

BAB II

LANDASAN TEORI

1.1 Definisi Aplikasi

Menurut Dhanta (2009: 32), aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu. Sedangkan menurut Anisyah (2000: 30), aplikasi adalah penerapan, penggunaan atau penambahan.

Dari pengertian diatas, dapat disimpulkan bahwa aplikasi merupakan *software* yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data.

1.2 Aplikasi Berbasis Internet

Menurut Dhanta (2009:136), Intranet adalah sumber daya informasi yang digunakan untuk kepentingan internal dari suatu instansi atau perusahaan dengan menggunakan jaringan komputer yang ada.

Intranet adalah jaringan komputer dalam sebuah perusahaan yang menggunakan teknologi internet, sehingga terbentuk lingkungan yang seperti internet tetapi bersifat privat bagi perusahaan bersangkutan (Kadir, 2003:533). Dalam hal ini pengaksesan informasi yang berkaitan dengan sistem basis data juga dilakukan melalui *web browser*, misalnya internet explorer.

Aplikasi berbasis internet merupakan suatu aplikasi pengolahan data menjadi sebuah informasi yang berkualitas dan dipergunakan untuk suatu alat bantu dalam pengambilan keputusan, yang menggunakan jaringan LAN (*Local*

Area Network). Kadangkala internet juga dihubungkan ke jaringan internet, dengan melengkapi *firewall* (Kadir, 2003:533).

Dengan cara seperti ini, akses dari seseorang dari luar perusahaan bisa ditolak, tetapi pemakai di dalam perusahaan tetap bisa mengakses internet.

1.3 Gravity Location Model

Menurut Pujawan (2008), “*Gravity Location Model* adalah suatu model *Supply Chain Management* yang digunakan untuk menentukan lokasi suatu fasilitas (misal gudang atau pabrik) yang menjadi penghubung antara sumber pasokan dan beberapa lokasi pasar”. Asumsi yang digunakan :

1. Ongkos transportasi naik sebanding dengan volume yang dipindahkan.
2. Sumber pasokan maupun pasar bisa ditentukan lokasinya pada suatu peta dengan koordinat x dan y yang jelas.

Tabel 2.1. Data Gudang Lokal

x_i	y_i	V_i	C_i
5	1	100	1,5
4	6	700	1,8
8	12	200	2,5
12	5	150	1,9
5	9	400	1,7
15	3	200	2,1

3. Pertama kali fasilitas diasumsikan berada di koordinat (0,0).

Notasi perhitungan yang digunakan :

x, y : Koordinat fasilitas yang dipertimbangkan (produsen).

x_n, y_n : Koordinat x dan y untuk lokasi sumber pasokan/pasar n

d_n : Jarak antara lokasi fasilitas dengan sumber pasokan/pasar n

C_n : Ongkos transportasi per unit beban per kilometer antara kandidat lokasi fasilitas dengan lokasi sumber pasokan/pasar n

V_n : Beban yang akan dipindahkan antara kandidat lokasi fasilitas dengan lokasi sumber pasokan/pasar n

Jarak antar lokasi dihitung sebagai jarak geometri dari dua lokasi, formula:

$$d_n = \sqrt{(x - x_n)^2 + (y - y_n)^2}$$

Tujuan model : meminimumkan total ongkos pengiriman, yang diformulasikan

sebagai : $TC = \sum_n C_n V_n d_n$

Tabel 2.2. Iterasi 1 Dengan Titik Awal(0,0).

x_n	y_n	j_n	V_n	C_n	$V_n C_n x_n / j_n$	$V_n C_n y_n / j_n$	$V_n C_n / j_n$
5	1	5,1	100	1,5	147,1	29,4	29,4
4	6	7,2	700	1,8	698,9	1048,4	174,7
8	12	14,4	200	2,5	277,4	416	34,7
12	5	13	150	1,9	263,1	109,6	21,9
5	9	10,3	400	1,7	330,2	594,4	66
15	3	15,3	200	2,1	411,8	82,4	27,5
Total					2128,5	2280,2	354,2

