

BAB II LANDASAN TEORI

2.1 Sistem

2.1.1 Pengertian Sistem

Menurut Herlambang (2005), definisi sistem dapat dibagi menjadi dua pendekatan, yaitu pendekatan secara prosedur dan pendekatan secara komponen. Berdasarkan pendekatan prosedur, sistem didefinisikan sebagai kumpulan dari beberapa prosedur yang mempunyai tujuan tertentu. Sedangkan berdasarkan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan mencapai tujuan tertentu.

McLeod dan Schell, dalam bukunya Sistem Informasi Manajemen, diterjemahkan oleh Wahyono. (2004) mengatakan, "Sistem adalah sekelompok elemen terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan."

Menurut Jogianto (2005), Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata, seperti tempat, benda dan orang-orang yang betul-betul ada dan terjadi.

Kesimpulan dari pengertian sistem diatas adalah sistem adalah sekumpulan unsur-unsur yang berada dalam satu organisasi yang saling berinteraksi dan bekerjasama untuk mencapai suatu tujuan.

2.1.2 Elemen Sistem

Menurut Wahyono (2004), Ada beberapa elemen yang membentuk sebuah sistem, yaitu: tujuan, masukan, proses, keluaran, batas, mekanisme pengendalian dan umpan balik serta lingkungan. Berikut penjelasan mengenai elemen-elemen yang membentuk sebuah sistem :

1. Tujuan

Setiap sistem memiliki tujuan (*Goal*), entah hanya satu atau mungkin banyak. Tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tak terarah dan tak terkendali. Tentu saja, tujuan antara satu sistem dengan sistem yang lain berbeda.

2. Masukan

Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan yang diproses. Masukan dapat berupa hal-hal yang berwujud (tampak secara fisik) maupun yang tidak tampak. Contoh masukan yang berwujud adalah bahan mentah, sedangkan contoh yang tidak berwujud adalah informasi (misalnya permintaan jasa pelanggan).

3. Proses

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai, misalnya berupa informasi dan produk, tetapi juga bisa berupa hal-hal yang tidak berguna, misalnya saja sisa pembuangan atau limbah. Pada pabrik kimia, proses dapat berupa bahan mentah.

4. Keluaran

Keluaran (*output*) merupakan hasil dari pemrosesan. Pada sistem informasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan, dan sebagainya.

4. Batas Yang disebut batas (*boundary*) sistem adalah pemisah antara sistem dan daerah di luar sistem (*lingkungan*). Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem. Sebagai contoh, tim bola basket mempunyai aturan permainan dan keterbatasan kemampuan pemain. Pertumbuhan sebuah toko eceran dipengaruhi oleh pembelian pelanggan, gerakan pesaing dan keterbatasan dana dari bank. Tentu saja batas sebuah sistem dapat dikurangi atau dimodifikasi sehingga akan mengubah perilaku sistem. Sebagai contoh, dengan menjual saham ke publik, sebuah perusahaan dapat mengurangi keterbatasan dana.

5. Mekanisme Pengendalian dan Umpan Balik

Mekanisme pengendalian (*control mechanism*) diwujudkan dengan menggunakan umpan balik (*feedback*), yang mencuplik keluaran. Umpan

balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya adalah untuk mengatur agar sistem berjalan sesuai dengan tujuan.

6. Lingkungan

Lingkungan adalah segala sesuatu yang berada diluar sistem. Lingkungan bisa berpengaruh terhadap operasi sistem dalam arti bisa merugikan atau menguntungkan sistem itu sendiri. Lingkungan yang merugikan tentu saja harus ditahan dan dikendalikan supaya tidak mengganggu kelangsungan operasi sistem, sedangkan yang menguntungkan tetap harus terus dijaga, karena akan memacu terhadap kelangsungan hidup sistem.

2.2 Keputusan

2.2.1 Pengertian Keputusan

Davis (2000) mengemukakan, Keputusan adalah hasil pemecahan masalah yang dihadapinya dengan tegas. Suatu keputusan merupakan jawaban yang pasti terhadap suatu pertanyaan. Keputusan harus dapat menjawab pertanyaan tentang apa yang dibicarakan dalam hubungannya dengan perencanaan. Keputusan dapat pula berupa tindakan terhadap pelaksanaan yang sangat menyimpang dari rencana semula.

