

BAB 2

LANDASAN TEORI

2.1 Multimedia

2.1.1 Definisi Multimedia

Definisi multimedia secara umum adalah pemanfaatan teknologi berbasis komputer sebagai sarana yang digunakan untuk menggabungkan antara teks, grafik, animasi, suara, dan video ke dalam suatu bentuk yang terintegrasi (<http://webopedia.internet.com/multimedia/multimedia.html>).

Pandangan serupa dikemukakan oleh Long (2000, p13) yang menyebutkan bahwa multimedia adalah kemampuan yang memungkinkan integrasi dari teks, grafik, resolusi tinggi, video, animasi, dan suara yang berbasiskan komputer.

Sedangkan menurut Vaughan (1994, pp5-6), multimedia adalah penggabungan dari teks, gambar, suara, animasi, dan elemen video yang ditampilkan di komputer atau peralatan elektronik lainnya. Jika pengguna dapat mengontrol apa dan kapan elemen yang disajikan, maka hal tersebut dikatakan multimedia interaktif.

Program multimedia yang umum dapat dijumpai pada games, ensiklopedia, dan materi pelatihan pada CD-ROM. Dengan demikian, setiap aplikasi yang disertai suara dan atau video dapat disebut program multimedia (<http://www.techweb.com/encyclopedia/defineterm?term=multimedia>).

Menurut Kamus Umum Bahasa Indonesia secara umum, media berarti suatu alat, perantara, atau penghubung. Media dalam cakupan lebih luas adalah sarana komunikasi yang digunakan untuk menyebarkan informasi.

Sesuai dengan namanya “MULTIMEDIA” bila diterjemahkan secara bebas dapat berarti lebih dari satu media. Media-media ini nantinya akan disinkronkan untuk menjadi satu kesatuan yang dapat dipergunakan. Artinya media yang akan dipilih nantinya harus dapat bekerja sama dengan media lain untuk membentuk satu kesatuan yang harmonis.

Adapun pengertian dari multimedia itu sendiri adalah suatu piranti lunak yang memakai lebih dari satu metode dalam berkomunikasi dengan *user*, seperti adanya pemakaian grafik, teks, suara, video, dan lain-lain.

Menurut Burger (1993, p3) multimedia adalah gabungan dari dua atau lebih media yang berbeda pada satu komputer personal. Sedangkan menurut Andleigh dan Thakrar (1996, p9) multimedia merupakan teknologi yang dikembangkan dengan penggabungan teks, grafik, animasi, suara, video dalam suatu produksi berbasis komputer yang dapat dinikmati secara interaktif.

2.1.2 Penerapan Teknologi Multimedia

Menurut Hofstetter (2001, pp30-105), aplikasi-aplikasi multimedia adalah :

- **Bisnis dan industri**

Multimedia menunjang bisnis dengan kekuatan untuk meraih dan melayani konsumen secara interaktif. Perdagangan dengan menggunakan multimedia menggantikan penjualan konvensional menggunakan toko dengan cara *home-shopping*. *Video conference* memungkinkan adanya tatap muka tanpa melakukan perjalanan. Sistem pengajaran menggunakan multimedia memungkinkan pengajar untuk hadir dimana saja ia diperlukan, dan pasti mengurangi biaya. Jaringan global membuat pelayanan konsumen tersedia dimana saja dan kapan saja.

- Pendidikan

Pengajar dapat membawa multimedia ke dalam kelas ajar mereka, seolah mereka berada pada situasi sebenarnya. Multimedia memungkinkan pengajar memiliki akses ke ratusan *slide*, video, suara, dan rencana pengajaran.

- Hiburan

Banyak sekali inovasi multimedia pada suara dan grafis yang digunakan oleh industri hiburan. Terdapat kompetisi antara *cinematographer* dengan produser game video dalam memberikan efek spesial yang menakjubkan.

- Pemerintahan dan Politik

Pemerintahan resmi telah beralih kepada multimedia untuk pemecahan solusi dalam masalah pemerintahan. Pemerintah menggunakan internet untuk mengetahui apa saja yang terjadi, dan PBB mendokumentasikannya.

- Pengobatan dan Perawatan

Ketika hidup bergantung pada sesuatu, manusia menjadi peduli akan hal itu. Jadi disinilah multimedia bekerja digabungkan dengan ilmu kedokteran. Multimedia memungkinkan orang untuk mendiagnosa apa penyakitnya, dan bagaimana cara meredakannya sebelum mereka pergi ke dokter ahli mereka.

- Sumber Daya Pengetahuan

Sekarang ini pengetahuan sudah ada yang dikemas dalam bentuk multimedia CD, DVD, dan online melalui internet.

- Paket Aplikasi Pengembangan

Multimedia telah membantu pengembangan dalam hal ilmu pengetahuan melalui produk-produk piranti lunaknya.

2.1.3 Elemen Multimedia

Menurut Hofstetter (2001, pp16-27), dalam multimedia terdapat enam buah objek :

- Teks

Meskipun penggunaan multimedia tanpa teks adalah hal yang biasa, kebanyakan sistem multimedia menggunakan teks karena hal tersebut merupakan cara yang efektif untuk mengkomunikasikan ide dan menyediakan instruksi untuk pengguna.

Terdapat empat macam teks, yaitu:

- a. Printed Text

Printed text adalah teks yang tercetak pada kertas.

- b. Scanned Text

Printed text yang telah diubah sedemikian rupa sehingga menjadi bentuk yang dapat dimengerti mesin komputer.

- c. Electronic Text

Teks dalam jumlah besar terapat dalam format yang dapat dibaca oleh mesin, sebab hampir semua orang yang menulis buku atau menerbitkan naskah hari ini mengerjakannya dengan pengolah kata dan alat penerbit elektronik.

- d. Hypertext

Huruf *hyper* menunjukkan proses penghubungan (linking), yang mana telah membuat multimedia menjadi bersifat interaktif. Kata hypertext dikenalkan oleh Ted Nelson (1965). Hypertext merujuk kepada text yang mempunyai link ke tempat ataupun dokumen lain.

- Grafik

Grafik seringkali muncul di belakang teks dan membuat kerangka bergambar. Gambar juga dapat berlaku sebagai *icon*, bercampur dengan teks, menunjukkan pilihan yang yang dapat dipilih; atau gambar dapat muncul *full-screen*, dengan menggunakan gambar sebagai pemicu, apabila pilihan dipilih, maka akan menampilkan objek multimedia yang lain.

- a. Bitmap

Bitmap adalah sebuah gambar yang disimpan sebagai satu kesatuan pixel yang terhubung dengan titik-titik pada layar komputer.

