. Memimpin, membina dan mengendalikan semua usaha,pekerjaan dan kegiatan agar sesuai dengan program kerja.
b. Mengawasi, memelihara dan meningkatkan materiil serta administrasi logistik.
c. Memelihara dan menegakan hokum, disiplin dan tata tertib di lingkungan Infolahtadam.
d. Memelihara, mengawasi, dan meningkatkan kesejahteraan, kemampuan kerja serta pengembangan personel dalam rangka kesiapan operasional satuan.
e. Mengawasi dan meningkatkan daya dan hasil guna serta keserasian kerja di lingkungan Infolahtadam.
f. Memimpin da mengendalikan pelaksanaan pembinaan territorial terbatas.
g. Menghimpun dan memelihara tradisi dan sejarah kesatuan.

**Sebagai Perwira Staf Khusus Pangdam**

a. Memberikan saran dan pertimbangan kepada pangdam mengenai hal-hal berkaitan dengan bidang tugasnya.
b. Melaksanakan bimbingan fungsi teknik pembinaan system informasi kepada personil di kesatuan jajaran kodam.
c. Melaksanakan pengawasan terhadap pelaksanaan fungsi pembinaan sistem informasi di kesatuan jajaran kodam.

**Sebagai penyelenggara fungsi pembinaan sistem informasi kodam.**

a. Melaksanakan fungsi teknis pembinaan sistem informasi di kodam berdasakan petunjuk teknik pembinaan sistem informasi TNI AD
b. Menentukan kebijakan teknis dalam penyelenggaraan fungsi pembinaan system informasi berdasarkan kebijakan umum pangdam dan pembinaan system informasi TNI AD
c. Memberikan bantuan pemeliharaan dan dukungan teknik fungsi pembinaan sistem informasi kodam.
d. Melaksanakan kegiatan lapangan kekuasaan teknis sesuai dengan fungsi pembinaan sistem informasi kodam
e. Melaksanakan penelitian dan pengkajian terhadap perkembangan teknologi sistem informasi untuk kepentingan pembinaan sistem informasi kodam.

Kainfolahta dalam melaksanakan tugas kewajibannya bertanggung jawab :

a. Secara taktis operasional kepada Pangdam dan dalam pelaksanaan tugas sehari-hari dikoordinasikan oleh Kasdam.
b. Secara teknis operasional kepada Kadisinfolahtad.

2. **Kasibinsisfomin**

Dijabat oleh seorang Pamen Angkatan Darat berpangkat mayor, merupakan pembantu Kainfolaahtadam yang bertanggung jawab menyelenggarakan kegiatan di bidang aplikasi Sistem

Administrasi Kodam, dengan tugas dan kewajiban sebagai berikut :

a. Menganalisis dan merancang aplikasi system informasi untuk mendukung pembinaan aplikasi adaministrasi di bidang personel, logistic, perencanaan dan anggaran kodam, berdasarkan sistem pembinaan yang berlaku serta menjaga keterpaduan dengan sistem aplikasi sisfomin yang dibangun di tingkat pusat.
b. Melaksanakan pemeliharaan evaluasi terhadap aplikasi yang telah dioperasikan agar dapat menyesuaikan dengan perubahan dan perkembangan yang terjadi.
c. Membina dokumentasi aplikasi guna menjamin kelangsungan operasional dan pengembangannya.
d. Melaksanakan analisis dan evaluasi terhadap aplikasi yang sedang berjalan untuk menemukan kemungkinan aplikasi sisfomin yang lebih efisien.
e. Melaksanakan penelitian dan pengkajian tehadap perkembangan teknologi informasi di bidang perangkat lunak dan database.
f. Memberikan saran dan pertimbangan kepada Kainfolahtadam sesuai dengan bidang tugasnya.

Kasibinsisfomin dalam melaksanakan tugasnya dibantu oleh 1(satu) Kepala UrusanAnalisa Sistem Informasi Administrasi disingkat Kurnalsisfomin yang dijabat oleh seorang Pama Angkatan Darat berpangkat Kapten. Kasibinsisfomin dalam melaksanakan tugasnya bertanggung jawab pada Kainfolahta.

3. **Kabinsisfokuat**

Dijabat oleh seorang Pamen Angkatan Darat berpangkat Mayor, merupakan pembantu Kainfolahtadam yang bertanggungjawab menyelenggarakan kegiatan di bidang aplikasi sistem informasi Kekuatan dan Lingkungan Operasi Kodam dengan tugas kewajiban sebagai berikut :

a. Menganalisis dan merancang aplikasi sistem informasi untuk mendukung pembinaan aplikasi kekuatan dan lingkungan operasi di bidang intelijen, operasi dan territorial, berdasarkan sistem pembinaan yang berlaku serta menjaga keterpaduan dengan sistem aplikasi sisfomin yang dibangun di tingkat pusat.

b. Melaksanakan pemeliharaan dan evaluasi terhadap aplikasi sisfokuat yang telah dioperasikan agar dapat menyesuaikan dengan perubahan atau perkembangan yang terjadi.

c. Membina dokumentasi aplikasi sisfokuat guna menjamin kelangsungan operasional dan pengembangannya.

d. Melaksanakan analisis dan evaluasi terhadap aplikasi sisfokuat yang sedang berjalan untuk menemukan kemungkinan aplikasi sisfokuat yang efisien.

e. Melaksanakan penelitian dan pengkajian terhadap perkembangan teknologi informasi di bidang perangkat lunak dan database.

f. Memberikan saran dan pertimbangan kepada Kainfolahtadam sesuai dengan bidang tugasnya.

Kasibinsisfokuat dalam melaksanakan tugasnya dibantu oleh 1 (satu) Kepala Urusan Analisa Sistem Informasi Kekuatan dan Lingkungan Operasi, disingkat Kaurnalsisfokuat & Lingops yang dijabat oleh seorang Pama Angkatan Darat berpangkat Kapten. Kasibinsisfokuat dalam melaksanakan tugasnya bertanggung jawab pada Kainfolahta.

4. **Kasibinmatsisfo**

Dijabat oleh seorang Pamen Angkatan Darat berpangkat Mayor, merupakan pembantu Kainfolahtadam yang bertanggungjawab menyelenggarakan kegiatan di bidang Pembinaan Materiil Sistem Informasi serta Dukungan Khusus Multimedia dan Presentasi dengan tugas kewajiban sebagai berikut:

a. Merencanakan kebutuhan materiil sistem informasi.
b. Menerima dan mendistribusikan materiil sistem informasi kepada kesatuan pengguna sesuai rencana.
c. Meninventarisir dan membuat laporan materiil sistem informasi yang dibiana Kainfolahtadam.
d. Melaksanakan pemeliharaan materiil sistem informasi sesuai dengan tingkat kewenangannya.
e. Menyusun daftar dan melaporkan materiil sistem informasi yang akan dihapus.
f. Memberikan dukungan khusus bidang multimedia dan presentasi.
g. Melaksanakan penelitian dan pengkajian terhadap perkembangan teknologi informasi di bidang perangkat lunak, perangkat keras, dan jaringan computer.
h. Memberikan saran dan pertimbangan kepada Kainfolahtadam sesuai dengan bidang tugasnya.

Kasibinmatsisfo dalam melaksanakan tugasnya dibantu oleh 1 (satu) Kepala Urusan-Materiil Analisa Sistem Informasi, disingkat Kaurmatsisfo yang dijabat oleh seorang Pama Angkatan Darat berpangkat Kapten dan 2 (dua) Perwira Urusan yang masing-masing dijabat oleh seorang Pama Angkatan berpangkat Letnan, terdiri dari :

- Kepala Urusan Materiil Sistem informasi, disingkat Kaurmatsisfo.
- Perwira Urusan Pemeliharaan Materiil Sistem Informasi, disingkat Paurharmatsisfo.
- Perwira Urusan Dukungan khusus, disingkat Paurduksus.

Kasibinsisfokuat dalam melaksanakan tugasnya bertanggung jawab pada Kainfolahta.

5. **Kasiduklahta**

Dijabat oleh seorang Pamen Angkatan Darat berpangkat Mayor, merupakan pembantu Kainfolahtadam yang bertanggung jawab menyelenggarakan kegiatan di bidang Dukungan Pengolahan Data dan Dukungan Teknis, dengan tugas kewajiban sebagai berikut :

a. Merencanakan, melaksanakan dan mengendalikan perekaman data sebagai bahan masukan pengolahan data.
b. Merencanakan, melaksanakan dan mengendalikan pelaksanaan, pengoperasian dan kesiapan sarana instalasi computer dalam rangka pengolahan data.
c. Melaksanakan pengolahan data berdasarkan skala prioritas dan pengujian informasi dengan menggunakan sarana computer dan sarana pendukungnya.
d. Membina dan menyelenggarakan administrasi dan perpustakaan data yang berkaitan dengan dukungan teknis pengolahan data elektronis.
e. Menyelenggarakan dukungan teknis operasional terhadap sarana pengolahan data dan sarana pendukungnya kepada kesatuan pengguna di jajaran Kodam.
f. Melaksanakan penelitian dan pengkajian terhadap perkembangan teknologi informasi di bidang perangkat lunak, perangkat keras, dan jaringan komputer.
g. Memberikan saran dan pertimbangan kepada Kainfolahtadam sesuai bidang tugasnya.

Kasiduklahta dalam melaksanakan tugasnya dibantu 2 (dua) Kepala Urusan yang masing-masing dijabat oleh seorang Pama Angkatan Darat berpangkat Kapten terdiri dari :

- Kepala Urusan Penyiapan dan Pengolahan Data disingkat Kaursiaplahta.
- Kepala Urusan Dukungan Teknis, disingkat Kaurduknis.

Kasiduklahta dalam melaksanakan tugasnya bertanggung jawab pada Kainfolahta.

6. **Kasituud**

Dijabat oleh seorang Pamen Angkatan Bersenjata berpangkat Mayor, merupakan unsur pelayanan Infolahtadam yang bertanggung jawab menyelenggarakan kegiatan di bidang Fungsi Organik serta Ketatausahaan dan Urusan Dalam dengan tugas dan kewajiban sebagau berikut :

- Memimpin, mengkoordinasikan dan mengendalikan segala usaha, pekerjaan dan kegiatan dalam pelaksanaan fungsi organik militer dan fungsi organik pembinaan.
- Menyelenggarakan kegiatan ketatausahaan dan urusan dalam.
- Merencanakan, mengkoordinasikan dan mengatur pelaksanaan pembinaan territorial terbatas.
- Memberikan saran dan pertimbangan kepada Kainfolahtadam sesuai bidang tugasnya.

Kasituud dalam melaksanakan tugasnya dibantu 2 (dua) Kepala Urusan yang masing-masing dijabat oleh seorang Pama Angkatan Darat berpangkat Kapten terdiri dari :

- Kepala Urusan Administrasi Personel dan Logistik, disingkat Kaurminperslog.
- Kepala Urusan Dalam, disingkat Kaurdal.

Kasituud dalam melaksanakan tugasnya bertanggung jawab pada Kainfolahta.

## 2.2 Landasan Teori

### 2.2.1 Konsep Sistem Informasi

#### 2.2.1.1 Konsep Dasar Sistem

Sistem : kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

**Menurut Jerry FithGerald** ; sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

Karakteristik Sistem :

1. Memiliki komponen : Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak perduli betapapun kecilnya, 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 sistem yang lebih besar yang disebut supra sistem, misalnya suatu perusahaan dapat disebut dengan suatu sistem dan industri yang merupakan sistem yang lebih besar dapat disebut dengan supra sistem. Kalau dipandang industri sebagai suatu sistem, maka perusahaan dapat disebut sebagai subsistem. Demikian juga bila perusahaan dipandang sebagai suatu sistem, maka sistem akuntansi adalah subsistemnya.
2. Batas sistem (boundary) : Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (scope) dari sistem tersebut.

3. Lingkungan luar sistem (environment) : Adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.
4. Penghubung sistem (interface) ; Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya.
5. Masukan sistem (input) ; Merupakan 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. Signal input adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam sistem komputer, program adalah maintanance input yang digunakan untuk mengoperasikan komputernya dan data adalah signal input untuk diolah menjadi informasi.
6. Keluaran sistem (Output) ; Merupakan hasil dari energi yang diolah oleh sistem.
7. Pengolah sistem (Process) ; Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.
8. Sasaran sistem ; Kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.

Klasifikasi Sistem :

1. Sistem abstrak ; sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik (sistem teologia)
2. Sistem fisik ; merupakan sistem yang ada secara fisik (sistem komputer, sistem akuntansi, sistem produksi dll.)
3. Sistem alamiah ; sistem yang terjadi melalui proses alam. (sistem matahari, sistem luar angkasa, sistem reproduksi dll.
4. Sistem buatan manusia ; sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan interaksi manusia dengan mesin disebut human- machine system (contoh ; sistem informasi)
5. Sistem Tertentu (deterministic system) ; beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi bagian-bagiannya

dapat dideteksi dengan pasti sehingga keluaran dari sistem dapat diramalkan (contoh ; sistem komputer)

6. Sistem tak tentu (probabilistic system) ; sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
7. Sistem tertutup (close system) ; sistem yang tidak berhubungan dan tidak terpengaruh dengan sistem luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luarnya. Secara teoritis sistem tersebut ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah relatively closed system (secara relatif tertutup, tidak benar-benar tertutup).
8. Sistem terbuka (open system) : sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

Lebih spesifik dikenal juga yang disebut dengan sistem terotomasi ; yang merupakan bagian dari sistem buatan manusia dan berineraksi dengan kontrol oleh satu atau lebih komputer sebagai bagian dari sistem yang digunakan dalam masyarakat modern.

Sistem terotomasi mempunyai sejumlah komponen yaitu ;

1. Perangkat keras (CPU, disk, printer, tape).
2. Perangkat lunak (sistem operasi, sistem database, program pengontrol komunikasi, program aplikasi).
3. Personil (yang mengoperasikan sistem, menyediakan masukan, mengkonsumsi keluaran dan melakukan aktivitas manual yang mendukung sistem).
4. Data (yang harus tersimpan dalam sistem selama jangka waktu tertentu).
5. Prosedur (instruksi dan kebijakan untuk mengoperasikan sistem).

Sistem terotomasi terbagi dalam sejumlah katagori :

1. On-line systems. Sistem on-line adalah sistem yang menerima langsung input pada area dimana input tersebut direkam dan menghasilkan output yang dapat berupa hasil komputasi pada area dimana mereka dibutuhkan. Area sendiri dapat dipisah-pisah dalam skala, misalnya ratusan kilometer. Biasanya digunakan bagi reservasi angkutan udara, reservasi kereta api, perbankan dll.
2. Real-time systems. Sistem real-time adalah mekanisme pengontrolan, perekaman data, pemrosesan yang sangat cepat sehinga output yang dihasilkan dapat diterima dalam waktu yang relatif sama. Perbedaan dengan sistem on-line adalah satuan waktu yang digunakan real-time biasanya seperseratus atau seperseribu detik sedangkan on-line masih dalah skala detik atau bahkan kadang beberapa menit. Perbedaan lainnya, on-line biasanya hanya berinteraksi dengan pemakai, sedangkan real-time berinteraksi langsung dengan pemakai dan lingkungan yang dipetakan.
3. Decision support system + strategic planning system. Sistem yang memproses transaksi organisasi secara harian dan membantu para manajer mengambil keputusan, mengevaluasi dan menganalisa tujuan organisasi. Digunakan untuk sistem penggajian, sistem pemesanan, sistem akuntansi dan sistem produksi. Biasanya berbentuk paket statistik, paket pemasaran dll. Sistem ini tidak hanya merekam dan menampilkan data tetapi juga fungsi-fungsi matematik, data analisa statistik dan menampilkan informasi dalam bentuk grafik (tabel, chart) sebagaimana laporan konvensional.
4. Knowledge-based system. Program komputer yang dibuat mendekati kemampuan dan pengetahuan seorang pakar. Umumnya menggunakan perangkat keras dan perangkat lunak khusus seperti LISP dan PROLOG.