Definisi keputusan pada umumnya adalah pilihan (*choice*), yaitu pilihan dari dua atau lebih kemungkinan. Jika berhubungan dengan proses, maka keputusan adalah keadaan akhir dari suatu proses yang lebih dinamis yang diberi label pengambilan keputusan. Keputusan dipandang sebagai proses karena terdiri atas satu rangkaian aktivitas yang berhubungan dan tidak hanya dianggap sebagai tindakan bijaksana. Dengan kata lain, keputusan merupakan kesimpulan yang dicapai sesudah dilakukan pertimbangan, yang terjadi setelah kemungkinan dipilih, sementara yang lain dikesampingkan (Davis, 2000).

Keputusan adalah suatu reaksi terhadap beberapa solusi alternatif yang dilakukan secara sadar dengan cara menganalisa kemungkinan-kemungkinan dari alternatif tersebut bersama konsekuensinya. Setiap keputusan akan membuat pilihan

terakhir, dapat berupa tindakan atau opini. Itu semua bermula ketika kita perlu untuk melakukan sesuatu tetapi tidak tahu apa yang harus dilakukan. Untuk itu keputusan dapat dirasakan rasional atau irrasional dan dapat berdasarkan asumsi kuat atau asumsi lemah.

Bila dikaitkan dengan suatu organisasi, keputusan ini disebut dengan Sistem Keputusan. Dan sistem keputusan ini adalah salah satu bagian dari sistem organisasi. Keputusan dapat diklasifikasikan menjadi 3 tingkatan (Davis, 2000).:

1. Strategis, keputusan dengan ciri kepastian besar dan orientasi masa depan.
2. Taktis, keputusan dengan ciri berhubungan dengan aktivitas jangka pendek dan alokasi sumber-sumber daya guna mencapai sasaran.
3. Teknik, keputusan dengan ciri standar-standar ditetapkan dan bersifat deterministik, mengusahakan agar tugas spesifik diimplementasikan dengan efektif dan efisien.

2.3 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan atau sering disebut *Decision Support System* (DSS) adalah Sistem berbasis model yang terdiri dari prosedur-prosedur dalam pemrosesan data dan pertimbangannya untuk membantu manajer dalam mengambil keputusan. Agar berhasil mencapai tujuannya maka sistem tersebut harus sederhana, robust, mudah untuk dikontrol, mudah beradaptasi lengkap pada hal-hal penting dan mudah berkomunikasi dengannya. Secara implisit juga berarti bahwa sistem ini harus berbasis komputer dan digunakan sebagai tambahan dari kemampuan penyelesaian masalah dari seseorang (Turban, 2005).

Sistem Pendukung Keputusan mendayagunakan *resources* individu-individu secara intelek dengan kemampuan komputer untuk meningkatkan kualitas keputusan. Jadi ini merupakan sistem pendukung yang berbasis komputer untuk manajemen pengambilan keputusan yang berhubungan dengan masalah-masalah yang semi terstruktur (Turban, 2005).

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

2.3.1 Proses Pengambilan Keputusan

Pengambilan keputusan adalah pemilihan beberapa tindakan alternatif yang ada untuk mencapai satu atau beberapa tujuan yang telah ditetapkan (Turban, 2005). Pengambilan keputusan meliputi beberapa tahap dan melalui beberapa proses. Proses pengambilan keputusan meliputi empat tahap yang saling berhubungan dan berurutan. Empat proses tersebut adalah (Turban, 2005):

a. *Intelligence*

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

b. *Design*

Tahap ini merupakan proses menemukan dan mengembangkan alternatif. Tahap ini meliputi proses untuk mengerti masalah, menurunkan solusi dan menguji kelayakan solusi.