- b. Gambar Vektor

Gambar vektor disimpan sebagai satu set persamaan matematis algoritma yang menggambarkan kurva, garis, dan membentuk suatu gambaran. Gambar vektor mempunyai dua keuntungan jika dibandingkan dengan bitmap, yaitu :

- Pertama, gambar vektor bersifat dapat dihitung, kita dapat menggunakan program grafis untuk memperbesar atau mengurangi ukuran gambar tanpa kehilangan mutunya.
- Kedua, gambar vektor secara normal mempunyai ukuran file lebih kecil dibanding grafik bitmap, gambar vektor dapat kita download dengan cepat melalui internet.

- c. *Clip Art*

Beberapa diantaranya memiliki lebar, kategori umum mencakup foto, *icon*, animasi, gambar latar belakang, dan tombol, sedangkan *library* lainnya berpusat pada suatu tema visual seperti natural (alami) atau gambaran ilmiah.

d. Digitized Picture

Peng-*capture* video yang menghubungkan suatu kamera video, VCR, videodisc player, atau video siaran langsung dihubungkan ke komputer anda dan meng-*capture* frame dengan cepat ke dalam bitmaps dan dapat digunakan untuk segala aplikasi multimedia.

e. Hyperpicture

Ketika bagian-bagian dari gambar digunakan sebagai pemicu dari aksi suatu multimedia, maka bagian tersebut dapat dikatakan sebagai *hyperpicture*. Pemicu dapat berasal dari berbagai ukuran atau bentuk, dan anda dapat membuatnya tidak terlihat.

- Sound

Suatu produksi multimedia juga mengandung objek sound, terdapat empat tipe dalam objek sound, yaitu:

a. Waveform Audio

Sama halnya dengan *video digitizers* yang dapat digunakan untuk menangkap gambar manapun yang dapat dilihat oleh suatu kamera, *waveform audio digitizers* dapat merekam bunyi apapun yang dapat didengar manusia. Tiap-tiap bunyi mempunyai suatu bentuk gelombang yang menguraikan frekuensi, amplitudo, dan bunyi harmonik. *Waveform audio digitizers* menangkap bunyi dengan cara mengambil contoh bentuk gelombang ini beribu kali per detik; hasilnya disimpan pada suatu harddisk komputer dalam suatu file yang pada umumnya berbentuk *.wav*, yang merupakan singkatan dari *waveform*.

b. MIDI

MIDI merupakan singkatan dari Musical Instrument Digital Interface.

c. Audio CD

Audio CD dapat menyimpan rekaman suara selama lebih dari 75 menit.

d. CD Plus, CD Extra, dan Enhanced CD

CD Plus, atau yang biasa dikenal dengan CD Extra atau Enhanced CD, adalah CD yang memuat musik yang dapat pula berfungsi sebagai CD-ROM, dengan data komputer yang termasuk ke dalam disk penyimpanan musik.

e. MP3

MP3 merupakan singkatan dari MPEG Audio Layer 3. MP3 adalah format file audio yang menggunakan MPEG audio yang memuat codec untuk encode (menekan besar file) dan decode (menampilkan kembali file kepada ukuran sebenarnya) bagi musik yang telah direkam. MP3 dapat menekan ukuran file CD audio menjadi suatu file yang berukuran lebih kecil yang membutuhkan *bandwidth* kecil untuk pengiriman melalui internet tanpa mengurangi kualitas file.

f. Hyperaudio

Musik dimainkan dari waktu ke waktu. Banyak alat-alat kreasi multimedia memungkinkan pengaturan waktu kejadian object pada penunjuk *sync* musik tersebut. Ketika audio digunakan untuk memicu objek multimedia, hal tersebut disebut *hyperaudio*.

- Video

Video merupakan aplikasi yang kaya akan sumber daya multimedia. Terdapat empat tipe video yang dapat digunakan sebagai objek dalam aplikasi multimedia.

a. Live Video

Live video menyediakan objek *real-time* yang menarik.

b. Videotape

Media video yang paling banyak digunakan adalah videotape. Videotape dapat menjadi objek multimedia yang menghubungkannya dengan dokumen lain (*link*). Bagaimanapun, media ini dibatasi oleh dua faktor. Pertama, videotape bersifat linier. Artinya bahwa informasi disimpan pada tape dalam urutan serial, pengaksesan data membutuhkan waktu yang cukup lama, sekitar tiga menit. Kedua, sebagian besar videotape player tidak dapat digunakan bersamaan dengan komputer.

c. Videodisc

Terdapat dua tipe format untuk video disc, yaitu CAV dan CLV. CAV dapat menyimpan lebih dari 54.000 frame tak bergerak atau video berdurasi 30 menit dengan musik. CLV dapat menyimpan video berdurasi lebih dari 1 jam pada tiap sisinya.

d. Digital Video

Digital video adalah media penyimpan video yang paling menantikan dan menarik. Karena dapat digunakan melalui jaringan, mengurangi penggunaan videotape dan videodisc player.

e. DVD

DVD merupakan singkatan dari Digital Versatile Disc, tapi ketika tujuan penggunaan DVD untuk memutar balik movie, DVD menjadi Digital Video Disc. DVD menggunakan MPEG-2 untuk menekan ukuran file menjadi disc berukuran 4.7 inch.

f. Hypervideo

Seperti file musik, video klip juga dimainkan dari waktu ke waktu. Banyak alat-alat kreasi multimedia memungkinkan anda mengatur waktu kejadian object pada penunjuk sync video tersebut. Ketika video digunakan untuk memicu objek multimedia, hal tersebut disebut hypervideo.

- Animasi

Dalam multimedia, animasi adalah penggunaan komputer untuk menciptakan gerakan pada layar. Animasi terdiri dari empat macam, yaitu :

a. Animasi Frame

Animasi frame membuat objek bergerak dengan cara menampilkan beberapa gambar, disebut frame, sebagaimana objek tampil di layar dan berpindah-pindah.

b. Animasi Vektor

Vektor adalah sebuah garis yang memiliki awal, arah, dan ukuran. Animasi vektor membuat objek bergerak dengan merubah tiga parameter vektor yang membentuk suatu objek. Macromedia adalah pemimpin dalam software animasi berbasis vektor. Macromedia Flash menggunakan grafik vektor dalam penciptaan animasi dan grafik interaktif pada Web.

c. Animasi Terkomputasi

Pada animasi terkomputasi (computational animation), anda menggerakkan objek di layar dengan mengubah koordinat x dan y. Koordinat x menunjukkan posisi horizontal objek, kiri-kanan layar. Koordinat y menunjukkan posisi vertikal, arah layar ke bawah.

d. Morphing

Morphing berarti perubahan bentuk dengan menampilkan sekumpulan frame yang menciptakan pergerakan halus.

