



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

- d. Memberikan informasi terstruktur dalam pengamlt keputusan perusahaan tiap periode.

BAB II

LANDASAN TEORI

2.1 Kajian Teori

2.1.1 *Decision Support System*

Decision Support System atau Sistem Pendukung Keputusan / SPK, secara umum didefinisikan sebagai sebuah sistem yang mampu memberikan kemampuan baik kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah semi-terstruktur. Secara khusus, SPK didefinisikan sebagai sebuah sistem yang mendukung kerja seorang manajer maupun sekelompok manajer dalam memecahkan masalah semi-terstruktur dengan cara memberikan informasi ataupun usulan menuju pada keputusan tertentu.

Konsep DSS diperkenalkan kira-kira pada kurun waktu 1970-an. Pada kurun waktu tersebut DSS masih dalam proses *Research* dan *Development*. Sedangkan aplikasinya secara meluas dimulai pada kira-kira akhir tahun 1980-an dan awal tahun 1990-an. Dan pada masa yang akan datang DSS masih akan berkembang terus dan memerlukan berbagai perbaikan dan penyempurnaan yang disesuaikan dengan keperluan dan perkembangan teknologi informasi. Di antara perkembangan DSS yang akan terjadi dimasa yang akan datang meputi aspek-aspek: *integrated architecture, connectivity, document data dan intelligence*. Andi (2007 : 65)

Definisi DSS sampai saat ini masih tergantung kepada dari sudut mana DSS tersebut dipandang. Namun pada umumnya DSS bisa didefinisikan dengan

melibatkan aspek-aspek sebagai berikut:

- a. Sistem yang berbasis komputer
- b. Membantu memecahkan masalah seorang manager
- c. Masalah semi terstruktur
- d. Interaktif di antara sistem dan manager
- e. Menggunakan analisis data

Kedua aspek yang terakhir adalah berasaskan aplikasi teknologi yang kemudian disebut dengan DDM (dialog, data dan modelling).

4

2.1.2 Kegunaan Sistem Pendukung Keputusan

Beberapa keuntungan penggunaan SPK antara lain adalah sebagai berikut **Kadir** (2009 : 25) :

- a. Mampu mendukung pencarian solusi dari berbagai permasalahan yang
- b. kompleks
- c. Dapat merespon dengan cepat pada situasi yang tidak diharapkan dalam konsisi yang berubah-ubah
- d. Mampu untuk menerapkan berbagai strategi yang berbeda pada konfigurasi berbeda secara cepat dan tepat
- e. Pandangan dan pembelajaran baru
- f. Sebagai fasilitator dalam komunikasi
- g. Meningkatkan kontrol manajemen dan kinerja
- h. Menghemat biaya dan sumber daya manusia (SDM)
- i. Menghemat waktu karena keputusan dapat diambil dengan cepat
- j. Meningkatkan efektivitas manajerial, menjadikan manager dapat bekerja lebih singkat dan dengan sedikit usaha
- k. Meningkatkan produktivitas analisis

2.1.3 Pengertian Sistem Informasi

Karakteristik dan Kemampuan Sistem Pendukung Keputusan.

Berikut Detailnya : **Kadir** (2009 : 43).

Beberapa karakteristik dari SPK, diantaranya adalah sebagai berikut:

- a. Mendukung seluruh kegiatan organisasi
- b. Mendukung beberapa keputusan yang saling berinteraksi
- c. Dapat digunakan berulang kali dan bersifat konstan
- d. Terdapat dua komponen utama, yaitu data dan model
- e. Menggunakan baik data eksternal maupun internal
- f. Memiliki kemampuan *what-if analysis* dan *goal seeking analysis*
- g. Menggunakan beberapa model kuantitatif. **Kadir** (2009 : 45).

Kemampuan yang harus dimiliki oleh sebuah sistem pendukung keputusan, diantaranya adalah sebagai berikut: **Kadir** (2009 : 48).

