

BAB 2

LANDASAN TEORI

2.1. Multimedia

Multimedia menjadi kunci yang penting dalam penyerapan informasi yang efektif.

2.1.1. Pengertian Multimedia

Multimedia adalah kemampuan yang memungkinkan integrasi dari teks, grafik beresolusi tinggi, video, animasi, dan suara yang berbasiskan pada komputer. (Long, 2000, p13)

Menurut Hofstetter (2001, p2), terdapat empat komponen penting dalam multimedia, antara lain:

- a. Harus menjadi komputer yang dapat mengkoordinasikan apa yang didengar, apa yang dilihat, dan dapat berinteraksi dengan pengguna.
- b. Ada penghubung (link) yang menghubungkan ke informasi yang ada.
- c. Ada alat penjelajah yang memperbolehkan pengguna berpindah ke informasi lain yang berhubungan dengan informasi sebelumnya.
- d. Adanya cara untuk mengumpulkan, memproses, dan menyampaikan informasi dan ide sendiri.

2.1.2. Elemen-elemen Multimedia

Menurut Hofstetter (2001, p16), elemen-elemen multimedia antara lain:

a. Teks

Teks merupakan suatu cara yang efektif untuk menyampaikan ide-ide dan memberikan instruksi kepada pengguna. Teks di sini berarti kata-kata yang muncul pada layar.

b. Gambar

Sering dikatakan bahwa gambar mewakili ribuan kata-kata. Bagaimanapun juga, hal itu benar jika kita dapat menampilkan gambar yang kita inginkan pada saat kita memerlukannya. Multimedia mengijinkan dilakukannya hal ini jika gambar menjadi objek dari sebuah penghubung (link). Gambar juga dapat disajikan sebagai sebuah ikon, dapat digabungkan dengan teks (seringkali muncul sebagai latar belakang dari teks), mewakili sebuah pilihan yang dapat dipilih, atau gambar dapat ditampilkan selayar penuh, di mana terdapat bagian dari gambar yang merupakan suatu pemicu yang jika dipilih akan meluncurkan suatu kejadian atau objek multimedia lain. Macam-macam gambar antara lain:

1) *Bitmaps*

Bitmaps adalah sebuah gambar yang tersimpan sebagai kumpulan dari *pixel*.

2) Gambar vektor (*vector image*)

Gambar vektor disimpan sebagai kumpulan algoritma yang mendefinisikan garis lengkung, garis lurus, dan bentuk dalam sebuah gambar. Untuk gambar yang tidak banyak mengandung perubahan warna, vektor merupakan suatu cara yang lebih efisien dalam menyimpan gambar dibandingkan dengan bitmaps. Sebagai contoh garis diagonal. Bitmaps akan menyimpan setiap titik sepanjang diagonalnya sebagai nilai warna RGB (*Red-Green-Blue*, sebuah

model warna). Sebaliknya gambar vektor hanya menyimpan titik awal garis, arahnya, panjangnya, dan warnanya. Gambar vektor memiliki dua keuntungan dibandingkan bitmaps. Pertama, gambar vektor dapat diubah-ubah skalanya tanpa kehilangan kualitas dari gambar. Kedua, karena umumnya file gambar vektor berukuran lebih kecil dibandingkan gambar bitmaps, maka gambar vektor lebih cepat di-*download* dari internet.

3) *Clip art*

Menciptakan gambar menggunakan tangan adalah membuang waktu. Untuk tidak membuang-buang waktu, terdapat banyak koleksi clip art yang dapat digunakan dalam memproduksi multimedia yang banyak disediakan di internet dengan gratis.

4) *Hyperpicture*

Saat bagian dari gambar digunakan sebagai pemicu kejadian multimedia, mereka disebut sebagai hyperpicture.

c. Suara

Terdapat bermacam-macam suara yang dapat digunakan untuk memproduksi suatu multimedia, antara lain:

1) *Waveform audio*

Waveform audio dapat merekam suara apa saja yang didengar. Setiap suara memiliki *waveform* yang menjelaskan frekuensi, amplitudo, harmoni suara tersebut. Waveform audio disimpan dalam piranti keras komputer dalam sebuah file yang memiliki *extension* .wav.

2) *MIDI (Musical Instrument Digital Interface)*

Merupakan salah satu cara yang paling efisien dalam merekam musik. Daripada merekam waveform dari suara, di mana memerlukan kapasitas penyimpanan yang besar, rekaman MIDI membutuhkan *chip* suara komputer untuk memainkan musik. Sebagai contoh, terdapat kode dalam MIDI untuk mengubah *on* dan *off*, membuat musik menjadi keras atau lembut, memberikan efek-efek pada suara. MIDI memiliki extension .mid.

3) Audio CD

Dapat menyimpan 75 menit rekaman suara dengan teliti dan murni.

4) *CD Plus, CD Extra, dan Enhanced CD*

CD Plus yang disebut juga CD Extra atau Enhanced CD dapat berfungsi seperti *CD-ROM*, dengan data komputer termasuk dalam *disk* musik. Jika CD Plus dimasukkan ke dalam *audio CD Player*, maka kita dapat mendengarkan musik seperti biasanya. Dengan memasukkan CD Plus dalam komputer multimedia maka program komputer akan mendukungnya dengan gambar yang mempesona, memiliki navigasi, dan dapat berinteraksi.

