

BAB II

LANDASAN TEORI

2.1 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan manipulasi data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat (Kusrini, 2007).

SPK adalah sekumpulan prosedur berbasis model untuk data pemrosesan dan penilaian, guna membantu para manajer mengambil keputusan. Little mempunyai argumen bahwa untuk berhasil maka sistem harus mudah, kuat, mudah dikontrol, mampu menyesuaikan diri, lengkap pada persoalan penting, dan mudah dikomunikasikan (Kusumadewi, 2006).

2.2 Metode *Preference Ranking Organization For Enrichment Evaluation* (PROMETHEE)

Promethee adalah suatu metode penentuan urutan (prioritas) dalam analisis multikriteria. Masalah adalah kesederhanaan, kejelasan, dan kestabilan. Dugaan dari dominasi kriteria yang digunakan dalam promethee adalah penggunaan nilai dalam hubungan *outranking* (Brans, 1998). Ini adalah metode peringkat yang cukup sederhana dalam konsep dan aplikasi dibandingkan dengan metode lain untuk analisis multikriteria (Goumas, 1998).

Prinsip yang digunakan adalah penetapan prioritas alternatif yang telah ditetapkan berdasarkan pertimbangan $(\forall i | f_i(.) \rightarrow R$ [real world], dengan kaidah dasar $\text{Max} \{ f_1(x), f_2(x), f_3(x), \dots, f_j(x), \dots, f_k(x) | x \in R \}$

Dimana K adalah sejumlah kumpulan alternatif, dan f_i ($i = 1, 2, \dots, K$) merupakan nilai/ukuran relatif kriteria untuk masing-masing alternatif. Dalam aplikasinya sejumlah kriteria telah ditetapkan untuk menjelaskan K yang merupakan penilaian dari R (real world). Promethee termasuk dalam keluarga dari metode *outranking* yang dikembangkan oleh B. Roy, dan meliputi dua fase:

- Membangun hubungan *outranking* dari K
- Eksploitasi dari hubungan ini memberikan jawaban optimasi kriteria dalam paradigma permasalahan multikriteria.

Dalam fase pertama, nilai hubungan *outranking* berdasarkan pertimbangan dominasi masing-masing kriteria. Indeks preferensi ditentukan dan nilai *outranking* secara grafis disajikan berdasarkan preferensi dari pembuat keputusan. Data dasar untuk evaluasi dengan metode Promethee disajikan sebagai berikut :

Tabel 2. 1 Data Dasar Analisis Promethee

	$f_1(.)$	$f_2(.)$...	$f_j(.)$...	$f_k(.)$
a_1	$f_1(a_1)$	$f_2(a_1)$...	$f_j(a_1)$...	$f_k(a_1)$
a_2	$f_1(a_2)$	$f_2(a_2)$...	$f_j(a_2)$...	$f_k(a_2)$
...
a_i	$f_1(a_i)$	$f_2(a_i)$...	$f_j(a_i)$...	$f_k(a_i)$
...
a_n	$f_1(a_n)$	$f_2(a_n)$...	$f_j(a_n)$...	$f_k(a_n)$

2.2.1. Nilai Hubungan Outranking dalam PROMETHEE

2.2.1.1. Dominasi Kriteria

Nilai f merupakan nilai nyata dari suatu kriteria : $f : K \rightarrow R$ dan tujuan berupa prosedur optimasi untuk setiap alternatif $a \in K$, $f(a)$ merupakan evaluasi dari alternatif tersebut untuk suatu kriteria. Pada saat dua alternatif dibandingkan, $a, b \in K$, harus dapat ditentukan perbandingan preferensinya. Penyampaian intensitas (P) dari preferensi alternatif a terhadap alternatif b sedemikian rupa sehingga :

- $P(a,b) = 0$, berarti tidak ada (indifferent) antara a dan b , atau tidak ada preferensi dari a lebih baik dari b .
- $P(a,b) \sim 0$, berarti lemah preferensi dari a lebih baik dari b .
- $P(a,b) \sim 1$, berarti kuat preferensi dari a lebih baik dari b .
- $P(a,b) = 1$, berarti mutlak preferensi dari a lebih baik dari b .