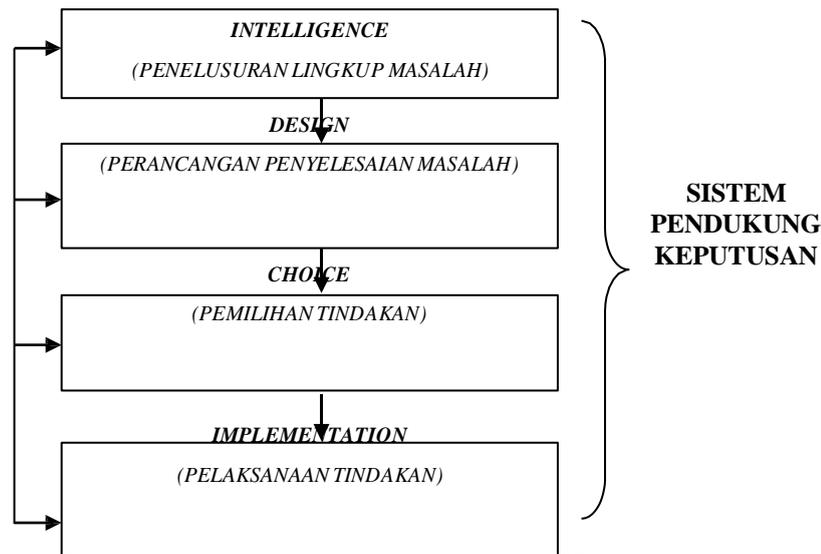
c. *Choice*

Pada tahap ini dilakukan proses pemilihan di antara berbagai alternatif tindakan yang mungkin dijalankan. Tahap ini meliputi pencarian, evaluasi, dan rekomendasi solusi yang sesuai untuk model yang telah dibuat. Solusi dari model merupakan nilai spesifik untuk variabel hasil pada alternatif yang dipilih.

d. *Implementation*

Tahap implementasi adalah tahap pelaksanaan dari keputusan yang telah diambil. Pada tahap ini perlu disusun serangkaian tindakan yang terencana, sehingga hasil keputusan dapat dipantau dan disesuaikan apabila diperlukan perbaikan.

Proses pengambilan keputusan, seperti terlihat pada gambar berikut.



Gambar 2.1 Fase Proses Pengambilan Keputusan

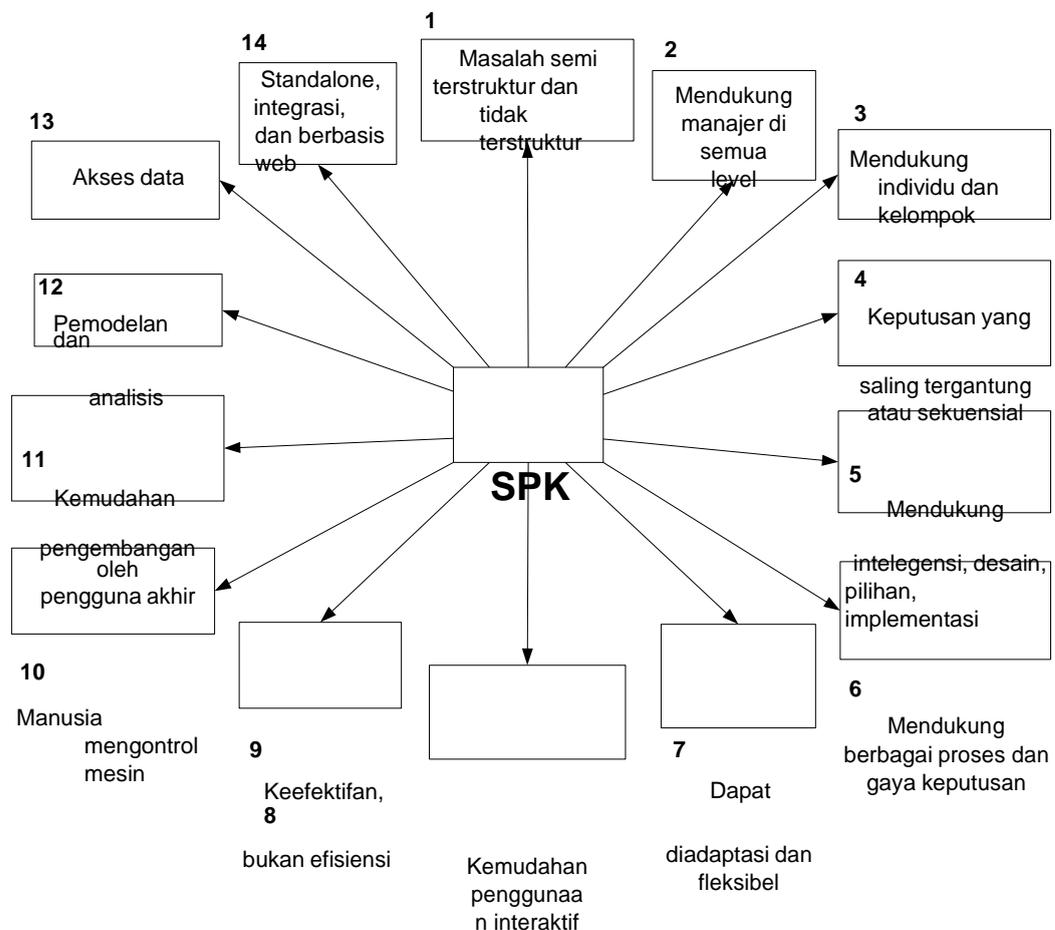
Sumber (Turban, 2005)

