

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK), adalah suatu sistem informasi berbasis komputer yang menghasilkan berbagai alternatif keputusan untuk membantu manajemen dalam menangani berbagai permasalahan yang terstruktur ataupun tidak terstruktur dengan menggunakan data dan model . Tujuan adanya SPK, untuk mendukung pengambil keputusan memilih alternatif hasil pengolahan informasi dengan model-model pengambil keputusan serta untuk menyelesaikan masalah yang bersifat semi terstruktur dan tidak terstruktur.

SPK dirancang untuk membantu pengambil keputusan dalam memecahkan masalah. SPK dirancang sedemikian rupa sehingga dapat digunakan atau dioperasikan dengan mudah oleh orang yang tidak memiliki dasar kemampuan pengoperasian komputer yang tinggi dan bersifat alternatif, serta SPK dirancang dengan menekankan pada aspek kemampuan adaptasi yang tinggi.

2.1.1 Manfaat Utama Sistem Pendukung Keputusan

- a. Mampu untuk mendukung permasalahan yang kompleks,
- b. Memberikan jawaban yang cepat untuk situasi yang tidak diharapkan dari hasil perubahan kondisi,
- c. Mampu untuk mencoba beberapa strategi yang berada di bawah konfigurasi yang berbeda, dengan cepat dan obyektif,

- d. Wawasan baru dan pengetahuan. Pemakai dapat terbuka dengan wawasan baru melalui komposisi model dan kepekaan yang luas analisis “*what-if*”,
- e. Memudahkan komunikasi. Kumpulan data dan pelaksanaan pembuatan model dijalankan dengan partisipasi aktif pemakai, sehingga sangat membantu diantara manajer,
- f. Memperbaiki kendali manajemen dan memperbaiki performansi organisasi,
- g. Keputusan bersifat obyektif. Keputusan yang dihasilkan SPK lebih konsisten dan obyektif daripada keputusan yang dibuat secara intuisi,
- h. Memperbaiki keefektifan manajerial. Memperbolehkan manajer untuk melakukan tugas dalam waktu yang singkat,
- i. Merperbaiki produktifitas analisis.

Tiga tingkatan teknologi yang digunakan dalam pengembangan SPK terdiri dari :

1. SPK Spesifik (*Spesific DSS*)

Paket yang terdiri dari pernakat keras dan lunak digunakan oleh sekelompok pengambil keputusan tertentu untuk menangani permasalahan khusus. SPK spesifik ini dikembangkan dari suatu *tools* SPK dan atau *Generator* SPK. Tujuan dari DSS yaitu ;

- a. Membantu manajer membuat keputusan untuk memecahkan *masalah semi terstruktur*.
- b. Mendukung penilaian manajer bukan mencoba menggantikannya.

c. Meningkatkan *efektifitas* pengambilan keputusan manajer daripada efisiensinya.

2. Peralatan SPK (*DSS Tools*)

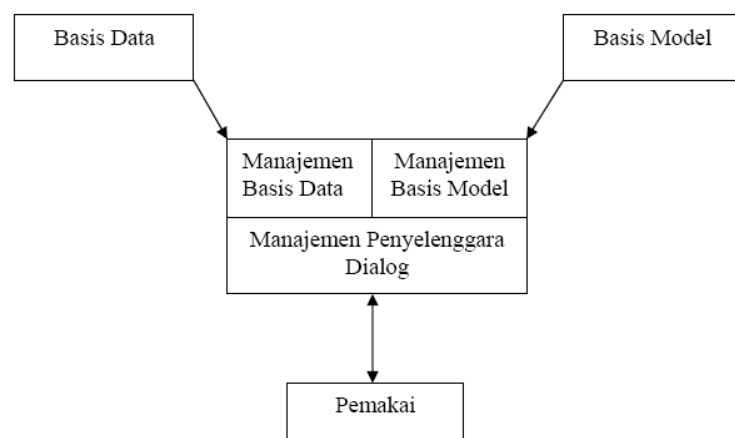
Merupakan elemen-elemen perangkat keras atau lunak yang dapat dipergunakan untuk mengembangkan SPK spesifik maupun pembangkit SPK. Meskipun peralatan ini mampu membuat SPK spesifik secara langsung, namun mengembangkan SPK spesifik dengan pembangkit SPK jauh lebih mudah dan efisien.

3. Pembangkit SPK (*DSS Generator*)

Merupakan “paket” dari kumpulan perangkat keras atau lunak yang menyediakan sekumpulan kemampuan untuk membuat SPK spesifik dengan cepat dan mudah.