Langkah-langkah :

1. Hitung jarak (d_n) untuk semua n
2. Tentukan koordinat lokasi dengan rumus

$$a = \frac{\sum_n \frac{C_n V_n x_n}{d_n}}{\sum_n \frac{C_n V_n}{d_n}} \quad b = \frac{\sum_n \frac{C_n V_n y_n}{d_n}}{\sum_n \frac{C_n V_n}{d_n}}$$

$$X_{(0)n} = 218,5/354,2 = 6,0$$

$$Y_{(0)n} = 2280,2/354,2 = 6,4$$

1. Apabila dua iterasi berurutan menghasilkan koordinat yang hampir sama, stop dan pilih koordinat sbg lokasi fasilitas. Jika tidak, ulangi iterasi dari langkah 1.

1.4 Analytical Hierarchy Process

Analytical Hierarchy Process (AHP) yang dikembangkan oleh Thomas Saaty merupakan suatu metode dalam pemilihan alternatif-alternatif dengan melakukan penilaian komparatif berpasangan sederhana yang digunakan untuk mengembangkan prioritas-prioritas secara keseluruhan berdasarkan *ranking*.

AHP adalah prosedur yang berbasis matematis yang sangat baik dan sesuai untuk evaluasi atribut-atribut kualitatif. Atribut-atribut tersebut secara matematik dikuantitatif dalam satu set perbandingan berpasangan, yang kemudian digunakan untuk mengembangkan prioritas-prioritas secara keseluruhan untuk penyusunan alternatif-alternatif pada urutan ranking / prioritas.

Kelebihan AHP dibandingkan dengan metode yang lainnya karena adanya struktur yang berhirarki, sebagai konsekuensi dari kriteria yang dipilih, sampai kepada sub-sub kriteria yang paling mendetail. Memperhitungkan validitas sampai dengan batas toleransi inkonsistensi berbagai kriteria dan alternatif yang dipilih oleh para pengambil keputusan (Saaty, 1990).

Karena menggunakan input persepsi manusia, model ini dapat mengolah data yang bersifat kualitatif maupun kuantitatif. Jadi kompleksitas permasalahan yang ada disekitar kita dapat didekati dengan baik oleh model AHP ini. Selain itu AHP mempunyai kemampuan untuk memecahkan masalah yang multi-objektif dan multi- kriteria yang didasarkan pada perbandingan preferensi dari setiap elemen dalam hierarki. Jadi model ini merupakan suatu model pengambilan keputusan yang komprehensif.

Ada beberapa prinsip yang harus dipahami dalam menyelesaikan persoalan dengan AHP, diantaranya adalah : *decomposition*, *comparative*

judgement, synthesis of priority dan logical consistency (Sri Mulyono, 2007 : 220).

2.4.1 Decomposition

Setelah persoalan didefinisikan, maka perlu dilakukan *decomposition* yaitu memecah persoalan yang utuh menjadi unsur-unsurnya. Jika ingin mendapatkan hasil yang akurat, pemecahan juga dilakukan terhadap unsur-unsurnya sampai tidak mungkin dilakukan pemecahan lebih lanjut, sehingga didapatkan beberapa tingkatan dari persoalan tadi. Karena alasan ini, maka proses analisis ini dinamakan hirarki (*hierarchy*). Ada dua jenis hirarki yaitu lengkap dan tak lengkap. Dalam hirarki lengkap, semua elemen pada suatu tingkat memiliki semua elemen yang ada pada tingkat berikutnya. Jika tidak demikian, dinamakan hirarki tak lengkap:

2.4.2 Comparative Judgement

Prinsip ini berarti membuat penilaian tentang kepentingan relatif dua elemen pada suatu tingkat tertentu dalam kaitannya dengan tingkat di atasnya. Penilaian ini merupakan inti dari AHP, karena ia akan berpengaruh terhadap prioritas elemen-elemen. Hasil dari penilaian ini akan tampak lebih baik bila disajikan dalam bentuk matriks yang dinamakan *matriks pairwise comparison*. Pertanyaan yang biasa diajukan dalam penyusunan skala kepentingan adalah :

- a. Elemen mana yang lebih (penting/disukai/mungkin) ?, dan
- b. Berapa kali lebih (penting/disukai/mungkin) ?

