BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

Pada landasan teori ini dijelaskan beberapa konsep dan dasar teori yang berkaitan dengan permasalahan yang akan dibahas sebagai dasar pemahaman dalam mengimplementasikan konsep-konsep tersebut kedalam semua kegiatan pengembangan sistem.

2.1.1 Analisis Sistem

Analisis sistem (*system analysis*) dapat didefinisikan sebagai berikut[1] :

"*Penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya*"

Tahap analisis sistem dilakukan setelah tahap perencanaan (*system planning*), dan sebelum tahap desain sistem (*system design*). Tahap analisis merupakan tahap yang sangat kritis dan sangat penting, karena kesalahan pada tahap ini akan menyebabkan kesalahan pada tahap selanjutnya.

Analsis yang dilakukan menggunakan metode analisis terstruktur, dimana metode analisis terstruktur ini banyak memakai *tool-tool* dalam mempresentasikan suatu sistem, sehingga penguraian sistem dapat terstruktur dengan baik, jelas dan lebih mudah dipahami.

2.1.2 Diagram Konteks

Diagram konteks merupakan diagram yang mengandung satu proses yang menggambarkan hubungan keterkaitan antara sistem dengan pihak-pihak diluar lingkungan sistem dan posisi sistem didalam lingkungan tersebut. Pihak-pihak tersebut merupakan pihak-pihak yang membutuhkan informasi dan data dari sistem ataupun pihak-pihak yang menjadi sumber informasi dan data bagi sistem. Hubungan keterkaitannya digambarkan sebagai aliran informasi dan data yang masuk ke dalam sistem dan keluar dari sistem.

2.1.3 Data Flow Diagram

DFD adalah sebuah teknik grafik yang menggambarkan aliran data dan transformasi yang digunakan sebagai perjalanan data dari masukan menuju ke keluaran. DFD dapat diartikan juga sebagai model jaringan dari sebuah sistem. DFD dapat menggambarkan proses-proses yang terjadi dan aliran data diantaranya.

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut disimpan. Secara umum DFD dapat diartikan sebagai salah satu *tools* untuk analisis sistem yang dapat bermanfaat untuk menggambarkan proses, aliran data, *entity* yang terlibat serta *data store* yang digunakan dalam sistem yang dipelajari. Dengan menuangkan hasil analisis ke dalam DFD, seorang analis dapat memahami sistem yang sedang dipelajari dengan mudah dan baik.

Dibawah ini terdapat simbol-simbol untuk DFD yaitu:

1. *Proses*, suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, prosedur atau alat yang digunakan untuk mentransformasikan data.
2. *Data Flow* (Arus Data), data yang mengalir dengan arah tertentu dari asal ke tujuan. Data yang mengalir dapat berupa dokumen, surat atau bentuk lainnya.
3. *Data Store* (Penyimpanan Data), digunakan untuk menyimpan dan mengambil data oleh proses. Data yang disimpan dapat berupa data yang terkomputerisasi maupun tidak terkomputerisasi.
4. *Source* atau *destination* atau dikenal juga dengan *external entity*, berupa orang, organisasi atau sistem lain yang berada diluar batas sistem yang berinteraksi dengan sistem yang sedang dikembangkan.

Dalam DFD terdapat beberapa aturan dasar penyusunan DFD yang dapat membantu untuk mempermudah penggambaran diagram arus data, diantaranya :

1. Setiap data yang dihasilkan atau keluar dari proses harus didasarkan pada data yang masuk ke proses tersebut.
2. Semua aliran data yang dihasilkan harus diberi nama, nama yang diberikan harus mencerminkan aliran data antara proses, *data store* dan *entity* yang ada.
3. Hanya data yang diperlukan untuk melakukan suatu proses saja yang harus digunakan sebagai input suatu proses.
4. Suatu proses harus tergantung pada input dan output yang masuk ke dalam proses itu saja, tidak perlu memperhatikan apa yang terjadi pada proses lainnya.
5. Setiap proses yang ada harus merupakan proses yang berjalan terus menerus, setiap proses harus diasumsikan siap menangani fungsi atau kerja setiap proses.

Agar representasi sistem dalam DFD mudah dipahami, maka DFD disusun dalam bentuk bertingkat (*leveled*) yang merupakan rincian lanjut dari proses pada level yang sebelumnya. Dimulai dari tingkat yang tertinggi dilakukan identifikasi proses apa saja yang ada dan data apa saja yang mengalir antar proses yang ada. Selanjutnya, dilakukan analisis terhadap setiap proses yang ada pada level tersebut, apakah sudah merupakan proses yang melakukan satu fungsi saja atau masih melakukan beberapa fungsi yang berbeda. Jika proses yang dianalisis masih melakukan lebih dari satu fungsi yang berbeda, maka perlu dilakukan pemecahan terhadap proses tersebut menjadi level lebih tinggi. Demikian seterusnya sampai didapat semua proses yang ada pada level paling tinggi dilakukan satu fungsi saja.

Jika DFD untuk suatu sistem yang dianalisis telah dibuat, sebelum melanjutkan kegiatan lainnya selalu dilakukan evaluasi atau pemeriksaan terhadap DFD tersebut. Pemeriksaan dilakukan terhadap munculnya kesalahan yang dapat diakibatkan oleh salah gambar, tidak konsistennya penggunaan simbol dan analisis terhadap proses yang terjadi.

2.1.4 Perancangan sistem

Perancangan sistem adalah suatu fase kreatif dalam siklus pengembangan sistem, dimana sistem yang dirancang harus memenuhi tujuan sistem secara teknis dan operasional. Pada tahap ini perlu dikembangkan spesifikasi untuk keluaran, database, operasi pemrosesan data, masukkan data, pengendalian serta pengamanan.