- a. Menunjang pembuatan keputusan manajemen dalam menangani masalah semiterstruktur dan tidak terstruktur.
- b. Membantu manajer pada berbagai tingkatan manajemen, mulai dari manajemen tingkat atas sampai manajemen tingkat bawah.
- c. Menunjang pembuatan keputusan secara kelompok dan perorangan.
- d. Menunjang pembuatan keputusan yang saling bergantung dan berurutan.
- e. Menunjang tahap-tahap pembuatan keputusan antara lain *intelligence, design, choice* dan *implementation*.
- f. Menunjang berbagai bentuk proses pembuatan keputusan dan jenis keputusan.
- g. Kemampuan untuk melakukan adaptasi setiap saat dan bersifat fleksibel.
- h. Kemudahan melakukan interaksi sistem.
- i. Meningkatkan efektivitas dalam pembuatan keputusan daripada efisiensi.

- j. Mudah dikembangkan oleh pemakai akhir.
- k. Kemampuan pemodelan dan analisis dalam pembuatan keputusan.
- l. Kemudahan melakukan pengaksesan berbagai sumber dan format data.

2.1.4 Pengertian *Fuzzy Multiple Attribute Decision Making* dan Jurnal Penelitian Terdahulu

2.1.4.1 Pengertian Metode *Fuzzy Multiple Attribute Decision Making*

Metode *Fuzzy Multiple Attribute Decision Making* Pada dasarnya dilakukan melalui 3 tahap, yaitu penyusunan komponen-komponen situasi, analisis, dan sintesis informasi (Rudolphi, 2010). Pada setiap penyusunan komponen, komponen situasi, akan dibentuk table taksiran yang berisi identifikasi alternatif dan spesifikasi tujuan, kriteria dan atribut.

Fuzzy Multiple Attribute Decision Making (FMADM) adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Inti dari FMADM 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 factor dalam proses perankingan alternatif bisa ditentukan secara bebas. Sedangkan pada pendekatan obyektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektifitas dari

pengambil keputusan.

Metode Fuzzy Multiple Attribute Decision Making ini menerapkan nilai pembobotan maksimal sebagai hasil terbaik yang akan dilakukan perankingan terhadap data yang akan disajikan sehingga user dapat dengan mudah memahami hasil terbaik mana yang disajikan.

Algoritma Fuzzy Multiple Attribute Decision Making :

Memberikan nilai setiap alternatif pada setiap kriteria yang sudah ditentukan, di mana nilai tersebut di peroleh berdasarkan nilai pembobotan. Memberikan nilai bobot sesuai dengan tingkat kepentingan. Melakukan proses perankingan dengan cara menghitung semua bobot penilaian yang menjadi kriteria sistem yang dibuat. (Kusumadewi, 2006).

2.1.4.2 Metode Fuzzy Multiple Attribute Decision Making Dalam Sistem Usulan

Metode *Fuzzy Multiple Attribute Decision Making* sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode FM ADM adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Metode FM ADM membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Berikut Rumusnya :

$$R_j = \frac{C_{ij}}{\text{Max}_{C_{ij}}}$$

Dalam metode penelitian ini ada bobot dan kriteria yang dibutuhkan untuk menentukan siapa yang akan terseleksi sebagai

penerima beasiswa. Adapun kriterianya adalah:

R = nilai parameter

C = point pembobotan

i = 1..3

j = 1..3

Max = point tertinggi

1 = selisih harga

2 = ongkos kirim

3 = waktu penawaran

Melakukan proses penilaian (R) dengan cara mencari ketiga kriteria ini dimana 1 = selisih harga berasal dari (Harga barang + ongkos kirim) ,

2 = Ongkos kirim berasal dari (input dari supplier) , dan 3 = waktu penawaran berasal dari (waktu input admin - waktu input supplier).

Dari 3 Kriteria ini maka ditotal $C1 + C2 + C3$ = mendapatkan hasil dari perhitungan.

Hasil dari R ini akan dibandingkan dengan R lainnya .Dan dipilih R yang memiliki nilai bobot point tertinggi.