5) *MP3*

MP3 ada untuk *MPEG Audio Layer 3*. Ini adalah sebuah format file audio yang menggunakan *MPEG audio codec* untuk *encode (compress)* dan *decode (decompress)* rekaman musik. MP3 dapat mengompres *CD audio track* menjadi ukuran yang lebih kecil.

6) *Hyperaudio*

Saat audio digunakan untuk memicu objek multimedia, mereka disebut hyperaudio.

d. Video

Video menyediakan sumber yang kaya dan hidup untuk aplikasi multimedia, seperti animasi, tetapi biasanya diperoleh dari adegan dunia nyata yang disimpan dalam *file*.

e. Animasi

Dalam multimedia, sebuah animasi digunakan untuk menciptakan suatu pergerakan di layar. Terdapat 4 macam animasi, antara lain :

1) *Frame*

Membuat objek bergerak dengan menggunakan beberapa frame.

2) Vektor

Sebuah vektor merupakan garis yang memiliki titik awal, arah, dan panjang. Animasi vektor membuat objek bergerak dengan menggunakan ketiga parameter di atas. *Macromedia* adalah industri yang mengawali piranti lunak berbasis vektor untuk membuat animasi. *Macromedia flash* menggunakan gambar vektor untuk menciptakan animasi dan gambar yang interaktif.

3) *Computational Animation*

Jika ingin membuat kata berjalan lurus di layar, dapat dilakukan 2 cara. Pertama, dapat dibuat sejumlah frame di mana setiap frame mewakili seiap inci pergerakan kata. Hal ini tidaklah efisien karena setiap frame menghabiskan banyak memori dan menghabiskan banyak waktu sebagian

orang untuk menggambar tiap-tiap frame. Kedua, dengan perhitungan animasi dapat memindahkan objek melewati layar dengan mengubah-ubah koordinasi titik x dan y.

4) *Morphing*

Morphing berarti peralihan dari suatu bentuk ke bentuk lain dengan memunculkan sejumlah frame yang dibuat dengan pergerakan yang sangat halus.

2.1.3. Aplikasi Multimedia

Menurut Hofstetter (2001, p30-115) multimedia dapat diterapkan di:

- a. Bisnis dan Industri
 - 1) presentasi bisnis
 - 2) kios informasi interaktif
 - 3) belanja secara online
- b. Pendidikan
 - 1) perangkat ajar bahasa
 - 2) perangkat ajar berhitung untuk anak-anak
 - 3) alat peraga biologi
 - 4) kamus
- c. Hiburan
 - 1) *video game*
 - 2) film animasi
 - 3) *virtual reality*

d. Kedokteran dan Pengobatan

Multimedia dapat menciptakan suatu paket informasi kedokteran seperti misalnya anatomi tubuh manusia, juga dapat digunakan dalam hal pelatihan kedokteran seperti latihan pembedahan.

e. Ensiklopedia

Dengan multimedia dapat dibuat suatu ensiklopedia dalam bentuk *CD* atau *DVD*. Ensiklopedia ini menyediakan suatu pencarian sehingga mudah menemukan informasi yang diinginkan.

2.1.4. Langkah-langkah Pengembangan Sistem Multimedia

Menurut Mcleod (1996, p139-140), langkah-langkah yang bisa diambil dalam pengembangan sebuah sistem multimedia antara lain :

1. Mendefinisikan masalah.

Analisis sistem mengidentifikasi kebutuhan pengguna dan menentukan bahwa pemecahannya memerlukan multimedia.

2. Merancang konsep.

Analisis sistem dan pengguna mungkin bekerja sama secara profesional di bidang multimedia dalam merancang konsep yang menentukan keseluruhan pesan dan memeriksa semua urutan utama.

3. Merancang isi.

Pengembang terlibat dalam merancang isi dengan menetapkan spesifikasi yang rinci, di sinilah media dipilih untuk menulis naskah.

4. Menulis naskah.

Dialog dari semua elemen dari urutan tertentu.

5. Merancang grafik.

Grafik dipilih untuk mendukung dialog, latar belakang / perlengkapannya perlu dipergunakan dalam merancang video.

6. Memproduksi sistem

Pengembang sistem memproduksi berbagai bagian dan menyatukannya dengan sistem, selain mengembangkan perangkat lunak aplikasi tugasnya mencakup kegiatan khusus seperti menyunting video.

7. Melakukan tes pengguna.

Analisis sistem melakukan pengetesan langsung terhadap sistem kepada calon pengguna dan jika belum memuaskan maka dilakukan *prototyping* ulang.

8. Menggunakan sistem.

Setelah pengguna setuju dan puas, dengan sistem yang dibuat, maka sistem tersebut segera diimplementasikan.

9. Memelihara sistem.

Seperti sistem berbasis lain, sistem multimedia harus diperlihora namun pengguna tidak dapat diharapkan untuk melakukan pemeliharaan. Ini adalah tugas spesialis dan profesional multimedia.

2.2. Interaksi Manusia dan Komputer

Bidang ini memungkinkan seolah-olah manusia dan komputer berinteraksi satu sama lain. Hal ini mendukung penyampaian informasi dengan baik.