Perancangan sistem informasi yang dilakukan, bertujuan untuk menghilangkan beberapa kelemahan sistem dan meningkatkan kemampuan sistem dalam menyelesaikan permasalahan yang ada dengan acuan hasil-hasil analisis yang telah dilakukan.

2.1.4.1 Basis data

1. Pengertian basis data

Basis data adalah kumpulan dari file atau tabel atau arsip yang saling berhubungan disimpan dalam media penyimpanan elektronik (disket atau hardisk). Prinsip utamanya adalah pengaturan data atau arsip dan tujuan utamanya adalah kemudahan dan kecepatan dalam mengambil kembali data atau arsip.[1]

2. Sistem pengelola basis data (*Database Management System*/DBMS)

Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak (sistem) yang khusus atau spesifik. Perangkat lunak ini disebut DBMS, dimana akan menentukan bagaimana data diorganisasi, disimpan, diubah, dan diambil kembali. DBMS juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, keakuratan data dan sebagainya.

Perangkat lunak yang termasuk DBMS seperti Dbase III+, dBase IV, FoxBase, MsAccess, Borland Paradox, Ms SQL Server, Oracle Borland Interbase. Salah satu tujuan DBMS adalah untuk menyediakan fasilitas antarmuka atau interface dalam melihat data (yang lebih ramah atau user friendly) pada pemakai.

3. Bahasa basis data

DBMS merupakan perantara bagi pemakai dengan basis data dalam disk. Cara berinteraksi atau berkomunikasi anatara pemakai dengan basis data tersebut diatur dalam suatu bahasa khusus yang ditetapkan oleh perusahaan pembuat DBMS. Bahasa itu dapat disebut sebagai bahasa basis data yang terdiri atas sejumlah perintah yang diformulasikan

dan dapat diberikan ke user dan dikenali atau diproses oleh DBMS untuk melakukan suatu arsip atau pekerjaan tertentu.

Sebuah bahasa basis data dapat dipalah ke dalam 2 bentuk :

a. *Data definition language* (DDL)

   DDL adalah struktur atau skema basis data yang menggambarkan desain basis data secara keseluruhan, dispesifikasikan dengan bahasa khusus. Dengan bahasa inilah dapat dibuat tabel baru, membuat index, mengubah tabel, menentukan struktur penyimpanan tabel, dan sebagainya. Hasil dari kompilasi perintah DLL adalah kumpulan tabel yang disimpan dalam file khusus yang disebut kamus data (Data dictionary).

b. *Data manipulation language* (DML)

   Sedangkan DML merupakan bentuk bahasa basis data yang berguna untuk melakukan manipulasi dan pengambilan data pada suatu data. Manipulasi data dapat berupa :

   1. Penyisipan atau penambahan data baru ke suatu basis data.
   2. Penghapusan data dari suatu basis data
   3. Pengubahan data dari suatu basis data

   DML merupakan bahasa yang bertujuan memudahakan pemakai, untuk mengakses data sebagai mana direpresentasikan oleh model baru.

2.1.4.2 Diagram E-R (*Entity Relationship Diagram*)

ERD Merupakan notasi grafis dalam pemodelan data konseptual, yang mendeskripsikan hubungan antara penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini relatif kompleks. Dengan ERD,

model dapat diuji dengan mengabaikan proses yang harus dilakukan. ERD menggunakan sejumlag notasi dan simbol untuk menggambarkan struktur dan hubungan antar data, pada dasarnya ada 3 macam simbol yang digunakan :

a. Entity

Adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat.

b. Atribut

Entity mempunyai elemen yang disebut atribut, dan berfungsi mendeskripsikan karakter entity.

c. Hubungan

Relationship sebagai mana Entity maka dalam hubungan pun harus dibedakan dalam hubungan atau bentuk hubungan antar Entity dengan isi dari hubungan itu sendiri.

Relasi antara 2 file atau 2 tabel dapat dikategorikan menjadi 3 macam, yaitu :

1. *One to One*

   Yang berarti entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan paling banyak dengan satu entitas pada himpunan entitas A.

2. *One to Many Realtionship*

   Yang berarti entitas pada himpunan entitas A berhubungan dengan banyak entitas pada satu himpunan entitas B, tetapi tidak sebaliknya setiap entitas pada himpunan entitas B berhubungan paling banyak dengan satu entitas pada himpunan entitas A.

3. *Many to Many Relationship*

   Yang berarti entitas pada himpunan entitas A berhubungan dengan banyak entitas pada banayak himpunan entitas B dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan banyak entitas pada himpunan A.

2.1.4.3 Kamus data (*Data dictionary*)

Kamus data merupakan katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data dibuat untuk dapat digunakan pada tahap analisis sistem dan saling berhubungan pada tahap perancangan sistem yang didasarkan pada aliran data di DFD.

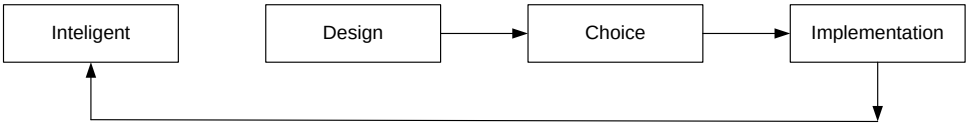
Pada tahap analisis dan perancangan sistem, kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem untuk dilakukannya proses perancangan sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan informasi yang dihubungkan oleh pemakai sistem.