Dari masing-masing bobot tersebut, maka dibuat suatu variabel-variabelnya. Dimana dari suatu variabel tersebut akan dirubah kedalam bilangan fuzzynya seperti dibawah ini :

Dari hasil wawancara yang dilakukan, maka menghasilkan usulan untuk dibuatnya, sistem pemilihan supplier terbaik dengan metode FM ADM yang akan dibuat penulis yang mana sistem yang dibuat akan memudahkan admin dalam menginputkan data pesanan barang yang dibutuhkan, yang mana nantinya data pesanan tersebut akan masuk ke akses supplier untuk diinputkan penawaran harganya, setelah penginputan harga tersebut, data yang disajikan sudah diurut berdasarkan poin pembobotan yang terdiri atas berdasarkan selisih harga dari harga perkiraan yang diinput admin, kemudian dari ongkos kirim yang dikenakan, dan berdasarkan waktu respon penawaran,

selain itu total harga penawaran juga tersaji, sehingga admin dalam mengambil keputusan bisa melihat dari poin pembobotan serta total harga terbaik yang akan dipilih oleh admin, berikut data pembobotannya :

Tabel 2.1 : pembobotan atau perankingan supplier terbaik

Pembobotan Berdasarkan :	
Berdasarkan Selisih Harga :	
Selisih Harga	Point
Rp 0 – Rp 500.000	100 point
Rp 500.001 – Rp 1.000.000	75 point
Rp 1.000.001 – Rp 2.000.000	50 point
>Rp 2.000.000	25 point
Berdasarkan Ongkos Kirim :	
Ongkos Kirim	Point
Rp 0 – Rp 200.000	100 point
Rp 200.001 – Rp 400.000	75 point
Rp 400.001 – Rp 600.000	50 point
>Rp 600.000	25 point
Berdasarkan Waktu Penawaran :	
Waktu Penawaran	Point
< = 3 jam	100 point
< = 6 jam	75 point
< = 9 Jam	50 point
>9Jam	25 point

A. Penerapan Algoritma FM ADM Dalam Sistem Usulan

Penerapan sistem FM ADM di sistem yang dibuat, dalam

pemilihan supplier terbaik atau optimal, mempunyai beberapa indikator pembobotan, dimana hasil dari pembobotan itu akan dirangking sesuai total penawaran harga dari yang terkecil sampai yang terbesar.

Dalam pemilihan supplier terbaik, para supplier akan dimintai terlebih dahulu untuk menginputkan penawaran harganya di sistem usulan yang akan dibuat nantinya.

Dalam pemilihan supplier terbaik, setelah supplier mengisi penawaran harga dari setiap barang yang dipesan oleh admin ditambah ongkos kirim di sistem usulan, setelah itu sistem akan menampilkan hasil perangkingan mulai dari total harga penawaran terendah sampai tertinggi , berikut pembobotan yang dihasilkan sistem berdasarkan.

1. Selisih Harga

Selisih Harga merupakan selisih harga dari total harga penawaran disertai ongkir dikurangi total harga perkiraan, yang mana dari selisih harga ini akan dihasilkan pembobotan. Untuk selisih harga sampai Rp 500.000 akan diberi nilai bobot 100 poin, untuk selisih harga lebih dari Rp 500.000 sampai Rp 1.000.000 akan diberi bobot 75 poin, untuk selisih harga lebih dari Rp 1.000.000 sampai Rp 2.000.000 akan diberi bobot penilaian sebesar 50 poin dan untuk selisih harga lebih dari Rp 2.000.000 akan mendapatkan bobot 25 poin.

2. Ongkos Kirim

Ongkos kirim adalah biaya yang dibebankan dari supplier kepada perusahaan untuk mengantarkan barang pesanan yang dipesan perusahaan yang telah diinputkan oleh admin. Ongkos kirim ini menjadi indikator selanjutnya yang akan menjadi penilaian

oleh admin dalam menentukan supplier terbaik yang akan dipilih untuk memenuhi pesanan dari barang yang telah dipesan oleh admin. Pembobotan indikator berdasarkan ongkos kirim (ongkir) ini diuraikan sebagai berikut, untuk ongkos kirim sampai dengan batas Rp 200.000, maka akan mendapatkan bobot penilaian 100 poin. Ongkir lebih dari Rp 200.000 sampai dengan Rp 400.000 akan mendapatkan bobot penilaian sebesar 75 poin. Ongkir lebih dari Rp 400.000 sampai dengan Rp 600.000 akan diberikan bobot penilaian sebesar 50 poin dan lebih dari Rp 600.000 akan mendapatkan bobot penilaian 25 poin.