Agar diperoleh skala yang bermanfaat ketika membandingkan dua elemen, seseorang yang akan memberikan jawaban perlu pengertian menyeluruh

tentang elemen-elemen yang dibandingkan dan relevansinya terhadap kriteria atau tujuan yang dipelajari. Dalam penyusunan skala kepentingan ini, digunakan acuan seperti pada tabel berikut:

Tabel 2.3. Skala Prioritas Dalam AHP

Nilai	Tingkat Kepentingan
1	Sama pentingnya (<i>Equal Importance</i>)
2	Sama hingga Sedikit Lebih penting
3	Sedikit Lebih penting (<i>Slightly more Importance</i>)
4	Sedikit Lebih hingga Jelas lebih penting
5	Jelas lebih penting (<i>Materially more Importance</i>)
6	Jelas hingga Sangat jelas lebih penting
7	Sangat jelas lebih penting (<i>Significantly more</i>)
8	Sangat jelas hingga Mutlak lebih penting
9	Mutlak lebih penting (<i>Absolutely more Importance</i>)

Dalam penilaian kepentingan relatif dua elemen berlaku *aksioma reciprocal* artinya jika elemen i dinilai 3 kali lebih penting daripada j, maka elemen j harus sama dengan $1/3$ kali pentingnya dibanding elemen i. Disamping itu perbandingan dua elemen yang sama akan menghasilkan angka 1, artinya sama pentingnya.

2.4.3 Synthesis of Priority

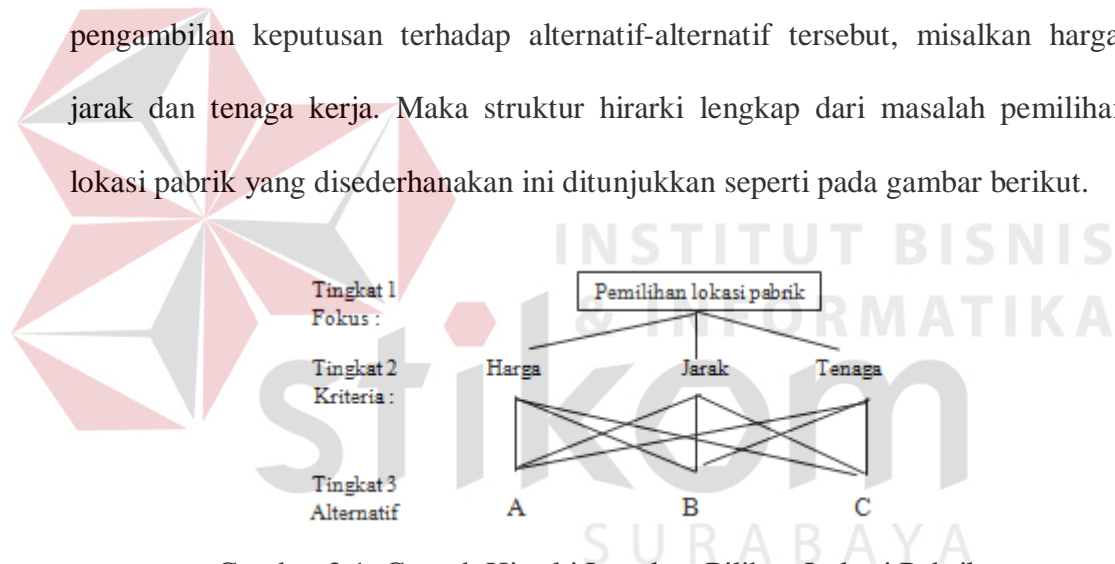
Dari setiap *pairwise comparison* kemudian dicari *eigen vectornya* untuk mendapatkan local priority. Karena matriks *pairwise comparison* terdapat pada setiap tingkat, maka untuk mendapatkan global priority harus dilakukan sintesa diantara *local priority*. Prosedur melakukan sintesis berbeda menurut bentuk hirarki. Pengurutan elemen-elemen menurut kepentingan relatif melalui prosedur sintesis dinamakan *priority setting*.