2.1.5 Sistem Pendukung Keputusan (SPK)

2.1.5.1 Teori Pengambilan Keputusan

Dalam era revolusi komunikai dan informasi saat ini, banyak sekali insan di dunia mampu mengumpulkan informasi sebanyak mungkin melalui media yang dia miliki namun belum tentu mampu mengolahnya dengan baik agar informasi tersebut dapat diamamfaatkan dalam waktu yang tepat. Keputusan dapat diambil dengan tepat (waktu dan sasaran) serta dapat dipertanggungjawabkan jika didukung oleh suatu sistem dengan tanpa melakukan aspek individu pengambil keputusan itu sendiri serta konteks permasalahan yang dihadapi.[2]

Secara garis besar seorang pengambil keputusan melewati beberapa alur/proses seperti pada gambar berikut ini untuk mengambil keputusan terbaiknya.



Gambar 2.1Tahapan Pengambilan Keputusan Proses Pemilihan Alternative

Tindakan/keputusan biasanya terdiri dari langkah-langkah sebagai berikut:

a. Tahap Intelligent
b. Tahap Design
c. Tahap choice

Terdapat dua kelompok pendekatan dalam mendefinisikan sistem. Pertama, pendekatan yang menekankan pada prosedur, yang mendefinisikan sistem sebagai berikut:

> "*Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menjelaskan suatu sasaran yang tertentu dan pengambilan keputusan merupakan aktivitas manajemen berupa pemilihan tindakan dari sekumpulan alternative yang telah dirumuskan sebelumnya untuk memecahkan suatu masalah atau suatu konflik dalam manajemen*".

Sistem Pendukung Keputusan merupakan pasangan dari intelektual sumber daya manusia dengan kemampuan dari komputer untuk memperbaiki kualitas dari keputusan, yaitu sistem pendukung keputusan yang terkomputerisasi bagi pembuat keputusan manajemen yang menghadapi masalah semi terstruktur.

Sistem pendukung keputusan suatu sistem interaktif, yang membantu pengambilan keputusan melalui penggunaan data dan model-model keputusan untuk memecahkan masalah-masalah yang sifatnya semi terstruktur.

Dari beberapa definisi tersebut dapat disimpulkan bahwa sistem pendukung keputusan merupakan suatu sistem terkomputerisasi yang dirancang untuk meningkatkan efektivitas pengambilan keputusan dalam memecahkan masalah yang bersifat semi terstruktur dan tidak terstruktur.

Berikut ini merupakan proses pengambilan keputusan, yaitu:

1. Penjelasan

   Tahap ini merupakan proses penelusuran dan pendeteksian dari lingkup problematika serta pengenalan masalah. Data masukan diperoleh, diproses dan diuji dalam rangka mengidentifikasi masalah.

2. Perancangan

   Tahap ini merupakan proses menemukan, mengembangkan dan menganalisis alternatif yang bisa dilakukan. Tahap ini meliputi proses menganalisis masalah, menguji kelayakan solusi.

3. Pemilihan

   Dilakukan proses pemilihan diantara berbagai alternatif tindakan yang mungkin dijalankan. Hasil pemilihan tersebut kemudian diimplementasikan dalam proses pengambilan keputusan.

4. Implementasi

   Tahap ini sebenarnya termasuk tahap ketiga, namun ada beberapa pihak berpendapat bahwa tahap ini perlu dipandang sebagai bagian yang terpisah guna menggambarkan hubungan antar fase secara lebih luas.

2.1.5.2 Tingkat teknologi SPK

Untuk merancang dan menggunakan SPK dikenal tiga tingkatan teknologi yang berupa perangkat keras (*hardware*) dan perangkat lunak (*software*). Ketiga tingkatan yang dimaksud adalah:

1. SPK spesifik (*spesific DSS*)

   Sistem pendukung keputusan khusus adalah sistem yang ditujukan untuk membantu pemecahan serangkaian masalah yang memiliki karakteristik tertentu.

2. Pembangkit SPK (*DSS generator*)

   Pembangkit sistem pendukung keputusan adalah perangkat lunak untuk pengembangan sistem pendukung keputusan yang berfungsi untuk menghubungkan perangkat keras dan perangkat lunak yang akan digunakan dalam merancang dan membangun sistem pendukung keputusan.

3. Perlengkapan SPK (*DSS tools*)

   Sistem ini merupakan teknologi yang paling mendasar dalam merancang dan membangun sistem pendukung keputusan, terdiri atas elemen perangkat keras dan perangkat lunak, diantaranya bahasa pemrograman, sistem operasi komputer, perangkat lunak pengakses data dan sebagainya.

2.1.5.3 Komponen SPK

Suatu sistem pendukung keputusan memiliki tiga subsistem utama, yaitu:

1. Manajemen basis data (*database*)
2. Subsitem manajemen basis model (Model base)
3. Subsistem penyelenggara dialog (*User sistem interface*).

2.1.5.4 Keuntungan dan keterbatasan SPK

Keuntungan sistem pendukung keputusan adalah sebagai berikut:

1. Sistem pendukung keputusan memperluas kemampuan untuk pengambil keputusan dalam memproses data atau inspirasi bagi pemakainya.
2. Sistem pendukung keputusan menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
3. Sistem pendukung keputusan membantu pengambil keputusan dalam hal penghematan waktu yang dibutuhkan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak teratur.
4. Walaupun suatu sistem pendukung keputusan tidak mampu memecah masalah yang dihadapi oleh pengambil keputusan, namun sistem pendukung keputusan dapat menjadi stimulan bagi pengambil keputusan dalam memahami persoalannya, karena sistem pendukung keputusan mampu menyajikan berbagai alternatif.
5. Sistem pendukung keputusan dapat menyediakan bukti tambahan untuk memberikan pembenaran sehingga dapat memperkuat posisi pengambil keputusan.