Dalam metode ini, fungsi preferensi seringkali menghasilkan nilai fungsi yang berbeda antara dua evaluasi, sehingga : $P(a,b) = P(f(a)-f(b))$. Untuk semua kriteria, suatu alternatif akan dipertimbangkan memiliki nilai kriteria yang lebih baik ditentukan oleh nilai f dan akumulasi dari nilai ini menentukan nilai preferensi atas masing-masing alternatif yang akan dipilih.

2.2.1.2. Rekomendasi Fungsi Preferensi Untuk Keperluan Aplikasi

Dalam Promethee disajikan enam bentuk fungsi preferensi kriteria. Hal ini tentu saja tidak mutlak, tetapi bentuk ini cukup baik untuk beberapa kasus. Untuk memberikan gambaran yang lebih baik terhadap area yang tidak sama, digunakan fungsi selisih nilai kriteria antara alternatif $H(d)$

dimana hal ini mempunyai hubungan langsung pada fungsi preferensi (Brans, 1998).

- **Kriteria Biasa (*Usual Criterion*)**

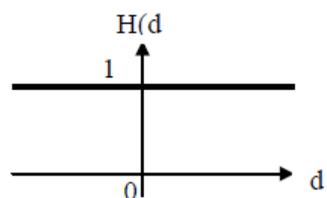
$$H(d) = \begin{cases} 0 & \text{jika } d \leq 0 \\ 1 & \text{jika } d > 0 \end{cases} \dots\dots\dots(1)$$

Keterangan :

$H(d)$: selisih kriteria antar alternatif

d : selisih nilai kriteria $\{d=f(a) - f(b)\}$

Pada kasus ini, tidak ada beda (sama penting) antara a dan b jika dan hanya jika $f(a) = f(b)$; apabila nilai kriteria pada masing-masing alternatif memiliki nilai berbeda, pembuat keputusan membuat preferensi mutlak untuk alternatif memiliki nilai yang lebih baik. Untuk melihat kasus preferensi pada kriteria biasa, ilustrasinya dapat dilihat dari perlombaan lari maraton, seorang peserta dengan peserta lain akan memiliki peringkat yang mutlak berbeda walaupun hanya dengan selisih nilai (waktu) yang teramat kecil, dan dia akan memiliki peringkat yang sama jika dan hanya jika waktu tempuhnya sama atau selisih nilai diantara keduanya sebesar nol. Fungsi $H(d)$ untuk fungsi preferensi ini disajikan pada gambar berikut:



Gambar 2. 1 Kriteria Biasa

- **Kriteria Quasi (*Quasi Criterion*)**

H(d) : 0 jika $d \leq q$

H(d) : 1 jika $d > q$ (2)

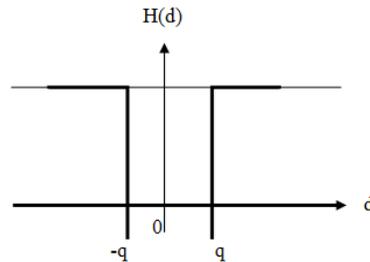
Keterangan:

H(d) : fungsi selisih kriteria

d : selisih nilai kriteria { $d = f(a) - f(b)$ }

Parameter (q) : harus merupakan nilai yang tetap

Seperti yang terlihat pada Gambar 2.2, dua alternatif memiliki preferensi yang sama penting selama selisih atau nilai H(d) dari masing-masing alternatif untuk kriteria tertentu tidak melebihi nilai q, dan apabila selisih hasil evaluasi untuk masing-masing alternatif melebihi nilai q maka terjadi bentuk preferensi mutlak.