3. Waktu Penawaran

Waktu penawaran adalah waktu respon akses supplier dalam menginputkan penawaran harga dari order atau pesanan yang telah diinputkan oleh admin. Waktu penawaran ini menjadi indikator terakhir dari 2 indikator sebelumnya, yang akan menjadi penilaian oleh admin dalam menentukan supplier terbaik yang akan dipilih untuk memenuhi pesanan dari barang yang telah dipesan oleh admin. Pembobotan indikator berdasarkan waktu penawaran ini diuraikan sebagai berikut, untuk waktu penawaran sampai dengan batas 3 jam dari order yang diinputkan admin, maka akan mendapatkan bobot penilaian 100 poin. Waktu Penawaran mulai dari 4 jam sampai dengan 6 jam setelah admin input order akan mendapatkan bobot penilaian sebesar 75 poin. Waktu penawaran mulai dari 5 jam sampai dengan 9 jam setelah admin input order akan diberikan bobot penilaian sebesar 50 poin dan lebih dari 9 jam akan mendapatkan bobot penilaian 25 poin.

2.1.4.3 Jurnal Penelitian Terdahulu

Menurut Heri Sulistyio dengan jurnalnya yang berjudul “ Sistem Pendukung Keputusan Untuk Menentukan Penerima

Beasiswa di SMA Negeri 6 Pandeglang Menggunakan Algoritma FMADM (2012 : p2), Heri Sulistyو mengemukakan bahwa untuk mengajukan permohonan beasiswa di SMA Negeri 6 Pandeglang siswa harus mengisi formulir yang berisikan data dari siswa, seperti nilai rata-rata raport siswa, penghasilan orang tua, jumlah saudara kandung, jumlah tanggungan orang tua, dan lain-lain. Khusus untuk pengajuan beasiswa kurang mampu harus melampirkan surat keterangan tidak mampu dari kelurahan. Data-data tersebut yang akan dijadikan pertimbangan untuk menentukan penerima beasiswa. Dari permasalahan diatas dapat diambil alternatif solusi yaitu dengan cara membangun sistem pendukung keputusan dengan menggunakan metode *Fuzzy Multiple Attribute Decision Making* (FMADM) dengan metode *Simple Additive Weighting* (SAW) untuk menentukan penerima beasiswa di SMA Negeri 6 Pandeglang yang diharapkan dapat menyelesaikan permasalahan yang ada dipenyialuran penerima beasiswa di SMA Negeri 6 Pandeglang. Dalam metode *Simple Additive Weighting* (SAW) dikelompokkan dalam kriteria-kriteria seperti nilai rata-rata raport siswa, penghasilan orang tua, jumlah saudara kandung, jumlah tanggungan orang tua, dan lain-lain, kemudian diterjemahkan dari bilangan *fuzzy* kedalam bentuk bilangan *crisp* sehingga nilainya akan bisa dilakukan proses perhitungan untuk mencari alternatif terbaik.

2.1.5 Analisis dan Perancangan Berorientasi Objek (*Object Oriented Programming*)

Analisa berorientasi objek (*object oriented programming*) adalah cara baru dalam memikirkan sebuah masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan penggabungan antar struktur data dan perilaku dalam sebuah entitas. Analisa berorientasi objek dimulai dengan

menyatakan suatu masalah, analisis menggambarkan model situasi dari dunia nyata, menggambarkan sifat yang penting.

Model analisa adalah abstraksi yang ringkas dan tepat dari apa yang harus dilakukan oleh sistem dan bagaimana cara melakukannya. Dalam pemrograman yang berorientasi objek, objek-objek diciptakan yang tidak hanya mencakup kode tentang data, namun juga instruksi tentang operasi yang harus ditampilkan atasnya. **Adi Nugroho** (2008 : 83)

Komponen utama pada analisa berorientasi objek adalah :

- Kelas yaitu definisi abstrak dari sebuah objek, dimana dijelaskan bahwa struktur dan perilaku dari tiap objek tergabung dalam satu kelas.
Sholih (2010 : 37)
- Objek: Merepresentasikan sebuah entitas, baik secara fisik, konsep ataupun secara software.
- Atribut: Nama-nama property dari sebuah kelas yang menjelaskan batasan nilainya dari properti yang dimiliki oleh sebuah kelas tersebut.