Disamping keuntungan sistem pendukung keputusan juga memiliki keterbatasan diantaranya:

1. Beberapa kemampuan manajemen dan bakat manusia yang tidak dapat dimodelkan, sehingga model yang ada dalam sistem tidak semuanya mencerminkan persoalan sebenarnya.
2. Kemampuan suatu pendukung keputusan terbatas pada pembendaharaan pengetahuan yang dimilikinya (pengetahuan dasar serta model dasar proses). Proses yang dapat dilakukan oleh sistem pendukung keputusan biasanya tergantung pada kemampuan perangkat lunak yang digunakannya.

2.1.5.5 Pemodelan sistem pengambilan keputusan

Seperti telah dijelaskan diatas system didefinisikan sebagai kumpulan objek yang memiliki keterkaitan fungsi dan prosedur untuk mencapai tujuan tertentu bersama-sama. Sistem pengambilan keputusan berkaitan dengan elemen-elemen keputusan seperti pengambilan keputusan, tool pengambilan keputusan, aturan dan ide atau prinsip dengan tujuan mencari solusi atas permasalahan keputusan yang dihadapi.

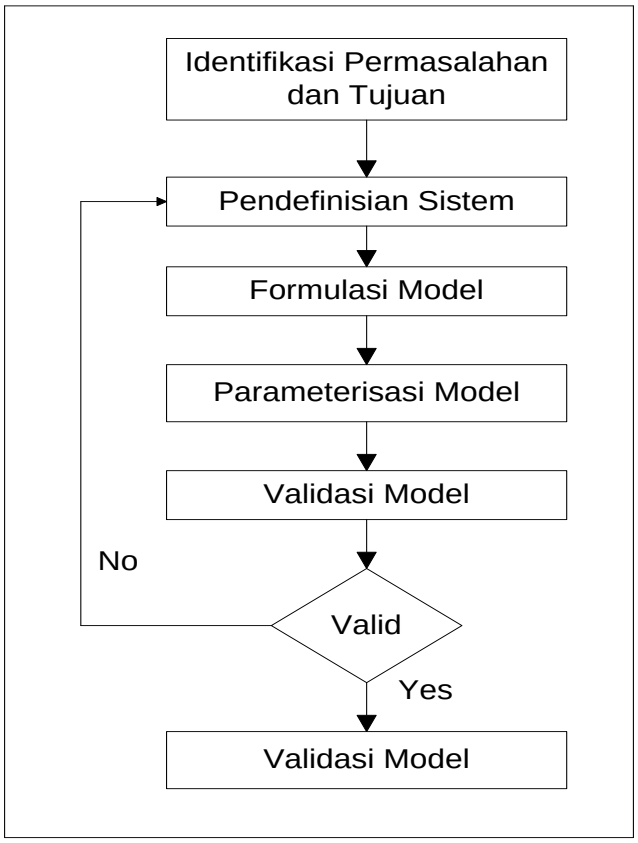
2.1.5.6 Metode Keputusan

Model keputusan relevan dengan model secara umum. Model didefinisikan sebagai representasi sederhana dari suatu keadaan nyata.

2.1.5.7 Tahapan Pemodelan

Pemodelan pada dasarnya merupakan proses membangun atau membentuk sebuah model, dalam bahasa formal tertentu, dari suatu system nyata berdasarkan sudut pandang tertentu menurut . Sistem nyata akan dilihat dan dibaca oleh pemodelan dan membentuk citra atau gambaran tertentu di dalam pikirannya.

Pemodelan dilakukan menurut beberapa tahapan seperti yang ditunjukan oleh gambar 2.2 Tahapan ini menjadi arah bagi pemodel untuk membuat model yang memiliki karakter dengan tingkat generalisasi tinggi, mekanisme transparan, berpotensi untuk dikembangkan peneliti lain, dan peka terhadap perubahan asumsi.



Gambar 2.2 Tahapan Pemodelan Sistem

Tahapan ini mengisyaratkan pemodelan untuk memasukkan komponen pada suatu system nyata yang benar – benar menentukan perilaku system untuk suatu persoalan yang sedang diamati dan mengisyaratkan bahwa pengguna model harus tetap mempertahankan validitasnya dan asumsinya.

2.1.5.8 Pengambilan Keputusan Kriteria Majemuk

Pengambilan keputusan kriteria majemuk pada prinsipnya adalah sebagai berikut : Model pengambilan keputusan untuk penentuan prioritas alternatif dengan menggunakan dua atau lebih kriteria atau atribut, yang satu sama lain terkadang memiliki konflik dan kriteria yang tidak sepadan untuk beberapa kepentingan kelompok**.**

Penggunaan model untuk pengambilan keputusan kriteria majemuk untuk suatu keputusan tertentu tergantung pada saat pemilihan kriteria yang digunakan sebagai kriteria satuan analisis. Pada saat pembuatan kriteria, pengambilan keputusan harus mencoba untuk menggambarkan dalam bentuk kuantifikasi jika hal ini memungkinkan, karena akan selalu ada faktor yang tidak dapat dikuantifikasikan yang juga tidak dapat diabaikan. Bila diabaikan maka hal ini dapat mengakibatkan semakin sulitnya membuat perbandingan kenyataan bahwa kriteria yang baik tidak bisa dikuantifikasikan itu sukar untuk diperkirakan dan diperbandingkan hendaknya tidak dapat menyebabkan pengambilan keputusan untuk tidak menggunakan kriteria tersebut, karena kriteria ini dapat saja relevan dengan masalah utama di dalam setiap analisis. Beberapa kriteria yang kemungkinan sangat penting, tetapi sulit dikuantifikasikan adalah seperti faktor-faktor sosial (seperti gangguan lingkungan), estetika, keadilan, faktor-faktor politis, serta kelayakan pelaksanaan, akan tetapi jika suatu kriteria dapat dikuantifikasikan tanpa merubah pengertiannya, maka hal ini dapat dilakukan.