2.3.2 Karakteristik Sistem Pendukung Keputusan

Turban (2005) mengemukakan karakteristik dan kapabilitas kunci dari Sistem Pendukung Keputusan adalah sebagai berikut (Gambar 2.2):

1. Dukungan untuk pengambil keputusan, terutama pada situasi semi terstruktur dan tak terstruktur.
2. Dukungan untuk semua level manajerial, dari eksekutif puncak sampai manajer lini.
3. Dukungan untuk individu dan kelompok.
4. Dukungan untuk semua keputusan independen dan atau sekuensial.
5. Dukungan di semua fase proses pengambilan keputusan: inteligensi, desain, pilihan, dan implementasi.
6. Dukungan pada berbagai proses dan gaya pengambilan keputusan.
7. Kemampuan sistem beradaptasi dengan cepat dimana pengambil keputusan dapat menghadapi masalah-masalah baru dan pada saat yang sama dapat menanganinya dengan cara mengadaptasikan sistem terhadap kondisi-kondisi perubahan yang terjadi.

8. Pengguna merasa seperti di rumah. *User-friendly*, kapabilitas grafis yang kuat, dan sebuah bahasa interaktif yang alami.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, *timelines*, kualitas) dari pada efisiensi (biaya).
10. Pengambil keputusan mengontrol penuh semua langkah proses pengambilan keputusan dalam memecahkan masalah.
11. Pengguna akhir dapat mengembangkan dan memodifikasi sistem sederhana.
12. Menggunakan model-model dalam penganalisisan situasi pengambilan keputusan.
13. Disediakkannya akses untuk berbagai sumber data, format, dan tipe, mulai dari sistem informasi geografi (GIS) sampai sistem berorientasi objek.
14. Dapat dilakukan sebagai alat *standalone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di satu organisasi keseluruhan dan di beberapa organisasi sepanjang rantai persediaan.



Gambar 2.2 Karakteristik dan Kapabilitas SPK

Sumber (Turban, 2005)

2.3.3 Keuntungan Sistem Pendukung Keputusan

Keuntungan dari sistem pendukung keputusan sebagai berikut :

1. Dapat memperluas kemampuan seseorang untuk mengambil keputusan dalam memproses data atau informasi pemakainya.
2. Membantu mengambil keputusan dalam hal penghematan waktu yang dibutuhkan untuk memecahkan masalah, terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
3. Dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
4. Dapat menjadi stimulan bagi pengambil keputusan dalam memahami permasalahannya, karena sistem pendukung keputusan mampu menyajikan berbagai alternatif.
5. Mampu menyediakan bukti tambahan untuk memberikan pembenaran, sehingga dapat memperluas posisi pengambilan keputusan.

Sistem Pendukung Keputusan terdiri atas 3 komponen atau subsistem, sebagai berikut:

1. Subsistem Data (*Data Subsystem*)

Subsistem data merupakan bagian yang menyeleksi data – data yang dibutuhkan oleh *Base Management Subsystem* (DBMS). DBMS sendiri merupakan subsistem data yang terorganisasi dalam suatu basis data. Data – data yang merupakan dalam suatu Sistem Pendukung Keputusan dapat berasal dari luar lingkungan. Keputusan pada manajemen level atas seringkali harus memanfaatkan data dan informasi yang bersumber dari luar perusahaan. Kemampuan subsistem data yang diperlukan dalam suatu Sistem Pendukung Keputusan, antara lain :

- a. Mampu mengkombinasikan sumber – sumber data yang relevan melalui proses ekstraksi data.
- b. Mampu menambah dan menghapus secara cepat dan mudah.
- c. Mampu menangani data personal dan non personal, sehingga user dapat bereksperimen dengan berbagai alternatif keputusan.
- d. Mampu mengolah data yang bervariasi dengan fungsi manajemen data yang luas.