2.2.1. Pengertian Interaksi Manusia dan Komputer

Definisi interaksi manusia dan komputer (Shneiderman, 1998, p11) adalah disiplin ilmu yang berhubungan dengan perancangan, evaluasi, dan implementasi sistem komputer interaktif yang digunakan oleh manusia serta studi fenomena-fenomena besar yang berhubungan dengannya.

Menurut Shneiderman (1992, p4) antarmuka pengguna (*user interface*) adalah bagian dari sistem komputer yang memungkinkan manusia berinteraksi dengan komputer.

2.2.2. Faktor-faktor manusia

Faktor-faktor manusia terukur yang perlu dievaluasi, menurut Shneiderman (1992, p4), yaitu :

a. Waktu belajar

Berapa lama orang awam dalam masyarakat pengguna mempelajari cara menggunakan suatu sistem?

b. Kecepatan kinerja

Berapa lama suatu tugas dilakukan?

c. Tingkat kesalahan pengguna

Berapa banyak kesalahan dan kesalahan-kesalahan apa saja yang dibuat oleh pengguna dalam melakukan tugas?

d. Retensi

Bagaimana kemampuan pengguna dalam mengingat pengetahuan setelah jangka waktu tertentu? Hal ini berhubungan dengan waktu belajar dari si pengguna (lihat poin a).

e. Kepuasan Subjektif

Bagaimana kesukaan pengguna terhadap berbagai aspek sistem?

2.2.3. Aturan Perancangan Antarmuka

Delapan aturan emas dalam perancangan antarmuka, menurut Shneiderman (1992, p74), yaitu:

a. Berusaha keras untuk konsisten

Aturan ini paling sering dilanggar, tetapi jika dituruti akan lebih rumit karena ada banyak bentuk konsistensi. Urutan aksi harus konsisten dalam situasi yang sama; istilah yang sama harus digunakan dengan tepat pada menu-menu dan layar bantuan; begitu juga dengan konsistensi dalam hal warna, tampilan, huruf besar, jenis huruf, dan sebagainya.

b. Memungkinkan pengguna lebih sering menggunakan *shortcut*.

Adanya peningkatan penggunaan shortcut untuk mengurangi jumlah interaksi dan meningkatkan kecepatan tampilan.

c. Memberikan umpan balik (*feedback*) yang informatif.

- d. Merancang dialog untuk menghasilkan keadaan akhir (selesai, sukses).

Urutan aksi harus diatur dalam grup di mana ada awal, tengah, dan akhir dengan adanya umpan balik yang dapat memberikan pilihan untuk menyiapkan grup aksi berikutnya.

- e. Memberikan penanganan kesalahan yang sederhana

Jika memungkinkan, sistem harus dirancang agar pengguna tidak membuat kesalahan yang serius; contohnya, pemilihan menu lebih disukai daripada pengisian *form* dan tidak mengizinkan pengisian karakter huruf pada tempat pengisian karakter angka. Jika pengguna melakukan kesalahan, sistem harus bisa mendeteksi kesalahan dan memberikan instruksi yang sederhana, membangun, dan khusus untuk melakukan perbaikan. Sebagai contoh, pengguna tidak harus mengetik ulang seluruh perintah, tetapi hanya perlu memperbaiki bagian yang salah.

- f. Mengizinkan pembalikan aksi (*undo*) dengan mudah.

Jika memungkinkan, aksi harus bisa dibalik. Ciri ini mengurangi kegelisahan, karena pengguna tahu bahwa kesalahan dapat diperbaiki, sehingga mendorong penjelajahan pilihan yang tidak biasa dipakai.

- g. Mendukung *internal locus of control*.

Pengguna berinisiatif dalam melakukan aksi daripada menunggu respon dari sistem untuk beraksi.

- h. Keterbatasan beban ingatan jangka pendek.

Keterbatasan pemrosesan informasi manusia dalam ingatan jangka pendek mengharuskan tampilan yang sederhana, tampilan banyak hal digabungkan,

jumlah window-motion dikurangi, dan jumlah waktu latihan yang cukup diberikan untuk pengkodean, membantu ingatan, dan mengetahui urutan aksi.

2.3. Rekayasa Piranti Lunak

Rekayasa piranti lunak (*software engineering*) membantu untuk pembuatan piranti lunak yang berkualitas.

2.3.1. Pengertian Rekayasa Piranti Lunak

Menurut Pressman (1997, p22), rekayasa piranti lunak adalah perkembangan dari piranti keras dan rekayasa sistem, yang meliputi tiga kunci elemen, yaitu: metode atau cara, alat, dan prosedur – sehingga manajer mampu untuk mengendalikan proses pengembangan piranti lunak dan menyediakan pelaksana dasar untuk membangun piranti lunak berkualitas tinggi dalam cara yang produktif.

2.3.2. Elemen-elemen Rekayasa Piranti Lunak

Menurut Pressman (1992, p24), elemen-elemen rekayasa piranti lunak antara lain:

a. Metode rekayasa piranti lunak

Metode ini meliputi: perencanaan proyek dan perkiraan, analisis kebutuhan sistem dan piranti lunak, perancangan struktur data, arsitektur pemrograman algoritma prosedur, pengkodean, pengujian, dan pemeliharaan.

b. Alat rekayasa piranti lunak.