2.1.5.9 Penentuan Kriteria

Sifat-sifat yang harus diperhatikan dalam memilih kriteria pada setiap persoalan pengambilan keputusan adalah sebagai berikut:

1. Lengkap

   Kriteria yang dipilih harus dapat mencakup seluruh aspek penting dalam persoalan tersebut. Suatu *set* kriteria disebut lengkap apabila *set* ini dapat menunjukkan seberapa jauh seluruh tujuan dapat dicapai.

2. Operasional

   Kriteria yang baik harus dapat digunakan dalam analisis. Sifat operasional ini mencakup beberapa pengertian, antara lain bahwa *set* kriteria ini harus mempunyai arti bagi pengambilan keputusan, sehingga ia dapat benar-benar menghayati implikasinya terhadap alternatif yang ada. Selain itu, jika tujuan pengambilan keputusan ini harus dapat digunakan sebagai sarana untuk meyakinkan pihak lain, maka *set* kriteria ini harus dapat digunakan sebagai sarana untuk memberikan penjelasan atau untuk berkomunikasi. Operasional ini juga mencakup sifat dapat diukur, tujuannya adalah untuk memperolah distribusi kemungkinan dari tingkat pancapaian kriteria yang mungkin diperoleh (untuk keputusan dalam ketidakpastian) dan mengungkapkan perferensi pengambilan keputusan atas pencapaian kriteria.

3. Tidak Berlebihan

   Kriteria yang dipilih tidak berlebihan untuk menghindari perhitungan yang berulang. Proses menentukan *set* kriteria diusahakan menghindari kriteria yang mengandung pengertian yang sama.

4. Minimum

   Jumlah kriteria harus minimum dengan tujuan agar lebih mengkonprehensifkan persoalan. Semakin banyak kriteria yang dilibatkan maka semakin sukar pula untuk

dapat menghayati permasalahan dengan baik, lebih jauh lagi, jumlah perhitungan yang diperlukan dalam analisis akan semakin banyak.

Adakalanya meskipun kita telah berusaha menjabarkan tujuan menjadi lebih spesifik, kita tetap dapat menemukan kriteria untuk sejumlah tujuan.

2.1.5.10 Jenis Metode Pengambilan Keputusan Kriteria Majemuk

Ada beberapa metode standar yang umum digunakan untuk pengambilan keputusan Kriteria majemuk adalah *Multi Attribute Utility Theory* (MAUT) (Edward, W, 1997), *Simple Multi Attribute Rating Tecnique* (SMART) (Edward, W dan Barron, FH, 1994), *Preference Ranking Organization Method for Enrichment Evaluation* (PROMETHEE) (Brans *et al*, 1984) dan *Analytic Hierarchy Process* (AHP) (Saaty, TL, 1980). Perkembangan ilmu pengambilan keputusan kriteria majemuk juga telah meluas dengan diperkenalkan metode yang lebih kompleks seperti *Analytic Network Process* (ANP).

Penelitian ini mengambil basis metode PROMETHEE sebagai metode untuk memecahkan permasalahan yang dihadapi dalam menentukan kelayakan penerimaan beasiswa.

2.1.5.11 Metode yang Digunakan dalam SPK

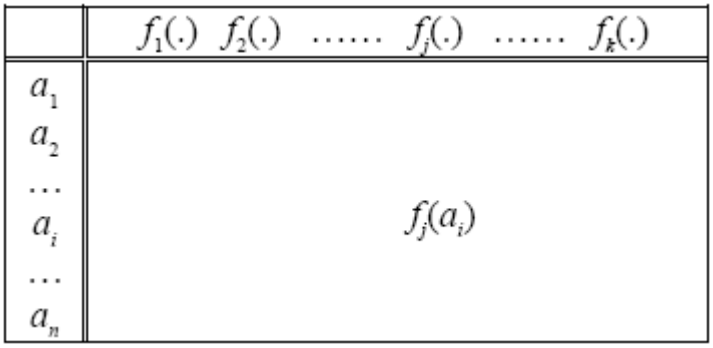
Metode yang digunakan dalam pengambilan keputusan penerima beasiswa ini yaitu menggunakan metode PROMETHEE.

2.1.5.11.1 Dasar PROMETHEE

PROMETHEE adalah salah satu metode penentuan urutan atau prioritas dalam analisis multikriteria atau MCDM (*Multi Criterion Decision Making*). Dugaan dari dominasi kriteria yang digunakan dalam PROMETHEE adalah penggunaan nilai dalam

hubungan *outrangking*. Masalah pokoknya adalah kesederhanaan, kejelasan dan kestabilan. Semua parameter yang dinyatakan mempunyai pengaruh nyata menurut pandangan ekonomi.

Data dasar untuk evaluasi dengan methode PROMETHEE disajikan pada Gambar 2.3 sebagai berikut :



Gambar 2.3 Data Dasar analisis PROMETHEE

2.1.5.11.2 Dominasi kriteria

Nilai *f* merupakan nilai nyata dari suatu kriteria, *f : K → Я (Real Word)* dan tujuannya berupa prosedur optimasi untuk setiap alternatif yang akan diseleksi, *a ε K*, *f(a)* merupakan evaluasi dari alternatif yang akan diseleksi tersebut untuk setiap kriteria. Pada saat dua alternatif dibandingkan *a,b ε 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 beda antara *a* dan *b*, atau tidak ada preferensi dari *a* lebih baik dari *b*.