Sistem berdasarkan prinsip dasar secara umum terbagi dalam :

1. Sistem terspesialisasi ; adalah sistem yang sulit diterakpan pada lingkungan yang berbeda (misalnya sistem biologi; ikan yang

dipindahkan ke darat)

2. Sistem besar ; adalah sistem yang sebagian besar sumber dayanya berfungsi melakukan perawatan harian (misalnya dinosaurus sebagai sistem biologi menghabiskan sebagian besar masa hidupnya dengan makan dan makan).
3. Sistem sebagai bagian dari sistem lain ; sistem selalu merupakan bagian dari sistem yang lebih besar, dan dapat terbagi menjadi sistem yang lebih kecil.
4. Sistem berkembang ; walaupun tidak berlaku bagi semua sistem tetapi hampir semua sistem selalu berkembang.

Pelaku sistem terdiri dari 7 kelompok :

1. Pemakai : Pada umumnya 3 ada jenis pemakai, yaitu operasional, pengawas dan eksekutif.
2. Manajemen : Umumnya terdiri dari 3 jenis manajemen, yaitu manajemen pemakai yang bertugas menangani pemakaian dimana sistem baru diterapkan, manajemen sistem yang terlibat dalam pengembangan sistem itu sendiri dan manajemen umum yang terlibat dalam strategi perencanaan sistem dan sistem pendukung pengambilan keputusan. Kelompok manajemen biasanya terlibat dengan keputusan yang berhubungan dengan orang, waktu dan uang, misalnya : " sistem tersebut harus mampu melakukan fungsi x,y,z, selain itu harus dikembangkan dalam waktu enam bulan dengan melibatkan programmer dari departemen w, dengan biaya sebesar x".
3. Pemeriksa : Ukuran dan kerumitan sistem yang dikerjakan dan bentuk alami organisasi dimana sistem tersebut diimplementasikan dapat menentukan kesimpulan perlu tidaknya pemeriksa. Pemeriksa biasanya menentukan segala sesuatunya berdasarkan ukuran-ukuran standar yang dikembangkan pada banyak perusahaan sejenis.
4. Penganalisa sistem ; Fungsi-fungsinya antara lain sebagai :

- Arkeolog ; yaitu yang menelusuri bagaimana sebenarnya sistem lama berjalan, bagaimana sistem tersebut dijalankan dan segala hal yang menyangkut sistem lama.
- Inovator ; yaitu yang membantu mengembangkan dan membuka wawasan pemakai bagi kemungkinan-kemungkinan lain.
- Mediator ; yaitu yang menjalankan fungsi komunikasi dari semua level, antara lain pemakai, manajer, programmer, pemeriksa dan pelaku sistem yang lainnya yang mungkin belum punya sikap dan cara pandang yang sama.
- Pimpinan proyek ; Penganalisa sistem haruslah personil yang lebih berpengalaman dari programmer atau desainer. Selain itu mengingat penganalisa sistem umumnya ditetapkan terlebih dahulu dalam suatu pekerjaan sebelum yang lain bekerja, adalah hal yang wajar jika penanggung jawab pekerjaan menjadi porsi penganalisa sistem.

5. Pendesain sistem : Pendesain sistem menerima hasil penganalisa sistem berupa kebutuhan pemakai yang tidak berorientasi pada teknologi tertentu, yang kemudian ditransformasikan ke desain arsitektur tingkat tinggi dan dapat diformulasikan oleh programmer.
6. Programmer : Mengerjakan dalam bentuk program dari hasil desain yang telah diterima dari pendesain.
7. Personel pengoperasian : Bertugas dan bertanggungjawab di pusat komputer misalnya jaringan, keamanan perangkat keras, keamanan perangkat lunak, pencetakan dan backup. Pelaku ini mungkin tidak diperlukan bila sistem yang berjalan tidak besar dan tidak membutuhkan klasifikasi khusus untuk menjalankan sistem.

Hal mendasar dalam pengembangan sistem

Penganalisa sistem merupakan bagian dari tim yang berfungsi mengembangkan sistem yang memiliki daya guna tinggi dan memenuhi kebutuhan pemakai akhir. Pengembangan ini dipengaruhi sejumlah hal,yaitu :

1. Produktifitas, saat ini dibutuhkan sistem yang lebih banyak, lebih

bagus dan lebih cepat. Hal ini membutuhkan lebih banyak programmer dan penganalisa sistem yang berkualitas, kondisi kerja ekstra, kemampuan pemakai untuk mengambangkan sendiri, bahasa pemrograman yang lebih baik, perawatan sistem yang lebih baik (umumnya 50 % sampai 70 % sumber daya digunakan untuk perawatan sistem), disiplin teknis pemakaian perangkat lunak dan perangkat pengembangan sistem yang terotomasi.

2. Realibilitas, waktu yang dihabiskan untuk testing sistem secara umum menghabiskan 50% dari waktu total pengembangan sistem. Dalam kurun waktu 30 tahun sejumlah sistem yang digunakan di berbagai perusahaan mengalami kesalahan dan ironisnya sangat tidak mudah untuk mengubahnya. Jika terjadi kesalahan, ada dua cara yang bisa dilakukan, yaitu melakukan pelacakan sumber kesalahan dan harus menemukan cara untuk mengoreksi kesalahan tersebut dengan mengganti program, menghilangkan sejumlah statement lama atau menambahkan sejumlah statement baru.

3. Maintabilitas, perawatan mencakup ;

- Modifikasi sistem sesuai perkembangan perangkat keras untuk meningkatkan kecepatan pemrosesan (yang memegang peranan penting dalam pengoperasian sistem),
- Modifikasi sistem sesuai perkembangan kebutuhan pemakai. Antara 50% sampai 80% pekerjaan yang dilakukan pada kebanyakan pengembangan sistem dilakukan untuk revisi, modifikasi, konversi,peningkatan dan pelacakan kesalahan.

### 2.2.1.2 Konsep Dasar Informasi

***Informasi***: *data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat.* Jadi ada suatu proses transformasi data menjadi suatu informasi == input - proses – output.

***Data*** merupakan raw material untuk suatu informasi. Perbedaan informasi dan data sangat relatif tergantung pada nilai gunanya bagi manajemen yang memerlukan. Suatu informasi bagi level manajemen

tertentu bisa menjadi data bagi manajemen level di atasnya, atau sebaliknya.

***Representasi informasi***: pelambangan informasi, misalnya: representasi biner.
***Kuantitas informasi:*** satuan ukuran informasi. Tergantung representasi. Untuk representasi biner satuannya: bit, byte, word dll.
***Kualitas informasi***: bias terhadap error, karena: kesalahan cara pengukuran dan pengumpulan, kegagalan mengikuti prosedur prmrosesan, kehilangan atau data tidak terproses, kesalahan perekaman atau koreksi data, kesalahan file histori/master, kesalahan prosedur pemrosesan ketidak berfungsian sistem.
***Umur informasi:*** kapan atau sampai kapan sebuah informasi memiliki nilai/arti bagi penggunanya. Ada *condition informasion* (mengacu pada titik waktu tertentu) dan *operating information* (menyatakan suatu perubahan pada suatu range waktu).

Kualitas Informasi tergantung dari 3 hal, yaitu informasi harus :

- Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan masudnya.
- Tetap pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat.
- Relevan, berarti informasi tersebut menpunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda.

Nilai Informasi ; ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Pengukuran nilai informasi biasanya dihubungkan dengan analisis cost effectiveness atau cost benefit.

### 2.2.1.3 Sistem Informasi

*Suatu sistem terintegrasi yang mampu menyediakan informasi yang bermanfaat bagi penggunanya.*
*Atau ;*
*Sebuah sistem terintegrasi atau sistem manusia-mesin, untuk menyediakan informasi untuk mendukung operasi, manajemen dalam suatu organisasi.*

Sistem ini memanfaatkan perangkat keras dan perangkat lunak komputer, prosedur manual, model manajemen dan basis data.

Dari definisi di atas terdapat beberapa kata kunci :

1. *Berbasis komputer dan Sistem Manusia/Mesin*
   - Berbasis Komputer : Perancang harus memahami pengetahuan komputer dan pemrosesan informasi
   - Sistem manusia mesin : Ada interaksi antara manusia sebagai pengelola dan mesin sebagai alat untuk memroses informasi. Ada proses manual yang harus dilakukan manusia dan ada proses yang terotomasi oleh mesin. Oleh karena itu diperlukan suatu prosedur/manual sistem.
2. *Sistem basis data terintegrasi*

   Adanya penggunaan basis data secara bersama-sama (sharing) dalam sebuah *data base manajemen system*.
3. *Mendukung Operasi*

   Informasi yang diolah dan di hasilkan digunakan untuk mendukung operasi organisasi.

Istilah Sistem Informasi
= *Manajemen Information System*
= *Information Processing System*
= *Information Decision System*
= *Information System.*

Semuanya mengacu pada sebuah sistem informasi berbasis komputer yang dirancang untuk mendukung operasi, manajemen dan fungsi pengambilan keputusan suatu organisasi.

Menurut Robert A. Leitch ; sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Komponen Fisik Sistem Informasi:

1. *Perangkat keras komputer*: CPU, Storage, perangkat Input/Output, Terminal untuk interaksi, Media komunikasi data
2. *Perangkat lunak komputer*: perangkat lunak sistem (sistem operasi dan utilitinya), perangkat lunak umum aplikasi (bahasa pemrograman), perangkat lunak aplikasi (aplikasi akuntansi dll).
3. *Basis data*: penyimpanan data pada media penyimpan komputer.
4. *Prosedur*: langkah-langkah penggunaan sistem
5. *Personil* untuk pengelolaan operasi (SDM), meliputi:
   - *Clerical personnel* (untuk menangani transaksi dan pemrosesan data dan melakukan inquiry = operator);
   - *First level manager*: untuk mengelola pemrosesan data didukung dengan perencanaan, penjadwalan, identifikasi situasi out-of-control dan pengambilan keputusan level menengah ke bawah.
   - *Staff specialist*: digunakan untuk analisis untuk perencanaan dan pelaporan.
   - *Management*: untuk pembuatan laporan berkala, permintaan khsus, analisis khusus, laporan khsusus, pendukung identifikasi masalah dan peluang.

### 2.2.1.4 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (building blok), 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 dimasukkan, 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 yag sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Komponen output

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, neghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

5. Komponen hardware

Hardware berperan penting sebagai suatu media 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

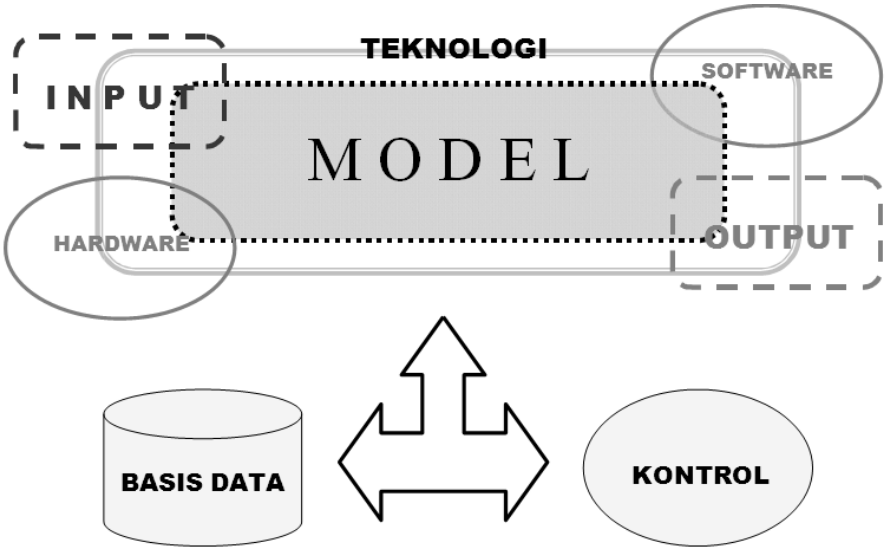
Software berfungsi sebagai tempat untuk mengolah,menghitung dan memanipulasi data yang diambil dari hardware untuk menciptakan suatu informasi.

7. Komponen basis data

Basis data (database) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di pernagkat 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 sistem 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 2.2 *Interaksi Sistem Informasi*

**2.2.1.5 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 di maksudkan yaitu operator komputer, analis sistem, programmer, personil data entry, dan manajer sistem informasi/EDP

2. Prosedur

Prosedur merupakan elemen fisik. Hal ini di sebabkan 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 komputer (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 pernagkat lunak yang terdiri atas program yang secara spesifik dibuat untuk setiap aplikasi.

5. Basis data

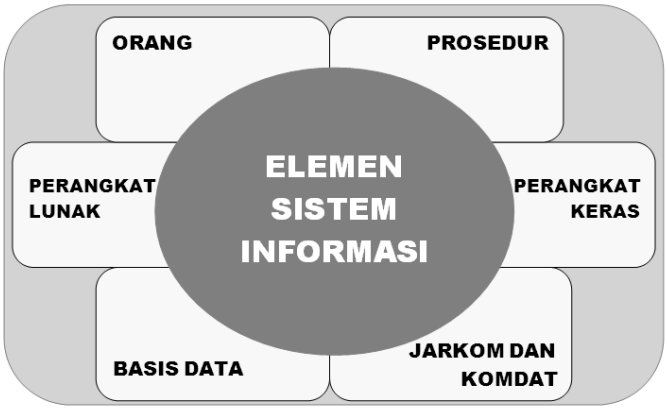
File yang berisi program dan data dibuktikan dengan adanya media penyimpanan secara fisik seperti diskette, hard disk, magnetic tape, dan sebagainya. File juga meliputi keluaran tercetak dan catatan lain diatas kertas, mikro film, dan lain sebagainya.

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 computer-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 2.3 *Hubungan Elemen Sistem Informasi*

**2.2.1.6 Elemen Sistem Informasi**

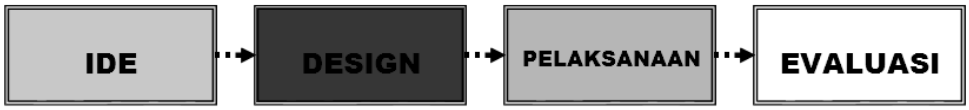
Sistem informasi dapat di bentuk sesuai kebutuhan organisasi masing-masing. Oleh karena itu, untuk dapat menerapkan sistem yang efektif dan efisiendiperlukan perencanaan, pelaksanaan, pengaturan, dan evaluasi sesuai keinginan masing-masing organisasi. Guna dari sistem yang efektif dan efisien tidak lain 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 ke dalam sistem.
Tingkat IV : Kontrol, memeriksa tingkat pelaksanaan dijalankan sesuai dengan design
Tingkat V : Evaluasi, memeriksa apakah perubahan yang terjadi sesuai dengan tujuan semula.
Tingkat VI : Tindak lanjut, melaksanakn 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 :




**2.2.1.7 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 di klasifikasikan 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. Sistem fisik merupakan sistem secara fisik, misalnya sistem komputer.

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 deterministik dan sistem probabilistik

Sistem deterministik adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi. Sistem probabilistik dalah 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 di pengaruhi 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 Rekayasa Perangkat Lunak

Perangkat Lunak Merupakan program-program komputer dan dokumentasi yang berkaitan. Produk perangkat lunak dibuat untuk pelanggan tertentu ataupun untuk pasar umum terdiri dari: Generik – dibuat untuk dijual ke suatu kumpulan pengguna yang berbeda Bespoke (custom) – dibuat untuk suatu pengguna tunggal sesuai dengan spesifikasinya. Rekayasa perangkat lunak berasal dari 2 kata yaitu Software ( Perangkat Lunak) dan Engineering (Rekayasa).

Perangkat Lunak (Software) adalah source code pada suatu program atau sistem. Perangkat lunak tidak hanya dokumentasi terhadap source code tapi juga dokumentasi terhadap sesuatu yang dibutuhkan selama pengembangan, instalasi, penggunaan dan pemeliharaan sebuah sistem.

*Engineering* atau Rekayasa adalah aplikasi terhadap pendekatan sistematis yang berdasar atas ilmu pengetahuan dan matematis serta aplikasi tentang produksi terhadap struktur,mesin, produk, proses atau sistem.

Rekayasa Perangkat Lunak adalah suatu disiplin rekayasa yang berkonsentrasi terhadap seluruh aspek.