2.1.2 Komponen-Komponen Sistem Pendukung Keputusan

SPK memiliki tiga sub sistem utama yang menentukan kapabilitas teknis SPK seperti pada gambar dibawah ini:



Gambar 2.1 Komponen Sistem Pendukung Keputusan

Komponen SPK dibagi atas tiga bagian yaitu :

i. Sub Sistem Basis Data

Merupakan komponen SPK penyedia data bagi sistem. Data dimaksud disimpan dalam suatu pangkalan data (*database*) yang diorganisasi yang disebut dengan system manajemen pangkalan data (Data Base Management System atau DBMS).

ii. Sub Sistem Basis Dialog

Model adalah suatu peniruan dari alam nyata. Kendala yang sering kali dihadapi dalam merancang suatu model adalah bahwa model yang disusun ternyata tidak mampu mencerminkan seluruh variable alam nyata. Sehingga keputusan yang diambil didasarkan pada model tersebut menjadi tidak akurat dan tidak sesuai dengan kebutuhan. Oleh karena itu, dalam menyimpan berbagai model pada sistem pangkalan model harus tetap dijaga fleksibilitasnya. Artinya harus ada fasilitas yang mampu membantu pengguna untuk memodifikasi atau menyempurnakan model seiring dengan perkembangan pengetahuan.

iii. Sub Sistem Penyelenggara Dialog (Basis Dialog)

Merupakan alat komunikasi antara pemakai dengan sistem atau dikenal dengan fasilitas antar muka. Bagian ini harus mampu memenuhi keinginan pemakai, bersifat komunikatif.

2.1.3 Tahap Pengambilan Keputusan

Tahap pengambilan keputusan, yaitu :

1. Penelusuran (*Intellegence*)

Merupakan tahap pendefinisian informasi yang dibutuhkan yang berkaitan dengan persoalan yang dihadapi serta keputusan yang akan diambil. Langkah ini sangat menentukan ketepatan keputusan yang akan diambil, karena sebelum suatu tindakan diambil, tentunya persoalan yang dihadapi harus dirumuskan terlebih dahulu secara jelas.

2. Perancangan (*Design*)

Merupakan tahap analisis dalam kaitan mencari atau merumuskan alternatif-alternatif pemecah masalah. Setelah permasalahan dirumuskan dengan baik, maka tahap berikutnya adalah merancang atau membangun model pemecahan masalahnya dan menyusun berbagai alternatif pemecah masalah.

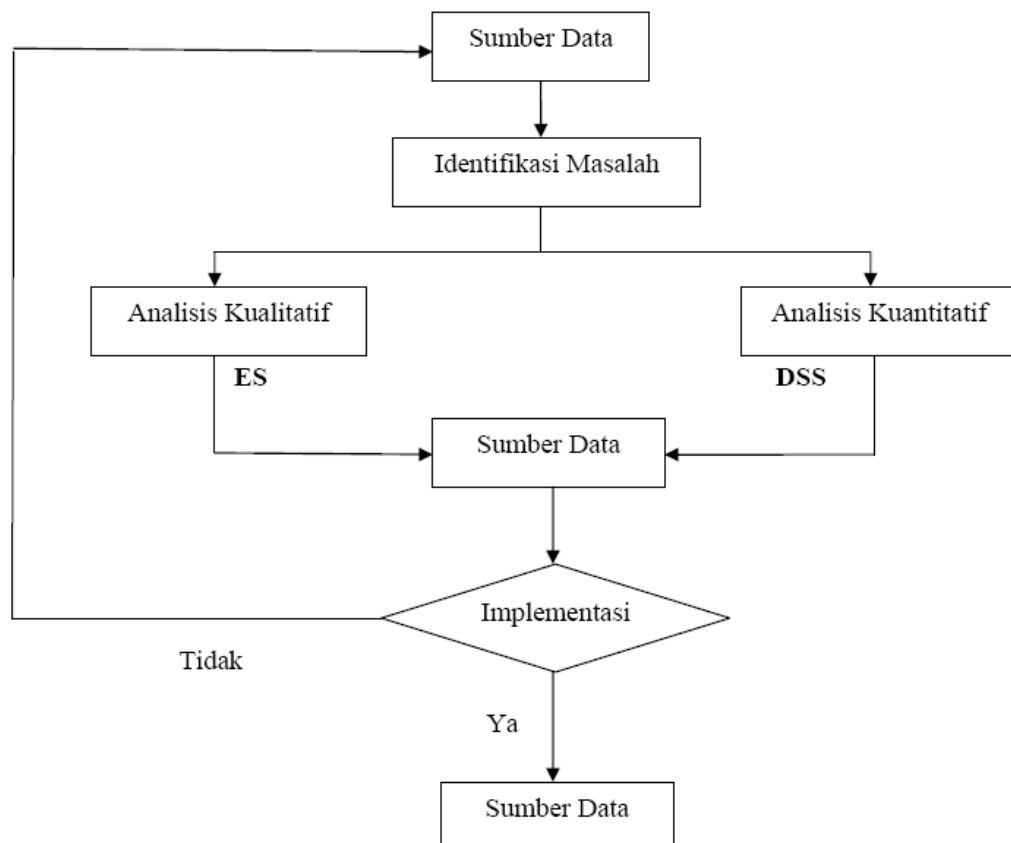
3. Pemilihan (*Choice*)