- *P(a,b) ≈ 0*, berarti lemah preferensi dari *a* lebih baik dari *b*.

- *P(a,b) = 1*, 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 obyek akan dipertimbangkan memiliki nilai kriteria yang lebih baik ditentukan nilai *f* dan akumulasi dari nilai ini menentukan nilai preferensi atas masing–masing obyek yang akan dipilih.

Setiap kriteria boleh memiliki nilai dominasi kriteria atau bobot kriteria yang sama atau berbeda, dan nilai bobot tersebut harus di atas 0 (Nol). Sebelum menghitung bobot untuk masing-masing kriteria, maka dihitung total bobot dari seluruh kriteria terlebih dahulu. Berikut rumus perhitungan bobot kriteria :

$$W_j = \frac{w_j}{\sum w_i} \quad \text{atau} \quad \sum W_j = 1$$

Maka didapat rumus perbandingan untuk setiap alternatif, sebagai berikut :

$$\pi(a_1, a_i) = \sum_{j=1}^{J} W_j \times P_j(a_1, a_i)$$

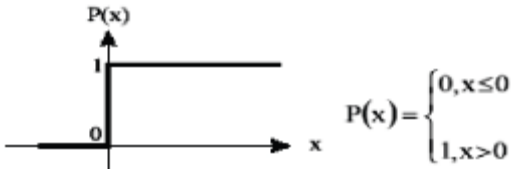
2.1.5.11.3 Rekomendasi fungsi preferensi untuk keperluan aplikasi

Dalam metode PROMETHEE ada Enam bentuk fungsi preferensi kriteria. Untuk memberikan gambaran yang lebih baik terhadap area yang tidak sama, maka digunakan tipe fungsi preferensi. Ke Enam tipe preferensi tersebut meliputi :

1. Tipe Biasa (*Usual Criterion*)

   Tipe *Usual* adalah tipe dasar, yang tidak memiliki nilai *threshold* atau kecenderungan dan tipe ini jarang digunakan. Pada tipe ini dianggap tidak ada beda antara alternatif *a*

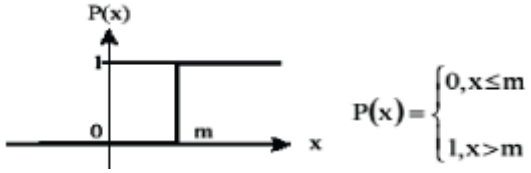
dan alternatif *b* jika *a=b* atau *f(a)=f(b)* , maka niliai preferensinya benilai 0 (Nol) atau *P(x)=0*. Apabila nilai kriteria pada masing-masing alternatif memiliki nilai berbeda, maka pembuat keputusan membuat preferensi mutlak benilai 1 (Satu) atau *P(x)=1* untuk alternatif yang memiliki nilai lebih baik. Fungsi *P(x)* untuk preferensi ini disajikan pada gambar 2.4.



Gambar 2.4 Tipe Preferensi *Usual* [7]

2. Tipe *Quasi* (*Quasi Criterion* atau U-Shape)

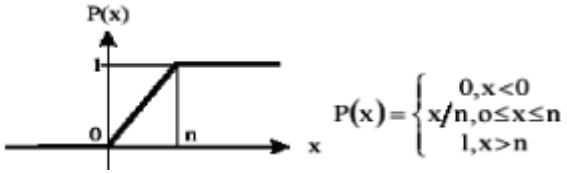
   Tipe *Quasi* sering digunakan dalam penilaian suatu data dari segi kwalitas atau mutu, yang mana tipe ini menggunakan Satu *threshold* atau kecenderungan yang sudah ditentukan, dalam kasus ini *threshold* itu adalah *indifference*. *Indifference* ini biasanya dilamabangkan dengan karakter *m* atau *q*, dan nilai *indifference* harus diatas 0 (Nol). Suatu alternatif memiliki nilai preferensi yang sama penting selama selisih atau nilai *P(x)* dari masing-masing alternatif tidak melebihi nilai *threshold*. Apabila selisih hasil evaluasi untuk masing-masing alternatif melebihi nilai *m* maka terjadi bentuk preferensi mutlak, jika pembuat memutuskan menggunakan kriteria ini, maka *decision maker* tersebut harus menentukan nilai *m*, dimana nilai ini dapat dijelaskan pengaruh yang signifikan dari sutau kriteria. fungsi *P(x)* untuk preferensi ini disajikan pada gambar 2.5.



Gambar 2.5 Tipe Preferensi *Quasi* [7]

3. Tipe Linier (*Linear Criterion* atau V-Shape)

   Tipe Linier acapkali digunakan dalam penilaian dari segi kuantitatif atau banyaknya jumlah, yang mana tipe ini juga menggunakan Satu *threshold* atau kecenderungan yang sudah ditentukan, dalam kasus ini *threshold* itu adalah *preference*. *Preference* ini biasanya dilamabangkan dengan karakter *n* atau *p*, dan nilai *preference* harus diatas 0 (Nol). Kriteria ini menjelaskan bahwa selama nilai selisih memiliki nilai yang lebih rendah dari *n*, maka nilai preferensi dari pembuat keputusan meningkat secara linier dengan nilai *x*, jika nilai *x* lebih besar dibandingkan dengan nilai *n*, maka terjadi preferensi mutlak. Fungsi *P(x)* untuk preferensi ini disajikan pada gambar 2.6.