2. Subsistem Model (*Model Subsystem*)

Subsistem model dalam Sistem Pendukung Keputusan memungkinkan pengambil keputusan menganalisa secara utuh dengan mengembangkan dan membandingkan alternatif solusi. Intergrasi model – model dalam Sistem Informasi yang berdasarkan integrasi data – data dari lapangan menjadi suatu Sistem Pendukung Keputusan.

Kemampuan subsistem model dalam Sistem Pendukung Keputusan antara lain :

- a. Mampu menciptakan model – model baru dengan cepat dan mudah
- b. Mampu mengkatalogkan dan mengelola model untuk mendukung semua tingkat pemakai
- c. Mampu menghubungkan model – model dengan basis data melalui hubungan yang sesuai
- d. Mampu mengelola basis model dengan fungsi manajemen yang analog dengan database manajemen

3. Subsistem Dialog (*User Sistem Interface*)

Subsistem dialog merupakan bagian dari Sistem Pendukung Keputusan yang dibangun untuk memenuhi kebutuhan representasi dan mekanisme kontrol selama proses analisa dalam Sistem Pendukung Keputusan ditentukan dari kemampuan berinteraksi antara sistem yang terpasang dengan user. Pemakai terminal dan sistem perangkat lunak merupakan komponen – komponen yang terlibat dalam subsistem dialog yang mewujudkan komunikasi antara user dengan sistem tersebut. Komponen dialog menampilkan keluaran sistem bagi pemakai dan menerima masukan dari pemakai ke dalam Sistem Pendukung Keputusan. Adapun subsistem dialog dibagi menjadi tiga, antara lain :

- a. Bahasa Aksi (*The Action Language*) Merupakan tindakan – tindakan yang dilakukan user dalam usaha untuk membangun komunikasi dengan sistem. Tindakan yang dilakukan oleh user untuk menjalankan dan mengontrol sistem tersebut tergantung rancangan sistem yang ada.
- b. Bahasa Tampilan (*The Display or Presentation Language*) Merupakan keluaran yang dihasilkan oleh suatu Sistem Pendukung Keputusan dalam

bentuk tampilan – tampilan akan memudahkan user untuk mengetahui keluaran sistem terhadap masukan – masukan yang telah dilakukan.

- c. Bahasa Pengetahuan (Knowledge Base Language) Meliputi pengetahuan yang harus dimiliki user tentang keputusan dan tentang prosedur pemakaian Sistem Pendukung Keputusan agar sistem dapat digunakan secara efektif. Pemahaman user terhadap permasalahan yang dihadapi dilakukan diluar sistem, sebelum user menggunakan sistem untuk mengambil keputusan.

2.4 *Technique for Order Performance by Similarity to Ideal Solution* (TOPSIS)

Topsis adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yonn dan Hwang pada tahun 1981. Ide dasar dari metode ini adalah bahwa alternatif yang dipilih memiliki jarak terdekat dengan solusi ideal dan yang terjauh dari solusi ideal negatif. TOPSIS memperhatikan jarak ke solusi ideal maupun jarak ke solusi ideal negatif dengan mengambil hubungan kedekatan menuju solusi ideal. Dengan melakukan perbandingan pada keduanya, urutan pilihan dapat ditentukan. Berikut ini adalah matriks keputusan C yang memiliki m alternatif dengan n kriteria, dimana x_{ij} adalah pengukuran pilihan dari alternatif ke-i dalam hubungannya dengan kriteria ke-j. (Sri Kusumadewi,2006)

$$C = \begin{matrix} & X_{11} & X_{12} & X_{13} & \dots & X_{1n} \\ X_{21} & X_{22} & X_{23} & \dots & X_{2n} \\ & & & & & \\ & & & & & \\ X_{m1} & X_{m2} & X_{m3} & \dots & X_{mn} \end{matrix}$$