Dengan mengacu pada rumusan tujuan serta hasil yang diharapkan selanjutnya manajemen memilih alternatif solusi yang diperkirakan paling sesuai. Pemilihan alternatif ini akan mudah dilakukan kalau hasil yang diinginkan terukur atau memiliki nilai kualitas tertentu.

4. Implementasi (*Implementation*)

Merupakan tahap pelaksana dari keputusan yang telah diambil. Pada tahap ini perlu disusun serangkaian tindakan yang terencana, sehingga hasil keputusan dapat dipantau atau diselesaikan apabila diperlukan perbaikan-

perbaikan. Dalam kejadiannya keputusan diterapkan suatu solusi diusulkan, satu Decision Support System memberikan dukungan seperti pada gambar 2.2.



Gambar 2.2 Dukungan Komputer Untuk Proses SPK

2.2. Logika *Fuzzy*

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang input kedalam suatu ruang output.

1. Himpunan Fuzzy

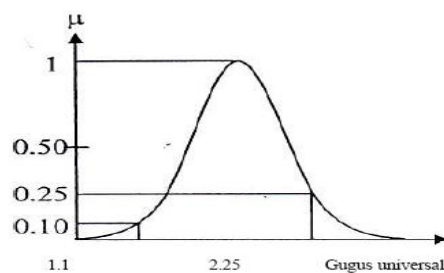
Dalam himpunan biasa (*crisp set*) keanggotaan setiap elemen himpunan *universal* pada suatu himpunan dinyatakan dengan anggota atau bukan anggota himpunan tersebut. Keanggotaan ini diberikan oleh suatu fungsi yang disebut fungsi keanggotaan. Fungsi keanggotaan memberikan nilai 1 untuk menyatakan anggota dan 0 untuk menyatakan bukan anggota.

Himpunan *fuzzy* merupakan pengembangan dari himpunan biasa. Fungsi keanggotaannya tidak hanya memberikan nilai 0 dan 1, tapi nilai yang berada pada suatu selang tertentu, biasanya dalam selang $[0,1]$, sehingga suatu elemen dapat memiliki derajat keanggotaan 0, 0.82 atau 1. Nilai yang diberikan oleh fungsi keanggotaan disebut derajat keanggotaan (*degree of membership*).

Apabila U menyatakan himpunan *universal* dan A adalah himpunan *fuzzy* dalam U , maka A adalah himpunan pasangan terurut sebagai berikut:

$$A = \{(u, \mu_A(u)) \mid u \in U\}$$

Dengan $\mu_A(u)$ adalah fungsi keanggotaan yang memberikan nilai derajat keanggotaan u terhadap himpunan *fuzzy* A , yaitu : $\mu_A : U \rightarrow [0,1]$. Misalkan fungsi keanggotaan himpunan *fuzzy* A seperti terlihat pada Gambar 2.5. Dari Gambar 2.3 dapat diketahui bahwa $\mu_A(1.1) = 0.10$, dan $\mu_A(2.25) = 0$.



Gambar 2.3 Fungsi keanggotaan himpunan fuzzy

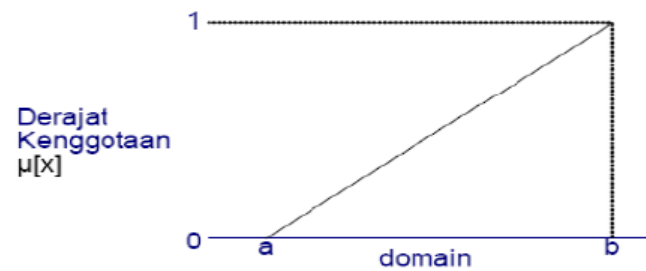
Himpunan *fuzzy* memiliki 2 atribut, yaitu :

- a. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti : BESAR, SEDANG, KECIL.
- b. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variable seperti: 12,10,8, dsb.

2. Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Ada dua cara mendefinisikan keanggotaan himpunan *fuzzy*, yaitu secara numeris dan fungsional. Definisi numeris menyatakan fungsi derajat keanggotaan sebagai vektor jumlah yang tergantung pada tingkat diskretisasi. Misalnya, jumlah elemen diskret dalam semesta pembicaraan. Definisi fungsional menyatakan derajat keanggotaan sebagai batasan ekspresi analitis yang dapat dihitung. Standar atau ukuran tertentu pada fungsi keanggotaan secara umum berdasar atas semesta X bilangan real.

Representasi Linear Ada 2 kemungkinan himpunan fuzzy linear yaitu: Kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol [0] bergerak kekanan menuju nilai domain yang memiliki derajat keanggotaan lebih tinggi.

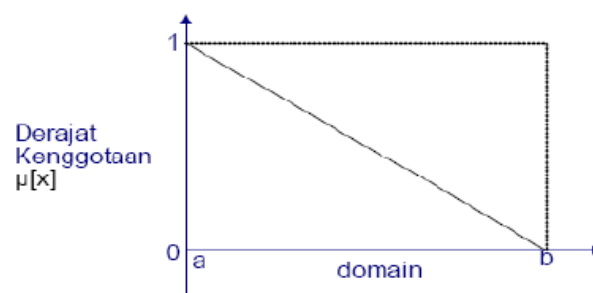


Gambar 2.4 Grafik Fungsi Keanggotaan Linear Naik

Fungsi Keanggotaan :

$$\mu[x] = \begin{cases} 0; & x \leq a \\ (x-a)/(b-a); & a \leq x \leq b \\ 1; & x \geq b \end{cases}$$

Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah.



Gambar 2.5 Grafik Fungsi Keanggotaan Linear Turun

Fungsi Keanggotaan:

$$\mu[x] = \begin{cases} (x-a)/(b-a); & a \leq x \leq b \\ 0; & x \geq b \end{cases}$$

1. Basis Data *Fuzzy* Model Tahani

Selama ini sudah ada beberapa penelitian tentang basis data *fuzzy*. Salah satu diantaranya adalah model Tahani. Basis *data, fuzzy* model Tahani masih tetap menggunakan relasi umum (standar), hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi pada *query*-nya. Sebagian besar basis data standar diklarifikasikan berdasarkan bagaimana data tersebut dipandang oleh pengguna. Misalkan data karyawan yang tersimpan pada tabel DT_KARYAWAN dengan *field* NIP, nama, tgl_lahir, gaji_per_bulan seperti pada table 2.1.

Tabel 2.1 Data Karyawan Mentah

NIP	Nama	Tgl Lahir	Thn Masuk	Gaji / Bln
01	Lia	03-06-1972	1996	750000
02	Iwan	23-09-1954	1985	1500000
03	Sari	12-12-1966	1988	1255000
04	Andi	06-03-1965	1998	1040000
05	Budi	04-12-1960	1990	950000
06	Amir	18-11-1963	1989	1600000
07	Rian	28-05-1965	1997	1250000
08	Kiki	09-07-1971	2001	550000
09	Alda	14-08-1967	1999	735000
10	Yoga	17-09-1977	2000	860000

Kemudian dari tabel DT_KARYAWAN, diperoleh suatu tabel temporer untuk menghitung umur karyawan dan masa kerjanya. Tabel tersebut diberi nama dengan tabel KARYAWAN.

Tabel 2.2 Data Karyawan Setelah diolah

NIP	Nama	Umur (th)	Masa Kerja (th)	Gaji / Bln
01	Lia	30	6	750000
02	Iwan	48	17	1500000
03	Sari	36	14	1255000
04	Andi	37	4	1040000
05	Budi	42	12	950000
06	Amir	39	13	1600000
07	Rian	37	5	1250000
08	Kiki	32	1	550000
09	Alda	35	3	735000
10	Yoga	25	2	860000

Dengan menggunakan basis data standar, maka dapat dicari data-data karyawan dengan spesifikasi tertentu menggunakan *query*. Misal bila ingin mendapatkan informasi tentang nama-nama karyawan yang usianya kurang dari 35 tahun, maka *query*-nya adalah :

```
SELECT NAMA FROM KARYAWAN WHERE (UMUR < 35)
```

Hasil : Lia, Kiki, Yoga.