Gambar 2. 2 Kriteria Quasi

- **Kriteria Dengan Preferensi Linier**

$$\mathbf{H(d)} = \begin{cases} 0 & \text{jika } d \leq 0 \\ \frac{d}{p} & \text{jika } 0 < d \leq p \\ 1 & \text{jika } d > p \end{cases} \dots\dots\dots (3)$$

Keterangan:

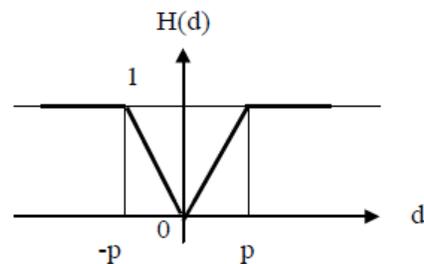
H(d) : fungsi selisih kriteria antar alternatif

d : selisih nilai kriteria { $d = f(a) - f(b)$ }

p : nilai kecenderungan atas

Kriteria preferensi linier dapat menjelaskan bahwa selama nilai selisih memiliki nilai yang lebih rendah dari p , preferensi dari pembuat keputusan meningkat secara linier dengan nilai d . Jika nilai d lebih besar dibandingkan dengan nilai p , maka terjadi preferensi mutlak (Brans, 1998). Fungsi kriteria ini disajikan pada Gambar 2.3.

Pada saat pembuat keputusan mengidentifikasi beberapa kriteria untuk tipe ini, dia harus menentukan nilai dari kecenderungan atas (nilai p). Dalam hal ini nilai d diatas p telah dipertimbangkan akan memberikan preferensi mutlak dari suatu alternatif.



Gambar 2. 3 Kriteria Dengan Preferensi Linier

- **Kriteria Level (Level Criterion)**

$$H(d) = \begin{cases} 0 & \text{jika } d \leq q \\ 0,5 & \text{jika } q < d \leq p \dots\dots\dots (4) \\ 1 & \text{jika } d > p \end{cases}$$

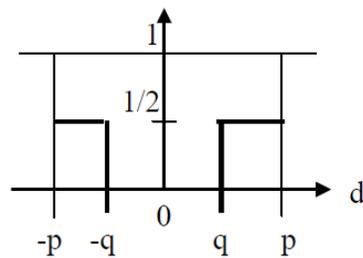
Keterangan :

$H(d)$: fungsi selisih kriteria antar alternatif

p : nilai kecenderungan atas

parameter (q) : harus merupakan nilai yang tetap

Dalam kasus ini, kecenderungan tidak berbeda q dan kecenderungan preferensi p adalah ditentukan secara simultan. Jika d berada di antara nilai q dan p, hal ini berarti situasi preferensi yang lemah ($H(d)=0.5$) (Brans, 1998) . Fungsi ini disajikan pada Gambar 2.4. dan pembuat keputusan telah menentukan kedua kecenderungan untuk kriteria ini.



Gambar 2. 4 Kriteria Level

- **Kriteria Dengan Preferensi Linier Dan Area Yang Tidak Berbeda**

$$H(d) = \begin{cases} 0 & \text{jika } d \leq q \\ (d - q) / (p - q) & \text{jika } q < d \leq p \\ 1 & \text{jika } d > p \end{cases} \dots\dots\dots(5)$$

Keterangan:

$H(d)$: fungsi selisih kriteria antara alternatif

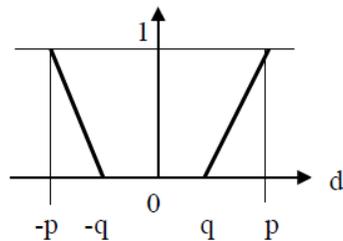
d : selisih nilai kriteria $\{d = f(a) - f(b)\}$

parameter (p) : nilai kecenderungan atas

parameter (q) : harus merupakan nilai yang tetap

Pada kasus ini, pengambil keputusan mempertimbangkan peningkatan preferensi secara linier dari tidak berbeda hingga preferensi mutlak dalam area antara dua kecenderungan q dan p [1].