2.2 Rekayasa Piranti Lunak

2.2.1 Definisi Rekayasa Piranti Lunak

Rekayasa Piranti Lunak (*Software Engineering*) dapat didefinisikan sebagai berikut (Pressman, 2001, p20), yaitu:

Pembuatan dan penggunaan prinsip-prinsip Rekayasa Piranti Lunak untuk mendapatkan software yang ekonomis yaitu software yang dapat diandalkan dan bekerja secara efisien pada mesin (komputer) sesungguhnya, definisi ini pertama kali diusulkan oleh Firtz Bauer.

Menurut Steward (1987, p3), rekayasa piranti lunak adalah mengatur hal-hal yang ingin dicapai, teknologi komputer, sumber daya manusia, dan biaya untuk menciptakan sebuah produk piranti lunak yang mampu memenuhi kebutuhan dengan jalan pemenuhan prosedur.

2.2.2 Karakteristik Piranti Lunak

Piranti lunak (*software*) secara logika berbeda dengan elemen sistem secara fisik oleh karena itu piranti lunak mempunyai karakteristik yang sangat berbeda dari perangkat keras (*hardware*) tersebut. Karakteristik-karakteristik tersebut (Pressman, 2001, pp6-9), yaitu:

a. Software dikembangkan dan direkayasa

Walaupun ada beberapa kesamaan diantara pengembangan piranti lunak dengan pembuatan perangkat keras, kedua aktivitas tersebut pada dasarnya berbeda. Dari kedua aktivitas tersebut kualitas tinggi dapat dicapai dengan desain yang baik, tetapi untuk pembuatan perangkat keras dapat terjadi masalah kualitas yang tidak dapat terjadi pada piranti lunak. Biaya piranti lunak dikhususkan pada rekayasanya. Maksudnya bahwa proyek tidak dapat dikendalikan seperti pada pembuatan piranti lunak.

b. Software tidak rusak

Berbeda dengan *hardware* yang dapat rusak karena debu, getaran, suhu yang terlalu tinggi atau rendah, dan banyak lagi masalah lainnya. Ketika komponen hardware rusak dapat digantikan oleh komponen cadangan atau pengganti, tetapi pada *software* tidak ada cadangan pengganti. Kesalahan *software* dapat terjadi pada desain atau *software* tersebut, oleh karena itu pemeliharaan *software* sangat rumit daripada pemeliharaan *hardware*.

c. Walaupun industri sedang bergerak ke arah perakitan berdasarkan komponen yang tersedia, kebanyakan perangkat lunak dibuat karena pesanan

Hardware untuk sebuah produk mikroprosesor didesain dan dirakit. Seorang insinyur desain menggambarkan sebuah skema sederhana dari sebuah sirkuit digital, oleh karena itu dibutuhkan analisa yang mendasar untuk memastikan fungsi yang sesuai yang akan dicapai, kemudian menunjuk pada sebuah katalog komponen digital. Sedangkan pada seorang desainer software tidak memberikan kemewahan seperti hardware, tidak ada katalog untuk katalog komponen software. Dapat dimungkinkan untuk membeli sebuah software tetapi hanya sebagai sebuah

unit software yang utuh, tidak seperti komponen yang dapat dirakit ke dalam sebuah program baru.

2.2.3 Lapisan Rekayasa Piranti Lunak

Rekayasa Piranti Lunak mempunyai tiga lapisan (*layer*) (Pressman, 2001, pp20-21), yaitu :

a. Proses (*process*)

Merupakan fondasi dari rekayasa piranti lunak. Proses mendefinisikan kerangka untuk sebuah *key process area* yang harus dibentuk untuk pengiriman efektif dari teknologi rekayasa piranti lunak. *Key process area* ialah dasar untuk mengatur dari proyek *software* dan membantu mendirikan *method* sehingga dapat diaplikasikan.

b. metode-metode (*method*)

Method memberikan cara bagaimana membangun suatu *software* secara teknis.

Method tersebut terdiri dari :

1. analisa kebutuhan (*requirements analysis*)
2. desain (*design*)
3. konstruksi program (*program construction*)
4. percobaan (*testing*)
5. pemeliharaan (*maintenance*)

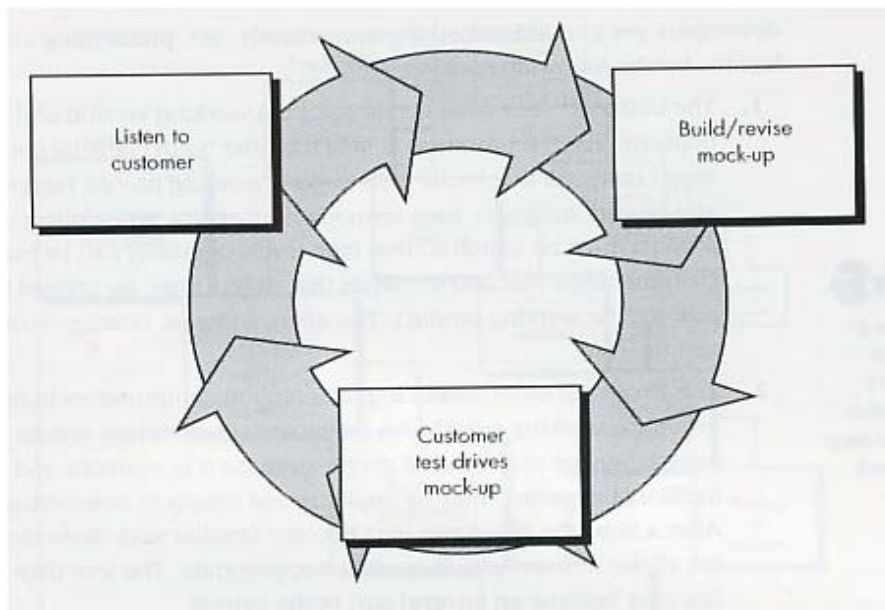
c. alat-alat bantu (*tools*)

Alat-alat bantu menyediakan dukungan otomatis atau semi otomatis untuk metode-metode. Salah satu contoh alat-alat bantu adalah *Computer Aided Software Engineering* (CASE) yang mengkombinasikan *software*, *hardware*, dan *software*

engineering database (suatu struktur data yang mengandung informasi penting mengenai analisis, desain, kode, dan testing).

2.2.4 Perancangan Piranti Lunak

2.2.4.1 Prototyping Model



Gambar 2.1 Prototyping Model

Dalam pengembangan piranti lunak, Prototyping model menggunakan tahapan sebagai berikut (Pressman, 2001, pp 30-32) :

- *Requirement Gathering*

Pengembang dan konsumen bertemu dan menggambarkan tujuan keseluruhan dari sistem software, kebutuhan, dan penggunaan software.