**Produk perangkat lunak** mengadopsi pendekatan yang sistematis dan terorganisir terhadap pekerjaannya dan menggunakan tool yang sesuai serta teknik yang ditentukan berdasarkan masalah yang akan dipecahkan, kendala pengembangan dan sumber daya yang tersedia

Rekayasa Perangkat Lunak (RPL) juga merupakan pendekatan sistematis dan matematis untuk membangun, memelihara dan mengenyahkan perangkat lunak. Dari cara pandang lain, RPL adalah pendekatan sistematis untuk merekayasa perangkat lunak yang handal / bermutu, tepat waktu dan dengan biaya yang optimal.

### 2.2.2.1 Mutu Perangkat Lunak

Terdapat 3 pihak (minimal) yang mempengaruhi mutu perangkat lunak yaitu :

- **Sponsor**

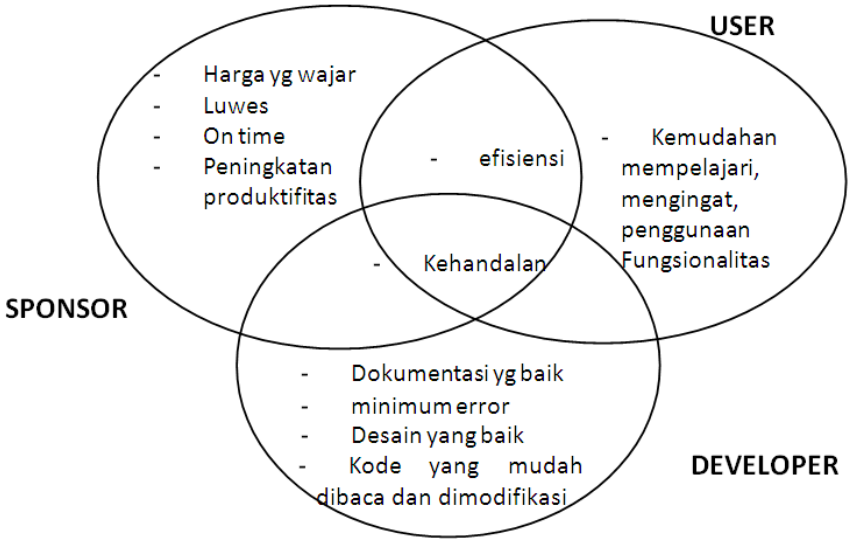
Seseorang atau organisasi yang membiayai/membayar selama pengembangan atau perantaraan sistem software dan biasanya mempunyai respon terhadap pengembangan sistem software itu sendiri dengan melibatkan perhitungan biaya yang optimal.

- **User**

Setiap orang yang secara langsung berinteraksi terhadap eksekusi software, yang secara langsung memberi input ke komputer dan menggunakan/menikmati output dari komputer.

- **Developer**

Seseorang atau organisasi yang memberikan modifikasi dan memelihara terhadap error serta mengembangkan sistem software tersebut.




Gambar 2.4 Sisi Pandang dari komponen kategori terhadap Mutu Perangkat Lunak

### 2.2.2.2 Kategori Perangkat Lunak

Kategori Perangkat lunak secara umum dapat dikelompokkan sebagai berikut:

1. **Perangkat Lunak Sistem**, Sekumpulan program yang ditulis untuk melayani program-program yang lain. Seperti kompiler, editor dan utilitas pengatur file.
2. **Perangkat Lunak Real-Time**, Program-program yang memonitor/menganalisi/mengontrol kejadian dunia nyata pada saat terjadinya ( real-time event)
3. **Perangkat Lunak Bisnis**, memroses informasi bisnis spt payroll, inventory dll.
4. **Perangkat Lunak Teknik dan Ilmu Pengetahuan**, ditandai dengan penggunaan algoritma *number crunching.*
5. ***Embedded Software***, produk yang ada dalam *read-only memory* dan dipakai untuk mengontrol hasil dan sistem untuk keperluan konsumen dan pasar industri
6. **Perangkat Lunak Komputer Personal**, sesuai kebutuhan personal spt pengolah kata,angka dan manajamen database
7. **Perangkat Lunak Kecerdasan Buatan**, menggunakan algoritma non-numeris untuk memecahkan masalah kompleks yang tidak sesuai untuk perhitungan atau analisis secara langsung.

### 2.2.2.3 Karakteristik Perangkat Lunak

Atribut Perangkat Lunak seharusnya memberikan pengguna kebutuhan fungsionalitas dan unjuk kerja yang dapat di rawat, berguna.

Dalam Buku Software Engineering Ian Sommerville, Perangkat Lunak mempunyai Karakteristik sebagai berikut:

1. Maintanability (Dapat Dirawat), Perangkat Lunak harus dapat memenuhi perubahan kebutuhan
2. Dependability, Perangkat Lunak harus dapat dipercaya
3. Efisiensi, Perangkat Lunak harus efisien dalam penggunaan resource
4. Usability, Perangkat Lunak harus dapat digunakan sesuai dengan yang direncanakan

### 2.2.2.4 Proses Perangkat Lunak

Proses Perangkat Lunak merupakan Sekumpulan aktifitas yang memiliki tujuan untuk pengembangan ataupun evolusi perangkat lunak.

Aktifitas umum dalam semua proses perangkat lunak terdiri dari:

1. *Software Specification* – apa yang harus dilakukan oleh perangkat lunak dan batasan/kendala pengembangannya
2. *Software Development* – proses memproduksi sistem perangkat lunak
3. *Software Validation* – pengujian perangkat lunak terhadap keinginan penggunak
4. *Software Evolution* – perubahan perangkat lunak berdasarkan perubahan keinginan.

Suatu proses model adalah suatu representasi abstrak suatu model. Proses model menampilkan suatu deskripsi suatu proses dari beberapa perspektif tertentu,

Proses Perangkat Lunak dapat dikatakan sebagai aktifitas yang saling terkait (koheren) untuk menspesifikasikan, merancang, implementasi dan pengujian sistem perangkat lunak.

### 2.2.2.5 Karakteristik Proses Perangkat Lunak

Karakteristik Proses Perangkat Lunak terdiri dari:

1. *Understandability*, membuat proses secara eksplisit didefinisikan dan bagaimana sehingga mudah untuk mengerti definisi proses
2. *Visibility,* Aktifitas proses menghasilkan hasil yang jelas sehingga tahapan proses yang dilakukan terlihat
3. *Supportability,* Aktifitas Proses dapat didukung atas CASE tools *Acceptability,* Penerimaan atas proses yang terdefinisi dan yang digunakan oleh Engineer selama pembangunan Produk Perangkat Lunak.
4. *Reliability,* Proses didesain dalam suatu metode untuk dihindarkan dari kesalahan
5. *Robustness,* Proses dapat meneruskan dalam masalah yang tidak diharpkan terjadi
6. *Maintainabiity,* Proses yang merefleksi atas perubahan thd permintaan

atau perbaikan proses yang diidentifikasi

7. *Rapidity,* bagaimana cepat dapat berjalan atas proses pengiriman atau implementasi sebuah sistem dari Spesifikasi yang ada sampai selesai

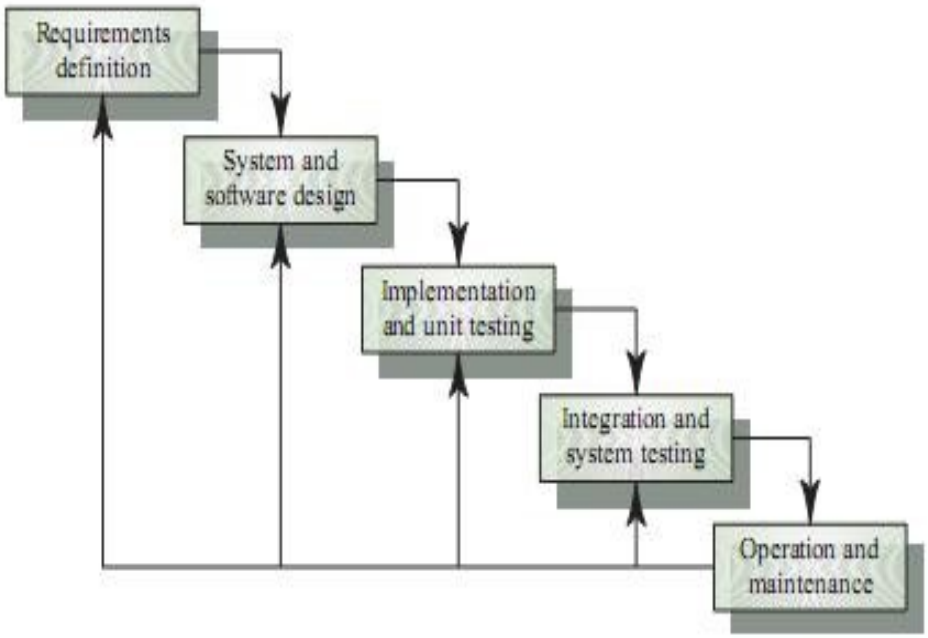
### 2.2.2.6 Daur Hidup Pembangunan Perangkat Lunak

Di dalam pengembangan rekayasa perangkat lunak biasanya dipandu dengan pemodelan dengan Daur Hidup Perangkat Lunak (*Software Development Life Cycle*).

Tak ada standar sehingga bervariasi model proses u/ menggambarkan rekayasa daur hidup p.l. Namun tahap-tahap yang prinsipal terhadap pemetaan model proses kedalam aktifitas pengembangan yang fundamental alalah sbb:

1. *Requirement Analysis and definition*
2. *System and Software Design*
3. *Implementation and unit testing*
4. *Integration and system Testing*
5. *Operation and maintenance*

**Model Air Terjun (Water Fall)**



Model di atas secara esensial sama dengan model „waterfall" atau air terjun. Kita dapat menganalogikan daur hidup p. L dengan daur hidup manusia dari benih s/d tua dan meninggal.

### 2.2.2.7 Model Proses Perangkat Lunak

Model Proses Perangkat Lunak merupakan suatu representasi proses perangkat lunak yang disederhanakan, dipresentasikan dar perspektif khusus. Contoh perspektif proses:

- Perspektif Alur-kerja (workflow) - barisan kegiatan
- Perspektif Alur Data (Data flow) – alur informasi
- Perspektif Peran/Aksi – siapa melakukan apa.

Menurut Ian Somerville, Model proses secara umum terdiri dari:

1. **Pendekatan Model Air terjun (Water fall)**, Menempatkan semua aktifitas sesuai dengan tahapan pada model Waterfall dengan memisahkan dan membedakan antara spesifikasi dan pengembangan
2. **Pengembangan yang berevolusi**, Pendekatan yang melanjutkan Aktifitas satu dan yang lainnya dari Spesifikasi dan pengembangan serta validasi secara cepat
3. **Pengembangan sistem Formal**, Pendekatan aktifitas bersasar suatu model sistem matematika yang ditransformasikan ke implementasi,
4. **Pengembangan Sstem berbasis Re-use (penggunaan ulang) komponen**, sistem dibangun dari komponen yang sudah ada dengan fokus integrasi sistem.

### 2.2.2.8 Model Waterfall (Air Terjun)

Fase Model Air Terjun

1. Analisis Kebutuhan dan pendefinisiannya
2. Perancangan sistem dan Perangkat Lunak
3. Implementasi dan unit testing
4. Integrasi dan pengujian sistem
5. Pengoperasian dan perawatan

Proses kembali ke state sebelumnya untuk mengantisipasi perubahan Setelah proses menuju ke suatu state di bawahnya adalah sangat sulit.

Masalah pada Model Air Terjun:

- Partisi projek ke stages yang berbeda tidak fleksibel.
- Hal ini mengakibatkan sulitnya untuk merespon perubahan kebutuhan pengguna
- Oleh sebab itu model ini hanya cocok digunakan apabila kebutuhan pengguna sudah dimengerti dengan baik,

**2.2.2.9 Biaya Perangkat Lunak**

Sekitar 60% untuk biaya pengembangan, 40% biaya pengujian. Untuk perangkat lunak berbasis pengguna (custom), biaya evolusi biasanya melebihi biaya pengembangan.

Biaya beragam tergantung pada tipe sistem yang akan dikembangkan dan kebutuhan sistem seperti unjuk kerja dan kehandalan sistem, Distribusi biaya bergantung pada model pengembangan yang digunakan.

**2.2.2.10Pemilihan Sebuah Bahasa Pemrograman**

Pemilihan terhadap penggunaan Bahasa Pemrograman dalam pengembangan sistem merupakan hal yang krusial. Hal ini sangat esensi dalam siklus Hidup Pengembangan Sistem. Pada tahapan Coding dan Testing difasilitasi bahasa yang terpilih dan pada Tahapan Pemeliharaan lebih mudah dikerjakan. Apalagi dihubungkan dengan biaya dalam pengembangan perangkat lunak.

Terdapat 2 faktor ang berhubungan terhadap Pemilihan Bahasa Pemrograman yaitu:

1. Perihal Pragmatik Pemilihan Bahasa
2. Bahasa Pemrograman yang dipilih

Pragmatik dalam Pemilihan Bahasa Pemrograman mengandung perihal berikut:

1. Sponsor Requirement, Permintaan Sponsor
2. Knowledge of coders, Pengetahuan yang mudah dipahami Programmer
3. Languages used in previous and/or concurrent projects, Bahasa Pemrograman yng digunakan proyek sebelumnya atau proyek yang berbarengan terkait dengan Pengetahuan Programmer
4. Availability and quality of language compiler, Ketersediaan dan kualitas Kompiler Bahasa Pemrograman yang sesuai target hardware yang digunakan
5. Availability of supporting software development tools, Ketersediaan Alat Bantu Perangkat lunak Pendukung Editor, Debugger, Linker dan yang lainnya
6. Portability, Sistem yang dikembangkan dapat beroperasi di berbagai mesinkomputer dan berjalan pada aneka Sistem Operasi yang berbeda

**2.2.2.11Programming-in-the Small Concerns & Programming-in-the Large Concerns**

Karakteristik Bahasa Pemrograman secara individu dipertimbangkan ketika memilih sebuah bahasa target yang akan digunakan pada Pengembangan Perangkat Lunak yang efisien, reliable dan pada Pemeliharaan perangkat lunak.

Seorang Software Engineer dapat memperlihatkan fitur-fitur bahasa pemrograman yang mendukung pada pengembangan Perangkat Lunak berasal dari 2 aspek yang berbeda (Rujukan Bell, Morrey & Pugh, 1987) yaitu:

1. Programming-in-the Small, Menguji atau mencoba fitur-fitur yang mendukung dengan Pengkodean program modul-modul tunggal dan program-program kecil oleh kepentingan Programmer secara individu.
2. Programming-in-the-Large, Pemrograman ini merujuk pada pengembangan sebuah system yang keseluruhannya dipengeruhi oleh koordinasi atas sekelompok orang (Sofware Engineer), dimana setiap engineer membuat respon komponen-komponen pada system dengan bagian yang berbeda-beda.

### *1. Programming-in-the Small*

Dalam Pemrograman kekuatan bahasa adalah kemampuan membawa suatu pekerjaan pembuatan program yang diinginkan secara mudah dan dengan membuat pengaruh kebutuhan pemrograman itu kecil. Kekuatan bahasa itu secara langsung berkaitan dengan fitur-fitur yang ada dan struktur data yang mendukung bahasa tersebut.

Karakteristik Fitur-fitur yang terkait dengan pemrograman dengan skala kecil (Programming-in-the Small) terdiri dari:

a. Sifat Simplicity, Clarity dan Orthogonality dari bahasanya

b. Sintaks dari bahasa pemrogramannya

c. Jumlah dan tipe kontrol struktur (Decision structure, loop control structure dan exception handling)

d. Abstraksi Data terhadap tipe struktur datanya

Simplicity sebuah bahasa pemrograman merupakan ukuran dari kamus data dari bahasa seperti jumlah operator, operan dan reserved word (seperti if..then dsb)dalam bahasa itu. Simplicity ini perihal yang diinginkan karena sifat ini membuat programer menjadi familier terhadap keseluruhan isi bahasa yang digunakan.

Clarity merupakan tingkat dari bahasa berkaitan dengan pengertian sintaks natural, pemahaman dan ketidakbingungan programmer dalam menggunakan bahasa pemrograman sampai berjalan.

Orthogonality merupakan tingkat dari programer bebas mengkombinasikan fitur-fitur yang ada di bahasa pemrograman itu seperti pemanfaatan fungsi yang memiliki nilai balik untuk diolah langsung dengan nilai yang lain dan tipe data tertentu.