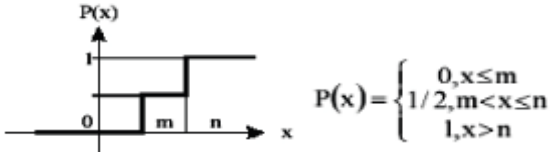


Gambar 2.6 Tipe Preferensi *Linear* [7]

4. Tipe Tingkatan (*Level Criterion*)

   Tipe ini mirip dengan tipe *Quasi* yang sering digunakan dalam penilaian suatu data dari segi kwalitas atau mutu. Tipe ini juga menggunakan *threshold indifference* (*m*) tetapi ditambahkan Satu *threshold* lagi yaitu *preference* (*n*). Nilai *indifference* serta *preference* harus diatas 0 (Nol) dan nilai *indifference* harus di bawah nilai *preference*. Apabila alternatif tidak memiliki perbedaan (*x*), maka nilai preferensi sama dengan 0

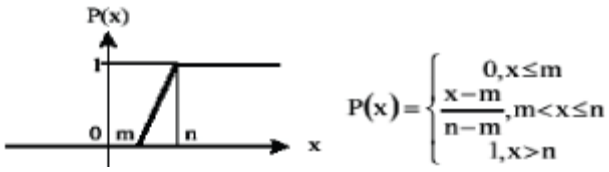
(Nol) atau *P(x)=0*. Jika *x* berada diatas nilai *m* dan dibawah nilai *n*, hal ini berarti situasi preferensi yang lemah *P(x)=0.5*. Dan jika *x* lebih besar atau sama dengan nilai *n* maka terjadi preferensi mutlak *P(x)=1*. Fungsi *P(x)* untuk preferensi ini disajikan pada gambar 2.7



Gambar 2.7 Tipe Preferensi *Level* [7]

5. Tipe *Linear Quasi* (*Linear Criterion with Indifference*)

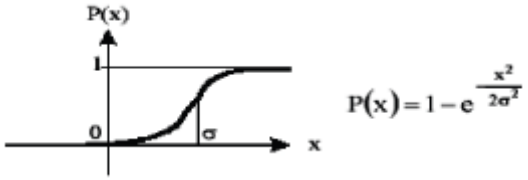
   Tipe *Linear Quasi* juga mirip dengan tipe *Linear* yang acapkali digunakan dalam penilaian dari segi kuantitatif atau banyaknya jumlah. Tipe ini juga menggunakan *threshold preference* (*n*) tetapi ditambahkan Satu *threshold* lagi yaitu *indifference* (*m*). Nilai *indifference* serta *preference* harus diatas 0 (Nol) dan nilai *indifference* harus di bawah nilai *preference*. Pengambilan keputusan mempertimbangkan peningkatan preferensi secara linier dari tidak berbeda hingga preferensi mutlak dalam area antara dua kecenderungan *m* dan *n*. Fungsi *P(x)* untuk preferensi ini disajikan pada gambar 2.8.



Gambar 2.8 Tipe Preferensi *Linear Quasi* [7]

6. Tipe *Gaussian*

   Tipe *Gaussian* sering digunakan untuk mencari nilai aman atau titik aman pada data yang bersifat *continue* atau berjalan terus.[8] Tipe ini memiliki nilai *threshold* yaitu *Gaussian threshold* (σ) yang berhubungan dengan nilai standar deviasi atau distribusi normal dalam statistik. fungsi *P(x)* untuk preferensi ini disajikan pada gambar 2.9.



Gambar 2.9 Tipe Preferensi *Gaussian* [7]

2.1.5.11.4 Nilai *threshold* atau kecenderungan

Enam tipe dari penyamarataan kriteria bisa dipertimbangkan dalam metode PROMETHEE, tiap-tiap tipe bisa lebih mudah ditentukan nilai kecenderungannya atau parameternya karena hanya Satu atau Dua parameter yang mesti ditentukan. Hanya tipe Usual saja yang tidak memiliki nilai parameter.

1. *Indifference threshold* yang biasa dilambangkan dalam karakter *m* atau *q*. Jika nilai perbedaan (*x*) di bawah atau sama dengan nilai *indifference* $x \leq m$ maka *x* dianggap tidak memiliki nilai perbedaan $x = 0$.
2. *Preference threshold* yang biasa dilambangkan dalam karakter *n* atau *p*. Jika nilai perbedaan (*x*) di atas atau sama dengan nilai *preference* $x \geq n$ maka perbedaan tersebut memiliki nilai mutlak $x = 1$.

3. *Gaussian threshold* yang biasa dilambangkan dalam karakter σ serta diketahui dengan baik sebagai parameter yang secara langsung berhubungan dengan nilai standar deviasi pada distribusi normal. [5]

2.1.5.11.5 Arah dalam grafik nilai *outrangking*

Perangkingan yang digunakan dalam metode PROMETHEE meliputi tiga bentuk antara lain :

1. *Entering flow*

   Entering flow adalah jumlah dari yang memiliki arah mendekat dari *node* a dan hal ini merupakan karakter pengukuran *outrangking*.