2.4.4 Logical Consistency

Konsistensi memiliki dua makna. Pertama adalah bahwa objek-objek yang serupa dapat dikelompokkan sesuai dengan keseragaman dan relevansi. Kedua adalah menyangkut tingkat hubungan antara objek-objek yang didasarkan pada kriteria tertentu.

2.4.5 Tahap-tahap pemecahan masalah dengan AHP

Misalkan kita akan memilih lokasi pabrik baru dengan tiga alternatif pilihan A, B dan C maka terlebih dahulu kita harus menetapkan kriteria pengambilan keputusan terhadap alternatif-alternatif tersebut, misalkan harga, jarak dan tenaga kerja. Maka struktur hirarki lengkap dari masalah pemilihan lokasi pabrik yang disederhanakan ini ditunjukkan seperti pada gambar berikut.



Gambar 2.1. Contoh Hirarki Lengkap Pilihan Lokasi Pabrik

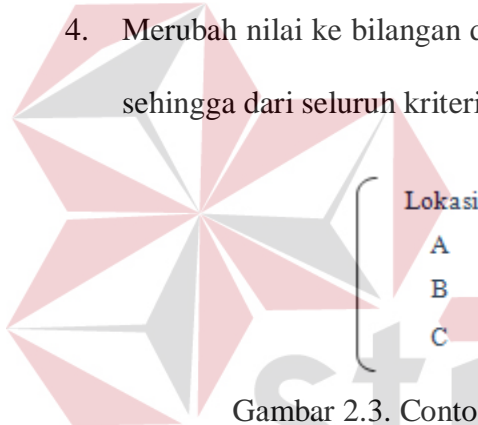
Langkah-langkah penyelesaian masalah selanjutnya adalah :

1. Membuat matrik hubungan perbandingan berpasangan antara tiap alternatif untuk setiap kriteria keputusan. Perbandingan dilakukan berdasarkan pilihan dari pembuat keputusan dengan menilai tingkat kepentingan / *preference level* suatu alternatif dibandingkan alternatif lainnya.

$$\left(\begin{array}{c|ccc} & \text{Harga} & & \\ \hline & A & B & C \\ \hline A & 1 & 3 & 2 \\ B & 1/3 & 1 & 1/5 \\ C & 1/2 & 5 & 1 \end{array} \right) \left(\begin{array}{c|ccc} & \text{Jarak} & & \\ \hline & A & B & C \\ \hline A & 1 & 6 & 1/3 \\ B & 1/6 & 1 & 1/9 \\ C & 3 & 9 & 1 \end{array} \right) \left(\begin{array}{c|ccc} & \text{Tenaga Kerja} & & \\ \hline & A & B & C \\ \hline A & 1 & 1/3 & 1 \\ B & 3 & 1 & 7 \\ C & 1 & 1/7 & 1 \end{array} \right)$$

Gambar 2.2 Contoh Matrik Alternatif Vs Preferensi Untuk Tiap Kriteria

2. Untuk setiap matriks kriteria, dilakukan penjumlahan nilai tiap kolom.
3. Membagi setiap nilai alternatif berpasangan dengan hasil penjumlahan pada kolom terkait, hasil pembagian kemudian dijumlahkan searah kolom, hasilnya seharusnya sama dengan 1 untuk menunjukkan konsistensinya.
4. Merubah nilai ke bilangan desimal dan mencari nilai rata-rata pada tiap baris, sehingga dari seluruh kriteria akan didapat matriks baru sebagai berikut.



$$\left(\begin{array}{c|cccc} & \text{Lokasi} & \text{Harga} & \text{Jarak} & \text{Tenaga Kerja} \\ \hline A & & .5012 & .2819 & .1790 \\ B & & .1185 & .0598 & .6850 \\ C & & .3803 & .6583 & .1360 \end{array} \right)$$