Alat rekayasa piranti lunak menyediakan dukungan otomatis atau semi-otomatis untuk metode rekayasa piranti lunak. Sistem yang mendukung perkembangan

piranti lunak adalah *computer-aided software engineering (CASE)*. CASE ini mengkombinasikan piranti lunak, piranti keras, dan basis data rekayasa piranti lunak (suatu struktur data yang berisi informasi penting tentang analisis, perancangan, pengkodean, dan pengujian) untuk menghasilkan lingkungan rekayasa piranti lunak yang analog dengan *computer-aided design* atau *computer-aided programming (CAD* atau *CAE)* untuk piranti keras.

c. Prosedur rekayasa piranti lunak

Prosedur rekayasa piranti lunak adalah gabungan dari metode dan alat. Prosedur mendefinisikan aturan kapan metode akan digunakan, *deliverability* (dokumen, laporan, formulir, dan sebagainya) yang dibutuhkan, kendali yang membantu memastikan kualitas dan mengkoordinasi perubahan, dan dengan adanya kejadian penting manajer piranti lunak dapat menilai kemajuan.

2.3.3. Paradigma Rekayasa Piranti Lunak

Paradigma (model) rekayasa piranti lunak dipilih berdasarkan sifat proyek dan aplikasi, metode dan alat yang digunakan, dan kendali dan *deliverability* yang dibutuhkan. Menurut Pressman (1992, p24-26) terdapat empat paradigma dalam rekayasa piranti lunak, yaitu:

a. Paradigma *Classic Life Cycle*

Paradigma rekayasa piranti lunak yang sering digunakan adalah paradigma Classic Life Cycle yang meliputi aktivitas-aktivitas sebagai berikut:

1) Analisis dan rekayasa sistem (*System engineering and analysis*)

Analisis dan rekayasa sistem meliputi pengumpulan kebutuhan pada puncak level perancangan dan analisis dengan jumlah yang sedikit.

2) Analisis kebutuhan piranti lunak (*System Requirements Analysis*)

Proses pengumpulan kebutuhan ditingkatkan dan difokuskan khususnya pada piranti lunak. Untuk memahami sifat program yang akan dibangun, seorang analis harus memahami informasi utama untuk piranti lunak, termasuk fungsi yang dibutuhkan, hasil, dan antarmuka. Kebutuhan untuk sistem dan piranti lunak didokumentasikan dan diperiksa bersama-sama dengan pengguna.

3) Perancangan (*Design*)

Perancangan piranti lunak adalah langkah-langkah yang difokuskan pada empat atribut program yaitu: struktur data, arsitektur piranti lunak, rincian prosedur, dan karakteristik interface. Perancangan ini merupakan proses untuk menerjemahkan kebutuhan menjadi representasi piranti lunak yang bisa diperkirakan kualitasnya sebelum dilakukan pengkodean.

4) Pengkodean (*Coding*)

Setelah dilakukan perancangan, maka langkah selanjutnya adalah dilakukan pengkodean. Pengkodean ini menerjemahkan perancangan menjadi bentuk yang bisa dibaca oleh mesin.

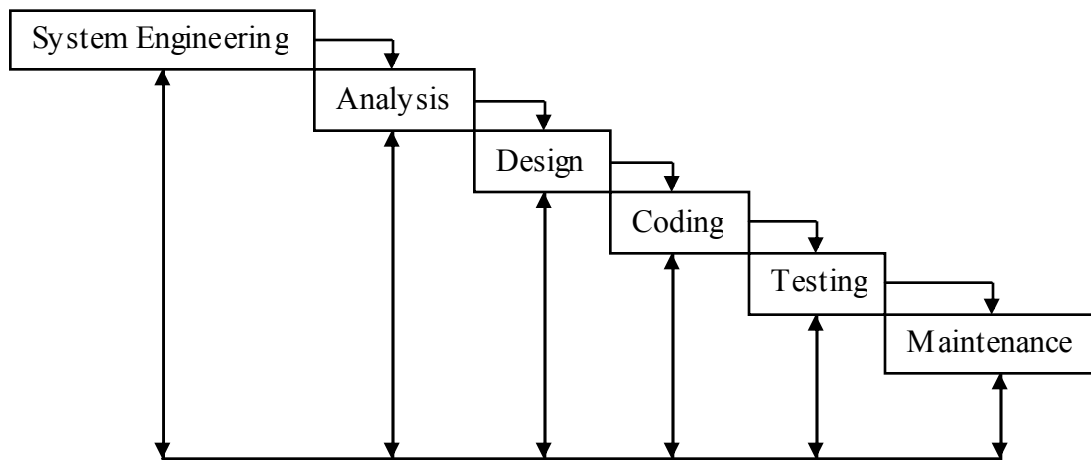
5) Pengujian (*Testing*)

Ketika pengkodean sudah selesai dibuat, maka program yang dibuat harus diuji. Dalam level pengujian ini bisa ditemukan kesalahan yang terjadi dan

dapat memastikan bahwa input yang didefinisikan akan menghasilkan hasil yang sebenarnya yang telah disetujui dengan hasil yang dibutuhkan.

