

BAB II

LANDASAN TEORI

2.1. PENGERTIAN SISTEM

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. (Jogiyanto HM, 2008)

Suatu sistem mempunyai tujuan atau sasaran (objektifitas). Tujuan biasanya dihubungkan dengan ruang lingkup yang lebih luas dan sasaran dalam ruang lingkup yang lebih sempit. Sasaran menentukan masukan dan keluaran yang dihasilkan. Sistem dikatakan berhasil jika mencapai suatu sasaran dan tujuan.

2.2. SISTEM PENDUKUNG KEPUTUSAN

Manusia merupakan bagian dari alam karena hidupnya yang tidak terlepas dari alam. Proses kehidupan manusia merupakan unsur yang semakin lama semakin mendominasi unsur-unsur lainnya di alam. Hal ini disebabkan karena manusia dibekali kemampuan-kemampuan untuk bisa berkembang. Segala proses yang terjadi di sekelilingnya dan dalam dirinya dirasakan dan diamatinya dengan menggunakan semua indera yang dimilikinya, dipikirkannya lalu ia berbuat dan bertindak.

Dalam menghadapi segala proses yang terjadi di sekelilingnya dan di dalam dirinya, hampir setiap saat manusia membuat atau mengambil keputusan dan melaksanakannya. Hal ini dilandasi dengan asumsi bahwa segala tindakan dilakukan secara sadar merupakan pencerminan hasil proses pengambilan keputusan dalam pikirannya, sehingga sebenarnya manusia sudah sangat terbiasa dalam membuat keputusan.

2.2.1. Definisi Sistem Pendukung Keputusan

Sistem pendukung keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan dan pemanipulasi data. Sistem ini digunakan untuk membantu mengambil keputusan dalam situasi yang semi terstruktur dan tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat.

DSS biasanya dibangun untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang. DSS yang seperti itu disebut aplikasi DSS. Aplikasi DSS digunakan dalam pengambilan keputusan. Aplikasi DSS menggunakan CBIS (*Computer Based Information Systems*) yang fleksibel, interaktif, dan dapat diadaptasi, yang dikembangkan untuk mendukung solusi atas masalah manajemen spesifik yang tidak terstruktur. (Kusrini, 2007)

2.2.2. Tujuan Sistem Pendukung Keputusan

Turban, dkk (dalam Kusri, 2007) mengungkapkan tujuan dari sistem pendukung keputusan yaitu :

- a. Membantu manager dalam pengambilan keputusan atas masalah semi terstruktur.
- b. Memberikan dukungan atas pertimbangan manager dan bukannya dimaksudkan untuk menggantikan fungsi manager.
- c. Meningkatkan efektivitas keputusan yang diambil manager lebih dari pada perbaikan efisiensinya
- d. Kecepatan komputasi, komputer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang rendah.
- e. Peningkatan produktivitas. Membangun satu kelompok pengambilan keputusan, terutama para pakar bisa sangat mahal.
- f. Dukungan kualitas. Komputer bisa meningkatkan kualitas keputusan yang dibuat.
- g. Berdaya saing. Manajemen dan pemberdayaan sumber daya perusahaan, tekanan persaingan menyebabkan tugas pengambilan keputusan menjadi sulit.
- h. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan.

2.2.3. Karakteristik dan Kapabilitas Sistem Pendukung Keputusan

Karakteristik dan kapabilitas kunci dari SPK menurut Turban dkk (2005), ialah:

1. Dukungan untuk pengambilan keputusan, terutama pada situasi semi terstruktur dan tak terstruktur, dengan menyertakan penilaian manusia dan informasi terkomputerisasi. Masalah-masalah tersebut tidak dapat dipecahkan oleh sistem komputer lain atau oleh metode atau alat kuantitatif standar.
2. Dukungan untuk semua *level* manajerial, dari eksekutif puncak sampai manajer lini.
3. Dukungan untuk individu dan kelompok. Masalah yang kurang terstruktur sering memerlukan keterlibatan individu dari departemen dan tingkat organisasional yang berbeda atau bahkan dari organisasi lain. SPK mendukung tim *virtual* melalui alat-alat Web kolaboratif.
4. Dukungan untuk keputusan independen dan atau sekuensial. Keputusan dapat dibuat satu kali, beberapa kali atau berulang (dalam interval yang sama).
5. Dukungan di semua fase proses pengambilan keputusan: intelegensi, desain, pilihan, dan implementasi.
6. Dukungan diberbagai proses dan gaya pengambilan keputusan.