Gambar 2.3. Contoh Matriks Nilai Alternatif Vs Kriteria

5. Membuat matriks nilai untuk kriteria, misalnya,

$$\left(\begin{array}{c|cccc} & \text{Kriteria} & \text{Harga} & \text{Jarak} & \text{Tenaga Kerja} \\ \hline \text{Harga} & & 1 & 1/5 & 3 \\ \text{Jarak} & & 5 & 1 & 9 \\ \text{Tenaga Kerja} & & 1/3 & 1/9 & 1 \end{array} \right)$$

Gambar 2.4. Contoh Matriks Nilai Kriteria

6. Mengulangi langkah 2 sampai dengan 4 untuk matriks baru ini. Nilai akhir yang didapat dari matriks baru ini merupakan *eigen vector* (vektor pengali) untuk matriks pada langkah 4.

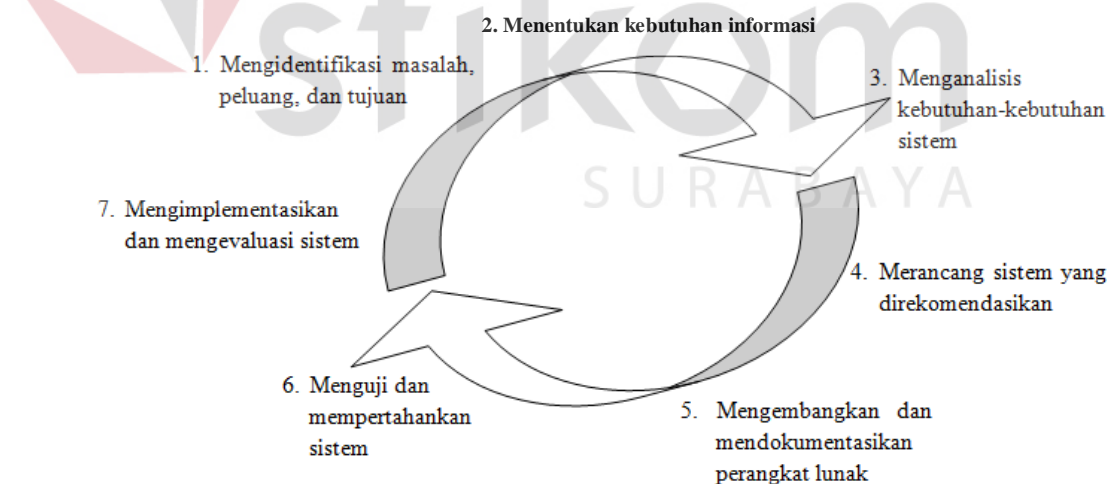
$$\begin{pmatrix}
 \text{Lokasi} & \text{Harga} & \text{Jarak} & \text{Tenaga Kerja} \\
 \text{A} & .5012 & .2819 & .1790 \\
 \text{B} & .1185 & .0598 & .6850 \\
 \text{C} & .3803 & .6583 & .1360
 \end{pmatrix}
 \times
 \begin{pmatrix}
 & \text{Kriteria} \\
 \text{Harga} & .1993 \\
 \text{Jarak} & .6535 \\
 \text{Tenaga Kerja} & .0860
 \end{pmatrix}$$

Gambar 2.5 Perkalian Matriks Akhir

7. Mengalikan kedua matriks pada Gambar 2.5 diatas. Alternatif dengan nilai terbesar merupakan alternatif yang harus dipilih.

1.5 Siklus Hidup Pengembangan Sistem (SHPS)

Siklus Hidup Pengembangan Sistem (SHPS) atau dalam bahasa asing disebut *System Development Life Cycle* (SDLC) adalah sebuah pendekatan, tentunya melalui beberapa tahap untuk menganalisis dan merancang sistem yang telah dikembangkan dengan baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik (Kendall & Kendall, 2003).