2.1.6 Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan sistem arsitektur yang bekerja dalam OOAD (*Object-Oriented Analysis/Design*) dengan satu bahasa yang konsisten untuk menentukan, visualisasi, mengkonstruksi, dan mendokumentasikan *artifact* (sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa *software*, dapat berupa model, deskripsi, atau *software*) yang terdapat dalam sistem *software*. UML merupakan bahasa pemodelan yang paling sukses dari tiga metode OO yang telah ada sebelumnya, yaitu *Booch*, *OMT* (*Object Modeling Technique*), dan *OOSE* (*Object-Oriented Software Engineering*). UML merupakan kesatuan dari ketiga pemodelan tersebut dan ditambah kemampuan lebih karena mengandung metode tambahan untuk mengatasi masalah pemodelan yang tidak dapat ditangani ketiga metode

tersebut. UML dikeluarkan oleh OMG (*Object Management Group, Inc*) yaitu organisasi internasional yang dibentuk pada 1989, terdiri dari perusahaan sistem informasi, *software developer*, dan para *user* sistem komputer.

Dengan adanya UML, diharapkan dapat mengurangi kekacauan dalam bahasa pemodelan yang selama ini terjadi dalam lingkungan industri. UML diharapkan juga dapat menjawab masalah penotasian dan mekanisme tukar menukar model yang terjadi selama ini. **Yuni Sugiarti** (2013 : 15)

Tujuan UML diantaranya adalah :

1. Memberikan model yang siap pakai, bahasa pemodelan *visual* yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.

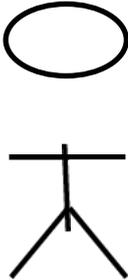
2.1.7 *Use case diagram*

Use case diagram adalah rangkaian/uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor.

Berikut pengertian *use case diagram* menurut **Yuni Sugiarti** (2013 : 20) "*Use Case Diagram* merupakan rangkaian tindakan yang dilakukan oleh sistem, aktor mewakili user atau sistem lain yang berinteraksi dengan sistem yang dimodelkan".

Tabel 2.2 Tipe Relasi pada *Use Case Diagram*

Simbol	Penjelasan
--------	------------

	<p><i>Actor</i></p> <p>Menspesifikasikan seperangkat peranan yang user sistem dapat diperankan ketika berinteraksi dengan <i>use case</i>.</p>
	<p><i>Association</i></p> <p>Menggambarkan interaksi antara <i>actor</i> dan <i>use case</i>.</p>
	<p><i>Generalization</i></p> <p>Relasi antar <i>use case</i>, dimana salah satunya dalam bentuk yang lebih umum dari yang lain.</p>
	<p><i>Use Case</i></p> <p>Sebuah deskripsi dari seperangkat aksi-aksi berurutan yang ditampilkan pada sebuah sistem.</p>
	<p><i>System</i></p> <p>Tempat seluruh aktivitas-aktivitas sistem yang sedang berjalan.</p>
	<p><i>Dependency</i></p> <p>Untuk menggambarkan ketergantungan sebuah <i>use case</i> dengan <i>use case</i> lainnya.</p>
<p><<Include>></p> 	<p><i>Include</i></p> <p>Menggambarkan bahwa keseluruhan dari sebuah <i>use case</i> merupakan fungsionalitas <i>use case</i> lainnya.</p>
<p><<Extend>></p> 	<p><i>Extend</i></p> <p>Menggambarkan hubungan antar <i>use case</i> dimana bahwa sebuah <i>use case</i> merupakan fungsionalitas <i>use case</i> lainnya apabila kondisi</p>

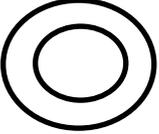
	tertentu terpenuhi.
--	---------------------

2.1.8 Activity Diagram

Diagram aktivitas adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. Diagram aktivitas mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah diagram aktivitas bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. Sebuah *activity diagram* memiliki : Yuni Sugiarti (2013 : 23)

Tabel 2.3 Tipe Relasi pada Activity Diagram

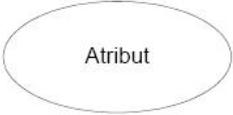
Simbol	Penjelasan
	<i>Initial State</i> Mempresentasikan dimulainya alur kerja suatu sistem dalam <i>activity diagram</i> .
	<i>Action State</i> Sebuah <i>state</i> yang menggambarkan eksekusi dari aksi <i>atomic</i> .
	<i>Transition Between Activities</i> Mengidentifikasi bahwa suatu objek dari state pertama akan menampilkan aksi-aksi tertentu dan memasuki state kedua ketika peristiwa terjadi pergerakan dari aksi ke aksi lainnya.