Query untuk mendapatkan informasi karyawan gajinya lebih dari 1 juta:

```
SELECT NAMA FROM KARYAWAN WHERE (GAJI > 1000000)
```

Hasil : Iwan, Sari, Andi, Amir, Rian.

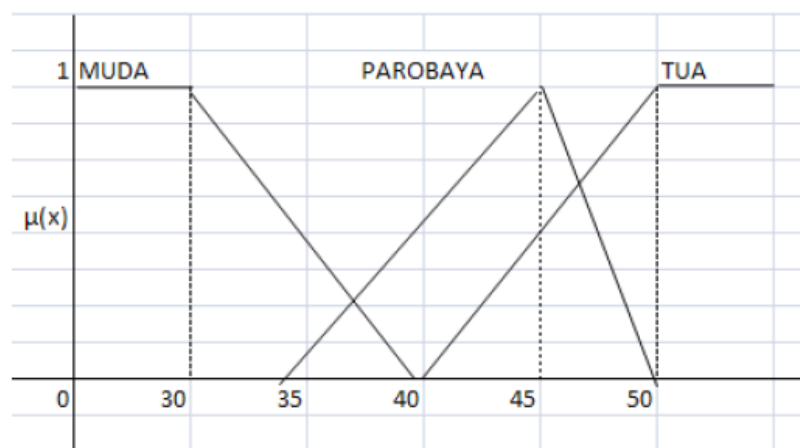
Query untuk mendapatkan informasi karyawan masa kerja kurang dari atau sama dengan 5 tahun tetapi gaji sudah lebih dari 1 juta :

```
SELECT NAMA FROM KARYAWAN WHERE (MASAKERJA <= 5) AND  
(GAJI > 1000000)
```

Hasil : Andi, Rian.

Pada kenyataannya seseorang kadang membutuhkan informasi dari data-data yang bersifat *ambiguous*, contoh “mencari data karyawan yang masih muda dan memiliki gaji yang tinggi”. Apabila hal ini terjadi, maka digunakanlah basisdata *fuzzy*. Salah satu diantaranya adalah model Tahani.

Basisdata *fuzzy* model Tahani masih tetap menggunakan relasi standar, hanya saja model ini menggunakan teori himpunan *fuzzy* untuk mendapatkan informasi query-nya. Misalkan mengkategorikan usia karyawan diatas kedalam himpunan : MUDA, PAROBAYA, dan TUA



Gambar 2.6 Fungsi Keanggotaan untuk variabel usia

Tabel 2.3 menunjukkan tabel karyawan berdasarkan umur dengan derajat keanggotaannya pada setiap himpunan.

Tabel 2.3 Karyawan Berdasarkan Umur

NIP	Nama	Umur (th)	Derajat Keanggotaan $\mu[x]$		
			MUDA	PAROBAYA	TUA
01	Lia	30	1	0	0
02	Iwan	48	0	0.4	0.8
03	Sari	36	0.4	0.1	0
04	Andi	37	0.3	0.2	0
05	Budi	42	0	0.7	0.2
06	Amir	39	0.1	0.4	0
07	Rian	37	0.3	0.2	0
08	Kiki	32	0.8	0	0
09	Alda	35	0.5	0	0
10	Yoga	25	1	0	0

Fungsi Keanggotaan :

$$a) \mu_{MUDA}[x] = \begin{cases} 1 & \text{jika } x \leq 30 \\ 40 - x/10 & \text{jika } 30 < x < 45 \\ 0 & \text{jika } x \geq 40 \end{cases}$$

jika $x \leq 30$
jika $30 < x < 45$
jika $x \geq 40$

$$b) \mu_{PAROBAYA}[x] = \begin{cases} 1 & \text{jika } x \leq 35 \text{ atau } x \geq 50 \\ (x - 35)/(45 - 35) & \text{jika } 35 < x < 45 \\ (50 - x)/(50 - 45) & \text{jika } 45 \leq x \leq 50 \end{cases}$$

jika $x \leq 35$ atau $x \geq 50$
jika $35 < x < 45$
jika $45 \leq x \leq 50$

$$c) \mu_{TUA}[x] = \begin{cases} 0 & \text{jika } x \leq 40 \\ (x - 40)/(50 - 40) & \text{jika } 40 < x < 50 \\ 1 & \text{jika } x \geq 50 \end{cases}$$

jika $x \leq 40$
jika $40 < x < 50$
jika $x \geq 50$

Variabel masa kerja dikategorikan dalam himpunan : BARU dan LAMA.