7. Adaptivitas sepanjang waktu. Pengambilan keputusan seharusnya reaktif, dapat menghadapi perubahan kondisi secara cepat dan dapat mengadaptasikan SPK untuk memenuhi perubahan tersebut. SPK bersifat fleksibel dan karena itu pengguna dapat menambahkan, menghapus, menggabungkan, mengubah, atau menyusun kembali elemen-elemen dasar. SPK juga fleksibel dalam hal dapat dimodifikasi untuk memecahkan masalah lain yang sejenis.
8. Pengguna merasa seperti di rumah. Ramah-pengguna, kapabilitas grafis yang sangat kuat dan antarmuka manusia-mesin interaktif dengan satu bahasa alami dapat sangat meningkatkan keefektifan SPK. Kebanyakan aplikasi SPK yang baru menggunakan antarmuka berbasis-Web.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, *timeliness*, kualitas) ketimbang pada efisiensinya (biaya pengambilan keputusan). Ketika SPK disebar, pengambilan keputusan sering membutuhkan waktu lebih lama, namun keputusannya lebih baik.
10. Kontrol penuh oleh pengambil keputusan terhadap semua langkah proses pengambilan keputusan dalam memecahkan suatu masalah. SPK secara khusus menekankan untuk mendukung pengambil keputusan, bukannya menggantikan.

11. Pengguna akhir dapat mengembangkan dan memodifikasi sendiri sistem sederhana. Sistem yang lebih besar dapat dibangun dengan bantuan ahli sistem informasi. Perangkat lunak OLAP dalam kaitannya dengan data *warehouse* membolehkan pengguna untuk membangun DSS yang cukup besar dan kompleks.
12. Biasanya model-model digunakan untuk menganalisa situasi pengambilan keputusan. Kapabilitas pemodelan memungkinkan eksperimen dengan berbagai strategi yang berbeda di bawah konfigurasi yang berbeda.
13. Akses disediakan untuk berbagai sumber data, format dan tipe mulai dari sistem informasi geografis (SIG) sampai sistem berorientasi objek.
14. Dapat dilakukan sebagai alat *stand alone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan disatu organisasi keseluruhan dan dibeberapa organisasi sepanjang rantai persediaan. Dapat diintegrasikan dengan SPK lain dan atau aplikasi lain, dan dapat didistribusikan secara internal dan eksternal dengan menggunakan *networking* dan teknologi Web.

2.2.4. **Komponen-Komponen Sistem Pendukung Keputusan**

Menurut Turban dkk (2005) komponen-komponen dalam Sistem Pendukung Keputusan terdiri dari:

a. Subsistem Manajemen Data

Subsistem manajemen data memasukkan satu database yang berisi data yang relevan untuk situasi dan dikelola oleh perangkat lunak yang disebut sistem manajemen database (DBMS). Subsistem manajemen data dapat diinterkoneksi dengan data *warehouse* perusahaan, suatu repositori untuk data perusahaan yang relevan untuk pengambilan keputusan. Biasanya data disimpan atau diakses via server Web database.

b. Subsistem Manajemen Model

Merupakan paket perangkat lunak yang memasukkan model keuangan, statistik, ilmu manajemen, atau model kuantitatif lainnya yang memberikan kapabilitas analitik dan manajemen perangkat lunak yang tepat. Bahasa-bahasa pemodelan untuk membangun model-model kustom juga dimasukkan. Perangkat lunak ini sering disebut Sistem Manajemen Basis Model (MBMS). Komponen ini dapat dikoneksikan ke penyimpanan korporat atau eksternal yang ada pada model. Sistem manajemen dan metode solusi model diimplementasikan pada sistem pengembangan Web (seperti *Java*) untuk berjalan pada server aplikasi.