### *2. Programming-in-the Large*

Fitur-fitur pada pemrograman ini terdapat pada pembuatan modul antarmuka, dukungan terhadap fungsional dan abstraksi data dan bagian kompilasi terhadap modul-modul program secara independen oleh linkage editor atau linker.

Programming ini dibutuhkan pada saat sebuah system mengandung banyak baris kode atau mempunyai skala yang besar yang membutuhkan pendekatan pengembangan sistem secara integrasi terhadap sekelompok proyek system.

Konsep penting dalam pemrograman terkait dengan penyelesaian masalah oleh seorang programmer adalah abstraksi prosedural yang berbentuk modul prosedur/fungsi dan abstraksi data. Pada Pemrograman berskala besar (Programming-in-the Large) meminta abstraksi pada level yang lebih tinggi dengan dukungan abstraksi data yang sesuai dengan modul prosedural yang terkait.

**2.2.2.12Pengembangan Perangkat Lunak**

Pengembangan perangkat lunak adalah suatu proses dimana kebutuhan pemakai diterjemahkan menjadi produk perangkat lunak. Proses ini mencakup aktivitas penerjemahan kebutuhan pemakai menjadi kebutuhan perangkat lunak, transformasi kebutuhan perangkat lunak menjadi desain, penerapan desain menjadi kode program, uji coba kode program, dan instalasi serta pemeriksaan kebenararan perangkat lunak untuk operasional [4].

Berdasarkan pengertian tersebut, secara umum dapat dikatakan bahwa proses pengembangan perangkat lunak mengikuti tahap-tahap:

1. Menentukan APA yang harus dikerjakan oleh perangkat lunak dalam satu rentang waktu tertentu.
2. Mendefinisikan BAGAIMANA perangkat lunak dibuat, mencakup arsitektur perangkat lunaknya, antarmuka internal, algoritma, dan sebagainya.
3. Penerapan (penulisan program) dan pengujian unit-unit program.
4. Integrasi dan pengujian modul-modul program.
5. Validasi perangkat lunak secara keseluruhan (pengujian sistem).

### 2.2.2.13Siklus Pengembangan Perangkat Lunak

Siklus pengembangan perangkat lunak atau sering disebut juga dengan siklus hidup perangkat lunak adalah:

- Periode waktu yang diawali dengan keputusan untuk mengembangkan produk perangkat lunak dan berakhir setelah perangkat lunak diserahkan. Umumnya siklus pengembangan ini terdiri dari tahap analisis kebutuhan, perancangan, penerapan, pengujian, dan instalasi serta pemeriksaan.
- Periode waktu yang diawali dengan keputusan untuk mengembangkan produk perangkat lunak dan berakhir saat produk tidak dapat ditingkatkan lebih jauh lagi oleh pengembang.

### 2.2.2.14Model Proses Pengembangan Perangkat Lunak

Model proses perangkat lunak (atau disebut juga paradigma rekayasa perangkat lunak) adalah suatu strategi pengembangan yang memadukan lapisan proses, metode, dan alat serta tahap-tahap generik. Model proses untuk rekayasa perangkat lunak dipilih berdasarkan sifat proyek dan aplikasi, metode dan alat yang digunakan, serta pengendalian dan hasil yang diinginkan. Berikut adalah beberapa model proses pengembangan perangkat lunak.

#### *1. Linear Sequential Model*

Linear sequential model (atau disebut juga "classic life cycle" atau "waterfall model") adalah metode pengembangan perangkat lunak dengan pendekatan sekuensial dengan cakupan aktivitas:

1. Pemodelan dan rekayasa sistem/informasi. Menetapkan kebutuhan untuk seluruh elemen sistem dan kemudian memilah mana yang untuk pengembangan perangkat lunak.
2. Analisis kebutuhan perangkat lunak
3. Perancangan
4. Pembuatan kode
5. Pengujian
6. Pemeliharaan

Beberapa kelemahan linear sequential model:

1. Proyek yang sebenarnya jarang mengikuti alur sekuensial, sehingga perubahan yang terjadi dapat menyebabkan hasil yang sudah didapat tim harus diubah kembali.
2. Linear sequential model mengharuskan semua kebutuhan pemakai sudah dinyatakan secara eksplisit di awal proses, tetapi kadang-kadang hal ini tidak dapat terlaksana karena kesulitan yang dialami pemakai saat akan mengungkapkan semua kebutuhannya tersebut.
3. Pemakai harus bersabar karena versi dari program tidak akan didapat sampai akhir rentang waktu proyek.
4. Adanya waktu menganggur bagi pengembang, karena harus menunggu anggota tim proyek lainnya menuntaskan pekerjaannya.

### *2. Prototyping Model*

Pendekatan prototyping model digunakan jika pemakai hanya mendefinisikan objektif umum dari perangkat lunak tanpa merinci kebutuhan input, pemrosesan dan outputnya, sementara pengembang tidak begitu yakin akan efisiensi algoritma, adaptasi sistem operasi, atau bentuk interaksi manusia- mesin yang harus diambil. Cakupan aktivitas prototyping model terdiri dari:

1. Mendefinisikan objetif secara keseluruhan dan mengidentifikasi kebutuhan yang sudah diketahui.
2. Melakukan perancangan secara cepat sebagai dasar untuk membuat prototype
3. Menguji coba dan mengevaluasi prototype dan kemudian melakukan

penambahan dan perbaikan-perbaikan terhadap prototype yang sudah dibuat.

Kelemahan prototyping model:

1. Walaupun pemakai melihat berbagai perbaikan dari setiap versi prototype, tetapi pemakai mungkin tidak menyadari bahwa versi tersebut dibuat tanpa memperhatikan kualitas dan pemeliharaan jangka panjang.
2. Pengembang kadang-kadang membuat kompromi implementasi dengan menggunakan sistem operasi yang tidak relevan dan algoritma yang tidak efisien.

### *3. RAD (Rapid Application Development) Model*

Merupakan model proses pengembangan perangkat lunak secara linear sequential yang menekankan pada siklus pengembangan yang sangat singkat. Pendekatan RAD model mempunyai cakupan:

1. Pemodelan bisnis
2. Pemodelan data
3. Pemodelan proses
4. Pembuatan aplikasi
5. Pengujian dan pergantian

Kelemahan RAD model:

1. Untuk proyek dengan skala besar, RAD membutuhkan sumber daya manusia yang cukup untuk membentuk sejumlah tim RAD.
2. RAD membutuhkan pengembang dan pemakai yang mempunyai komitmen untuk melaksanakan berbagai aktivitas melengkapi sistem dalam kerangka waktu yang singkat.
3. Akan menimbulkan masalah jika sistem tidak dapat dibuat secara modular.
4. RAD tidak cocok digunakan untuk sistem yang mempunyai resiko teknik

yang tinggi.

### *4. Incremental Model*

Merupakan kombinasi linear sequential model (diaplikasikan secara berulang) dan filosofi pengulangan dari prototyping model. Setiap tahapan linear sequential menghasilkan *deliverable increment* bagi perangkat lunak, dimana *increment* pertamanya merupakan sebuah produk inti yang mewakili kebutuhan dasar sistem. Produk inti ini nantinya dikembangkan menjadi *increment-increment* selanjutnya setelah digunakan dan dievaluasi sampai didapat produk yang lengkap dan memenuhi kebutuhan pemakai.

Kelemahan incremental model:

1. Hanya akan berhasil jika tidak ada *staffing* untuk penerapan secara menyeluruh.
2. Penambahan staf dilakukan jika hasil *incremental* akan dikembangkan lebih lanjut.

### *5. Spiral Model*

Merupakan model proses perangkat lunak yang memadukan wujud pengulangan dari model prototyping dengan aspek pengendalian dan sistematika dari linear sequential model. Dalam model ini perangkat lunak dikembangkan dalam suatu seri *incremental release*. Spiral model dibagi menjadi 6 aktivitas kerangka kerja sebagai berikut:

1. Komunikasi dengan pemakai
2. Perencanaan
3. Analsis resiko
4. Rekayasa
5. Konstruksi dan pelepasan
6. Evaluasi

Kelemahan spiral model:

1. Sulit untuk meyakinkan pemakai (saat situasi kontrak) bahwa penggunaan pendekatan ini akan dapat dikendalikan.

2. Memerlukan tenaga ahli untuk memperkirakan resiko, dan harus mengandalkannya supaya sukses.
3. Belum terbukti apakah metode ini cukup efisien karena usianya yang relatif baru.

### *6. Component Assembly Model*

Menggabungkan berbagai karakteristik dari spiral model. Pembuatan aplikasi dengan pendekatan model ini dibangun dari komponen-komponen perangkat lunak yang sudah dipaketkan sebelumnya dengan cakupan aktivitas sebagai berikut:

1. Mengidentifikasi calon-calon komponen (kelas objek)
2. Melihat komponen-komponen dalam pustaka
3. Mengekstrak komponen jika ada
4. Membangun komponen jika tidak ada
5. Menyimpan komponen baru pada pustaka
6. Mengkontruksi iterasi ke-n dari sistem

### *7. Fourth Generation Techniques (4GT)*

Menggunakan perangkat bantu yang akan membuat kode sumber secara otomatis berdasarkan spesifikasi dari pengembang perangkat lunak. Hanya digunakan untuk mengembangkan perangkat lunak yang menggunakan bentuk bahasa khusus atau notasi grafik yang diselesaikan dengan syarat yang dimengerti pemakai. Cakupan aktivitas 4GT:

1. Pengumpulan kebutuhan.
2. Translasi kebutuhan menjadi prototype operasional, atau langsung melakukan implementasi secara langsung dengan menggunakan bahasa generasi keempat (4GL) jika aplikasi relatif kecil.
3. Untuk aplikasi yang cukup besar, dibutuhkan strategi perancangan sistem walaupun 4GL akan digunakan.
4. Pengujian.
5. Membuat dokumentasi.

6. Melaksanakan seluruh aktivitas untuk mengintegrasikan solusi-solusi yang membutuhkan paradigma rekayasa perangkat lunak lainnya.

Salah satu keuntungan penggunaan model 4GT adalah pengurangan waktu dan peningkatan produktivitas secara besar, sementara kekurangannya terletak pada kesulitan penggunaan perangkat bantu dibandingkan dengan bahasa pemrograman, dan juga kode sumber yang dihasilkannya tidak efisien.

### 2.2.3 Jaringan Komputer

#### 2.2.3.1 Pengertian Jaringan Komputer

Jaringan komputer merupakan sekumpulan komputer berjumlah banyak yang terpisah-pisah akan tetapi saling berhubungan dalam melaksanakan tugasnya. Dua buah komputer misalnya dikatakan terkoneksi bila keduanya dapat saling bertukar informasi. Bentuk koneksi dapat melalui: kawat tembaga, serat optik, gelombang mikro, satelit komunikasi.

Dalam suatu jaringan computer, pengguna harus secara eksplisit:

- masuk atau log in ke sebuah mesin
- menyampaikan tugas dari jauh
- memindahkan file-file
- menangani sendiri secara umum seluruh manajemen jaringan

Jaringan komputer menjadi penting bagi manusia dan organisasinya karena jaringan komputer mempunyai tujuan yang menguntungkan bagi mereka. Tujuan jaringan komputer adalah untuk:

1. Resource sharing/ berbagi sesumber: seluruh program, peralatan dan data yang dapat digunakan oleh setiap orang yang ada dijaringan tanpa dipengaruhi lokasi sesumber dan pemakai. Misalnya: Staff BIRO Akademik mengirimkan daftar mahasiswa baru ke perpustakaan dalam bentuk print out dengan langsung mencetaknya di printer perpustakaan dari computer di BIRO akademik. Atau sebaliknya staff perpustakaan mendapatkan langsung file daftar mahasiswa baru yang disimpan di komputer staff BIRO akademik.

2. High reliability/kehandalan tinggi: tersedianya sumber-sumber alternatif kapanpun diperlukan. Misalnya pada aplikasi perbankan atau militer, jika salah satu mesin tidak bekerja, kinerja organisasi tidak terganggu karena mesin lain mempunyai sumber yang sama.
3. Menghemat uang: membangun jaringan dengan komputer-komputer kecil lebih murah dibandingkan dengan menggunakan mainframe. Data disimpan di sebuah komputer yang bertindak sebagai server dan komputer lain yang menggunakan data tersebut bertindak sebagai client. Bentuk ini disebut Client-server.
4. Scalability/ skalabilitas: meningkatkan kinerja dengan menambahkan komputer server atau client dengan mudah tanpa mengganggu kinerja komputer server atau komputer client yang sudah ada lebih dulu.
5. Medium komunikasi: memungkinkan kerjasama antar orang-orang yang saling berjauhan melalui jaringan komputer baik untuk bertukar data maupun berkomunikasi.
6. Akses informasi luas: dapat mengakses dan mendapatkan informasi dari jarak jauh
7. Komunikasi orang-ke-orang: digunakan untuk berkomunikasi dari satu orang ke orang yang lain
8. Hiburan interaktif

Dalam pengenalan jaringan komputer, pembahasan dilihat dari dua aspek: **perangkat keras** dan **perangkat lunak**. Dalam perangkat keras pengenalan meliputi jenis transmisi, dan bentuk-bentuk jaringan komputer atau topologi. Sedangkan dalam pembahasan perangkat lunaknya akan meliputi susunan protokol dan perjalanan data dari satu komputer ke komputer lain dalam suatu jaringan.

### 2.2.3.2 Perangkat Keras

Ada dua klasifikasi jaringan komputer yaitu dibedakan berdasarkan teknologi transmisi dan jarak.

**1. Teknologi Transmisi**

Secara garis besar ada dua jenis teknologi transmisi:

a. Jaringan broadcast

Memiliki saluran komunikasi tunggal yang dipakai bersama-sama oleh semua mesin yang ada pada jaringan. Pesan-pesan berukuran kecil, disebut paket dan dikirimkan oleh suatu mesin kemudian diterima oleh mesin-mesin yang lainnya. Bagian alamat pada paket berisi keterangan

tentang kepada siapa paket ditujukan. Saat menerima sebuat paket, mesin akan cek bagian alamat, jika paket tersebut untuk mesin itu, maka mesin akan proses paket itu. Jika bukan maka mesin mengabaikannya.

b. Jaringan point-to-point

Terdiri dari beberapa koneksi pasangan individu dari mesin-mesin. Untuk pergi dari satu sumber ke tempat tujuan, sebuah paket pada jaringan jenis ini mungkin harus melalui satu atau lebih mesin-mesin perantara. Seringkali harus melalui banyak rute (route) yang mungkin berbeda jaraknya. Karena itu algoritma routing memegang peranan penting pada jaringan point-to-point.

Sebagai pegangan umum (walaupun banyak pengecualian), jaringan yang lebih kecil dan terlokalisasi secara geografis cenderung memakai broadcasting, sedangkan jaringan yang lebih besar umumnya mengunakan point-to-point.

Sebagai pegangan umum (walaupun banyak pengecualian), jaringan yang lebih kecil dan terlokalisasi secara geografis cenderung memakai broadcasting, sedangkan jaringan yang lebih besar umumnya mengunakan point-to-point.

**2. Jarak**

Jarak adalah hal yang penting sebagai ukuran klasifikasi karena diperlukan teknik-teknik yang berbeda untuk jarak yang berbeda. Tabel berikut

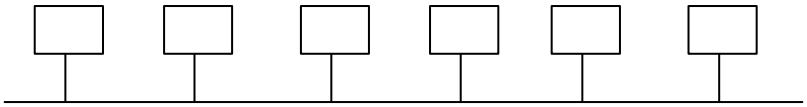
menggambarkan hubungan antar jarak dan prosessor yang ditempatkan pada tempat yang sama.

| **Jarak antar prosessor** | **Prosesor di tempat yang sama** | **Jenis jaringan** |
|---|---|---|
| 0.1 m | Papan rangkaian | Data flow machine: komputer-komputer paralel, memiliki beberapa unit fungsi yang semuanya bekerja untuk program yang sama |
| 1 m | Sistem | Multicomputer, sistem yang berkomunikasi dengan cara mengirim pesan-pesannya melalui bus pendek dan sangat cepat. |
| 10 m | Ruangan | Local Area Network (LAN) |
| 100 m | Gedung | Local Area Network (LAN) |
| 1 km | Kampus | Local Area Network (LAN) |
| 10 km | Kota | Metropolitan area Network (MAN) |
| 100 km | Negara | Wide Area Network (WAN) |
| 1.000 km | Benua | Wide Area Network (WAN) |
| 10.000 km | Planet | Internet |

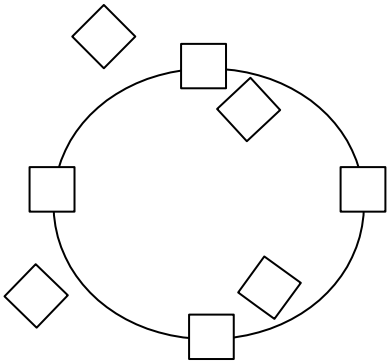
**Jalan data elektrik yang mana bit dikirimkan dalam CPU, antar CPU dan komponen-komponen lain di mainboard*.