- *“Quick Design”*

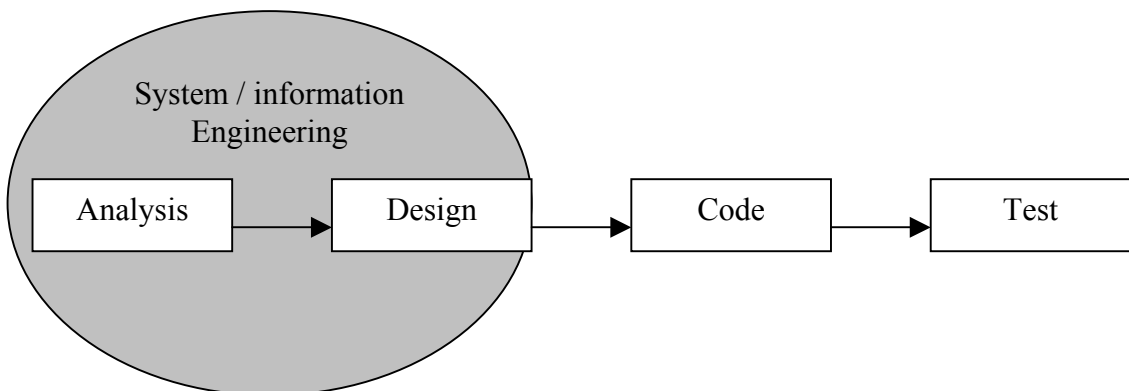
Bertitik berat pada penggunaan simbol-simbol yang mewakili semua aspek software dan mampu dimengerti oleh konsumen.

- *Prototype*

Dilakukan evaluasi oleh konsumen atau pengguna dan digunakan untuk menyaring kebutuhan dari pengembangan piranti lunak.

2.2.4.2 Linear Sequential Model

Biasa disebut dengan *classic life cycle* atau model *waterfall*, model linear sequential memberikan suatu model perancangan piranti lunak dengan pendekatan sequential yang sistematis, dimulai dari tahap sistem dan berlanjut ke analisis, desain, coding, testing, dan sistem pendukung.



Gambar 2.2 Linear Sequential Model

Tahapan dalam linear sequential model terletak pada (Pressman, 2001, pp 28-30) :

- *System/information engineering and modelling*

Karena suatu piranti lunak selalu menjadi bagian dari satu sistem besar (atau bisnis), maka perancangan pada model ini akan dimulai dengan pengumpulan

keperluan seluruh sistem, lalu menempatkan beberapa bagian dari keperluan piranti lunak tersebut.

- *Software Requirements Analysis*

Analisis harus mampu menerjemahkan segala hal yang berkaitan dengan piranti lunak, seperti *function*, *behavior*, *performance*, dan antar muka.

- *Design*

Desain piranti lunak merupakan proses yang terdiri dari beberapa proses yang berfokus kepada empat atribut program, yaitu :

- Struktur data
- Arsitektur piranti lunak
- *Interface representation*
- Detil proses (algoritma)

- *Code Generation*

Desain yang telah dibuat harus diterjemahkan ke dalam satu bentuk yang dapat dimengerti oleh mesin.

- *Testing*

Setelah algoritma di-generate, saat itu pulalah dimulainya tahap *testing*.

- *Support*

Sebuah piranti lunak akan berubah seiring dengan perjalanannya ke tangan konsumen. Perubahan akan dilakukan karena adanya *error* yang telah ditangani, piranti lunak harus beradaptasi dengan lingkungan luar.

2.2.4.3 The RAD Model

Rapid Application Development (RAD) adalah sebuah model proses pengembangan piranti lunak yang menekankan pada siklus pengembangan yang pendek.

RAD terdiri dari beberapa fase (Pressman, 2001, pp32-34), yaitu :

- *Business Modelling*

Informasi yang mengalir diantara fungsi-fungsi bisnis dibangun dengan cara menjawab pertanyaan-pertanyaan berikut :

- Informasi apa saja yang menggerakkan proses bisnis ?
- Informasi apa yang dihasilkan ?
- Siapa yang menghasilkan informasi tersebut ?
- Informasi tersebut akan diarahkan kemana ?
- Siapa yang memproses informasi tersebut ?

- *Data Modelling*

Aliran data yang digambarkan sebagai fase dalam model bisnis diolah menjadi satu set objek data yang diperlukan dalam mendukung bisnis.

- *Process modelling*

Objek-objek data digambarkan dalam fase data modelling diubah untuk mencapai aliran informasi yang penting untuk pelaksanaan fungsi bisnis.

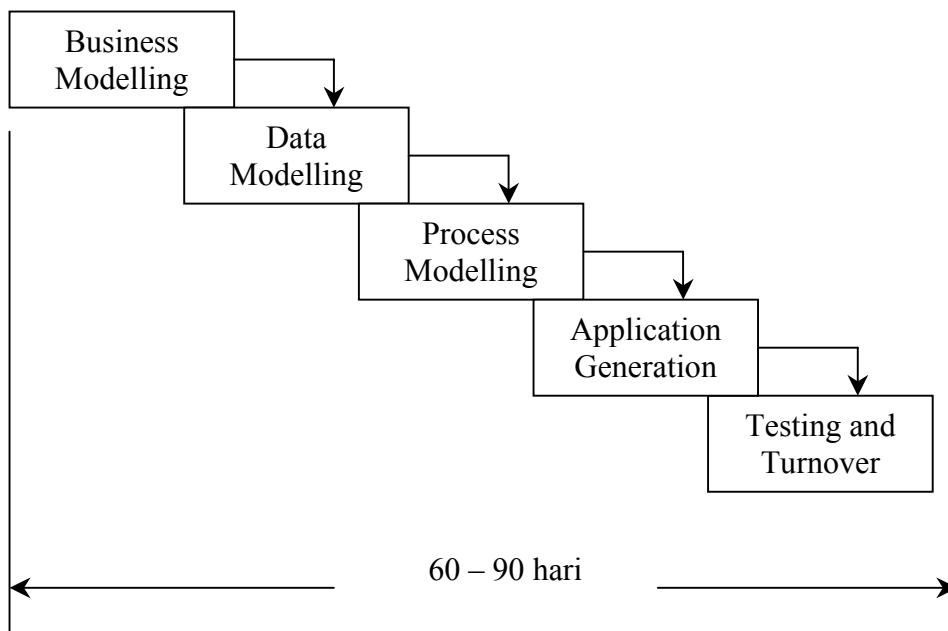
- *Application Generation*

Dilihat dari penggunaan teknik generasi ke-empat, daripada menciptakan piranti lunak dengan menggunakan bahasa pemrograman generasi ke-tiga, RAD bekerja dengan menggunakan kembali komponen program yang telah ada (jika

memungkinkan) atau menciptakan komponen yang dapat digunakan kembali (jika diperlukan).