c. Subsistem Antarmuka Pengguna

Pengguna berkomunikasi dengan dan memerintahkan SPK melalui subsistem ini. Pengguna adalah bagian yang dipertimbangkan dari sistem. Para peneliti menegaskan bahwa beberapa kontribusi unik dari SPK berasal dari interaksi yang intensif antara komputer dan pembuat keputusan. *Browser Web* memberikan struktur antarmuka pengguna grafis yang familier dan konsisten bagi kebanyakan SPK.

d. Subsistem Manajemen Berbasis-Pengetahuan

Subsistem ini dapat mendukung semua subsistem lain atau bertindak sebagai suatu komponen independen. Ia memberikan inteligensi untuk memperbesar pengetahuan si pengambil keputusan. Subsistem ini dapat diinterkoneksi dengan repositori pengetahuan perusahaan (bagian dari sistem manajemen pengetahuan), yang kadang-kadang disebut basis pengetahuan organisasional. Pengetahuan dapat disediakan *via server Web*. Banyak metode kecerdasan tiruan diimplementasikan dalam sistem pengembangan Web seperti Java, dan mudah untuk diintegrasikan dengan komponen SPK lainnya.

Berdasarkan definisi, SPK harus mencakup tiga komponen utama dari DBMS, MBMS, dan antarmuka pengguna. Subsistem manajemen berbasispengetahuan adalah opsional, namun dapat memberikan banyak manfaat karena

memberikan inteligensi bagi tiga komponen utama tersebut. Seperti pada semua sistem informasi manajemen, pengguna dapat dianggap sebagai komponen SPK.

2.2.5. Langkah – langkah Pemodelan Sistem Pendukung Keputusan

Saat melakukan pemodelan dalam pembangunan DSS dilakukan langkah-langkah (Kusrini. 2007), yaitu sebagai berikut :

a. Studi Kelayakan (*Intelligence*)

Pada langkah ini, sasaran ditentukan dan dilakukan pencarian prosedur, pengumpulan data, identifikasi masalah, identifikasi kepemilikan masalah, klasifikasi masalah, hingga akhirnya terbentuk sebuah pernyataan masalah.

b. Perancangan (*Design*)

Pada tahapan ini akan diformulasikan model yang akan digunakan dan kriteriankriteria yang ditentukan. Setelah itu, dicari alternatif model yang bisa menyelesaikan permasalahan tersebut. Langkah selanjutnya adalah memprediksi keluaran yang mungkin. Kemudian ditentukan variabel-variabel model.

c. Pemilihan (*Choice*)

Setelah pada tahap perancangan ditentukan berbagai alternatif model berserta variable-variabelnya. Pada tahapan ini akan dilakukan pemilihan modelnya, termasuk solusi dari model tersebut. Selanjutnya, dilakukan analisis sensitivitas, yakni dengan mengganti beberapa variabel.

- d. Membuat DSS Setelah menentukan modelnya, berikutnya adalah mengimplementasikannya dalam aplikasi DSS.

2.3. FUZZY MULTIPLE ATTRIBUT DECISION MAKING (FMADM)

Fuzzy Multiple Attribute Decision Making (FMADM) adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu (Muley A.A, 2010).

Inti dari *Fuzzy Multiple Attribute Decision Making* adalah menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan. Pada dasarnya, ada 3 pendekatan untuk mencari nilai bobot atribut, yaitu pendekatan subyektif, pendekatan obyektif dan pendekatan integrasi antara subyektif dan obyektif. Masing- masing pendekatan memiliki kelebihan dan kelemahan. Pada pendekatan subyektif, nilai bobot ditentukan berdasarkan subyektifitas dari para pengambil keputusan, sehingga beberapa faktor dalam proses perankingan alternatif bisa ditentukan secara bebas. Sedangkan pada pendekatan obyektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektifitas dari pengambil keputusan. Salah satu metode yang digunakan untuk menyelesaikan masalah *Fuzzy Multiple Attribute Decision Making* (FMADM) adalah metode *Simple Additive Weighting* (SAW) (Deni, 2013).