Disini secara terbatas dan sederhana dijelaskan secara singkat LAN, MAN, WAN
dan Internet.

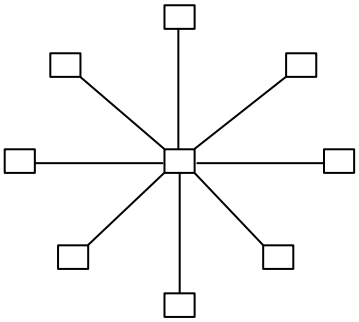
a. **LAN**: menghubungkan komputer-komputer pribadi dalam kantor perpusahaan, pabrik atau kampus: LAN dapat dibedakan dari jenis jaringan lainnya berdasarkan 3 karakteristik: ukuran, teknologi transmisi dan topologi jaringan.



Gambar 2.6 Topologi LAN jenis broadcast : BUS (kabel linier)



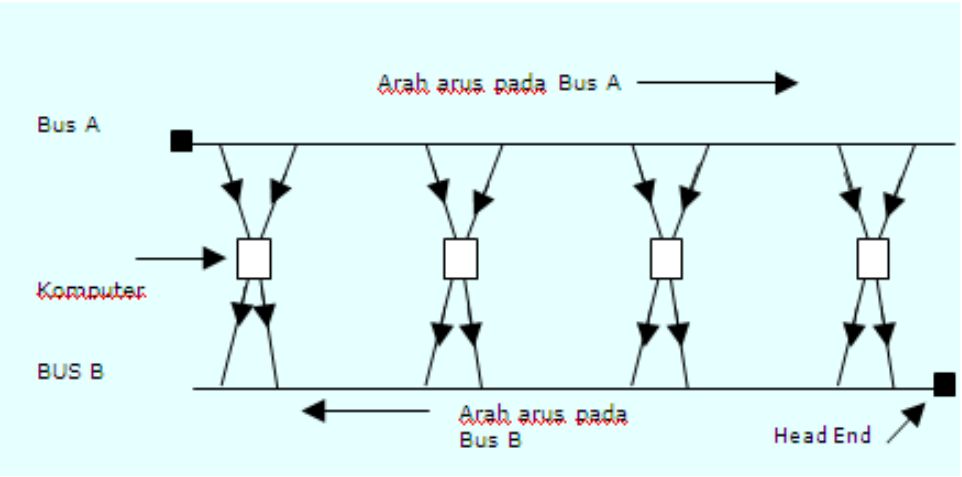
Gambar 2.7 Topologi LAN jenis broadcast: RING



Gambar 2.8 Topologi LAN jenis point-to-point : STAR

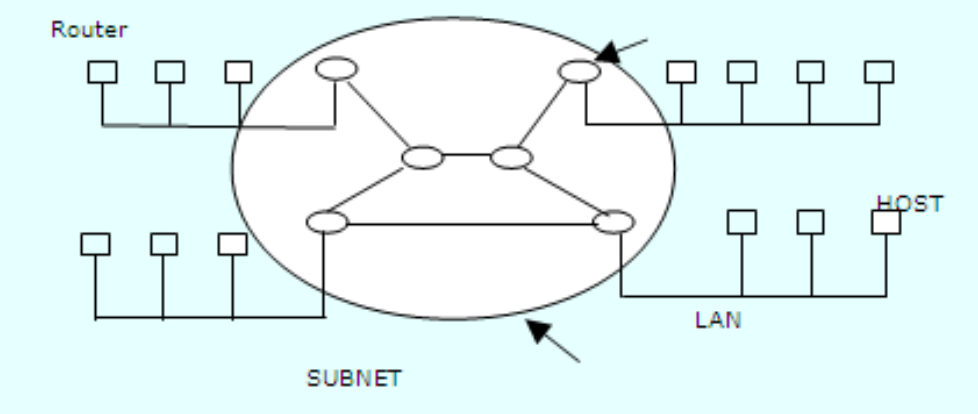
**b. MAN**

Merupakan versi LAN ukuran lebih besar dan biasanya memakai teknologi yang sama dengan LAN. MAN mampu menunjang data dan suara, dan bahkan dapat berhubungan dengan jaringan televisi kabel. MAN hanya memiliki sebuah atau dua buah kabel dan tidak mempunyai elemen switching, yang berfungsi untuk mengatur paket melalui beberapa kabel output.




**c. WAN**

Mencakup daerah geografis yang luas, seringkali mencakup negara atau benua. WAN terdiri dari kumpulan mesin yang bertujuan untuk menjalankan program-program (aplikasi) pemakai. Mesin ini disebut HOST. HOST dihubungkan oleh sebuah subnet komunikasi atau cukup disebut SUBNET. Tugas subnet adalah membawa pesan dari satu host ke host lainnya. Pada sebagian besar WAN subnet terdiri dari 2 komponen: kabel transmisi dan elemen switching.

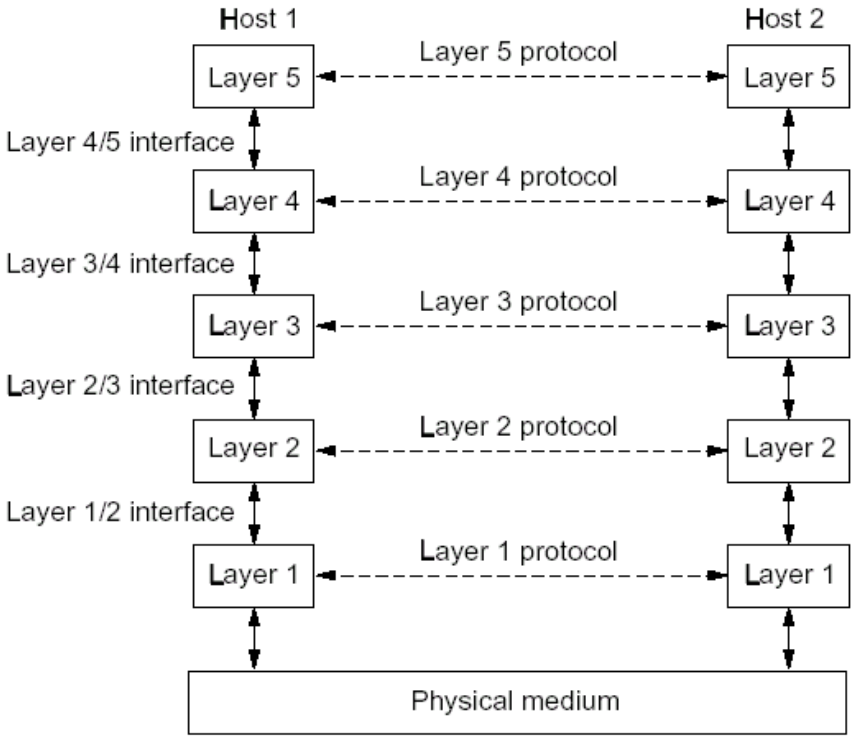



**d. Internet**

Terdapat banyak jaringan di dunia ini, seringkali dengan perangkat keras dan perangkat lunak yang berbeda-beda. Orang yang terhubung ke jaringan sering berharap untuk dapat komunikasi dengan orang lain yang terhubung ke jaringan lainnya. Keinginan seperti ini memerlukan hubungan antar jaringan yang seringkali tidak kompatibel dan berbeda. Kadang menggunakan mesin yang disebut GATEWAY sebagai penerjemah antar jaringan yang tidak kompatibel. Kumpulan jaringan yang terkoneksi disebut INTERNETWORK atau INTERNET. Bentuk INTERNET yang umum adalah kumpulan dari LAN yang dihubungkan oleh WAN.

**2.2.3.3** Perangkat Lunak

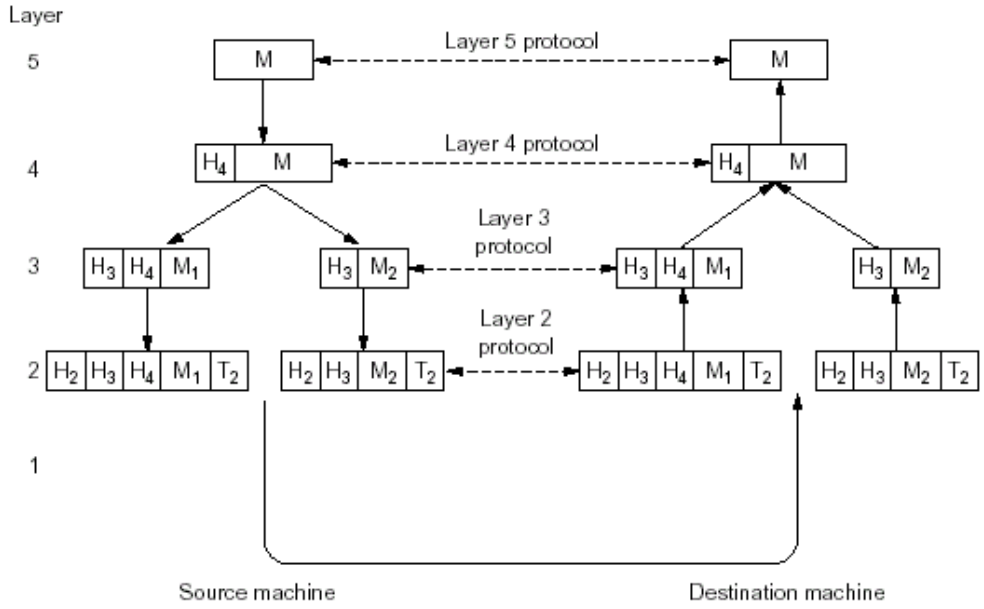
Jaringan diorganisasikan sebagai suatu tumpukan lapisan (layer). Tujuan tiap lapisan adalah memberikan layanan kepada lapisan yang berada di atasnya. Misal lapisan 1 memberi layanan terhadap lapisan 2. Masing-masing lapisan memiliki protokol. Protokol adalah aturan suatu "percakapan" yang dapat dilakukan. Protokol mendefinisikan format, urutan pesan yang dikirim dan diterima antar sistem pada jaringan dan melakukan operasi pengiriman dan penerimaan pesan. Protokol *lapisan n* pada satu mesin akan berbicara dengan protokol *lapisan n* pula pada mesin lainnya. Dengan kata lain, komunikasi antar pasangan lapisan N, harus menggunakan protokol yang sama. Misal, protokol lapisan 3 adalah IP, maka akan ada pertukaran data secara *virtual* dengan protokol lapisan 3, yaitu IP, pada stasiun lain.




Gambar 2.9 Susunan lapisan (layer)

Pada kenyataannya protokol *lapisan n+1* pada satu mesin tidak dapat secara langsung berbicara dengan protokol *lapisan n+1* di mesin lain, melainkan harus melewatkan data dan kontrol informasi ke lapisan yang berada di bawahnya (*lapisan n*), hingga ke lapisan paling bawah. Antar lapisan yang "berkomunikasi", misal lapisan n dengan lapisan n+1, harus menggunakan suatu *interface*(antar muka) yang mendefinisikan layanan-layanannya. Himpunan lapisan dan protokol disebut ***arsitektur protokol***. Urutan protokol yang digunakan oleh suatu sistem, dengan satu protokol per lapisan, disebut ***stack protocol***. Agar suatu paket data dapat saling dipertukarkan antar lapisan, maka paket data tersebut harus ditambahkan suatu **header** yang menunjukkan karakteristik dari protokol pada lapisan tersebut.

Satu stasiun dapat berhubungan dengan stasiun lain dengan cara mendefinisikan spesifikasi dan standarisasi untuk segala hal tentang media fisik komunikasi dan juga segala sesuatu menyangkut metode komunikasi datanya. Hal ini dilakukan pada lapisan 1.

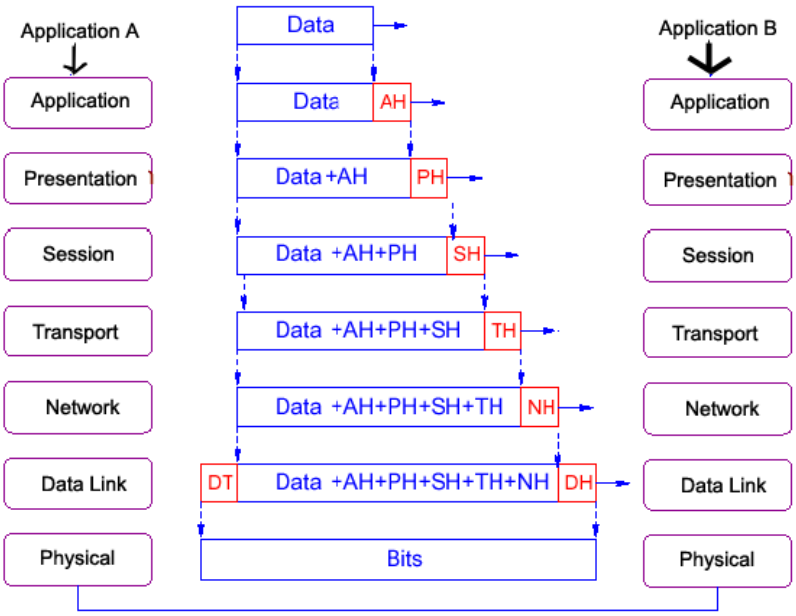



Gambar 2.10 Pemberian Header pada Lapisan-lapisan

Karena begitu kompleknya tugas-tugas yang harus disediakan dan dilakukan oleh suatu jaringan komputer, maka tidak cukup dengan hanya satu standard protokol saja. Tugas yang komplek tersebut harus dibagi menjadi bagian- bagian yang lebih dapat di atur dan diorganisasikan sebagai suatu **arsitektur komunikasi**.

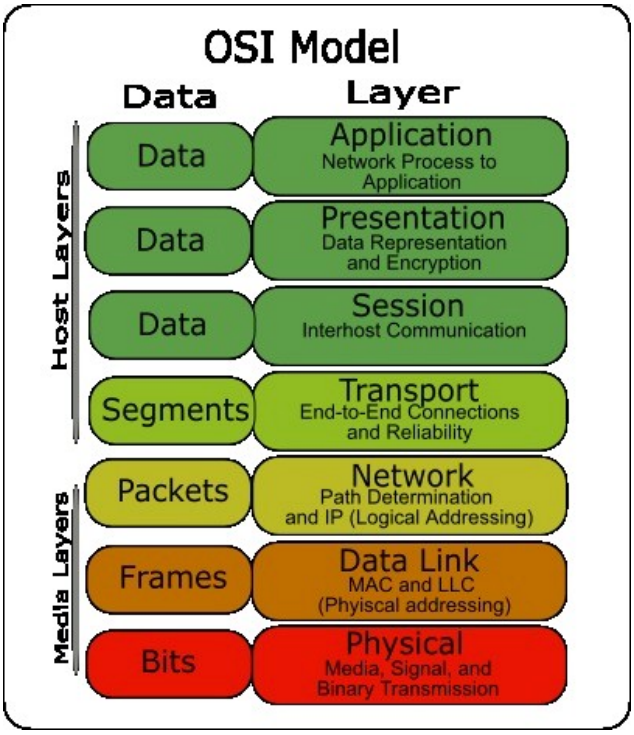
Menanggapi hal tersebut, suatu organisasi standard **ISO (International Standard Organization)** pada tahun 1977 membentuk suatu komite untuk mengembangkan suatu arsitektur jaringan. Hasil dari komite tersebut adalah **Model Referensi OSI (Open Systems Interconnection).** Model Referensi OSI adalah System Network Architecture (SNA) atau dalam bahasa

Indonesianya Arsitektur Jaringan Sistem. Hasilnya seperti pada Gambar OSI Layer dan Header yang menjelaskan ada 7 lapisan (layer) dengan nama masing-masing.