Gambar 2.7 Fungsi Keanggotaan untuk masa kerja

2.3. Konsep Dasar Sistem

Terdapat dua kelompok pendekatan di dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem sebagai berikut ini:

Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

Pendekatan sistem yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai berikut ini:

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

2.3.1. Karakteristik Sistem

Suatu sistem mempunyai karakteristik yang tertentu yaitu :

1. Komponen Sistem (*Components*)

Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian sistem, yang mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem keseluruhan.

2. Batas Sistem (*Boundary*)

Batas sistem (*boundary*) merupakan daerah yang membatasi antara suatu system dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas suatu system menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environments*)

Lingkungan luar (*environments*) dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan merugikan sistem.

4. Penghubung Sistem (*Interface*)

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan susbsistem lainnya sehingga memungkinkan sumber-sumber daya mengalir antara subsistem yang satu dengan yang lain.

5. Masukan Sistem (*Input*)

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan jadi keluaran.

8. Sasaran Sistem (*Objectives*)

Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan system dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.3.2. Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandangan, diantaranya adalah sebagai berikut :

1. Sistem diklasifikasikan sebagai sistem abstrak (*abstract system*) dan sistem fisik (*physical system*). Sistem abstrak adalah sistem yang berupa pemikiran atau ide yang tidak tampak secara fisik, misalnya sistem teologia dan sistem fisik merupakan sistem yang ada secara fisik, misalnya sistem komputer.
2. Sistem diklasifikasikan sebagai sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*). Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia, misalnya sistem

perputaran bumi dan system buatan manusia adalah sistem yang dirancang oleh manusia.

3. Sistem diklasifikasikan sebagai sistem tertentu (*deterministic system*) dan system tak tentu (*probabilistic system*). Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi dan sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
4. Sistem diklasifikasikan sebagai sistem tertutup (*closed system*) dan sistem terbuka (*open system*). Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya dan sistem terbuka adalah system yang berhubungan dan terpengaruh dengan lingkungan luarnya

2.4 Basis Data (*Database*)

Basis data terdiri atas dua kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.

Prinsip utama dalam basis data adalah pengaturan data/arsip dan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data/arsip yang menggunakan media penyimpanan elektronis seperti *disk* (disket

atau *harddisk*). Basis data dikelola/ditangani melalui perantara alat/mesin pintar elektronis yang kita kenal sebagai komputer.

Basis data bukan hanya sekedar penyimpanan data secara elektronis dengan bantuan komputer. Artinya, tidak semua bentuk penyimpanan data secara elektronis bisa disebut basis data. Yang sangat ditonjolkan dalam basis data adalah pengaturan/pemilahan/pengelompokkan/pengorganisasian data yang akan kita simpan sesuai fungsi/jenisnya. Pemilahan/pengelompokkan/pengorganisasian ini dapat berbentuk sejumlah file/tabel terpisah atau dalam bentuk pendefinisian kolomkolom/*field-field* data dalam setiap file/tabel.

Operasi-operasi dasar yang dapat dilakukan berkenaan dengan basis data dapat meliputi pembuatan basis data baru (*create database*), penghapusan basis data (*drop database*), pembuatan file/tabel baru ke suatu basis data (*create table*), penghapusan file/tabel dari suatu basis data (*drop table*), penambahan/pengisian data baru ke sebuah file/tabel di sebuah basis data (*insert*), pengambilan data dari sebuah file/tabel (*retrieve/search*), pengubahan data dari sebuah file/tabel (*update*), dan penghapusan data dari sebuah file/tabel (*delete*).

2.4.1. Database Management System (DBMS)

Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah Perangkat Lunak (Sistem) yang khusus/spesifik. Perangkat lunak inilah (disebut DBMS) yang akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. Ia juga

menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan/konsistensi data, dan sebagainya.

Perangkat lunak yang termasuk DBMS seperti dBase III+, dBase IV, FoxBase, Rbase, MS-Access dan Borland-Paradox (untuk kelas sederhana) atau Borland-Interbase, MS-SQLServer, CA-Open Ingres, Oracle, Informix dan Sybase (untuk kelas kompleks/berat).