2.3.1. METODE *SIMPLE ADDITIVE WEIGHTING* (SAW)

Metode SAW sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Formula untuk melakukan normalisasi adalah sebagai berikut :

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i x_{ij}} & \text{jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\text{Min}_i x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases} \dots\dots(2.1)$$

dengan r adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_{ij} dengan $i=1,2,\dots,m$ dan $j=1,2,\dots,n$. Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai :

$$V_i = \sum_{j=1}^n W_j r_{ij} \dots\dots\dots(2.2)$$

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih.

Kelebihan dari model *Simple Additive Weighting* (SAW) dibandingkan dengan model pengambilan keputusan yang lain terletak pada kemampuannya untuk melakukan penilaian secara lebih tepat karena didasarkan pada nilai kriteria dan bobot preferensi yang sudah ditentukan, selain itu SAW juga dapat menyeleksi alternatif terbaik dari sejumlah alternatif yang ada karena adanya

proses perankingan setelah menentukan nilai bobot untuk setiap atribut.

2.3.2. Langkah Penyelesaian Metode SAW

Dalam penelitian ini menggunakan model FMADM metode SAW. Adapun langkah-langkahnya adalah:

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu C_i .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria (C_i), kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan ataupun atribut biaya) sehingga diperoleh matriks ternormalisasi R .
4. Hasil akhir diperoleh dari proses perankingan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik (A_i) sebagai solusi (Kusumadewi, 2006).

2.4. BEASISWA

Beasiswa adalah pemberian berupa bantuan keuangan yang diberikan kepada perorangan yang bertujuan untuk digunakan demi keberlangsungan pendidikan yang ditempuh. Beasiswa tidak bisa diberikan kepada semua peserta didik, ada syarat-syarat yang telah ditentukan yang harus di penuhi untuk bisa mendapatkan beasiswa.

2.5. SEKILAS TENTANG *Microsoft Visual Basic .NET*

Menurut Sibero (2010), VB .NET adalah bahasa pemrograman yang dikembangkan oleh perusahaan *Microsoft*. *Visual basic .NET* merupakan pengembangan dari versi sebelumnya, yaitu *visual basic 6.0*. Perbedaan mendasar antara *visual basic .NET* dengan versi-versi sebelumnya adalah kemampuan OOP (*object oriented programming*) yang telah ditanamkan pada *visual basic .NET*.

Aplikasi *Visual Basic* memiliki keunggulan antara lain *visual basic* memiliki kemampuan untuk mengkompilasi program dalam bentuk *native code*, yaitu optimisasi pada saat prosesor mengkompilasi dan menjalankan program, sehingga menjadikan *visual basic* memiliki kemudahan dan kecepatan di dalam mengakses program.

2.6. SEKILAS TENTANG MySQL

MySQL sebenarnya merupakan turunan dari SQL. SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam *query data*. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query* MySQL bisa sepuluh kali lebih cepat dari

PostgreSQL dan lima kali lebih cepat dibandingkan Interbase. (Andri Kristanto, 2010).

MySQL menawarkan berbagai keunggulan dibandingkan database server lain. Berikut ini adalah beberapa keunggulan MySQL :

- a. Mampu menangani jutaan user dalam waktu yang bersamaan.
- b. Mampu menampung lebih dari 50.000.000 *record*.
- c. Sangat cepat mengeksekusi perintah.
- d. Memiliki *user privilege* sistem yang mudah dan efisien.

2.7. DESAIN SISTEM

2.7.1 Data Flow Diagram

Diagram yang menggunakan notasi simbol untuk menggambarkan arus data sistem. (Hartono, 2010)

Menurut Hartono, tahun 2010 dalam bukunya Basis Data ada beberapa simbol yang digunakan pada DFD untuk mewakili:

1. Kesatuan Luar

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar system yang dapat berupa orang, organisasi, atau system lain yang berada pada lingkungan luarnya yang memberikan *input* atau menerima *output* dari sistem.