- *Testing and Turnover*

Sejak proses RAD menekankan penggunaan kembali semua komponen piranti lunak, banyak komponen program yang telah dites.



Gambar 2.3 The RAD Model

2.2.4.4 Evolusi dalam Perancangan Piranti Lunak

Seiring dengan berkembangnya proses bisnis, maka piranti lunak pun berkembang mengikuti kegiatan bisnis yang menggunakan piranti lunak tersebut. Hal tersebut mengakibatkan ikut berkembangnya pula proses perancangan piranti lunak. Evolusi dalam perancangan piranti lunak (Pressman, 2001, pp34-42) yaitu :

- *The Incremental Model*
Pengkombinasian elemen-elemen dalam model *linear sequential* dengan *prototype* terdapat pada *incremental model*.
- *The Spiral Model*
Merupakan sebuah evolusi proses perancangan piranti lunak yang menggabungkan *prototype* dengan aspek-aspek sistematis dan terkontrol dari model *linear sequential*.
- *The WINWIN Spiral Model*
Dalam model ini, terdapat pencapaian kebutuhan oleh konsumen dan pengembang. Konsumen mendapatkan piranti lunak yang mampu memenuhi kebutuhan konsumen. Pengembang mampu menghasilkan produk atau sistem tepat waktu dan tidak melebihi biaya.
- *The Concurrent Development Model*
Model ini sering digunakan sebagai paradigma atas pengembangan aplikasi *client* atau *server*.

2.2.5 Kualitas Piranti Lunak

Beberapa faktor pengukur kualitas piranti lunak (Pressman, 2001, pp96-97) sebagaimana diungkapkan Gilb adalah sebagai berikut :

- *Correctness* – apakah sistem mengerjakan suatu perintah sesuai dengan yang diperintahkan?

- *Reliability* – berapa lama produk akan terus melakukan sesuai dengan spesifikasi, khususnya ditujukan untuk pemberian input baru dan lingkungan baru yang harus mampu ditangani oleh spesifikasi piranti lunak
- *Usability* – seberapa mudah penggunaan piranti lunak oleh pengguna tanpa, atau sebagai alternatif, seberapa baik keandalan sistem dalam melindungi pengguna yang baru pertama kali menggunakan sistem
- *Understandability / simplicity* - bagaimana seseorang dapat memahami dengan mudah apa yang dikerjakan sistem, bagaimana cara kerjanya, dan bagaimana cara mengendalikan dan menggunakan sistem
- *Robustness* - bagaimana nantinya sistem mengatasi masukan yang buruk dengan mengidentifikasinya dan membantu dengan mengoreksi mereka
- *Efficiency* - bagaimana sistem menggunakan sumber daya secara efektif untuk menangani permasalahan sama besar dan sama cepat seperti yang telah ditetapkan
- *Survivability* - seberapa baik sistem produk melaksanakan fungsinya dengan baik ketika lingkungan (sebagai contoh, perangkat keras) telah memburuk atau tidak mendukung
- *Integrity* - seberapa baikkah sistem melindungi dirinya sendiri terhadap tindakan buruk dari pengguna, atau akses yang tidak sah
- *Verifiability* – seberapa mudah sistem melakukan verifikasi bahwa sistem telah melakukan dengan tepat apa yang harus dilakukan
- *Maintainability* – seberapa murah perbaikan ataupun perubahan sistem
- *Versatility / flexibility* – seberapa murah melakukan perubahan prosedur pada produk agar melakukan sesuatu yang berada di luar kebiasaan dan tidak dilakukan antisipasi terhadap kebutuhan

- *Expandability* – seberapa mudah memperluas kemampuan sistem
- *Portability* – seberapa mudah sistem beroperasi pada lingkungan yang lain (sebagai contoh, tentang perangkat keras yang berbeda atau OS yang berbeda)
- *Reusability* – seberapa mudah penggunaan sistem ataupun bagian-bagiannya pada sistem yang lain
- *Interoperability* – seberapa mudah menggabungkan sistem dengan sistem yang lain

2.3 Data Flow Diagram (DFD)

2.3.1 Definisi Data Flow Diagram

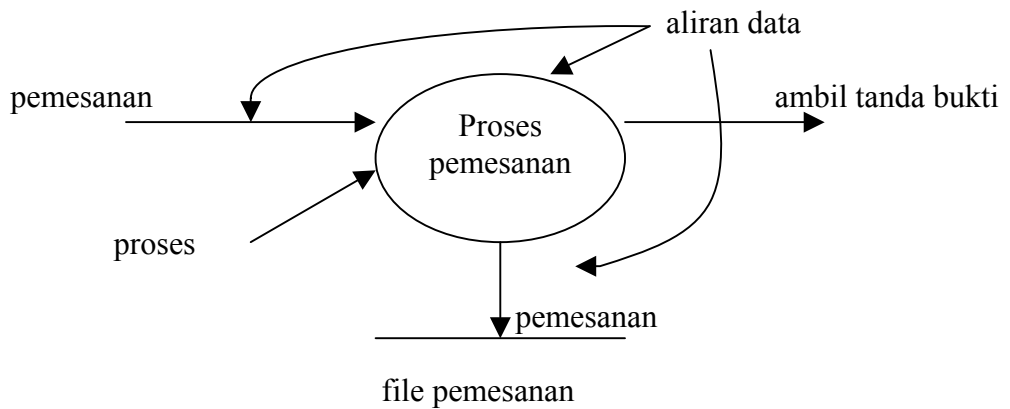
Menurut Pressman (1997, p47) *Data Flow Diagram* mengandung pengertian :

- DFD, merupakan penyederhanaan dari lukisan yang menggambarkan pergerakan data melalui sebuah sistem, menggunakan bahasa umum untuk kedua klien dan juga desainer dalam menggambarkan sistem.
- Menghindari penggunaan banyak simbol, DFD menggunakan simbol-simbol berbeda yang menggambarkan proses, aliran data, dan data file.
- DFD bertindak sebagai tabel isi, yang berisikan simbol berserta label tentang penjelasan lebih lanjut satu informasi dinamakan catatan tambahan.

2.3.2 Konsep Dasar DFD

Sebuah aliran data (data flow) adalah pergerakan informasi. Hal tersebut dapat menyebabkan sesuatu terjadi pada saat penerimaan informasi. Sebuah proses merubah input aliran data menjadi output aliran data. Ketika sebuah aliran data berhenti,

menunggu untuk dipakai pada lain waktu, maka ia menjadi data store, atau disebut juga dengan file.