6) Pemeliharaan (*Maintenance*)

Perubahan akan terjadi karena kesalahan yang dialami, karena piranti lunak harus diadaptasikan dengan perubahan yang terjadi dalam lingkungan eksternal (seperti contoh: kebutuhan berubah karena adanya sistem operasi yang baru), atau karena kebutuhan fungsional pengguna atau perbaikan peningkatan hasil.



Gambar 2.1. The Classic Life Cycle (Pressman 1992, p25)

b. Paradigma Prototyping

Prototyping adalah suatu proses di mana para pengembang mampu menciptakan suatu model piranti lunak yang harus dibangun. Model ada 3 bentuk, yaitu:

- 1) *Paper prototype* atau model yang berbasiskan *PC* yang menggambarkan interaksi mesin dengan manusia dalam bentuk yang bisa membuat pengguna mengerti bagaimana interaksi akan terjadi.

- 2) *Working prototype* yang mengimplementasikan beberapa *subset* fungsi yang dibutuhkan dari piranti lunak yang diinginkan.
- 3) Keberadaan program yang menunjukkan bagian atau semua fungsi yang diinginkan tetapi punya beberapa ciri yang akan ditingkatkan pada usaha pengembangan yang baru.

Prototyping dimulai dengan pengumpulan kebutuhan. Pengembang dan pengguna bertemu dan mendefinisikan semua fungsi untuk piranti lunak, mengidentifikasi apa yang dibutuhkan, dan menggambarkan daerah di mana penjelasan lebih jauh diperlukan. Setelah itu dilakukan perancangan yang cepat. Perancangan yang cepat difokuskan pada representasi dari aspek-aspek dari piranti lunak yang akan dijelaskan kepada pengguna. Setelah dilakukan perancangan dengan cepat, maka dibangun prototype. Pengguna mengevaluasi prototype dan menyaring kebutuhan untuk piranti lunak yang akan dikembangkan. Proses iterasi terjadi ketika prototype dipusatkan untuk memuaskan kebutuhan pengguna sementara pada saat yang sama pengembang lebih mengetahui apa yang dibutuhkan.

c. Paradigma *Spiral Model*

Model spiral untuk rekayasa piranti lunak yang dikembangkan meliputi ciri-ciri yang terbaik dari paradigma Classic Life Cycle dan Prototyping. Model spiral mendefinisikan 4 aktivitas umum, yaitu:

1) Perencanaan

Menemukan tujuan atau sasaran, pilihan dan batasan-batasan.

2) Resiko Analisis

Menganalisis pilihan dan mengidentifikasikan atau mengenali resiko.

3) Rekayasa

Mengembangkan produk “level selanjutnya”.

4) Evaluasi dari Pengguna

Penilaian hasil rekayasa.

d. Paradigma *Fourth – Generation Techniques*

Paradigma 4GT ini difokuskan pada kemampuan untuk menspesifikasikan piranti lunak dari suatu mesin pada tingkat yang dekat dengan *natural language* atau menggunakan notasi yang memberikan fungsi yang penting.

Paradigma 4GT dimulai dengan pengumpulan kebutuhan. Pengguna menggambarkan kebutuhan dan kemudian dikembangkan strategi perancangan untuk sistem. Penggunaan 4GT tanpa perancangan akan menyebabkan kesulitan-kesulitan (kualitas yang kurang, kemampuan pemeliharaan yang kurang, dan kurang diterima pengguna) seperti yang sudah disebutkan pada pengembangan piranti lunak menggunakan pendekatan konvensional.

Dengan menggunakan Bahasa Generasi Keempat (*4GL*), para pengembang mampu menggambarkan hasil yang diinginkan dalam cara di mana hasil secara otomatis diterjemahkan dalam kode. Jelasnya struktur data dengan informasi yang relevan harus ada dan bisa dibaca oleh 4GL.

Untuk mentransformasikan implementasi 4GT menjadi suatu produk, pengembang harus melakukan pengujian, mengembangkan dokumentasi yang

berarti, dan menunjukkan semua kegiatan “transisi” lainnya yang dibutuhkan dalam paradigma rekayasa piranti lunak lainnya.

2.4. State Transition Diagram

State Transition Diagram (STD) adalah bagian penting dalam perancangan sistem.

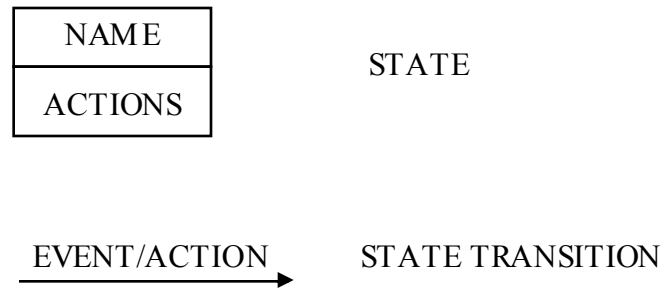
2.4.1. Pengertian State Transition Diagram