Gambar 2.6. Siklus Hidup Pengembangan Sistem (Kendall & Kendall, 2003:11)

SHPS terbagi menjadi tujuh tahap seperti yang ditunjukkan pada gambar 9 di atas. Menurut Kendall & Kendall (2003), masing-masing tahap ditampilkan secara terpisah, namun tidak pernah tercapai sebagai satu langkah terpisah. Melainkan,

beberapa aktivitas muncul secara simultan, dan aktivitas tersebut dilakukan berulang-ulang. Pernyataan tersebut berkesimpulan bahwa dalam gambar SHPS tersebut tahap satu dengan lainnya secara visual terlihat terpisah, akan tetapi pada kenyataannya, proses yang dilakukan oleh tahap tersebut dilakukan secara bertahap dan ada keterkaitan antara tahap satu dengan lainnya, tentunya dilakukan secara bertahap dan berkesinambungan. Berikut penjelasan lebih lanjut mengenai tujuh tahap yang terdapat pada gambar 9 di atas:

a. Mengidentifikasi Masalah, Peluang, dan Tujuan

Tahap ini merupakan tahap yang berpengaruh pada keberhasilan proyek, karena jika ada kekeliruan menentukan masalah, peluang, dan tujuan maka proyek tersebut akan sia-sia jika dikerjakan. Pada tahap identifikasi masalah terdapat beberapa langkah, yaitu diantaranya: (1) Melihat apa yang terjadi didalam bisnis. (2) Menentukan masalah dengan tepat.

Setelah masalah didapat maka langkah selanjutnya menentukan peluang yang ada pada bisnis tersebut. peluang disini dimaksudkan bahwa penganalisis sistem yakin bahwa dengan akan ada peningkatan jika ada sistem informasi yang terkomputerisasi. Jika sudah menemukan masalah dan peluang, langkah selanjutnya yaitu menentukan tujuan. Menentukan tujuan juga mempunyai beberapa langkah diantaranya: (1) Menemukan apa yang sedang terjadi dalam bisnis. (2) Menentukan aspek dalam aplikasi-aplikasi sistem informasi. (3) Menyebutkan problem atau peluang-peluang tertentu.

Ada beberapa aktivitas yang dilakukan pada tahap ini, diantaranya yaitu :

1. Wawancara terhadap manajemen pengguna.
2. Menyimpulkan pengetahuan yang diperoleh.

3. Mengestimasi cakupan proyek.

4. Mendefinisikan hasil-hasilnya.

Output dari tahap ini laporan feasible yang berisikan definisi problem dan ringkasan tujuan.

b. Menentukan Kebutuhan Informasi

Pada tahap ini penganalisis menentukan kebutuhan informasi untuk pengguna yang terlibat. Adapun langkah-langkah yang dilakukan dalam menentukan kebutuhan informasi yaitu: (1) Menentukan sampel dan memeriksa data mentah. (2) Wawancara. (3) Mengamati perilaku pembuat keputusan dan lingkungan kantor. (4) *Prototyping*. Tahap ini mempunyai tujuan untuk menampilkan informasi yang dibutuhkan dalam bisnis terkait serta membentuk kerangka pendekatan untuk memikirkan ulang bisnis dengan cara lebih kreatif. Penganalisis akan bisa memahami fungsi-fungsi bisnis dan melengkapi informasi tentang masyarakat, tujuan, data, dan prosedur yang terlibat.

c. Menganalisis Kebutuhan Sistem

Menganalisis kebutuhan-kebutuhan sistem dapat menggunakan sebuah perangkat untuk menentukan kebutuhan. Perangkat tersebut dapat berupa diagram alir data dan kamus data. Maksud dari perangkat tersebut yaitu untuk menggambarkan dan menyusun *input*, proses, dan *output*.