Gambar 2.11 OSI Layer dan Header

Gambar OSI Layer dan Header juga menggambarkan header-header yang diberikan pada setiap lapisan kepada data yang dikirimkan dari lapisan ke lapisan.




Gambar 2.12 OSI Model: Gambaran Tiap Layer

Setiap lapisan memiliki tugas yang berbeda satu sama lain. Berikut masing-masing tugas dari tiap lapisan:

**7) Application Layer** : menyediakan layanan untuk aplikasi misalnya transfer file, email, akses suatu komputer atau layanan.

**6) Presentation Layer** : bertanggung jawab untuk menyandikan informasi. Lapisan ini membuat dua host dapat berkomunikasi.

**5) Session Layer** : membuat sesi untuk proses dan mengakhiri sesi tersebut. Contohnya jika ada login secara interaktif maka sesi dimulai dan kemudian jika ada permintaan log off maka sesi berakhir. Lapisan ini juga menghubungkan lagi jika sesi login terganggu sehingga terputus.

**4) Transport Layer** : lapisan ini mengatur pengiriman pesan dari hos-host di jaringan. Pertama data dibagi-bagi menjadi paket-paket sebelum pengiriman dan kemudian penerima akan menggabungkan paket-paket tersebut menjadi data utuh kembali. Lapisan ini juga memastikan bahwa pengiriman data bebas kesalahan dan kehilangan paket data.

**3) Network Layer** : lapisan bertanggung jawab untuk menerjemahkan alamat logis jaringan ke alamat fisik jaringan. Lapisan ini juga memberi identitas alamat, jalur perjalanan pengiriman data, dan mengatur masalah jaringan misalnya pengiriman paket-paket data.

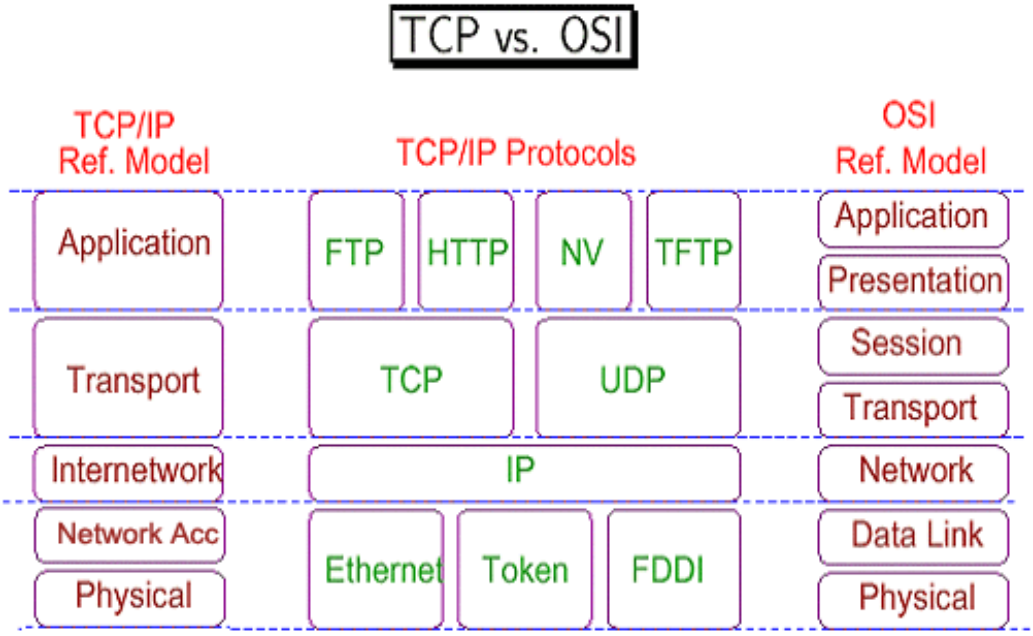
**2) Data Link Layer** :lapisan data link mengendalikan kesalahan antara dua komputer yang berkomunikasi lewat lapisan physical. Data link biasanya digunakan oleh hub dan switch.

**1) Physical Layer** : lapisan physical mengatur pengiriman data berupa bit lewat kabel. Lapisan ini berkaitan langsung dengan perangkat keras seperti kabel, dan kartu jaringan (LAN CARD).

Selain referensi model arsitektur protokol OSI, ada model arsitektur protokol yang umum digunakan yaitu **TCP/IP (Transfer Control Protokol/Internet Protocol).** Arsitektur **TCP/IP** lebih sederhana dari pada tumpukan protokol OSI, yaitu berjumlah 5 lapisan protokol. Jika diperhatikan pada Gambar Perbandingan TCP/IP dan OSI, ada beberapa lapisan pada model OSI yang dijadikan satu pada arsitektur TCP/IP. Gambar tersebut juga menjelaskan protokol-protokol apa saja yang digunakan pada setiap lapisan di

TCP/IP model.

Beberapa protokol yang banyak dikenal adalah FTP (File Transfer Protocol) yang digunakan pada saat pengiriman file. HTTP merupakan protokol yang dikenal baik karena banyak digunakan untuk mengakses halaman-halaman web di Internet.




Gambar 2.13 Perbandingan TCP/IP dan OSI

Berikut penjelasan lapisan layanan pada TCP/IP:

a. Lapian **Application**, menyediakan komunikasi antar proses atau aplikasi pada host yang berjauhan namun terhubung pada jaringan.

b. Lapisan **Transport (End-to-End)**, menyediakan layanan transfer end-to-end. Lapisan ini juga termasuk mekanisme untuk menjamin kehandalan transmisi datanya. Layanan ini tentu saja akan menyembunyikan segala hal yang terlalu detail untuk lapisan di atasnya.

c. Lapisan **Internetwork**, fokus pada pemilihan jalur (*routing*) data dari host sumber ke host tujuan yang melewati satu atau lebih jaringan yang berbeda dengan menggunakan *router*.

d. Layanan **Network Access/Data link**, mendefinisikan antarmuka logika antara sistem dan jaringan.

e. Lapisan **Physical**, mendefinisikan karakteristik dari media transmisi, pensinyalan dan skema pengkodean sinyal

**2.2.3.3** Aplikasi Jaringan Komputer

Jaringan komputer saat ini diterapkan hampir dalam semua tempat seperti: bank, perkantoran, universitas, rumah sakit, bidang pariwisata, hotel, dan bahkan rumah. Semua ini diawali dengan komputerisasi. Komputerisasi memberikan kemudahan dalam penyelesaian banyak tugas dan meningkatkan kebutuhan untuk saling berbagi informasi antar bagian terkait, dan kebutuhan untuk pengamanan dan penyimpanan data. Kebutuhan tersebut kemudian dijawab oleh teknolgi jaringan komputer.

Hingga saat ini jaringan komputer sudah menjadi kebutuhan umum masyarakat, dan karena itu pemahaman dasar tentang jaringan komputer diperlukan, terutama bagi orang-orang yang berkecimpung dalam dunia teknolgi informasi.

## 2.2.4 Basis Data

### 2.2.4.1 Pengertian Basis Data

Basis data terdiri dari kata basis dan data. Basis dapat diartikan gudang atau tempat bersarang dan data yang berarti representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, hewan, peristiwa, konsep dan sebagiannya 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.

### 2.2.4.2 Konsep Dasar Basis Data

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

### 2.2.4.3 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).

### 2.2.4.4 Database Management System (DBMS)

Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak yang khusus. Perangkat lunak inilah disebut DBMS yang akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. DBMS juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, keakuratan atau konsistensi data, dan sebagainya.

### Tujuan Basis Data

Tujuan awal dan utama dalam pengelolaan data dalam sebuah basis data adalah agar dapat memperoleh menemukan kembali data (yang dicari) dengan mudah dan cepat. Di samping itu, pemanfaatan basis data untuk pengelolaan data, juga memiliki tujuan-tujuan lain.

Secara lebih lengkap, pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (objektif) seperti berikut ini:

1. Kecepatan dan kemudahan (*speedy*)
2. Efisiensi ruangan penyimpanan (*space*)
3. Keakuratan (*accuracy)*
4. Ketersediaan (*availability*)
5. Kelengkapan (*completeness*)
6. Keamanan (*security*)
7. Kebersamaan pemakaian (*sharability*)

## 2.2.5 Sistem Analisis

### 2.2.5.1 Pengantar Sistem Analisis

Merancang sistem komputerisasi adalah tugas pokok dari seorang *Systems Analyst*. Hasil rancangan tersebut selanjutnya akan ditindaklanjuti dengan pembuatan program aplikasi oleh *programmer.* Sistem komputerisasi yang telah dibuat selanjutnya akan diimplementasikan oleh *user*.

Pada kenyataannya, banyak sekali pertimbangan yang harus dilakukan seseorang dalam membuat sistem komputerisasi, misalkan spesifikasi *hardware* dan *software* (teknologi) apa saja yang dibutuhkan, berapa anggaran yang disediakan, siapa saja yang terlibat dan harus di*training*, waktu yang tersedia, dan sebagainya.

Karenanya, perancangan sistem komputerisasi akan melibatkan banyak orang di dalamnya. Hal ini mengharuskan dibuatnya '*master plan*,' '*blue print*,' atau skenario umum yang harus disepakati bersama terlebih dulu.

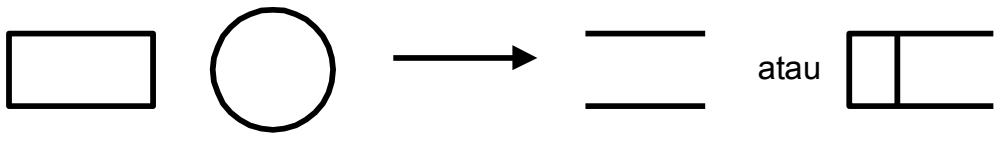
Catatan ini hanya memberikan sedikit gambaran dari perancangan sistem komputerisasi yang sangat rumit, yaitu hanya membahas tentang *Data Flow Diagram*, *Entity Relationship Diagram*, dan Normalisasi Data.

### 2.2.5.2 Pengantar DFD

DFD merupakan salah satu komponen dalam serangkaian pembuatan perancangan sebuah sistem komputerisasi. DFD menggambarkan aliran data dari sumber pemberi data (input) ke penerima data (output). Aliran data itu perlu diketahui agar si pembuat sistem tahu persis kapan sebuah data harus disimpan, kapan harus ditanggapi (proses), dan kapan harus didistribusikan ke bagian lain.

1. Komponen-komponen DFD

   Komponen-komponen DFD terdiri atas :



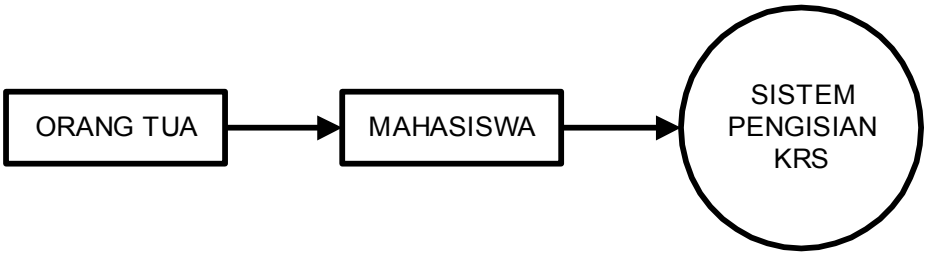

Terminator Proses Alur Data Penyimpan Data (*data store*)

Gambar 2.14 Komponen-komponen DFD

a. Terminator

Terminator dapat disebut juga ‘Kesatuan Luar,’ yaitu suatu unit kerja / jabatan, atau sejenisnya yang berada di luar sistem tetapi memberi andil atas pemberian atau penerimaan data dari sistem secara langsung. Terminator dapat pula disebut dengan ‘Sumber Pemberi Data (input),’ maupun ‘Tujuan Pemberian Data (output).

Pemberi data dan penerima data yang dimaksud adalah pihak yang sangat dekat dan memiliki hubungan langsung dengan sistem. Adapun pihak luar yang berhubungan dengan pihak luar lainnya tidak boleh digambarkan. Misalkan, dalam pengisian KRS, mahasiswa berhubungan dengan sistem. Orang tua berhubungan dengan mahasiswa, tetapi tidak berhubungan dengan sistem, karenanya, kesatuan luar ‘orang tua’, tidak boleh digambarkan.




Gambar 2.15 Contoh Hubungan Terminator yang Salah

b. Proses

Proses adalah suatu tindakan yang akan diambil terhadap data yang masuk. Karena proses adalah tindakan, maka proses berisi kata kerja, Proses diberikan identifikasi (nomor) agar mempermudah sekuen untuk diagram detilnya.




Gambar 2.16 Contoh Proses

c. Alur Data

Alur data menggambarkan data yang mengalir dari terminator ke proses atau dari proses ke proses lainnya. Data yang dibawa oleh alur data harus disebutkan dan diletakkan di atas lambang alur data dan bila alur data digambar panjang, sebaiknya penulisan data mendekati lambang anak panahnya.



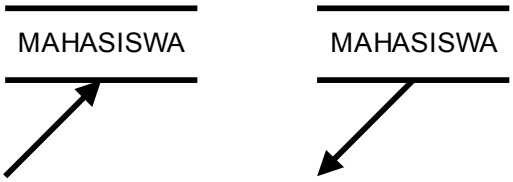

Gambar 2.17 Contoh Alur Data Searah dan Dua Arah

Data yang menempati alur data dapat berupa elemen data tunggal, maupun kumpulan elemen data. Misalkan, pada kumpulan elemen data : 'Jawaban Ujian', dapat ditulis secara lengkap dengan menyebutkan setiap elemen data yang ada di sana, yaitu : 'Lembar Jawaban', dan 'Naskah Soal'.

d. Penyimpan Data (*Data Store*)

Data yang akan disimpan perlu ditempatkan ke satu tempat penyimpanan data. Data yang disimpan dapat berupa data manual maupun data digital. Untuk data digital, penyimpan data tersebut kelak akan dijadikan *file* data di komputer. Alur data yang anak panahnya menuju penyimpan data, kegiatannya adalah 'menulis/ merekam' data, sehingga isi *file* data akan berubah karenanya. Sedangkan alur data yang anak panahnya menuju ke proses dari penyimpan data, kegiatannya adalah 'membaca' data, sehingga isi *file* data tidak akan berubah karenanya.

Penyimpan data harus diberi nama, misalkan data yang berisi biodata mahasiswa diberi nama 'MAHASISWA'.




Gambar 2.18 Menulis dan Membaca data di Penyimpan Data

2. LEVELISASI DFD

DFD digambarkan secara bertingkat, dari tingkat yang global berturut-turut hingga tingkat yang sangat detil. Tingkat yang global (umum) disebut dengan 'Diagram Konteks' atau '*Context Diagram'*. Ini termasuk *level* 0.

Selanjutnya, dari diagram konteks, prosesnya dijabarkan lebih rinci lagi di 'Diagram Nol' atau '*Zero Diagram.*' Ini disebut *level* 1. Pada diagram nol ini yang berkembang hanya proses dan alur data yang menghubungkan proses-prosesnya, sedangkan jumlah terminator dan alur data yang masuk atau keluar dari terminator, tetap.

Bila, masih dirasakan perlu memerinci proses berikutnya, maka diagram selanjutnya disebut dengan 'Diagram Detil' atau 'Diagram primitif.' Ini disebut dengan *level* 2. Dalam diagram detil, yang digambar cukup proses (nomor berapa) yang perlu didetilkan saja, selain itu (proses lainnya, atau terminatornya) tidak perlu digambarkan.

Bila masih dapat lebih didetilkan lagi, maka *level* 3, dan seterusnya bisa dibuat.