State Transition Diagram digunakan untuk menunjukkan keadaan tempat dari kelas yang diberikan, kejadian yang menyebabkan perubahan dari 1 state ke state lainnya, dan aksi yang menghasilkan perubahan pada state. Sebuah *state diagram* mewakili sebuah penampilan dari model yang dinamik dari bagian atau keseluruhan sistem. Tidak setiap bagian memiliki kejadian yang penting, jadi kita memenuhi State Transition Diagram hanya untuk bagian-bagian yang ada; dan menunjukkan kejadian dari sistem secara keseluruhan. Selama analisis, kita menggunakan State Transition Diagram untuk menunjukkan kejadian yang terus berubah dari sistem. Sedangkan saat disain, kita menggunakan State Transition Diagram untuk menangkap kejadian yang berubah dari setiap bagian atau menggabungkan setiap bagian. (Booch 1994, p199)

Menurut pendapat Yourdon (1989) yang dimaksud dengan keadaan adalah node perubahan yang dapat ditinjau. Sebagai contoh : keadaan untuk memantau, keadaan bahaya, keadaan pelepasan tekanan, dan sebagainya. Dengan mempelajari State Transition Diagram, perancangan piranti lunak dapat menentukan perilaku sistem yang lebih penting, dapat memastikan apakah terdapat lubang dalam perilaku yang telah ditentukan.

2.4.2. Simbol State Transition Diagram

Menurut Booch (1994, p200-201) yang sependapat dengan Harel, dua elemen penting dari State Transition Diagram adalah state dan state transition.



Gambar 2.2. Simbol State Transition Diagram (Booch 1994, p200-201)

State adalah sebuah objek yang mewakilkan hasil kumulatif dari setiap kejadian. Pada suatu waktu, state dapat meliputi semua bagian, nilai dari *attribute* dan hubungan dengan objek lain. Kita dapat menggeneralisasikan konsep dari sebuah objek state untuk menerapkan ke objek kelas, karena semua bagian pada kelas yang sama berada pada lokasi state yang sama, di mana meliputi semua hal yang pasti atau tidak dari sebuah kemungkinan state.

Setiap state memiliki nama yang biasanya khusus dan unik. Setiap lambang state memiliki nama yang biasanya khusus dan unik. Setiap lambang state yang memiliki nama sama pada diagram mengacu pada state yang sama.

State transition adalah sebuah kejadian dari beberapa kejadian yang menyebabkan state dari sistem berubah. Perubahan ini yang kita gambar menggunakan lambang dari Gambar 2.2.. Setiap state transition menghubungkan 2 state. Sebuah state dapat memiliki

sebuah state transition untuk dirinya sendiri, dan dapat memiliki beberapa state transition yang berbeda dari state yang sama, meskipun setiap perubahan harus unik, berarti tidak akan ada keadaan tertentu yang menyebabkan lebih dari satu state transition dari state yang sama.

2.5. Basis Data

Basis data (*database*) adalah bagian yang sangat penting dalam penyimpanan data.

2.5.1. Pengertian Basis Data

Menurut pendapat Connolly (2003, p140), sebuah basis data merupakan sebuah kumpulan dari data yang berhubungan secara *logical* yang dapat digunakan secara bersama, dan sebuah deskripsi dari data tersebut, dirancang untuk mendapatkan info yang diperlukan dari sebuah organisasi.

2.5.2. Elemen-elemen Basis Data

Menurut Kruglinsky (1986, p12), elemen-elemen yang terkandung dalam basis data antara lain:

a. Basis data teks

Adalah koleksi teks yang terorganisasi, biasanya berada pada pengatur komputer dan perlengkapan komunikasi yang dapat diakses secara *online*.

b. File data

File data biasanya tersedia pada *DBMS (Database Management System)*, tidak dalam jaringan dan produk hirarki. File data tampak dalam bentuk dua dimensi di

mana setiap garis adalah *record* dan setiap kolom adalah *field*. Tabel biasanya disimpan pada file *PC-DOS*. Sebuah kamus merinci semua field, dan satu atau lebih indeks tidak diperbolehkan masuk ke record manapun. Tabel, kamus, dan semuanya merupakan bagian dari file data.

c. File PC-DOS

Adalah sebuah koleksi data dalam sebuah disk, dikendalikan sebagai sebuah unit dalam sistem operasi PC-DOS.

d. Record

Adalah sekelompok field informasi yang saling berhubungan dan diperlakukan sebagai sebuah unit.

e. Field

Sebuah field mengidentifikasikan suatu lokasi dalam sebuah record di mana sebuah unit data disimpan. Field memiliki karakteristik tertentu, seperti panjang (jumlah digit atau karakter) dan tipe (numerik atau alfabet), dan ditampilkan dalam bentuk kolom-kolom.

f. *Occurance*

Adalah hal yang sebenarnya dari field atau record dalam basis data.

g. *Schema*

Adalah sebuah diagram skematik dari tampilan hubungan bermacam-macam tipe record dalam basis data.

h. Kamus Data

Adalah penjelasan secara penuh dari field dalam basis data. Untuk hirarki dan jaringan basis data, kamus data menunjukkan hubungan antara tipe record.

i. *Form*

Adalah sebuah jendela menuju ke file data.

j. *Key*

Sebuah key dapat unik atau tidak unik untuk mengidentifikasi sebuah record dan biasanya merupakan komponen dari indeks. Sebuah key dapat merupakan field tunggal atau sekelompok field.

k. Indeks

Adalah sebuah tabel penomoran record yang disusun berdasarkan key. Penomoran record ini, disebut pointer, disusun untuk membantu DBMS menemukan bagian dari record secara cepat dengan menggunakan key.