2.4.1 Prosedur Topsis

1. Normalisasi matriks keputusan

Setiap elemen pada matriks X dinormalisasi untuk mendapatkan matriks normalisasi R. Setiap normalisasi dari nilai r_{ij} dapat dilakukan dengan perhitungan sebagai berikut:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}, \text{ untuk } i = 1, 2, 3, \dots, m \text{ dan } j = 1, 2, 3, \dots, n$$

Dimana:

r_{ij} = matriks ternormalisasi [i][j]

x_{ij} = matriks keputusan [i][j]

2. Pembobotan pada matriks yang telah dinormalisasi

Diberikan bobot $W = (W_1, W_2, \dots, W_n)$, sehingga *weighted normalised matrix* V dapat dihasilkan sebagai berikut:

$$V = \begin{pmatrix} w_1 r_{11} & w_2 r_{12} & \dots & w_n r_{1n} \\ w_1 r_{21} & w_2 r_{22} & \dots & w_n r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_1 r_{m1} & w_2 r_{m2} & \dots & w_n r_{mn} \end{pmatrix}$$

Secara matematis, *weighted normalised matrix* ini dapat diperoleh dengan rumus berikut ini:

$$V_{ij} = W_j \cdot r_{ij}$$

Dimana:

$v_{i,j}$ = matriks normalisasi terbobot [i][j]

w_j = vektor bobot [j]

r_{ij} = matriks ternormalisasi [i][j]

3. Menentukan solusi ideal positif dan solusi ideal negatif

Solusi ideal positif dinotasikan dengan A^+ dan solusi ideal negatif dinotasikan dengan A^- . Untuk lebih jelasnya, dapat dilihat dibawah ini:

$$A^+ = \{(\max V_{ij} | j \in J), (\min V_{ij} | j \in J'), i = 1, 2, 3, \dots, m\} = \{V_1^+, V_2^+ \dots V_n^+\}$$

$$A^- = \{(\min V_{ij} | j \in J), (\max V_{ij} | j \in J'), i = 1, 2, 3, \dots, m\} = \{V_1^-, V_2^-, \dots V_n^-\}$$

Dimana:

$J = \{1, 2, \dots, n \text{ dan } j \text{ berhubungan dengan } \textit{benefit criteria}\}$

$J' = \{1, 2, \dots, n \text{ dan } j \text{ berhubungan dengan } \textit{cost criteria}\}$

V_j^+ = solusi ideal positif [j]

V_j^- = solusi ideal negatif [j]

Pembangunan A^+ dan A^- adalah untuk mewakili alternatif yang *most preferable* ke solusi ideal dan yang *least preferable* secara berurutan.

4. Menghitung *Separation Measure*

Separation measure ini merupakan pengukuran jarak dari suatu alternatif ke solusi ideal positif dan solusi ideal negatif

Perhitungan matematisnya adalah sebagai berikut:

e. Rumus pengukuran jarak dari suatu alternatif ke solusi ideal positif

$$S_i^+ = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^+)^2}, \text{ untuk } i=1,2,3,\dots,m$$

Dimana:

S_i^+ = jarak alternatif A_i dengan solusi ideal positif

V_{ij} = matriks normalisasi terbobot $[i][j]$

V_j^+ = solusi ideal positif $[j]$

f. Rumus pengukuran jarak dari suatu alternatif ke solusi ideal negatif

$$S_i^- = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^-)^2}, \text{ untuk } i=1,2,3,\dots,m$$

Dimana:

S_i^- = jarak alternatif A_i dengan solusi ideal negatif

V_{ij} = matriks normalisasi terbobot $[i][j]$

V_j^- = solusi ideal negatif $[j]$

5. Menghitung kedekatan relatif dengan solusi ideal

Kedekatan relatif dari alternatif A_i dengan solusi ideal positif A^+ direpresentasikan dengan:

$$C_i^+ = \frac{S_i^-}{S_i^- + S_i^+}, \text{ dimana } 0 < C_i^+ < 1 \text{ dan } i = 1, 2, 3, \dots, m$$

Dimana:

C_i^+ = kedekatan tiap alternatif terhadap solusi ideal positif

S_i^+ = jarak alternatif A_i dengan solusi ideal positif

S_i^- = jarak alternatif A_i dengan solusi ideal negatif

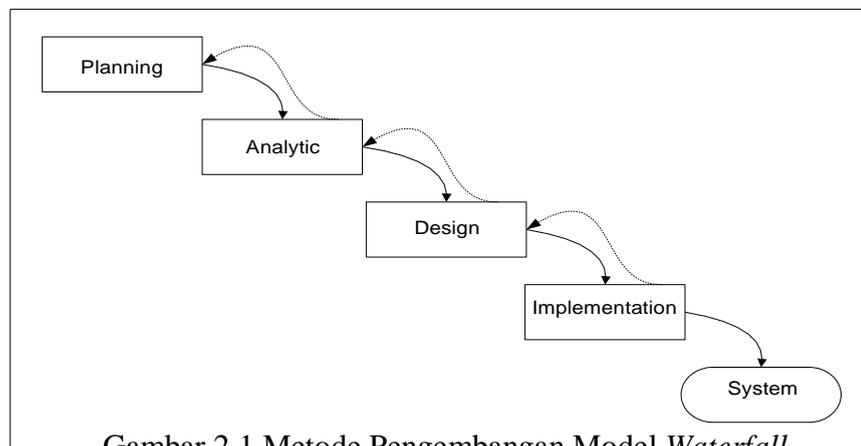
Dikatakan alternatif A_i dekat dengan solusi ideal positif apabila C_i^+ mendekati 1. Jadi $C_i^+ = 1$ jika $A_i = A^+$ dan $C_i^+ = 0$ jika $A_i = A^-$

6. Mengurutkan pilihan

Pilihan akan diurutkan berdasarkan pada nilai C_i^+ sehingga alternatif yang memiliki jarak terpendek dengan solusi ideal positif adalah alternatif yang terbaik. Dengan kata lain, alternatif yang memiliki nilai C_i^+ yang lebih besar itulah yang lebih dipilih.

2.5 Metode pengembangan perangkat lunak menggunakan metode Metode *System Development Life Cycle model Waterfall*

Pada metode penelitian ini dilakukan rekayasa perangkat lunak yang digunakan adalah model *Waterfall* seperti pada gambar berikut ini:



Gambar 2.1 Metode Pengembangan Model *Waterfall*

Sumber (Dennis dan Wixom. 2003)

Keterangan:

1. *Planning (Perencanaan)*

Tahap perencanaan merupakan proses penting untuk mengetahui mengapa sistem harus dibuat dan menentukan bagaimana cara membangun sistem tersebut. Langkah pertama dari proses tersebut adalah dengan mengidentifikasi peluang apakah dapat memberikan kemungkinan biaya rendah tetapi menghasilkan keuntungan.

2. *Analytic (Analisis)*

Analisis sistem dilakukan untuk memberikan jawaban pertanyaan siapa yang akan menggunakan sistem. Apa yang akan dilakukan oleh sistem, dimana dan kapan sistem tersebut digunakan. Pada tahap ini pembuat sistem akan

melakukan observasi dan pengamatan terhadap sistem yang lama, kemudian mengidentifikasi, memanfaatkan dan mengembangkan peluang, dan membangun konsep untuk sebuah sistem baru.

3. *Design (perancangan)*

Tahap perancangan dilakukan untuk menetapkan bagaimana sistem akan dioperasikan. Hal ini berkaitan dengan menentukan perangkat keras, perangkat lunak, jaringan, tampilan program, *form* dan laporan yang akan dipakai. Selain itu perlu juga menspesifikasi program, database dan file yang dibutuhkan.

4. *Implementation*

Merupakan tahap berikutnya untuk menerjemahkan data atau pemecahan masalah yang telah dirancang ke dalam bahasa pemrograman komputer yang telah ditentukan. Semua tahap ini desain perangkat lunak sebagai sebuah program lengkap atau unit program.

5. *System*

Tahapan ini, merupakan hasil sistem yang telah dibuat dalam bentuk perangkat lunak yang telah dipasang dan digunakan, termasuk didalamnya proses pemeliharaan dan perbaikan kesalahan. Perangkat lunak yang telah selesai dibuat dapat mengalami perubahan-perubahan atau penambahan sesuai dengan permintaan *user* atau perubahan sistem.

2.6 *Unified Modelling Language (UML)*

UML yang merupakan singkatan dari Unified Modelling Language adalah sekumpulan pemodelan konvensi yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak dalam kaitannya dengan objek. (Nugroho, 2008).