2.4.2. Tujuan Basis Data

Tujuan awal dan utama dalam pengelolaan data dalam sebuah basis data adalah agar dapat memperoleh menemukan kembali data (yang dicari) dengan mudah dan cepat. Secara lebih lengkap, pemanfaatan basis data dilakukan untuk memenuhi sejumlah tujuan (objektif) seperti berikut ini:

- a. Kecepatan dan kemudahan (*speed*)
- b. Efisiensi ruang penyimpanan (*space*)
- c. Keakuratan (*accuracy*)
- d. Ketersediaan (*availability*)
- e. Kelengkapan (*completeness*)
- f. Keamanan (*security*)
- g. Kebersamaan pemakaian (*sharability*)

2.4.3 Pemakai (*user*) Basis Data

Ada beberapa jenis/tipe pemakai suatu sistem basis data yang dibedakan berdasarkan cara mereka berinteraksi terhadap sistem:

1. Programmer Aplikasi

Pemakai yang berinteraksi dengan basis data melalui *Data Manipulation Language* (DML), yang disertakan (*embedded*) dalam program yang ditulis dalam bahasa pemrograman induk (seperti C, Pascal, Cobol, dan lain-lain).

2. User Mahir (*Casual User*)

Pemakai yang berinteraksi dengan sistem tanpa menulis modul program. Mereka menyatakan *query* (untuk akses data) dengan bahasa *query* yang telah disediakan oleh suatu DBMS.

3. User Umum (*End User Naive User*)

Pemakai yang berinteraksi dengan sistem basis data melalui pemanggilan satu program aplikasi permanen (*executable program*) yang telah ditulis/disediakan sebelumnya.

4. User Khusus (*Specialized User*)

Pemakai yang menulis aplikasi basis data non konvensional, tetapi untuk keperluan-keperluan khusus, seperti untuk aplikasi AI, Sistem Pakar, Pengolahan Citra, dan lain-lain, yang bisa saja mengakses basis data dengan/tanpa DBMS yang bersangkutan.

Untuk sebuah sistem basis data yang *stand-alone*, maka pada suatu saat hanya ada satu pemakai yang dapat bekerja. Sedang untuk sistem basis data dalam

jaringan, maka pada suatu saat ada banyak pemakai yang dapat berhubungan (menggunakan) basis data yang sama.

2.5. Alat Pemodelan Sistem

Alat-alat pemodelan sistem informasi sangat dibutuhkan dalam proses analisis dan perancangan sistem. Alat-alat pemodelan sistem informasi terdiri dari:

1. Bagan Alir Dokumen (*Document Flowmap*)

Bagan alir dokumen (*document flowmap*) atau disebut juga bagan alir formulir (*form flowmap*) atau *paperwork flowmap* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

2. *Entity-Relationship Diagram* (ERD)

ERD adalah diagram yang memperlihatkan entitas-entitas yang terlibat dalam suatu sistem serta hubungan-hubungan (*relation*) antar entitas.

Komponen-komponen pembentuk model ERD yaitu:

- a. Entitas (*entity*)

Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain. Entitas dapat berupa orang, tempat, benda, peristiwa atau konsep yang bisa memberikan atau mengandung informasi.

- b. Atribut (*attributes/properties*)

Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik (*properti*) dari entitas tersebut.

- c. Relasi (*relationship*)

Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

d. Kardinalitas/derajat

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi yang terjadi di antara dua himpunan entitas dapat berupa :

- 1) *One to One* (1-1), relasi yang terjadi jika sebuah *entry* dalam sebuah *object data store* dihubungkan dengan hanya sebuah *entry* dalam *object data store* yang lain.
- 2) *One to Many* (1-M), relasi yang terjadi jika sebuah *entry* dalam sebuah *object data store* dihubungkan dengan satu atau lebih *entry* dalam *object data store* yang lain.
- 3) *Many to Many* (M-M), relasi yang terjadi jika satu atau lebih *entry* dalam sebuah *object data store* dihubungkan dengan satu atau lebih *entry* dalam *object data store*.
- 4) Kunci (*key*)

Sebuah atribut atau set atribut yang nilainya mengidentifikasi entitas secara unik dalam set entitas.

3. Diagram Konteks (*Context Diagram*)

Diagram konteks merupakan diagram aliran data pada tingkat paling atas yang merupakan penggambaran yang berfungsi untuk memperlihatkan interaksi/hubungan langsung antara sistem dengan lingkungannya.

Diagram konteks menggambarkan sebuah sistem berupa sebuah proses yang berhubungan dengan satu atau beberapa entitas/*entity*.