2.5.3. Normalisasi

Menurut Brathwaite (1990, p35-38), selama analisis data, atribut-atribut yang relevan disimpan dan didefinisikan untuk setiap tipe *entity*. Hal ini mungkin akan mengacu pada identifikasi tipe entity yang baru atau subdivisi dari entity yang sudah ada. Sekali model entity memiliki alasan yang kuat untuk dikatakan telah lengkap, pemeriksaan secara tegas harus dilakukan untuk mendeteksi *relationship* yang tidak diperlukan. Pemeriksaan ini meliputi suatu proses yang disebut normalisasi.

Dalam normalisasi dibutuhkan tiga aksi yang harus dilakukan pada atribut dalam sebuah entity, antara lain:

a. *First Normal Form*

Suatu relasi dikatakan bentuk normal pertama jika semua grup yang berulang dihapus.

b. *Second normal form*

Suatu relasi dikatakan bentuk normal kedua jika setiap atribut bukan key tergantung fungsional pada *primary key*.

c. *Third Normal Form*

Suatu relasi dikatakan bentuk normal ketiga jika setiap atribut bukan key tergantung langsung dan fungsional pada primary key.

2.5.4. E/R Diagram

Menurut Date (2000, p424) *entity or relationship diagram* (E/R Diagram) merupakan suatu teknik untuk mewakili struktur logika dari basis data dengan cara menggambarannya.

2.6. Sistem Informasi

Sistem informasi menjadi rangka pemikiran dasar untuk pengembangan aplikasi.

2.6.1. Pengertian Sistem

Menurut O'Brien (2003, p8), sistem adalah sekumpulan komponen yang saling berhubungan dan bekerja sama untuk mencapai tujuan dengan menerima input dan menghasilkan output dengan proses transformasi yang terorganisasi. Komponen-komponen sistem terdiri dari:

a. Input

Mengambil dan mengumpulkan elemen yang memasuki sistem untuk diproses.

b. Proses

Proses transformasi yang mengubah input menjadi output.

c. Output

Pengiriman elemen yang dihasilkan melalui proses transformasi.

2.6.2. Pengertian Informasi

Menurut O'Brien (2003, p13) informasi adalah data yang telah diubah agar menjadi bermakna dan berguna bagi pengguna.

2.6.3. Pengertian Sistem Informasi

Menurut O'Brien (2003, p7) sistem informasi merupakan kombinasi dari orang, piranti keras, piranti lunak, jaringan komunikasi dan sumber data yang terkumpul ditransformasikan, disebarakan dalam organisasi.

2.7. Ensiklopedia

Ensiklopedia adalah salah satu bentuk dokumentasi untuk menyimpan informasi.

2.7.1. Pengertian Ensiklopedia

Menurut Mulyono (1990, p232), ensiklopedia adalah buku atau serangkaian buku yang menghimpun keterangan atau uraian yang tergantung dari berbagai hal dalam bidang seni dan ilmu pengetahuan yang disusun menurut abjad atau menurut lingkungan ilmu.

Enkoklion paideia (bahasa Yunani) yang secara harfiah berarti pendidikan di dalam lingkungan yaitu unsur-unsur dari pengetahuan yang membentuk suatu pendidikan lengkap. (The, 1998, p4)

Ensiklopedia didefinisikan sebagai karya himpunan dari berbagai ilmu (ensiklopedia umum) dan ilmu-ilmu bidang tertentu (ensiklopedia khusus). (Anonymous, 1991, p309).

2.7.2. Sistem Ensiklopedia Multimedia

Sistem ensiklopedia multimedia adalah suatu sistem yang memanfaatkan media elektronika yang bertumpu pada kecanggihan teknologi komputer dengan memadukan video, audio, dan teks, yang berisikan uraian ringkas mengenai berbagai hal yang berhubungan dengan ilmu pengetahuan.

Sistem ensiklopedia multimedia selain memiliki tampilan yang menarik juga mudah digunakan serta mampu berinteraksi dengan pengguna. Sistem ensiklopedia multimedia harus dapat memenuhi kebutuhan informasi pengguna dan menggantikan sistem konvensional lainnya seperti buku-buku ensiklopedia yang ada.

Komponen-komponen pendukung sistem ensiklopedia multimedia ini adalah komputer-komputer pada umumnya seperti *processor*, media penyimpanan, monitor, *keyboard*, *mouse*, dan *speaker*. Untuk mendapatkan tampilan yang lebih menarik lagi maka dapat ditambahkan kamera video, mikrofon, dan peralatan-peralatan lain yang sesuai dengan kebutuhan.