Gambar 2.4 Konsep DFD (Pressman, 1997, p48)

Terdapat dua tipe DFD (Pressman, 1997, p 48) , yaitu :

- DeMarco

Pendekatan DeMarco dijelaskan dalam konsep membuat diagram yang dapat dimengerti oleh klien sehingga mereka dapat mengatakan dari awal apakah yang dirancang sudah sesuai dengan yang diinginkan.


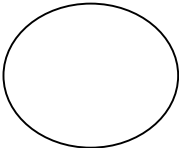

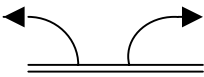
- Two-Entity

Dapat digunakan sebagai penyederhanaan dari versi DeMarco, tapi dapat pula diperluas tergantung dengan keinginan proyek.

2.3.3 Elemen DFD

Karena informasi bergerak melalui piranti lunak, maka dimodifikasi oleh beberapa informasi. DFD adalah teknik grafis yang menggambarkan aliran informasi dan perubahan yang terjadi saat data bergerak dari input ke output. Aliran informasi tersebut

mempunyai simbol-simbol yang dapat dimengerti, adapun simbol-simbol tersebut adalah (Pressman, 1997, p 166) :

- *External entity*
 Digunakan untuk mewakili elemen sistem (misal perangkat keras, manusia) atau sistem lain yang memproduksi informasi ketika terjadi perubahan oleh piranti lunak
- *Proses*
 Mewakili sebuah proses atau perubahan yang diterapkan kepada data dan merubahnya dengan beberapa cara.
- *Data Flow*
 Menunjukkan satu atau lebih item data. Seluruh panah dalam DFD diberi label.
- *Data Store*
 Dua garis menunjukkan *data store*, penyimpan informasi yang nantinya akan digunakan oleh piranti lunak.

2.3.4 Pedoman Penyusunan DFD

Dalam penyusunan DFD yang baik, diperlukan pemahaman terhadap aturan-aturan perancangan DFD. Berikut ini merupakan pedoman dalam penyusunan suatu DFD :

- Pedoman pemberian nama proses
 - Nama proses terdiri dari kata kerja dan kata benda yang mencerminkan fungsi proses tersebut
 - Jangan menggunakan kata ‘proses’ sebagai bagian dari nama suatu proses

- Tidak boleh ada beberapa proses yang memiliki nama yang sama
- Proses harus diberi nomor, urutan nomor sedapat mungkin mengikuti aliran atau urutan proses
- Penomoran proses pada tingkat pertama (Diagram Nol) adalah 1.0, 2.0, dan seterusnya
- Penomoran proses pada tingkat kedua dari proses 1.0 (rincian dari proses 1.0) adalah 1.1, 1.2, dan seterusnya.
- *Context Diagram* tidak perlu diberi nomor
- Pedoman pemberian nama aliran data (*Data Flow*)
 - Nama aliran data yang terdiri dari beberapa kata dihubungkan dengan garis sambung
 - Tidak boleh ada aliran data yang namanya sama dan pemberian nama harus mencerminkan isinya
 - Aliran data yang terdiri dari beberapa elemen dapat dinyatakan dengan grup elemen
 - Hindari penggunaan kata 'data' dan 'informasi' untuk memberi nama dengan grup elemen
 - Sedapat mungkin nama aliran data ditulis lengkap
- Pedoman pemberian nama *data store*
 - Nama harus mencerminkan isi data store tersebut
 - Bila namanya lebih dari satu kata, maka harus diberi tanda sambung
- Pedoman pemberian nama *external entity*
 - Nama terminal berupa kata benda

- Terminal tidak boleh memiliki nama yang sama kecuali memang objeknya sama. Bila demikian, maka terminal ini perlu diberi garis miring pada pojok kiri atas
- Ketentuan lain
 - Nama aliran data yang masuk ke dalam suatu proses tidak boleh sama dengan nama aliran data yang keluar dari proses tersebut
 - Data flow yang masuk ke atau keluar dari data store tidak perlu diberi nama bila :
 - Aliran datanya sederhana dan mudah dipahami
 - Aliran data menggambarkan seluruh item (satu record utuh)
 - Tidak boleh ada aliran data dari terminal ke data store atau sebaliknya karena terminal bukan bagian dari sistem. Hubungan terminal dengan data store harus melalui proses.

2.4 State Transition Diagram (STD)

2.4.1 Definisi STD

State Transition Diagram (STD) adalah grafik yang menggambarkan hasil dari perubahan state selama proses. STD merupakan alat model (*modelling tool*) yang menggambarkan sifat ketergantungan pada waktu dari suatu sistem *real-time* dan *user interface* pada suatu sistem *on-line*. Contohnya : proses kendali, telepon, dan lain sebagainya. Menurut Kowal (1989, p327), untuk menjelaskan atau menggambarkan proses yang berlangsung dalam suatu program, maka digunakan STD.

STD pada dasarnya merupakan sebuah diagram yang terdiri dari *state* dan transisi atau perpindahan *state*. Transisi terdiri dari kondisi dan aksi. Kondisi adalah suatu

kejadian yang dapat diketahui oleh sistem sedangkan aksi adalah tindakan yang dilakukan oleh sistem bila terjadi pembahasan *state* atau dapat juga merupakan reaksi terhadap kondisi.

2.4.2 Komponen STD

Komponen-komponen yang digunakan dalam STD antara lain (Kowal, 1989):

- *State*

State adalah kumpulan keadaan atau atribut yang mencirikan seseorang atau suatu benda pada waktu, bentuk keadaan, atau kondisi tertentu. State dinotasikan dengan empat persegi panjang berisi keadaan yang terjadi di dalam sistem.

keadaan

Keadaan sistem seperti berupa menunggu pengguna memasukkan password, menunggu konfirmasi dari pengguna, menunggu perintah selanjutnya, dan lain-lain.

- Perubahan State

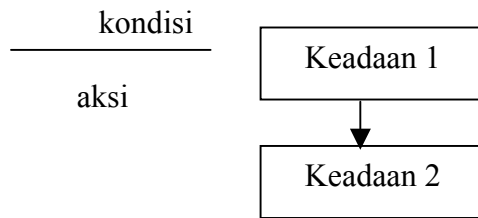
Perubahan state merupakan perubahan kondisi yang digambarkan dalam STD dengan menghubungkan dua keadaan yang berkaitan dengan suatu garis panah.