	<p><i>Decision Point</i></p> <p>Menentukan kapan alur dalam aktivitas menjadi bercabang.</p>
	<p><i>Final State</i></p> <p>Mempresentasikan bahwa telah diakhirinya alur suatu sistem dalam <i>activity diagram</i>.</p>

2.1.9 ERD (Entity Relationship Diagram)

Menurut Satzinger (2012 : 37), ERD adalah model konseptual yang mendeskripsikan hubungan antara penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data. Dengan ERD, model dapat diuji dengan mengabaikan proses yang dilakukan. ERD pertama kali dideskripsikan oleh Peter Chen yang dibuat sebagai bagian dari perangkat lunak CASE. Notasi yang digunakan dalam ERD dapat dilihat pada Tabel di bawah ini :

Tabel 2.4 Tabel Notasi ERD

Notasi	Keterangan
	Entitas , adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	Relasi , menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
	Atribut , berfungsi mendeskripsikan karakter entitas (atribut yg berfungsi sebagai key diberi garis bawah)
	Garis , sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

A. Kardinalitas Relasi

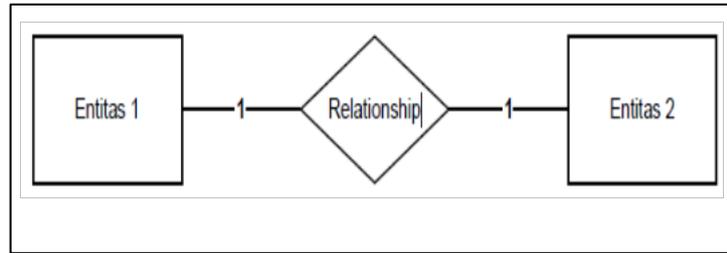
Dalam ERD hubungan (relasi) dapat terdiri dari sejumlah entitas yang disebut dengan derajat relasi. Derajat relasi maksimum disebut dengan kardinalitas, sedangkan derajat minimum disebut dengan modalitas. Jadi kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas lain.

1. One to One Relationship

Hubungan antara file pertama dan file kedua adalah satu berbanding satu.

Contoh :

- pada pengajaran private satu guru satu siswa.
- seorang guru mengajar seorang siswa, seorang siswa diajar oleh seorang guru.



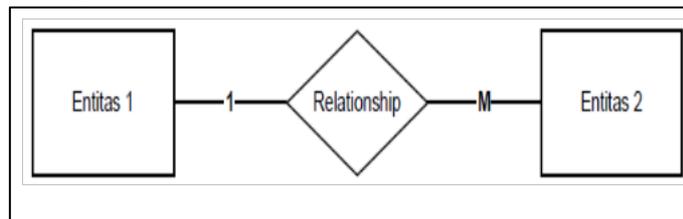
Gambar 2.1 Gambar One toOne Relationship

2. One to Many atau Many to One Relationship

Hubungan antara file pertama dan file kedua adalah satu berbanding banyak atau banyak berbanding satu.

Contoh :

- Dalam suatu perusahaan satu bagian mempekerjakan banyak pegawai.
- Satu bagian mempekerjakan banyak pegawai, satu pegawai kerja dalam satu bagian.



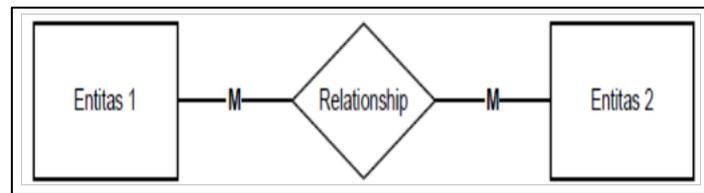
Gambar 2.2 One to Many Relationship

3. Many to Many Relationship

Hubungan file pertama dan file kedua adalah banyak berbanding banyak.

Contoh :

- Dalam universitas seorang mahasiswa dapat mengambil banyak matakuliah
- Satu mahasiswa mengambil banyak matakuliah dan satu matakuliah diambil banyak mahasiswa.