2.7.3. Penyusunan Ensiklopedia

Untuk membuat atau merevisi ensiklopedia membutuhkan banyak waktu dan tenaga, selain tujuan dan sasaran ensiklopedia haruslah jelas. Berikut adalah beberapa hal yang harus diperhatikan dalam menyusun sebuah ensiklopedia:

a. Sasaran dan tujuan

Sebelum editor memutuskan subjek apa yang harus diikutsertakan, mereka harus mengetahui fungsi umum yang disediakan ensiklopedia. Pengembangan subjek materi juga tergantung pada apakah ensiklopedia mencakup satu atau banyak hal.

b. Pengguna

Memainkan peran penting dalam menyusun ensiklopedia. Beberapa ensiklopedia ditujukan khususnya bagi anak-anak sementara yang lain ditujukan bagi cendekiawan dan spesialis, juga ada yang diperuntukkan bagi keluarga. Ada ensiklopedia yang didesain sebagai referensi harian, alat bagi seluruh keluarga, dan sebagai guru, pustakawan, profesional, atau bisa juga untuk orang-orang bisnis.

c. Ruang lingkup ensiklopedia

Jumlah hal dan volume pada ensiklopedia tergantung pada isi yang dicakup, siapa yang menggunakan, dan harga. Walaupun ensiklopedia memiliki banyak hal dan volume, tidak ada ensiklopedia yang mencakup segala macam hal. Biasanya ensiklopedia memberikan daftar buku yang dapat dibaca untuk pengetahuan lebih lanjut.

d. Pengaturan isi

Kebanyakan ensiklopedia menggunakan salah satu dari metode penyusunan subjek, yaitu pengaturan secara alfabetis dan pengaturan berdasarkan topik.

e. Ilustrasi

Sangat penting dalam membantu pembaca dalam memahami hal yang sedang dibahas. Ilustrasi ini termasuk gambar, peta, dan diagram. Penggunaan warna menjadi sangat penting dalam mengilustrasikan ensiklopedia.

f. Penggunaan yang mudah

Adalah penting bagi semua org yang ingin mencari informasi dalam ensiklopedia. Kemudahan ini meliputi pencarian mengenai suatu topik yang dapat ditemukan pada indeks.

g. Format penulisan

Adalah tampilan secara fisik dari tiap hal, dapat membuat ensiklopedia menarik atau membosankan. Pendesain harus menentukan ukuran dari hal, ukuran dari tipe tulisan dan panjang dari tiap baris. (Anonymous, 1995, p4-5).

2.8. Mobil

Mobil tidak diragukan menjadi bagian penting dalam kehidupan.

2.8.1. Fakta-fakta Mobil

Menurut Mulyono (2000, p750), mobil adalah kendaraan darat yang digerakkan oleh tenaga mesin, beroda empat atau lebih (biasanya genap) dan menggunakan bahan bakar minyak untuk menghidupkan mesinnya.

Mobil mengubah dunia di abad ke-20, terutama di Amerika Serikat dan negara-negara industri lainnya. Dari pertumbuhan di wilayah pedesaan dan pinggiran kota sampai ke pengembangan sistem jalan dan jalan raya, yang disebut sebagai ‘kereta tanpa kuda’ telah selamanya mengubah pandangan modern. Pembuatan, penjualan, dan pelayanan dari mobil-mobil telah menjadi elemen utama dari ekonomi perindustrian. Tetapi sejalan dengan mobilitas yang lebih baik dan pembukaan lapangan kerja, mobil telah membawa kebisingan dan polusi udara, dan kecelakaan mobil mendapat peringkat yang cukup tinggi dalam hal penyebab kematian dan luka-luka di seputar dunia. Tetapi, 1900an telah menjadi abad dari otomotif, dan mobil-mobil tidak diragukan akan terus membentuk kebudayaan dan perekonomian dengan baik ke abad-abad mendatang.

Di tahun 2001, pabrik-pabrik pembuatan mobil di lebih dari 35 negara memproduksi sekitar 39.5 juta mobil untuk kelas *passenger cars*. Sekitar 7.3 juta dari mobil kelas ini diproduksi di Amerika Utara sepanjang tahun 2001.

Sebuah mobil dibuat di sekeliling sebuah mesin. Sistem-sistem yang bervariasi mendukung mesin tersebut dengan bahan bakar, mendinginkannya selama operasi, melumas bagian-bagian yang bergerak, dan membuang gas buangan yang dihasilkan. Mesin menghasilkan tenaga mekanik yang disalurkan ke roda mobil melalui *drivetrain*, yang salah satu bagiannya adalah transmisi. Sistem suspensi, yang mencakup per dan penyerap bantingan, melapisi kendaraan dan membantu melindungi mesin dari kerusakan oleh bantingan, beban-beban berat, dan tekanan-tekanan lain. Roda dan ban mendukung mobil di jalan dan berputar untuk mengerakkan mobil maju atau mundur. Sistem setir dan rem memberikan kontrol atas arah dan kecepatan. Sebuah sistem listrik memulai pergerakan dan pengoperasian mesin, memonitor, dan mengendalikan banyak aspek dari

operasi kendaraan, dan memberikan tenaga untuk komponen-komponen seperti lampu dan radio. Fitur-fitur seperti *bumper*, *air bag*, dan *seat belt* membantu melindungi pengendara bila sebuah kecelakaan terjadi. (Encarta, 2004)

2.8.2. Ensiklopedia Mobil

Suatu ensiklopedia mobil berarti suatu ensiklopedia yang menghimpun keterangan atau uraian mengenai mobil sebagai salah satu produk ilmu pengetahuan dan teknologi dalam bidang transportasi. Ensiklopedia ini tidak hanya bisa berupa buku saja tetapi bisa berupa program aplikasi untuk komputer.