4. *Data Flow Diagram* (DFD)

DFD/DAD adalah suatu alat pemodelan yang digunakan untuk memodelkan fungsi dari sistem, menggambarkan secara rinci mengenai sistem sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan menunjukkan dari dan ke mana data mengalir serta penyimpanannya. Beberapa simbol digunakan di DFD:

- a. Kesatuan luar (*external entity*) atau batas sistem (*boundary*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan masukan atau menerima keluaran dari sistem.
- b. Arus data (*data flow*) ini mengalir diantara proses (*process*), simpanan data (*data store*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.
- c. Proses (*process*) merupakan kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.
- d. Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa suatu file atau *database* di sistem komputer, suatu arsip atau

catatan manual, suatu kotak tempat data di meja seseorang, suatu tabel acuan manual, dan suatu agenda atau buku.

5. Spesifikasi Proses (*Process Specification* (PSPEC))

Spesifikasi proses (PSPEC) digunakan untuk menggambarkan semua proses model aliran yang nampak pada tingkat akhir penyaringan. Kandungan dari spesifikasi proses dapat termasuk teks naratif, gambaran bahasa desain program (*Programme Design Language* (PDL)) dari algoritma proses, persamaan matematika, tabel, diagram, atau bagan.

6. Kamus Data (*Data Dictionary*)

Kamus data (*data dictionary*) atau disebut juga dengan istilah *systems data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. Kamus data harus memuat hal-hal berikut ini:

- a. Nama arus data
- b. Alias atau nama lain dari data dapat dituliskan bila nama lain ini ada.
Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan yang lainnya.
- c. Bentuk data, dapat berupa dokumen dasar atau formulir, dokumen hasil cetakan komputer, laporan tercetak, tampilan di layar monitor, variabel, parameter, dan *field*.

- d. Arus data, menunjukkan dari mana data mengalir dan ke mana data akan menuju.
 - e. Penjelasan, dapat diisi dengan keterangan-keterangan tentang arus data tersebut.
 - f. Periode, menunjukkan kapan terjadinya arus data.
 - g. Volume, digunakan untuk mengidentifikasi besarnya simpanan luar yang akan digunakan, kapasitas dan jumlah dari alat input, alat pemroses dan alat output.
 - h. Struktur data, menunjukkan arus data yang dicatat di kamus data terdiri dari item-item data apa saja.
7. Skema Relasi

Skema relasi adalah untuk presentasi atribut-atribut dari *entity* yang terdapat dalam sistem dan hubungan antar *entity* pada model ERD. Skema relasi merupakan turunan dari ERD.

2.6. Perangkat Lunak Penunjang

2.6.1. Borland Delphi

Delphi adalah suatu program berbasis bahasa pascal yang berjalan dalam lingkungan windows. Delphi telah memanfaatkan suatu teknik pemrograman yang disebut RAD yang telah membuat pemrograman menjadi lebih mudah. Delphi adalah suatu bahasa pemrograman yang telah memanfaatkan metode pemrograman *Object Oriented Programming* (OOP). Secara ringkas, *object* adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat (*visual*).

Object biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu.

Khusus untuk pemrograman *database*, Delphi menyediakan object yang sangat kuat, canggih, dan lengkap, sehingga memudahkan pemrogram dalam merancang, membuat dan menyelesaikan aplikasi *database* yang diinginkan. Selain itu, Delphi juga dapat menangani data dalam berbagai format *database*, misalnya format MSAccess, SyBase, Oracle, FoxPro, Informix, DB2, MySQL dan lain-lain. Format *database* yang dianggap asli dari Delphi adalah Paradox dan dBase.

2.6.2. MySQL

MySQL adalah sebuah *database* server yang dibuat oleh Tcx Data KonsultAB. Saat ini MySQL telah digunakan oleh perusahaan-perusahaan terkemuka di seluruh dunia, diantaranya Silicon Graphics (<http://www.sgi.com>), Siemens Nixdorf (<http://www.siemens.com>), Alesis Digital Studio Electronics (<http://www.alesis.com>) dan masih banyak perusahaan-perusahaan terkemuka lainnya yang menggunakan MySQL.

Perusahaan-perusahaan tersebut dapat dilihat pada MySQL user's list di <http://www.mysql.com/information/userlist.htm>. MySQL adalah sebuah *text based database server*, artinya MySQL tidak dibuat dalam bentuk aplikasi yang memiliki *Graphical User Interface*.