Dua parameter tersebut telah ditentukan. Fungsi H selanjutnya disajikan pada Gambar 2.5

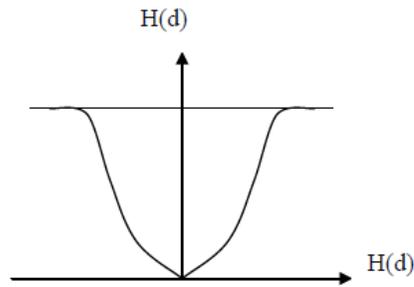


Gambar 2. 5 Kriteria Dengan Preferensi Linier Dan Area Yang Tidak Berbeda

- **Kriteria Gaussian (*Gaussian Criterion*)**

$$H(d) = \begin{cases} 0 & \text{jika } d \leq 0 \\ 1 - \exp\left(-\frac{d^2}{2g^2}\right) & \text{jika } d > 0 \end{cases} \dots\dots\dots (6)$$

Fungsi ini disajikan dalam Gambar 2.5 Fungsi ini bersyarat apabila telah ditentukan nilai σ , dimana dapat dibuat berdasarkan distribusi normal dalam statistik (Brans, 1998).



Gambar 2. 6 Kriteria Gaussian

2.2.1.3. Indeks Preferensi Multikriteria

Tujuan pembuat keputusan adalah menetapkan fungsi preferensi P_i , dan π_i untuk semua kriteria f_i ($i = 1, \dots, k$) dari masalah optimasi kriteria majemuk. Bobot (*weight*) π_i merupakan ukuran relatif dari kepentingan kriteria f_i ; jika semua kriteria memiliki nilai kepentingan yang sama dalam pengambilan keputusan maka semua nilai bobot adalah sama. Indeks preferensi multikriteria (ditentukan berdasarkan rata-rata bobot dari fungsi preferensi P_i)

$$\delta(a, b) = \sum_{i=1}^n \pi_i P_i(a, b): \forall a, b \in A \quad \dots\dots\dots(7)$$

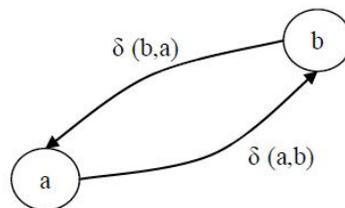
$\delta(a, b)$ merupakan intensitas preferensi pembuat keputusan yang menyatakan bahwa alternatif a lebih baik dari alternatif b dengan pertimbangan secara simultan dari seluruh kriteria.

Hal ini dapat disajikan dengan nilai antara 0 dan 1, dengan ketentuan sebagai berikut :

- $\delta(a, b) = 0$, menunjukkan preferensi yang lemah untuk alternatif a lebih dari alternatif b berdasarkan semua kriteria.

- $\delta(a, b) = 1$, menunjukkan preferensi yang kuat untuk alternatif a lebih dari alternatif b berdasarkan semua kriteria.

Indeks preferensi ditentukan berdasarkan nilai hubungan *outranking* pada sejumlah kriteria dari masing-masing alternatif. Hubungan ini dapat disajikan sebagai grafik nilai *outranking*, node-nodenya merupakan alternatif berdasarkan penilaian kriteria tertentu. Diantara dua node (alternatif) a dan b, merupakan garis lengkung yang mempunyai nilai $\delta(b, a)$, dan $\delta(a, b)$ (tidak ada hubungan khusus antara $\delta(b, a)$, dan $\delta(a, b)$). Hal ini dapat dilihat pada gambar dibawah



Gambar 2. 7 Hubungan Antara Node

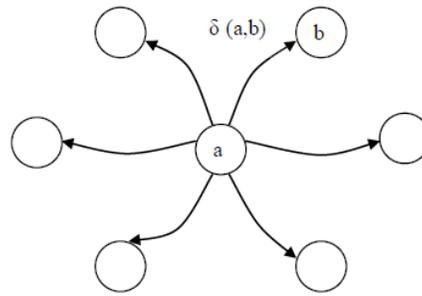
2.2.2. Promethee Ranking

2.2.2.1. Arah Dalam Grafik Nilai Outranking

Untuk setiap node a dalam grafik nilai outranking ditentukan berdasarkan *leaving flow*, dengan persamaan :