Notasinya : \longrightarrow

- Kondisi dan Aksi



Untuk membuat STD menjadi lengkap diperlukan dua hal tambahan, yaitu kondisi dan aksi. Kondisi adalah suatu kejadian pada lingkungan luar yang dapat dideteksi oleh sistem, sedangkan aksi adalah yang dilakukan sistem bila terjadi perubahan state atau merupakan reaksi terhadap kondisi. Aksi akan menghasilkan keluaran, tampilan pesan di layar, melakukan kalkulasi, dan lain-lain. Kondisi dan aksi

digambarkan di dekat panah yang menghubungkan dua keadaan yang berkaitan seperti gambar berikut :



Gambar 2.5 Komponen Dasar STD

Berikut ini adalah tabel notasi STD :

Notasi	Fungsi
	Melambangkan state atau keadaan atau kondisi dari suatu sistem
	Melambangkan perubahan state atau keadaan atau kondisi dari suatu sistem

Tabel 2.1 Tabel Notasi State Transition Diagram (STD)

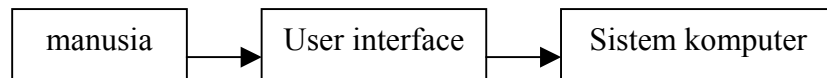
Notasi STD ini berguna dalam memudahkan pembuatan suatu sistem dengan benar dan jelas. Manfaat lain yang dapat diperoleh dalam penggunaan STD adalah agar suatu sistem dapat dipahami dengan benar.

2.5 Interaksi Manusia dan Komputer

2.5.1 Pengertian User Interface

Berikut ini beberapa pengertian tentang *user interface* :

- Menurut Burger (1993, p100) *user interface* adalah cara untuk mengkomunikasikan keinginan orang dalam sistem operasi.
- Menurut Angel (1990, p4) *user interface* adalah antarmuka antara manusia dengan komputer yang telah diubah secara radikal dengan menggunakan grafik komputer.



Gambar 2.5 Hubungan antara manusia, user interface, dan sistem komputer

2.5.2 Pengertian Interaksi Manusia dan Komputer (IMK)

Pengertian IMK adalah ilmu tentang bagaimana interaksi antara manusia dengan komputer dan perlu tidaknya komputer dikembangkan untuk kesuksesan interaksi dengan manusia (<http://searchwebservice.techtarget.com/definition/human-computer-interaction.html>,2002).

2.5.3 Perancangan User Interface

Terdapat delapan aturan emas dalam perancangan *user interface* (Shneiderman, 1998, pp 74-75), yaitu :

1. Konsistensi.

Aturan ini adalah paling sering dilanggar. Urutan tindakan yang konsisten harus diperlukan situasi serupa.; istilah serupa harus yaitu digunakannya *prompt*, menu,

dan layar bantuan, dan warna konsisten, tata ruang, kapitalisasi, huruf, dan seterusnya harus dipekerjakan dalam keseluruhannya.

2. Sering penggunaan *shortcut*

Seperti frekuensi pengguna yang meningkat, demikian juga keinginan pemakai untuk mengurangi banyaknya interaksi dan untuk meningkatkan langkah interaksi.

3. Tawarkan umpan balik informatif.

Karena pada tiap tindakan pemakai harus terdapat umpan balik sistem. Tanggapan harus bersifat fleksibel dan dinamis, untuk tindakan kecil dan sering, tanggapan dapat bersifat sederhana, sedangkan untuk tindakan utama dan jarang, tanggapan harus bersifat lebih utama.

4. Desain dialog untuk menghasilkan penutup.

Urutan tindakan harus terorganisir ke dalam kelompok dengan terdiri dari permulaan, pertengahan, dan akhiran. Umpan balik yang informatif di penyelesaian suatu kelompok tindakan memberikan operator kepuasan dalam memenuhi keinginan konsumen.

5. Pencegahan satu kesalahan dan memberikan solusinya.

Sedapat mungkin, mendesain sistem dimana para pemakai tidak akan membuat suatu kesalahan serius; sebagai contoh, menyukai pemilihan menu untuk membentuk suatu karakter dan tidak mengijinkan memasuki sistem. Jika para pengguna melakukan kesalahan, sistem akan mendeteksi kesalahan tersebut dan menawarkan solusi sederhana, bersifat membangun, dan instruksi spesifik untuk pemulihan kondisi sistem.

6. Memungkinkan adanya umpan balik.

Sebisa mungkin setiap tindakan diikuti dengan aksi balik dengan cepat dari komputer.

7. Pergunakan kontrol internal.

Operator mengetahui dengan pasti bahwa mereka bertanggung jawab terhadap kelangsungan hidup sistem, hal tersebut dilakukan dengan adanya kontrol internal.

8. Kurangi penggunaan memori dalam jangka pendek.

Keterbatasan dalam pemrosesan informasi membuat mesin komputer bekerja lebih berat dari yang seharusnya.

2.5.4 Tujuan Perancangan User Interface

Adapun tujuan dari perancangan *user interface* (Shneiderman, 1998, pp14-16) adalah sebagai berikut :

1. Waktu untuk belajar.

Berapa lama waktu yang dibutuhkan untuk anggota dari masyarakat pemakai untuk belajar bagaimana cara menggunakan perintah yang relevan bagi satu set tugas?

2. Kecepatan tampilan

Berapa lama waktu yang dibutuhkan untuk melakukan aksi terhadap pengguna ?

3. Tingkat kesalahan para pemakai

Berapa banyak dan kesalahan macam apa dilakukan pengguna saat menggunakan sistem? Walaupun waktu untuk membuat dan mengoreksi kesalahan akan disatukan dengan kecepatan tampilan, penanganan kesalahan adalah komponen kritis yang patut diperhitungkan.

4. Ingatan dari waktu ke waktu

Seberapa baikkah para pemakai memelihara pengetahuan mereka setelah satu jam, satu hari, satu minggu?

5. Kepuasan Subyektif

Berapa banyak pengguna menggunakan berbagai aspek sistem?

2.5.5 Tiga Pilar dalam Desain

Tiga pilar dalam desain dapat membantu perancang antar muka dalam mengubah ide menjadi suatu sistem. Tiga pilar tersebut adalah sebagai berikut (Shneiderman, 1998, pp100-103) :

- Pedoman proses dan dokumen

Setiap proyek memiliki kebutuhan yang berbeda, tapi pedoman ini harus berisi :

- Kata dan *icon*

Terdiri dari penggunaan terminologi, karakter huruf, ukuran huruf, tipe huruf, *icon*, grafik, garis, penggunaan warna, latar belakang, dan lain-lain.

- *Screen-layout*

Terdiri dari seleksi menu, umpan balik, pesan *error*, margin, *data entry* dan *display*, *header* dan *footer*, dan lain-lain.

- Peralatan input dan output

Terdiri dari *keyboard*, *sound system*, dan lain sebagainya.

- Urutan aksi

Seperti *clicking*, maupun sintaks yang digunakan.

- Training

Seperti menu *help* yang tersedia, juga tutorial, dan materi-materi yang mendukung.