Gambar 2.3 Many to Many Relationship

2.1.10 DFD (Data Flow Diagram)

Pengertian Data Flow Diagram (DFD) menurut Jogiyanto Hartono adalah :

“Diagram yang menggunakan notasi simbol untuk menggambarkan arus data system”. (Jogiyanto Hartono, 2015, 701).

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem yang baru yang akan dikembangkan secara logika dan menjelaskan arus data dari mulai pemasukan sampai dengan keluaran data tingkatan diagram arus data mulai dari diagram konteks yang menjelaskan secara umum suatu system atau batasan system dari level 0 dikembangkan menjadi level 1 sampai system tergambar secara rinci. Gambaran ini tidak tergantung pada perangkat keras, perangkat lunak, struktur data atau organisasi file.

Menurut Jogiyanto Hartono, tahun 2015 dalam bukunya Basia Data ada beberapa simbol digunakan pada DFD untuk mewakili :

1. Kesatuan Luar (External Entity)

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

2. Arus Data (Data Flow)

Arus Data (data flow) di DFD diberi simbol suatu panah. Arus data ini

mengalir di antara 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 (Process)

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. Simpanan Data (Data Store)

Data Store merupakan simpanan dari data yang dapat berupa suatu file atau database pada sistem komputer.

2.1.11 SDLC (*System Development Life Cycle*)

Menurut **Kenneth, E. Kendall, Julie E. Kendall**(2011 : 7) **System Development Life Cycle** (SDLC) adalah fase pendekatan untuk analisa dan desain untuk sistem yang dikembangkan melalui daur tertentu dari analisis dan aktifitas pengguna (*user*). Siklus kehidupan klasik (*The Classiclife Cycle*) merupakan salah satu metode penerapan dari SDLC. Metode perancangan SDLC (*System Development Life Cycle*).

- a. Perencanaan yaitu menyangkut estimasi dari kebutuhan – kebutuhan fisik tenaga kerja dan dana yang dibutuhkan untuk mendukung pengembangan sistem yang telah diterapkan.
- b. Analisis yaitu menganalisa sistem yang telah berjalan, kemudian dilakukan perbaikan dalam sistem yang baru.
- c. Desain dan pemrograman yaitu Tahapan untuk merancang dan membuat desain berdasarkan hasil rancangan yang ada. *Tool* yang digunakan dalam perancangan adalah *Xampp, Macromedia Dreamweaver* dan *IBM Rational Rose Enterprise Edition*.
- d. Implementasi yaitu tahapan pengujian aplikasi apakah berjalan

- dengan maksimal sesuai dengan rancangan yang dibuat.
- e. Testing yaitu tahapan melakukan test terhadap sistem yang telah dibuat.

2.1.12 Internet

Menurut **Kadir** (2009 : 98), Internet (*Interconnection Networking*) sebenarnya merupakan contoh sebuah jaringan komputer. Jaringan ini menghubungkan jutaan komputer yang tersebar di seluruh dunia. Internet banyak memberikan keuntungan pada pemakai. Keuntungan pertama yang diperoleh melalui internet adalah kemudahan dalam memperoleh informasi.

Menurut sejarahnya, Internet lahir pada era 60-an atau tepatnya tahun 1969. Internet berawal dari proyek riset yang disponsori oleh DARPA (*Defence Advance Projects Agency*). Riset ini bertujuan untuk mengembangkan suatu jaringan komputer yang :

- Bekerja secara transparan, melalui bermacam – macam jaringan komunikasi data yang terhubung satu dengan lainnya.
- Tahan terhadap gangguan (bencana alam, serangan, nuklir, dan lain – lain).

Pengembangan jaringan ini ternyata sukses dan melahirkan ARPANET .Jadi, ARPANET dapat dikatakan sebagai jaringan computer yang pertama didunia. ARPANET dibangun pada akhir tahun 1969 dan awal tahun 1970. Pada waktu itu, Universitas UCLA menjadi *node* atau titik pertama pada jaringan ARPANET. Sedangkan node kedua adalah Universitas Stanford. Perangkat yang digunakan untuk menghubungkan kedua *node* tersebut disebut IMP (*Interface Message Processor*). Baik Stanford maupun UCLA memiliki perangkat IMP. Untuk menghubungkan kedua universitas tersebut digunakan saluran telepon yang dibangun oleh AT&T.