$$\Phi^+(a) = \frac{1}{n-1} \sum_{x \in G} \delta(A, x) \dots\dots\dots(8)$$

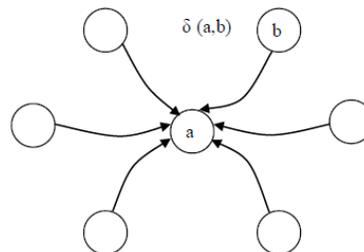
Dimana $\delta(a, x)$ menunjukkan preferensi bahwa alternatif a lebih baik dari alternatif x. *Leaving flow* adalah jumlah dari nilai garis lengkung yang memiliki arah menjauh dari node a dan hal ini merupakan karakter pengukuran *outranking* (gambar dibawah)

Gambar 2. 8 *Leaving Flow*

Secara simetris dapat ditentukan *entering flow* dengan persamaan :

$$\Phi^- (a) = \frac{1}{n-1} \sum_{x \in Q} \delta(A, x) \dots\dots\dots (9)$$

Entering flow diukur berdasarkan karakter *outranked* dari a (Gambar dibawah)

Gambar 2. 9 *Entering Flow*

Sehingga pertimbangan dalam penentuan *net flow* diperoleh dengan persamaan :

$$\Phi (a) = \Phi^+ (a) - \Phi^- (a) \dots\dots\dots(10)$$

Penjelasan dari hubungan *outranking* dibangun atas pertimbangan untuk masing-masing alternatif pada grafik nilai *outranking*, berupa urutan parsial (Promethee I) atau urutan lengkap (Promethee II) pada sejumlah alternatif.

1. Promethee I

Nilai terbesar pada *leaving flow* dan nilai yang kecil dari *entering flow* merupakan alternatif yang terbaik. *Leaving flow* dan *entering flow* menyebabkan:

$$aP^+b \text{ jika } \Phi^+(a) > \Phi^+(b)$$

$$aI^+b \text{ jika } \Phi^+(a) = \Phi^+(b)$$

$$aP^-b \text{ jika } \Phi^-(a) < \Phi^-(b)$$

$$aI^-b \text{ jika } \Phi^-(a) = \Phi^-(b)$$

Promethee I menampilkan partial preorder (PI, II, RI) dengan mempertimbangkan interaksi dari dua preorder:

$$\left\{ \begin{array}{ll} a P_1 b \text{ (a outrank b)} & \text{jika } a P^+ b \text{ dan } a P^- b \\ & \text{atau } a P^+ b \text{ dan } a I^- b \\ & \text{atau } a I^+ b \text{ dan } a P^- b \\ a I_1 b \text{ (a tidak beda dengan b)} & \text{jika } a I^+ b \text{ dan } a I^- b \\ a R_1 b \text{ (a dan b incomparable)} & \text{jika pasangan lain} \end{array} \right.$$

2. Promethee II

Dalam kasus *complete preorder* dalam K adalah penghindaran dari bentuk incomparable, Promethee II *complete preorder* (PII, III) disajikan dalam bentuk *net flow* disajikan berdasarkan pertimbangan persamaan :
 $aPIIb$ jika $(a) > \Phi(b)$ $aPIIb$ jika $(a) = \Phi(b)$ Melalui *complete preorder*, informasi bagi pembuat keputusan lebih realistik. Terdapat beberapa kelebihan dari metode Promethee yaitu :

- ✓ Lebih jelas dan lebih sederhana / mudah dipahami oleh para praktisi.
- ✓ Memperhitungkan data kualitatif sebaik data kuantitatif.
- ✓ Menyediakan enam tipe preferensi terhadap kriteria.

- ✓ Memperhitungkan kriteria berbeda pada saat yang sama, yang tidak mungkin dengan keputusan berbasis proses yang didasarkan hanya pada satu kriteria.
- ✓ Dapat menggunakan kriteria yang berbeda untuk setiap dimensi.
- ✓ Perangkingan alternatif dapat dilakukan secara parsial maupun lengkap.