2. Arus Data

Arus data (*data flow*) di DFD diberi symbol suatu anak panah. Arus data ini mengalir diantara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

3. Proses

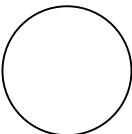
Proses (*process*) menunjukkan pada bagian yang mengubah input menjadi output, yaitu menunjukkan bagaimana satu atau lebih input diubah menjadi beberapa output. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

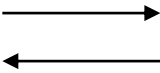


4. Data Store

Simpanan data (*data store*) merupakan simpanan dari data yang dapat berupa suatu file atau data base pada system computer.

Dalam menggambarkan diagram arus data atau *data flow diagram* menggunakan simbol-simbol seperti dibawah ini:

Tabel 2.1 simbol-simbol DFD


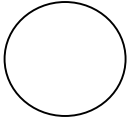
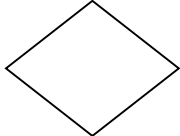
No.	Simbol	Keterangan
1.		Simbol proses. Menunjukkan proses kemputerisasi.


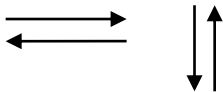

No.	Simbol	Keterangan
2.		Simbol aliran data. Menunjukkan arah ke bagian lain atau ke proses sebaliknya.
3.		Simbol penyimpanan. Menunjukkan sebagai komponen untuk memudahkan kumpulan data atau informasi.
4.		Simbol terminator. Menunjukkan organisasi (kelompok organisasi) atau organisasi diluar system lain yang memberi atau menerima data.

2.7.2 Flowchart

Flowchart (bagan alir data) adalah bagan yang menunjukkan alir didalam program atau prosedur sistem secara logika. Bagan alir ini digunakan terutama untuk mendefinisikan hubungan antara bagian (pelaku proses), proses manusia maupun proses komputer dan aliran data (dalam bentuk masukan dan keluaran). (Yatini, 2010)

Tabel 2.2 simbol-simbol Flowchart

No.	Simbol	Keterangan
1.		Dokumen, digunakan untuk menunjukkan dokumen <i>input</i> dan <i>output</i> baik untuk proses manual, mekanik, atau komputer.
2.		Penghubung, digunakan untuk menunjukkan hubungan dengan bagian lain dalam satu halaman.
3.		<i>Symbol decision</i> , yaitu menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya/tidak.

No.	Simbol	Keterangan
4.		Proses, digunakan untuk menunjukkan kegiatan proses dari operasi program computer.
5.		Garis alir, digunakan untuk menunjukkan arus proses.
6.		Terminator yang berfungsi untuk eksekusi suatu data.

2.7.3 Entitas Relationship Diagram

Entity Relationship Diagram (ERD) merupakan suatu kumpulan file-file yang saling berkaitan satu dengan yang lainnya pada model data relation hubungan antar file direlasikan dengan kunci relasi (*relation key*) yang merupakan kunci utama dimasing-masing file. Perancangan database yang tepat akan menyebabkan MySql/paket program lainnya akan bekerja dengan optimal. (Yatini, 2010).

Entity Relationship Diagram atau disebut ERD, adalah mendokumentasikan data perusahaan dengan mengidentifikasi jenis dan hubungannya. Komponen-komponen ERD yaitu:

1. Entitas

Jenis entitas (*entity type*) dapat berupa suatu elemen lingkungan, sumber daya, atau transaksi yang begitu pentingnya bagi perusahaan sehingga didokumentasikan dengan data jenis entitas didokumentasikan dengan symbol persegi panjang.


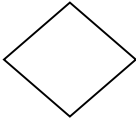
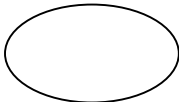
2. Hubungan

Hubungan adalah suatu asosiasi yang ada antara dua jenis entitas. Hubungan digambarkan dengan bentuk belah ketupat. Setiap belah ketupat diberi label kata kerja.

3. Atribut

Atribut adalah karakteristik dari suatu entitas. Atribut-atribut tersebut sebenarnya adalah elemen-elemen data dan masing-masing diberikan satu nilai tunggal, yang disebut nilai atribut digambarkan dalam bentuk elips.

Tabel 2.3 Simbol – simbol Atribut

No.	Symbol	Keterangan
1.		Entitas
2.		Hubungan
3.		Atribut