Tahun 1972, ARPANET didemonstrasikan didepan peserta *The First International Conference on Computer Communicatons* dengan

menghubungkan 40 node. Kemudian pada tahun 1990, ARPANET diubah menjadi internet. Secara singkat, sejarah perkembangan ARPANET hingga kemunculan internet sebagai berikut : **Iwan Sofana (2009,p44)**

- Tahun 1957, DoD membentuk ARPA (*Advanced Research Projects Agency*).
- Tahun 1969, ARPANET terbentuk.
- Tahun 1970 ARPANET menghubungkan UCLA, UCSB, U-Utah & Stanford.
- Tahun 1973, ARPANET menghubungkan *University College London* dan *Royal University Norway*.
- Tahun 1982, model jaringan internet dikembangkan.
- Tahun 1990, ARPANET kemudian dikenal dengan internet.

Tabel 2.5. Tipe Domain

Tipe Domain	Keterangan
com atau co	Perusahaan komersial
edu atau ac	Institusi pendidikan
gov atau go	Biro atau badan pemerintah
mil	Militer
net	Penyedia jaringan
org atau or	Organisasi lain-lain

2.1.13 Spesifikasi Basis Data atau Database

Secara umum, database berarti koleksi data yang saling terkait. Secara praktis, basis data dapat dianggap sebagai suatu penyusunan data yang terstruktur yang disimpan dalam media pengingat (*hard disk*) yang tujuannya adalah agar data tersebut dapat diakses dengan mudah dan cepat.

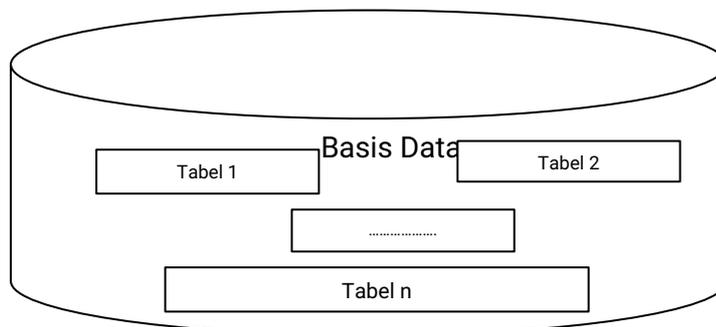
Sesungguhnya ada beberapa macam database, antara lain yaitu *database* hierarkis, *database* jaringan, dan database relasional. *Database* relasional merupakan *database* yang populer saat ini dan telah diterapkan

pada berbagai platform dari PC hingga *mini computer*.

Sebuah *database* relasional tersusun atas sejumlah tabel. Sebagai contoh *database* akademis mencakup tabel – tabel seperti dosen, mahasiswa, KRS, nilai, dan lain – lain. Basis data tentang bintang film bisa mencakup info pribadi (nama, jenis kelamin, tanggal lahir, dan sebagainya) dan film – film yang pernah dibintangi. **Jogiyanto** (2008 : 25)

2.1.14 Istilah Tabel, Baris, dan Kolom

Dalam terminologi *database* relasional, dikenal istilah seperti tabel, baris, dan kolom. Tabel (biasa juga disebut relasi) menyatakan bentuk berdimensi dua yang mewakili suatu kelompok data sejenis. Sebagai contoh, pada gambar dibawah ini memperlihatkan keberadaan tiga buah tabel.



Gambar 2.4 Sebuah *database* tersusun atas sejumlah table

Sebuah table berisi sejumlah kolom. Sebagai contoh, pada table yang berisi informasi pribadi masing – masing bintang film terdapat data dengan nama *id_bin*, *nama*, *tgl_lahir*, dan *sex*. Yang secara berturut – turut menyatakan data kode atau identitas bintang film, nama bintang film, tanggal lahir, dan jenis kelamin. Masing – masing data tersebut disebut sebagai kolom atau *field*.

Secara individual, data *id_bin*, *nama*, *tgl_lahir*, dan *sex* adalah nama – nama kolom. Sebaiknya pasangan data yang mencakup kode bintang film, nama bintang film, tanggal lahir dan jenis kelamin disebut baris atau *record*. **Kadir** (2009 : 69)