Disamping kelebihan diatas terdapat juga beberapa kekurangan dari metode Promethee yaitu :

- Membutuhkan informasi tambahan berupa fungsi preferensi tertentu yang harus didefinisikan / dijelaskan.
- Tidak mampu menangani masalah optimasi terhadap kendala yang sangat mungkin ada dalam permasalahan pemilihan alternatif optimal.

2.3 Perancangan Sistem

Perancangan sistem adalah suatu fase dimana diperlukan suatu keahlian perencanaan untuk elemen-elemen komputer yang akan menggunakan sistem baru. Ada 2 hal yang perlu diperhatikan dalam perancangan sistem yang baru. Dalam pembuatan sistem pendukung keputusan pemberian kredit ini, penulis menggunakan alat bantu yang digunakan dalam perancangan sistem yaitu Diagram Kontek (*Context Diagram*), DFD (*Data Flow Diagram*), dan ERD (*Entity Relationship Diagram*)

2.3.1. Context Diagram (CD)

Context Diagram adalah sebuah diagram sederhana yang menggambarkan hubungan antara entity luar, masukan dan keluaran dari sistem. Diagram konteks dipresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem.

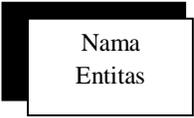
2.3.2. Data Flow Diagram (DFD)

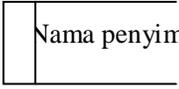
DFD adalah diagram yang menggunakan simbol-simbol untuk menyajikan entitas, proses, arus data, dan penyimpanan data yang berkaitan dengan suatu sistem. (James A. Hall, 2007).

DFD adalah penyajian grafis dari sebuah sistem yang mempergunakan empat bentuk simbol untuk mengilustrasikan bagaimana data mengalir melalui proses-proses yang saling tersambung.

Simbol-simbol tersebut mencerminkan (1) unsur-unsur lingkungan dengan mana sistem berinteraksi, (2) proses, (3) arus data, dan (4) penyimpanan data. (Raymond McLeod, Jr, George P. Schell, 2008)

Tabel 2. 2 Simbol-Simbol DFD

Simbol	Deskripsi
 <p>Nama Entitas</p>	Sumber input atau output data
 <p>N Deskripsi Proses</p>	Proses yang dipicu atau didukung oleh data

Simbol	Deskripsi
 Nama penyimpanan data	Penyimpanan data seperti file transaksi, file utama, atau file referensi
	Arah arus data

2.3.3. Entity Relationship Diagram (ERD)

ERD adalah teknik dokumentasi yang digunakan untuk menyajikan relasi antara entitas. (James A. Hall, 2007)

2.4 Borland Delphi 7

Delphi 7 merupakan sebuah peranti pengembangan Aplikasi berbasis Windows yang dikeluarkan oleh Borland International. (Abdul Kadir, 2005).

Delphi dikembangkan oleh perusahaan perangkat lunak yang bernama Borland. Borland sudah banyak mengembangkan bahasa-bahasa pemrograman, seperti Turbo C dan Turbo Pascal. Delphi adalah aplikasi database yang memakai bahasa Pascal yang dipakai Delphi dengan Object Pascal, tetapi Delphi versi 7 menyebutnya dengan bahasa Delphi.

2.5 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache *HTTP* Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*.

Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam GNU (*General Public License*) dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

2.6 KSP (Koperasi Simpan Pinjam)

Koperasi simpan pinjam atau biasa disebut dengan koperasi kredit adalah koperasi yang bergerak dalam bidang pemupukan simpanan dari para anggotanya, untuk kemudian dipinjamkan kembali kepada anggota yang memerlukan bantuan modal (Revrison, 2000). Pelaksanaan simpan pinjam oleh koperasi dan tata cara pendiriannya telah diatur dalam Peraturan Pemerintah Nomor 9 Tahun 1995 Tentang Pelaksanaan Kegiatan Usaha Simpan Pinjam Oleh Koperasi.