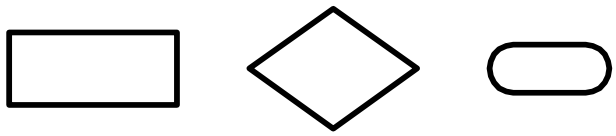
### 2.2.5.3 Pengantar ERD

ERD adalah gambaran mengenai berelasinya antarentitas. Sistem adalah kumpulan elemen yang setiap elemen memiliki fungsi masing-masing dan secara bersama-sama mencapai tujuan dari sistem tersebut. 'Kebersama-sama'-an dari sistem di atas dilambangkan dengan saling berelasinya antara satu entitas dengan entitas lainnya.

Entitas (*entity/ entity set*), memiliki banyak istilah di dalam ilmu komputer, seperti tabel (*table*), berkas (*data file*), penyimpan data (*data store*), dan sebagainya.

1. Komponen-komponen ERD

ERD memiliki komponen-komponen :



Entitas Relasi Atribut

Gambar 2.19 Komponen-komponen ERD.

a. Entitas dan atribut.

Seperti telah dijelaskan di atas, entitas adalah tempat penyimpan data, maka entitas yang digambarkan dalam ERD ini merupakan *data store* yang ada di DFD dan akan menjadi *file* data di komputer.

Entitas adalah suatu objek dan memiliki nama. Secara sederhana dapat dikatakan bahwa jika objek ini tidak ada di suatu *enterprise* (lingkungan tertentu), maka *enterprise* tersebut tidak dapat berjalan normal. Contoh, entitas 'MAHASISWA' harus ada di lingkungan perguruan tinggi, begitu juga dengan entitas 'DOSEN', 'MT_KULIAH', dan sebagainya.

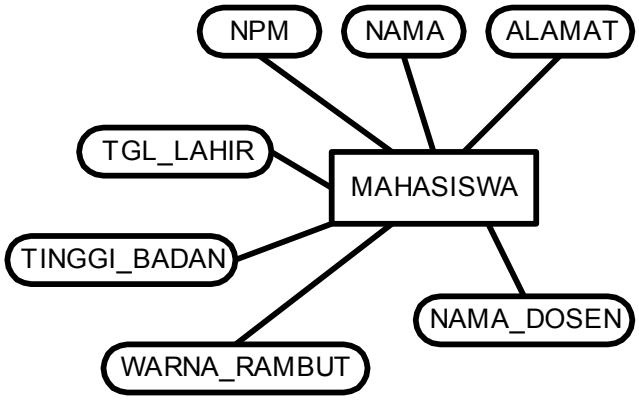
Di dalam entitas 'MAHASISWA' berisi elemen-elemen data (biodata mahasiswa) yang terdiri atas NPM, NAMA, KELAS, ALAMAT, dan sebagainya. NPM, NAMA, KELAS, dan ALAMAT disebut dengan atribut (*field*).

Apa saja atribut yang bisa menjadi ciri dari entitas, secara sederhana dapat dilakukan dengan melakukan pertanyaan logis. Misalkan, di entitas MAHASISWA ada atribut NILAI. Tanyakan ke mahasiswa, 'berapa nilai anda ?.' Tentu si mahasiswa akan berbalik tanya : 'nilai apa ?,' karena pertanyaan tersebut tidak dapat dijawab langsung, maka NILAI bukanlah atribut dari mahasiswa.

Jika, pertanyaan yang diajukan dapat dijawab entitas secara logis dan benar, maka ia merupakan atributnya. Misalkan 'berapa

tinggi badan anda ?,' maka tinggi badan adalah atribut dari mahasiswa, meskipun hal itu tidak perlu digunakan karena tidak berfungsi apa-apa dalam kemahasiswaannya.

Jadi, ada atribut yang harus ada, ada atribut yang boleh ada, dan ada atribut yang tidak boleh ada, di dalam sebuah entitas. Contoh penggambaran entitas dan atributnya.




Gambar 2.20 Hubungan Atribut dan Entitasnya

Gambar 14. memperlihatkan bahwa atribut-atribut NPM, NAMA, ALAMAT, dan TGL_LAHIR harus ada di dalam biodata seorang mahasiswa. Atribut-atribut TINGGI_BADAN, dan WARNA_RAMBUT adalah atribut-atribut yang boleh tidak ada di dalam biodata mahasiswa (karena tidak penting). Sedangkan atribut NAMA_DOSEN adalah atribut yang tidak boleh ada di entitas mahasiswa.

Pada akhirnya, entitas ini akan menjadi *file* data (yang bersifat *master file*) di dalam komputer. *Master file* adalah *file* utama (yang harus ada, dan sifatnya jarang berubah).

b. Relasi

Relasi adalah penghubung antara satu entitas (*master file*) dengan entitas lain di dalam sebuah sistem komputer. Pada akhirnya, relasi akan menjadi *file* transaksi (*transaction file*) di komputer.

Secara kalimat logis, contoh relasi yang terjadi di sebuah perpustakaan adalah : "Anggota meminjam buku," atau "Anggota mengembalikan buku." Dalam hal ini, Anggota dan Buku adalah entitas, meminjam dan mengembalikan adalah transaksi (relasi antara anggota dan buku).

c. Derajat Kardinalitas (*Cardinality Degree*)

Hubungan antarentitas ditandai pula oleh derajat kardinalitas. Fungsi dari derajat kardinalitas ini adalah untuk menentukan entitas kuat dan entitas lemah. Tiga jenis derajat kardinalitas adalah :

*(1)* *One to one*, dilambangkan dengan 1 : 1

*(2)* *One to many*, dan sebaliknya, yang dilambangkan dengan 1 : M dan sebaliknya

*(3)* *Many to many*, dilambangkan dengan M : M atau M : N

Entitas dengan derajat kardinalitas 1 adalah entitas lemah sehingga entitas tersebut boleh digabung saja dengan entitas yang kuat (derajat kardinalitas M).

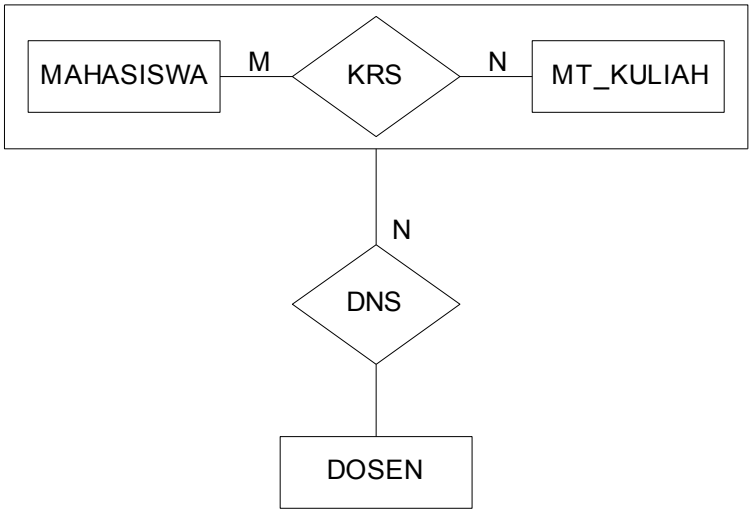
Misalkan kalimat bolak-balik berikut ini :

"Satu mahasiswa memiliki satu kelas"

"Satu kelas memiliki lebih dari satu (banyak) mahasiswa"

Kata (entitas) "KELAS" selalu disebut dengan kata "satu", sedangkan kata (entitas) "MAHASISWA" pernah disebut dengan lebih dari satu (banyak). Maka, di *file* MAHASISWA boleh berisi atribut KELAS, dan KELAS tidak perlu menjadi *file* sendiri.

(4) Gabungan/ kombinasi ketiga bentuk di atas, misalkan *many to many to many*. Tapi, di sini tidak akan dibahas secara lebih lanjut. Contoh :




Gambar 2.21 Relasi *many to many to many*

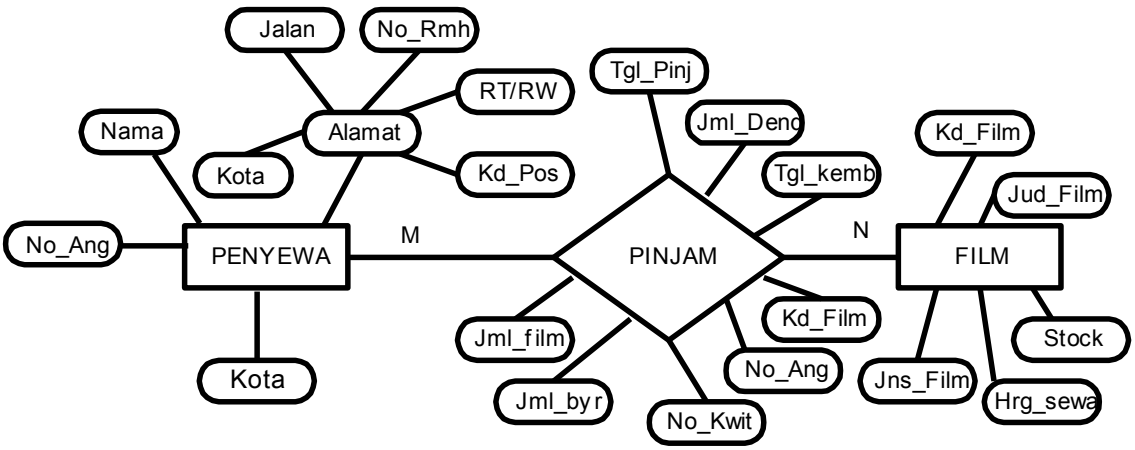
Penggambaran ER dari kasus di atas (lanjutan dari DFD) dilakukan dengan cara :

(1) *Data Store* (penyimpan data) yang ada di DFD akan menjadi *Entity* di dalam ERD;

(2) Tentukan atribut-atribut (secara logika) yang harus ada di dalam setiap entitasnya;

(3) Tentukan serajat kardinalitasnya sesuai dengan peraturan yang berlaku di rental tersebut (dalam hal ini, setiap Penyewa boleh menyewa film lebih dari satu);

(4) Tentukan kunci atribut di setiap entitasnya.

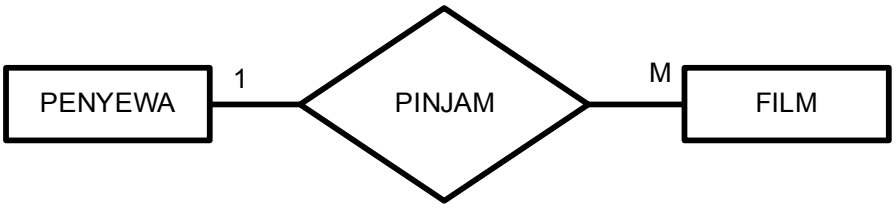



Gambar 2.22 ERD dari Kasus di Atas

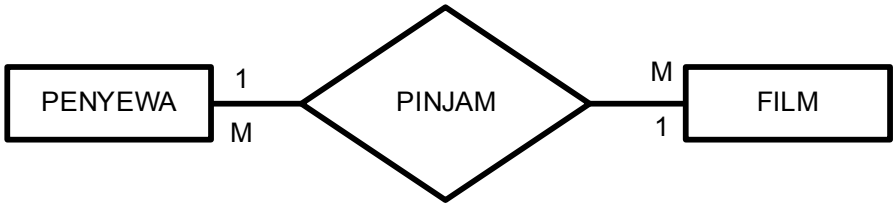
Derajat kardinalitas yang terjadi adalah M : N (*many to many).* Mengapa tidak M : M, karena belum tentu "10 orang penyewa pasti menyewa 10 film." Karena tidak selalu M = M, maka dipilih M : N saja, jadi, suatu saat M boleh = N, dan di saat lain boleh M ≠ N.

Cara menentukan *many to many*-nya adalah dengan membuat dua kalimat bolak-balik. Derajat kardinalitas kalimat pertama diletakkan di atas, dan derajat kardinalitas kalimat kedua diletakkan di bawah.

Kalimat pertama : "satu orang penyewa boleh pinjam satu atau lebih (judul) film."




Kalimat kedua : "satu (judul) film boleh dipinjam oleh satu atau lebih penyewa"




Gambar 2.23 Penentuan *Many to Many*

Di antara derajat kardinalitas yang berada di atas dan bawah, dicari yang terbesar (yang kecil dihapus), maka akan didapatkan M : M yang pada akhirnya dilambangkan dengan M : N.

d. Penentuan *Primary Key*

Di setiap entitas di dalam ERD (di gambar 12 di atas), seharusnya ada atribut (*field*) yang dipilih untuk dijadikan kunci utama atribut (*primary key*/ *key field*), yaitu atribut yang dijadikan identitas yang menjamin keunikan (tidak ada yang sama) isi datanya.

Misalkan, untuk entitas mahasiswa dipilih atribut NPM sebagai kunci utama atributnya karena tidak ada satupun mahasiswa yang memiliki NPM yang sama.

Penulisan kunci utama atribut di dalam ERD harus dibedakan dengan atribut lainnya, misalkan dengan pemberian tanda '*' di depan nama atributnya, atau digarisbawahi atributnya.

Secara logika, memang mudah menentukan sebuah atribut kunci, namun sesungguhnya, kunci utama diperoleh dari kunci kandidat, dan kunci kandidat diperoleh dari kunci super.

*(1) SUPER KEY*

*Super key* adalah satu atau lebih *field* yang dapat dipilih untuk membedakan (mengkarakteristikkan) antara satu *record* dengan *record* lainnya. Bila *file*nya adalah MAHASISWA, maka satu atau lebih *field* yang dipilih agar dapat membedakan antara satu orang mahasiswa dengan mahasiswa lainnya.

NPM jelas bisa membedakan, NAMA juga bisa, namun dengan syarat tidak ada nama yang sama, gabungan NPM dan NAMA pasti bisa membedakan, apalagi gabungan NPM, NAMA dan TGL_LAHIR. Sehingga, *super key* bisa merupakan kombinasi dari satu atau gabungan *field* yang dapat mencirikan suatu *record*.

*Super key*nya :
NPM
NAMA (dengan syarat tidak ada nama yang sama)
ALAMAT (dengan syarat alamat tidak ada yang sama)
TGL_LAHIR (dengan syarat tidak ada tanggal lahir yang sama)
NPM+NAMA

NPM+NAMA+ALAMAT
NPM+TGL_LAHIR
NPM+ALAMAT+TGL_LAHIR
dan berbagai kombinasi lainnya

*(2) CANDIDATE KEY*

Kunci kandidat adalah kunci super dengan jumlah *field* paling sedikit, maka diperoleh : NPM, NAMA, ALAMAT, TGL_LAHIR (karena masing-masing hanya terdiri dari 1 *field* saja).

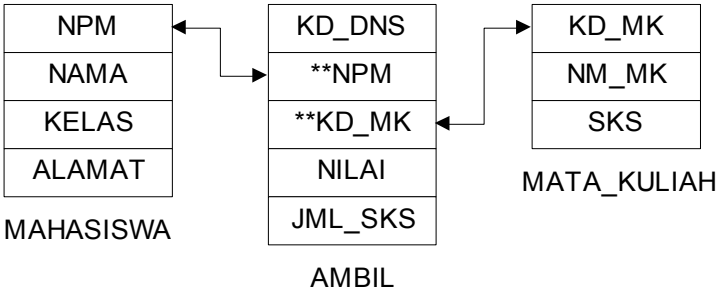
*(3) PRIMARY KEY*

Kunci utama adalah kunci kandidat yang dipilih dengan kemungkinan kepemilikan nilai data *field* yang berbeda antara satu *record* dengan *record* lainnya. Maka dipilih NPM karena tidak ada mahasiswa yang memiliki NPM yang sama. Jelaslah, kunci utama pastilah merupakan kunci kandidat dan juga kunci super, tetapi sebaliknya, kunci super dan kunci kandidat belum tentu merupakan kunci utama.

*(4) ALTERNATE KEY*

Kunci kandidat yang tidak terpilih menjadi kunci utama disebut dengan kunci alternatif.

Berikut, akan digambarkan di mana atribut (*field*) NILAI dimasukkan ke dalam suatu entitas (*file*) dan apa yang disebut dengan kunci tamu (*foreign key*).




Gambar 2.24 Hubungan Antarentitas

Di entitas MAHASISWA, *key field* yang dipilih adalah NPM;
Di entitas AMBIL, *key field* yang dipilih adalah KD_DNS;
Di entitas MATA_KULIAH, *key field* yang dipilih adalah KD_MK;

Di entitas AMBIL, yang merupakan *transaction file*, dimasukkan pula atribut NPM dan atribut KD_MK yang merupakan kunci-kunci utama dari entitas-entitas lain. Karenanya, NPM dan KD_MK di entitas AMBIL merupakan kunci tamu (*foreign key*).