- Piranti lunak *user-interface*

Penggunaan versi cetak untuk rancangan memang sangat membantu, tapi akan lebih membantu apabila rancangan tersebut dibuat semenarik mungkin dengan adanya interaksi melalui keyboard dan mouse.

- Tinjauan tenaga ahli dan testing penggunaan

Metode ini membutuhkan tinjauan dari tenaga ahli dalam hal kritik dan saran. Test penggunaan dilakukan oleh pengguna yang akan memakai sistem, survei, dan analisa secara otomatis juga diperlukan.

2.6 Kios Informasi

2.6.1 Pengertian Informasi

Menurut Davis (1982, p200) informasi adalah data yang diproses ke dalam bentuk yang berarti bagi penerimanya dan memiliki nilai yang *real* maupun yang dipersepsikan dalam mengambil tindakan dan keputusan pada saat ini maupun yang akan datang. Definisi tersebut memperlihatkan nilai dari informasi dalam memotivasi, pembuatan model, dan pembentukan latar belakang yang mempengaruhi pengambilan tindakan dan keputusan pada masa yang akan datang.

Informasi adalah data yang telah dikumpulkan oleh pengguna yang mana dalam mengumpulkan data guna menjadi informasi dan data tersebut dimasukkan ke dalam variabel masalah jika dapat mendukung masalah atau penyelesaian masalah maka kumpulan data tersebut adalah informasi.

Dimensi dari informasi yang mempengaruhi nilai dari informasi, menurut McLeod (2001, p107) adalah :

- *Relevancy*

Informasi harus menunjang dan berhubungan dengan masalah yang sedang dihadapi

- *Accurancy*

Informasi yang menyangkut finansial seperti misalnya piutang, penggajian, dan lainnya harus mempunyai keakuratan yang tinggi.

- *Timeliness*

Informasi harus tersedia untuk membantu pemecahan masalah sebelum situasi menjadi kritis, informasi harus dapat menggambarkan situasi pada saat ini sebagai tambahan dari gambaran situasi yang telah lampau.

- *Completeness*

Informasi dapat merepresentasikan gambaran yang lengkap dari situasi atau problem.

2.6.2 Pengertian Kios Informasi

Dengan semakin berkembangnya teknologi informasi yang diiringi dengan perkembangan media. Penyampaian informasi yang berupa teks, gambar, grafik, animasi, suara, dan video, maka jika digabungkan dapat menjadi suatu sistem yang dapat berinteraksi dengan pemakai untuk memberikan perintah, mengendalikan, dan memanipulasi suatu objek dengan dibantu suatu alat masukan tertentu dan sistem akan

memberikan respon. Sistem yang dapat berinteraksi dengan pemakai dimana dalam kios informasi akan menampilkan informasi disertai tampilan yang menarik.

Kios adalah struktur fisik yang kecil, biasanya termasuk sebuah komputer dan tampilan layar, dimana tampilan mengenai informasi untuk manusia dan termasuk suara dan video (<http://whatis.techtarget.com/definition/kiosk-definition.html>, 2002)

Kios informasi adalah sebuah bangunan fisik (sering kali berupa sebuah komputer dengan layar tampilan yang menarik) untuk menyajikan informasi untuk orang yang hanya sekedar lewat saja. (<http://iroi.seu.edu.cn/books/whatis/kiosk.htm>,2002).

Sistem kios informasi biasanya ditempatkan di tempat umum, yang mudah dilihat oleh pemakai. Sistem yang ada harus dapat dengan mudah digunakan oleh pengguna dan *user-friendly*, waktu akses sistem tersebut juga harus cepat sehingga pengguna tidak perlu menunggu lama untuk mendapatkan informasi.

2.6.3 Manfaat Kios Informasi

Manfaat utama yang dapat diperoleh pemakai suatu kios informasi adalah pemakai dapat memperoleh informasi yang diinginkan dengan mudah, cepat, dan sesuai dengan apa yang diharapkan tentang informasi tersebut. Informasi yang diperoleh juga merupakan informasi yang teraktual karena informasi yang tersedia dalam suatu kios informasi selalu diperbaharui sesuai dengan keadaan yang ada.

2.7 Database

2.7.1 Pengertian Database

Menurut C.J. Date (2000, p2) database adalah koleksi yang berisi tentang data logis terkait (dan uraian tentang data tersebut), yang dirancang untuk memenuhi kebutuhan informasi dari suatu organisasi.

Suatu sistem database pada dasarnya hanya sistem pencatatan terkomputerisasi. Database sendiri dapat digunakan sebagai satu alat pengarsipan elektronik. Para pemakai sistem dapat melaksanakan berbagai operasi pada file-file yang tersimpan, sebagai contoh penambahan baru, memasukkan atau menyisipkan data, pengambilan data, mengubah data, menghapus data, pemindahan data yang telah tersimpan.

2.7.2 Komponen Database

Sistem Database melibatkan empat komponen utama (Date, 2000, pp 5-9), yaitu :

- Data

Secara umum, data yang tersimpan dalam database - sedikitnya di dalam suatu sistem - kedua-duanya akan terintegrasi dan dapat digunakan bersama.
- Perangkat Keras

Komponen perangkat keras terdiri dari:

 - *Secondary storage volume*

Kebanyakan berupa disk magnetis, yang digunakan untuk mengatur data tersimpan, bersama-sama dengan alat *input-output (disk drive)*, pengontrol, saluran input-output dan sebagainya

- Perangkat keras *prosesor* dan berhubungan dengan *main memori* yang digunakan untuk mendukung kerja *software* sistem database

- Piranti Lunak

Antara database fisik sendiri - yaitu bagaimana sebuah data disimpan secara fisik - dan para pemakai sistem adalah suatu lapisan piranti lunak, dikenal dengan berbagai cara sebagai database manager atau server database atau kebanyakan, paling umum, sistem management database. Semua permintaan pengaksesan database ditangani oleh DBMS (*Database Management System*)

- Para Pemakai

Para pengguna sistem dikelompokkan dalam tiga kelas, yaitu :

- Pertama, para programmer aplikasi yang bertanggung jawab dalam penulisan program aplikasi database dalam beberapa bahasa program seperti COBOL, PL/I, C++, Java, atau beberapa bahasa pemrograman tingkat lebih tinggi "generasi keempat".
- Kelas pemakai yang kedua, kemudian, adalah mereka yang disebut sebagai *end-user*, yang berhubungan dengan sistem dari terminal komputer. Mereka dapat mengakses database melalui salah satu aplikasi yang telah disebutkan sebelumnya, atau menggunakan suatu alat penghubung yang berfungsi sebagai suatu bagian integral perangkat lunak sistem database.
- Kelas pemakai yang ketiga, adalah pengurus database atau DBA (*Database Administrator*).