   Untuk setiap nilai *node* a dalam grafik nilai *outrangking* ditentukan berdasarkan e*ntering flow* dengan persamaan:

$$\phi^{+}(a_1) = \sum_{i=1}^{I} \pi(a_1, a_i')$$

2. *Leaving flow*

   Sedangkan *Leaving flow* adalah jumlah dari yang memiliki arah menjauh dari *node* a. dan hal ini merupakan pengukuran *outrangking*. Adapun persamaannya:

$$\phi^{-}(a_1) = \sum_{i=1}^{I} \pi(a_1', a_i)$$

3. *Net Flow*

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

$$\phi(a_1) = \phi^{+}(a_1) - \phi^{-}(a_1)$$

Semakin besar nilai *Entering flow* dan semakin kecil *Levaing flow* maka alternatif tersebut memiliki kemungkinan dipilih yang semakin besar. Perangkingan dalam PROMETHEE I dilakukan secara parsial, yaitu didasarkan pada nilai *Entering flow* dan *Levaing flow*. Sedangkan PROMETHEE II termasuk perangkingan komplek karena didasarkan pada nilai *Net flow* masing-masing alternatif yaitu alternatif dengan nilai *Net flow* lebih tinggi menempati satu rangking yang lebih baik.

2.1.5.11.6 Langkah-langkah perhitungan dengan metode PROMETHEE

Langkah-langkah perhitungan dengan metode PROMETHEE adalah sebagai berikut:

1. Menentukan beberapa alternatif
2. Menentukan beberapa kriteria
3. Menentukan dominasi kriteria
4. Menentukan tipe penilaian, dimana tipe penilaian memiliki 2 tipe yaitu; tipe minimum dan maksimum.
5. Menentukan tipe preferensi untuk setiap kriteria yang paling cocok didasarkan pada data dan pertimbangan dari *decision maker*. Tipe preferensi ini berjumlah Enam (*Usual, Quasi, Linear, Level, Linear Quasi* dan *Gaussian*).
6. Memberikan nilai *threshold* atau kecenderungan untuk setiap kriteria berdasarkan preferensi yang telah dipilih.
7. Perhitungan *Entering flow, Leaving flow dan Net flow*
8. Hasil pengurutan hasil dari perangkingan

Dalam metode promethee ada 2 macam perangkingan yang disandarkan pada hasil perhitungan, antara lain :

a. Perangkingan parsial yang didasarkan pada nilai *Entering flow* dan *Leaving flow.*

b. Perangkingan lengkap atau komplit yang didasarkan pada nilai *Net flow.*

2.1.6 Perangkat Lunak Penunjang

2.1.6.1 Borland Delphi 7

Borland Delphi 7.0. Secara umum Borland Delphi memiliki kemiripan dengan Visual Basic kecuali bahasa dasarnya adalah Object Pascal (suatu versi Pascal yang mengadopsi konsep pemrograman berorientasi objek). Delphi juga mengadopsi teknologi COM+/MTS (*Conponent Object Model / Microsoft Transaction Server*), yaitu teknologi Microsoft yang memungkinkan sekumpulan komponen yang dikembangkan dengan bahasa-bahasa pemrograman yangberbeda digunakan oleh suatu aplikasi dengan syarat bahasa-bahasa pemrograman itu menggunakan platform COM.

Salah satu kelebihan Delphi adalah aplikasinya bisa dikembangkan diatas berbagai macam sistem operasi, misalnya UNIX, LINUX dan sebagainya. Kelebihan lainnya adalah pada umumnya aplikasi yang dikembangkan dengan Delphi akan berjalan lebih cepat , selain itu tipe data yang dimiliki oleh Delphi lebih lengkap.

Delphi merupakan perangkat pengembangan aplikasi yang sangat terkenal dilingkungan Window. Dengan menggunkan perangkat lunak ini kita dapat membangun berbagai aplikasi Windows dengan cepat dan mudah. Dengan pendekatan visual, kita dapat menciptakan aplikasi yang canggih tanpa banyak menuliskan kode.

Delphi menggunakan bahasa Objek Pascal sebagai dasar. Jika kita telah menguasai Pascal, kita dapat lebih mudah memahami program Delphi. Untuk mempermudah pemograman dalam membuat program aplikasi, Delphi menyediakan fasilitas

pemograman yang sangat lengkap. Khusus untuk pemograman database, Delphi menyediakan objek yang sangat kuat, canggih dan lengkap, sehingga memudahkan pemograman dalam merancang, membuat dan menyelesaikan aplikasi database yang diinginkan. Selain itu Delphi juga dapat menangani data dalam berbagai format database, misalnya MS.Accses, SyBase, Oracle, FoxPro, Informix, InterBase, SQL Server dll. Format database yang dianggap asli dari Delphi adalah Paradox dan dBase.

2.1.6.2 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL.[3]

Tidak sama dengan proyek-proyek seperti Apache, dimana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius.

MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk

turunan yang bersifat *closed source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem *database* (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. 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*. Selain itu MySQL juga memiliki beberapa keistimewaan, antara lain :

1. *Portability*

   MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

2. *Open Source*

   MySQL didistribusikan secara *open source* (gratis), dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.

3. *Multiuser*

   MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance tuning*

   MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. *Column types*

   MySQL memiliki tipe kolom yang sangat kompleks, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.

6. *Command dan functions*

   MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam *query*.

7. *Security*

   MySQL memiliki beberapa lapisan sekuritas seperti level *subnetmask*, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

8. *Scalability dan limits*

   MySQL mampu menangani *database* dalam skala besar, dengan jumlah *records* lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

9. *Connectivity*

   MySQL dapat melakukan koneksi dengan *client* menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).

10. Localisation

MySQL dapat mendeteksi pesan kesalahan pada client dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk didalamnya.

11. Interface

MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

12. Clients dan tools

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database, dan pada setiap tool yang ada disertakan petunjuk online.

13. Struktur tabel

MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan database lainnya semacam *PostgreSQL* ataupun *Oracle*.