d. Merancang Sistem yang Direkomendasikan

Pada tahap ini penganalisis merancang sistem yang direkomendasikan setelah mengumpulkan data yang didapat. Langkah-langkahnya diantaranya: (1) Merancang data *entry*. Pada tahap ini penganalisis mendata seluruh input yang

akan dimasukkan dalam *Graphical User Interface* (GUIs) agar informasi yang didapatkan adalah informasi yang akurat. (2) Merancang file-file atau basis data. Tahap ini berfungsi sebagai penyimpanan data agar data terorganisir serta dapat melakukan pengelolaan keluaran yang bermanfaat. (3) Merancang prosedur-prosedur *back-up* dan kontrol. Fungsinya agar data dan informasi yang tersimpan dapat terselamatkan jika terjadi sesuatu bencana atau hal-hal yang tidak diinginkan. (4) Membuat paket spesifikasi program bagi pemrogram. Paket tersebut bisa digambarkan dengan flowchart sistem, diagram alir data, dan lain sebagainya.

e. Mengembangkan dan Mendokumentasikan Perangkat Lunak

Penganalisis perlu menggunakan salah satu teknik terstruktur dalam mengembangkan perangkat lunak. Teknik tersebut yaitu rencana terstruktur, *Nassi-Shneiderman charts*, dan *pseudocode*. Pendokumentasian dilakukan untuk menjelaskan pengembangan dan kode program serta bagian-bagian kompleks dari program.

f. Menguji dan Mempertahankan Sistem

Tahap ini merupakan tahap yang dilakukan secara berkesinambungan ketika program sudah dibuat dan diuji yaitu diperthankan dengan cara memperbaharui program. Pengujian juga diperlukan untuk menemukan adanya kendala maupun masalah yang terjadi ketika adanya pengujian.

g. Mengimplementasikan dan Mengevaluasi Sistem

Penganalisis bekerjasama dengan pengguna dalam melakukan implementasi sistem. Keterlibatan tersebut yakni dalam hal pelatihan dalam mengendalikan sistem serta perencanaan konversi sistem lama ke sistem yang baru. Setelah

melakukan implementasi maka dilakukan adanya evaluasi yang bertujuan untuk mengetahui pemenuhan kriteria bahwa pengguna benar-benar menggunakan sistem.

1.6 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama *Bubble chart*, *Bubble diagram*, model proses, diagram alur kerja, atau model fungsi.

DFD ini adalah salah satu alat pembuatan model yang sering digunakan, khususnya bila fungsi-fungsi sistem merupakan bagian yang lebih penting dan kompleks dari pada data yang dimanipulasi oleh sistem. Dengan kata lain, DFD adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem.

DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisis maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

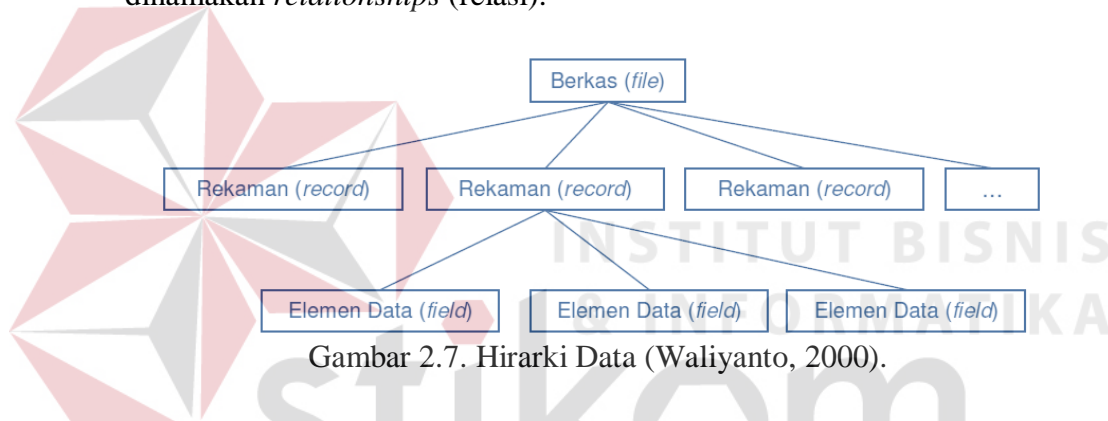
1.7 Entity Relationship Diagram (ERD)

Menurut Oetomo (2002:130), ERD berfungsi untuk menggambarkan relasi dari dua *file* atau dua tabel yang dapat digolongkan dalam tiga macam bentuk relasi, yaitu satu-satu, satu-banyak, dan banyak-banyak. ERD

menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data.