UML dapat juga diartikan sebuah bahasa grafik standar yang digunakan untuk memodelkan perangkat lunak berbasis objek. UML pertama kali dikembangkan pada pertengahan tahun 1990an dengan kerjasama antara James Rumbaugh, Grady Booch dan Ivar Jacobson, yang masing-masing telah mengembangkan notasi mereka sendiri di awal tahun 1990an. (Nugroho, 2008).

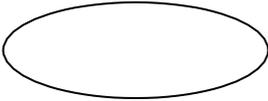
2.6.1 Komponen-komponen UML

UML mendefinisikan diagram-diagram berikut ini: (Nugroho, 2008).

a. Use case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah use case merepresentasikan sebuah interaksi antara aktor dengan sistem.

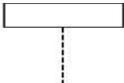
Tabel 2.1 Simbol *Use Case*

No.	Simbol	Keterangan
1	Aktor 	Merupakan kesatuan <i>eksternal</i> yang berinteraksi dengan sistem.
2.	<i>Use Case</i> 	Rangkaian/uraian sekelompok yang saling terkait dan membentuk sistem.
3.	<i>Generelation</i> 	Menggambarkan hubungan khusus atau interaksi dalam objek.

b. Class Diagram

- 1) Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).
- 2) Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.

Tabel 2.2. Simbol Class Diagram

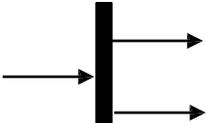
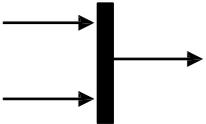
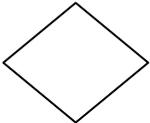
NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

c. Activity Diagram

- 1) Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.
- 2) Activity diagram merupakan state diagram khusus, yang sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karena itu, activity diagram tidak menggambarkan perilaku internal sebuah sistem dan interaksi antar subsistem, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Tabel 2.3 Simbol Activity Diagram

No.	Simbol	Keterangan
1	<i>Start State</i> 	<i>Start state</i> adalah sebuah kondisi awal sebuah <i>object</i> sebelum ada perubahan keadaan. Start state digambarkan dengan sebuah lingkaran solid.

2.	<i>End State</i> 	<i>End state</i> adalah menggambarkan ketika objek berhenti memberi respon terhadap sebuah event. <i>End state</i> digambarkan dengan lingkaran solid di dalam sebuah lingkaran kosong.
3.	<i>State/Activities</i> 	<i>State</i> atau <i>activities</i> menggambarkan kondisi sebuah entitas, dan digambarkan dengan segiempat yang pinggirnya.
4.	<i>Fork (Percabangan)</i> 	<i>Fork</i> atau percabangan merupakan pemisalah beberapa aliran konkuren dari suatu aliran tunggal.
5.	<i>Join (Penggabungan)</i> 	<i>Join</i> atau penggabungan merupakan penggabungan beberapa aliran konkuren dalam aliran tunggal.
6.	<i>Decision</i> 	<i>Decision</i> merupakan suatu logika aliran konkuren yang mempunyai dua cabang aliran konkuren.

d. Sequence Diagram

Sequence diagram secara grafis menggambarkan bagaimana objek berinteraksi antara satu sama lain melalui pesan pada sebuah use case atau operasi.

Tabel 2.4 Simbol Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
----	--------	------	------------

1		Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		Bagaimana objek dibentuk atau diawali
3		Bagaimana objek dibentuk dan dihancurkan
4		Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2.7 Basis Data

Sistem basis data merupakan penyimpanan informasi yang terorganisasi secara komputerisasi sehingga memudahkan pemakai dalam pengolahannya dan penggunaannya (Sutanta, 2004). Pada saat ini sistem basis data sudah dikembangkan pada mesin - mesin komputer kecil sampai komputer yang lebih besar seperti mainframe. Tujuan dari hal tersebut secara keseluruhan adalah untuk melakukan perawatan informasi dan menyajikannya kapan saja dibutuhkan oleh pengguna.

Beberapa istilah dalam *database* yang sering dipakai antara lain:

1. *File*

Pengertian *file* adalah kumpulan *record-record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda-beda data *valuenya*.

2. *Record*

Record adalah kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang *entry* secara lengkap

3. *Field*

Pengertian *field* adalah suatu item informasi di antara item informasi lain yang membentuk *record*.