### 2.2.6 Software Pendukung

#### 2.2.6.1 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

XAMPP dikembangkan dari sebuah tim proyek bernama *Apache Friends*, yang terdiri dari Tim Inti (*Core Team*), Tim Pengembang (*Development Team*) & Tim Dukungan (*Support Team*).

#### 2.2.6.2 Apache

*Server* HTTP *Apache* atau *Server Web*/WWW *Apache* adalah *server web* yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Windows, Novotel Netware dan lainnya) yang berguna untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas web/www ini menggunakan HTTP.

Apache memiliki fitur-fitur yang canggih seperti pesan kesalahan yang dapat dikonfigur, autentikasi berbasis basis data dan lain-lain. Apache juga

didukung oleh sejumlah antar muka pengguna berbasis grafik (GUI) yang memungkinkan penanganan *server* menjadi mudah.

Apache merupakan *software open source* dikembangkan oleh komunitas terbuka yang terdiri dari pengembang-pengembang dibawah naungan *Apache Software Foundation.*

**2.2.6.3 My SQL**

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 memerlukan sistem manajemen database seperti MySQL. Karena komputer sangat unggul dalam menangani sejumlah besar data, sistem manajemen database memainkan suatu peranan yang penting dalam komputasi, baik sebagai *utility stand-alone* maupun bagian dari aplikasi lainnya.

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

Pada MySQL terdapat beberapa perintah. Perintah-perintah pada MySQL ini hampir sama dengan perintah-perintah pada database server lainnya. Perintah-perintah MySQL itu antara lain adalah sebagai berikut :

1. Create database, digunakan untuk membuat *database* pada *database server*.

Sintaksnya adalah :

```
Create database database_name
```

*Database_name* adalah nama *database* yang akan dibuat.

2. Use database, digunakan untuk menunjuk *database* yang akan digunakan.

Sintaksnya adalah :

```
Use database_name
```

*Database_name* adalah nama *database* yang akan digunakan.

3. Create table, digunakan untuk membuat tabel pada *database* yang digunakan.

Sintaksnya adalah :

```
Create table table_name
(
Column1 column_type column_attributes,
Column2 column_type column_attributes,
Primary_key (column_name)
);
```

*Table_name* adalah nama tabel yang akan dibuat. *Column1* adalah nama kolom yang akan dibuat pada tabel. *Column_type* adalah tipe dari kolom tersebut , dapat berupa *char, varchar, tinytext, mediumtext, longtext, enum, int, tinyint, mediumint, bigint, float, decimal, time, date, datetime, timestamp, year.*

4. Insert, digunakan untuk menambahkan record pada tabel.

Sintaksnya adalah :

```
Insert into table_name(column1, column2,..) values (value1,value2,..)
```

*Table_name* adalah nama tabel yang akan ditambahkan *record*-nya.

*Column1, column2* adalah kolom yang akan ditambahkan data. *Value1,value2* adalah data yang akan ditambahkan.

5. Update, digunakan untuk mengubah record yang sudah ada pada tabel.

Sintaksnya adalah :

```
Update table_name set column1=value1, column2=value2
where column=value
```

*Table_name* adalah nama tabel yang akan dirubah *record*-nya. *Column1, column2* adalah kolom yang akan dirubah data. *Value1,value2* adalah data yang akan digantikan.

6. Drop table, digunakan untuk menghapus tabel.

Sintaksnya adalah :

```
Drop table table_name
```

Table_name adalah nama tabel yang akan dihapus.

7. Show tables, digunakan untuk menampilkan tabel-tabel yang telah dibuat dalam *database* yang aktif.

Sintaksnya adalah :

```
Show tables
```

8. Show field, digunakan untuk menampilkan seluruh field dalam suatu tabel.

Sintaksnya adalah :

```
Show field from table_name
```

*Table_name* adalah nama tabel yang akan ditampilkan *field*-nya.

9. Alter table, digunakan untuk menambah,merubah, dan menghapus field dalam suatu tabel.

Sintaksnya adalah :

a. Untuk menambahkan

```
Alter table_name add column column1 column_type
column_attributes
```

*Table_name* adalah nama tabel yang akan ditambahkan *field*-nya. *Column1* adalah nama *field* baru, *column_type* adalah tipe kolom dan *column_attributes* adalah atribut kolom yang akan ditambahkan.

b. Untuk mengubah

```
Alter table_name change column1 column2
column_type column_attributes
```

*column1* adalah mana *field* yang akan dirubah, *column2* adalah nama *field* baru, *column_type* adalah tipe kolom dan *column_attributes* adalah atribut kolom.

c. Untuk menghapus

```
Alter table table_name drop column column1
```

*Table_name* adalah tabel yang akan dihapus *field*-nya. *Column1* adalah nama *field* yang akan dihapus.

Ada beberapa kelebihan yang dimiliki *MySQL* sehingga dapat menarik banyak pengguna. Kelebihan tersebut yaitu:

1. Fleksibilitas.

   Saat ini, *MySQL* telah dioptimasi untuk duabelas platform seperti *HP-UX, Linux, Mac OS X, Novell Netware, OpenBSD, Solaris, Microsoft Windows* dan lain-lain. *MySQL* juga menyediakan *source code* yang dapat diunduh secara gratis, sehingga pengguna dapat mengkompilasi sendiri sesuai platform yang digunakan. Selain itu, *MySQL* juga dapat dikustomisasi sesuai keinginan penggunanya, misalnya mengganti bahasa yang digunakan pada antarmukanya.

2. Performansi.

   Sejak rilis pertama, pengembang *MySQL* fokus kepada performa. Hal ini masih tetap dipertahankan hingga sekarang dengan terus meningkatkan fiturnya.

3. Lisensi.

   *MySQL* menawarkan berbagai pilihan lisensi kepada penggunanya. Lisensi *open source* yang ditawarkan yaitu lisensi *GNU General Public License* dan *Free/Libre and Open Source Software (FLOSS) License Exception*. Selain itu ditawarkan juga lisensi komersil berbayar yang memiliki fasilitas dukungan teknis.

### 2.2.7 Bahasa Pemograman

Dokumen dan aplikasi yang dapat berjalan di *web browser* pada umumnya memiliki format *hypertext markup language* (HTML). Oleh karenanya, untuk dapat membangun sebuah *website e-commerce* diperlukan pengetahuan tentang HTML. Tetapi pengetahuan tentang HTML saja tidak cukup, karena untuk melakukan transaksi di internet diperlukan sebuah web yang dinamis, artinya antara *customer* dan perusahaan dapat melakukan trading secara *online* [2].

Ada berbagai macam bahasa pemrograman yang bisa digunakan untuk dapat membangun sebuah *web* dinamis. Berikut contoh bahasa pemrograman yang dapat digunakan dalam membangun *website e-commerce* :

### 2.2.7.1 HTML (*Hypertext Markup Language*)

HTML (*Hyper Text Markup Language*) adalah file teks yang terdiri dari tag-tag markup yang bisa digunakan dalam semua file-file yang berjalan di atas *web browser.*

HTML merupakan pengembangan dari standar pemformatan dokumen teks, yaitu *Standard Generalized Makrup Language* (SGML). HTML dibuat pertama kali oleh Tim Berners-Lee dan dipopulerkan pertama kali oleh *browser* Mosaic. Selama awal tahun 1990, HTML mengalami perkembangan yang sangat pesat dan setiap pengembangannya HTML akan menambahkan kemampuan dan fasilitas yang lebih baik sebelumnya. Namun dalam perkembangannya HTML tidak mengubah cara kerjanya. HTML 2.0 secara resmi dikeluarkan pada bulan November 1995 oleh IETF (*Internet Engineering Task Force*). Kemudian HTML 3.0 (1995) memberikan kemampuan yang lebih dari versi sebelumnya. Dengan usaha dari *Word Wide Web Consortium's* HTML. Working Group pada tahun 1996, dihasilkan HTML 3.2. HTML versi terbaru adalah HTML 4.01 yang dikeluarkan secara resmi oleh W3C pada tanggal 24 April 1998.

File HTML dapat dibuat dalam sebuah teks editor, misalnya yang sederhana di Notepad. Teks editor Notepad ini dapat ditemukan pada hampir semua sistem operasi seperti Windows 98 SE, ME, 2000, NT ataupun Xp, karena secara otomatis ketika meng-*install* sistem opersi *default* dapat mencarinya dari menu Start  All Program  Accessories  Notepad.
Pembaca dapat menggunakan teks editor yang lain, seperti SideKick, MS WritePad, UltraEdit, Carimson Editor, Edit Plus, dan sebagai'y [2].

### 2.2.7.2 *Personal Home Page Tool* (PHP)

PHP diciptakan oleh Rasmus Lerdorf pada tahun 1994 dan bersifat *open source.* Sampai bulan januari 2007, PHP sudah digunakan oleh kurang lebih 20 juta *domain* dan terus berkembang sampai saat ini.

PHP merupakan singkatan dari *Hypertext Preprocessor*, adalah sebuah bahasa scripting yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik.

Tujuan utama bahasa ini adalah untuk memungkinkan perancang *web* menulis halaman *web* dinamis dengan cepat.

PHP merupakan bahasa pemograman *web* yang bersifat *server-side HTML=embedded scripting*, di mana script-nya menyatu dengan HTML dan berada si *server*. Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di *server* tetapi disertakan HTML biasa. PHP dikenal sebgai bahasa scripting yang menyatu dengan *tag* HTML, dieksekusi di *server* dan digunakan untuk membuat halaman *web* yang dinamis seperti ASP (*Active Server Pages*) dan JSP (*Java Server Pages*).

Seluruh aplikasi berbasis *web* dapat dibuat dengan PHP. Namun kekuatan yang paling utama PHP adalah pada konektivitasnya dengan system *database* di dalam *web*.

Sistem *database* yang dapat didukung oleh PHP adalah :

1. Oracle
2. MySQL
3. Sybase
4. PostgreSQL

PHP dapat berjalan di berbagai system operasi seperti windows 98/NT, UNIX/LINUX, solaris maupun macintosh. PHP merupakan *software* yang *open source* yang dapat download secara gratis dari situs resminya yaitu http://www.php.net, ataupun dari situs-situs yang menyediakan *software* tersebut seperti di ftp://gerbang.che.itb.ac.id.

*Software* ini juga dapat berjalan pada *web server* seperti PWS (*Personal Web Server*), Apache, IIS, AOLServer, fhttpd, phttpd dan sebagainya. PHP juga merupakan bahasa pemograman yang dapat kita kembangkan sendiri seperti menambah fungsi-fungsi baru. Keunggulan lainnya dari PHP adalah PHP juga mendukung komunikasi dengan layanan seperti protocol IMAP, SNMP, NNTP, POP3 bahkan HTTP. PHP dapat diinstal sebagai bagian atau modul dari apache web *server* atau sebagai CGI script yang mandiri.

PHP memiliki empat kelebihan utama yang menarik minat banyak pengguna, diantaranya sebagai berikut:

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah
kompilasi dalam penggunaanya.
2. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai
apache, IIS, Lighttpd, nginx, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pemahamanan, PHP adalah bahasa scripting yang paling mudah karena
memiliki referensi yang banyak.
4. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan.

Banyak keuntungan yang dapat diperoleh jika menggunakan PHP sebagai modul dari apache di antaranya adalah :

1. Tingkat keamanan yang cukup tinggi
2. Waktu eksekusi yang lebih cepat dibandingkan dengan bahasa pemograman web lainnya yang berorientasi pada *server-side scripting*.
3. Akses ke sistem *database* yang lebih fleksibel seperti MySQL.

#### 2.2.7.3 *Cascanding Style Sheet* (CSS)

**a. Mengenal CSS**

CSS merupakan bentuk kependekan dari *Cascading Style Sheets,* istilah ini apabila diartikan memiliki arti "hamparan pasir melata". Persepsi baru bahwa dokumen ini merupakan bentuk Script yang sangat membantu untuk membuat desain lebih menarik. CSS bukan merupakan bahasa pemograman, akan tetapi dapat disebut dengan spesial yang berguna untuk melakukan pengaturan pada setiap komponen (bagian-bagian seperti tabel, form, dan lain – lain) web. Inti dari dokumen CSS adalah Tag HTML, di dalam CSS tag HTML tidak lagi disebut sebagai Tag akan tetapi akan disebut sebagai Selector.

Fungsi terpenting dari dokumen CSS adalah melakukan pengaturan halaman web dengan sistem Style, karena dokumen CSS

merupakan bentuk turunan dari elemen HTML (perintah html) maka dokumen ini merupakan bentuk kode yang embeded dengan Tag HTML, dalam penggunaannya kedua objek ini akan sangat berhubungan dan seakan tidak pernah terpisahkan.

**b. Bagian Penting Dalam CSS**

Dalam Penulisannya CSS memiliki sedikit perbedaan antara penulisan Tag HTML dengan Style CSS, telah dijelaskan bahwa pada penulisan CSS menggunakan Tag HTML, dan dalam dokumen CSS akan disebut dengan Selector. Secara garis besar, penulisan CSS sebagai berikut. :

```
Selector {
    Properti1: Value;
    Properti2: Value;
    }
```

Dari penggambaran penulisan CSS dapat dijelaskan bagian-bagiannya adalah:

1. Selector : Bagian ini merupakan kumpulan dari Elemen HTML, jadi semua elemen yang ada pada HTML akan digunakan pada tag ini.
2. Propertie: Bagian ini merupakan artibut yang digunakan pada setiap elemen yang digunakan, tidak semua atribut akan dapat digunakan pada semua Selector, yaitu tergantung dukungan yang ada pada HTML.
3. Value: Merupakan nilai yang diberikan pada setiap Properti dalam Selector, penentuan nilainya dapat diset sama persis dengan pada saat menggunakan nilai pada atribut HTML.

### 2.2.7.4 Javascript

*Javascript* merupakan *cross-platform* yang dikembangkan oleh Netscape dan pertama kali digunakan dalam *browser* Netscape. Javascript dibuat agar mudah diintegrasikan kedalam program dan aplikasi lain, misalnya *browser*. Sebagian besar *browser* saat ini sudah mendukung javascript. Oleh

karena itu, *script* dari javascript biasanya dimasukkan kedalam suatu HTML dan dieksekusi di*browser* lain.

Javascript bergantung kepada browser(navigator) yang memanggil halaman web yang berisi skrip skrip dari Javascript dan tentu saja terselip di dalam dokumen HTML. Javascript juga tidak memerlukan kompilator atau penterjemah khusus untuk menjalankannya (pada kenyataannya kompilator Javascript sendiri sudah termasuk di dalam browser tersebut).

### 2.1.4 Macromedia Dreamweaver

Merupakan sebuah *software* HTML editor profesional yang digunakan untuk mendesain secara visual dan mengelola situs web maupun halaman web. Bilamana kita menyukai untuk berurusan dengan kode-kode HTML secara manual atau lebih menyukai bekerja dengan lingkungan secara visual dalam melakukan editing, Dreamweaver membuatnya menjadi lebih mudah dengan menyediakan tools yang sangat berguna dalam peningkatan kemampuan dan pengalaman kita dalam mendesain web.

Dreamweaver 8 dalam hal ini digunakan untuk web desain. Dreamweaver 8 mengikutsertakan banyak tools untuk kode-kode dalam halaman web beserta fasilitas-fasilitasnya, antara lain : Referensi HTML, CSS dan Javascript, Javascript debugger, dan editor kode ( tampilan kode dan Code inspector) yang mengizinkan kita mengedit kode Javascript, XML, dan dokumen teks lain secara langsung dalam Dreamweaver. Teknologi Dreamweaver Roundtrip HTML mampu mengimpor dokumen HTML tanpa perlu memformat ulang kode tersebut dan kita dapat menggunakan Dreamweaver pula untuk membersihkan dan memformat ulang HTML bila kita menginginkannya. Selain itu Dreamweaver juga dilengkapi kemampuan manajemen situs, yang memudahkan kita mengelola keseluruhan elemen yang ada dalam situs. Kita juga dapat melakukan evaluasi situs dengan melakukan pengecekan broken link, kompatibilitas browser, maupun perkiraan waktu download halaman web.