Pada dasarnya ada tiga macam simbol yang digunakan yaitu :

- a. *Entity* : adalah suatu obyek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat.
- b. *Atribut* : Atribut berfungsi untuk mendeskripsikan karakter entiti.
- c. *Hubungan* : Entiti dapat berhubungan satu sama lain. Hubungan ini dinamakan *relationships* (relasi).



Gambar 2.7. Hirarki Data (Waliyanto, 2000).

1.8 Testing

Menurut Hetzel (1983), *Testing* adalah tiap aktivitas yang digunakan untuk dapat melakukan evaluasi suatu atribut atau kemampuan dari program atau sistem dan menentukan apakah telah memenuhi kebutuhan atau hasil yang diharapkan.

Testing software adalah proses mengoperasikan *software* dalam suatu kondisi yang di kendalikan, untuk (1) verifikasi apakah telah berlaku sebagaimana telah ditetapkan (menurut spesifikasi), (2) mendeteksi *error*, dan (3) validasi apakah spesifikasi yang telah ditetapkan sudah memenuhi keinginan atau kebutuhan dari pengguna yang sebenarnya.

- a. Verifikasi adalah pengecekan atau pengetesan entitas-entitas, termasuk *software*, untuk pemenuhan dan konsistensi dengan melakukan evaluasi hasil terhadap kebutuhan yang telah ditetapkan. (*Are we building the system right?*)
- b. Validasi melihat kebenaran sistem, apakah proses yang telah ditulis dalam spesifikasi adalah apa yang sebenarnya diinginkan atau dibutuhkan oleh pengguna. (*Are we building the right system?*)
- c. Deteksi *error*: Testing seharusnya berorientasi untuk membuat kesalahan secara intensif, untuk menentukan apakah suatu hal tersebut terjadi bilamana tidak seharusnya terjadi atau suatu hal tersebut tidak terjadi dimana seharusnya mereka ada.

1.8.1 Functional Test

Dasar utama pemikiran teknik *test case* adalah melakukan analisis terhadap fungsi-fungsi yang terdapat pada suatu sistem, apakah fungsi-fungsi tersebut mempunyai kinerja sebagaimana yang diharapkan atau dispesifikasikan.

Teknik ini membutuhkan jawaban atas pertanyaan sebagai berikut:

- a. Fungsi utama apa saja yang harus ada pada sistem?
- b. Berdasarkan fungsi-fungsi yang ada, keluaran apa saja yang harus dihasilkan untuk membuktikan bahwa fungsi tersebut telah dipenuhi?
- c. Apa saja masukan dan inialisasi yang dibutuhkan sistem untuk menghasilkan keluaran pada tiap fungsi yang bersangkutan?

Karena itu, pendekatan pertama adalah mendapatkan informasi spesifikasi dari fungsi yang diharapkan dapat disediakan oleh sistem. Informasi ini umumnya terdapat pada dokumentasi spesifikasi fungsional sistem.

Bagaimana bila tidak ada dokumentasi spesifikasi fungsional sistem? Pengguna harus membuat spesifikasi fungsional. Proses pembuatan dapat dimulai dari struktur menu program atau buku panduan untuk pengguna (misal *Help File* atau *User Manual*).

1.8.2 Desain *Functional Test*

Menganalisa tiap fungsi secara terpisah untuk mendefinisikan sekumpulan kriteria tes analisis tiap pemrosesan fungsi masukan dan keluaran untuk menentukan apa yang dibutuhkan tes. Fungsi yang perlu dianalisa adalah

sebagai berikut:

- a. Kriteria fungsi
- b. Keluaran-keluaran fungsi
- c. Masukan-masukan fungsi
- d. Kondisi-kondisi fungsi
- e. Status-status fungsi

