

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Sistem Pendukung Keputusan**

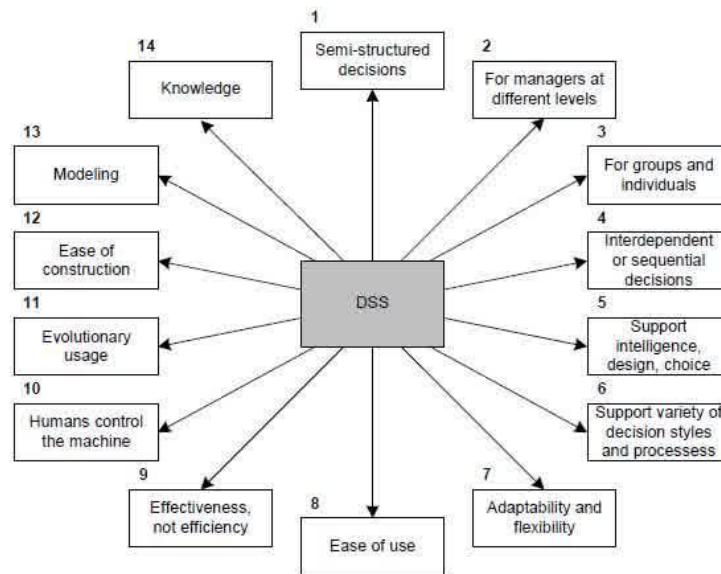
##### **2.1.1. Pengertian Sistem Pendukung Keputusan**

*Decision Support System* atau Sistem Pendukung Keputusan, 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 (Hermawan, 2005).

Pembuatan keputusan merupakan fungsi utama seorang manajer atau administrator. Kegiatan pembuatan keputusan meliputi pengidentifikasian masalah, pencarian alternatif penyelesaian masalah, evaluasi dari alternatif-alternatif tersebut dan pemilihan alternatif keputusan yang terbaik. Kemampuan seorang manajer dalam membuat keputusan dapat ditingkatkan apabila ia mengetahui dan menguasai teori dan teknik pembuatan keputusan. Dengan peningkatan kemampuan manajer dalam pembuatan keputusan diharapkan dapat ditingkatkan kualitas keputusan yang dibuatnya, dan hal ini tentu akan meningkatkan efisiensi kerja manajer yang bersangkutan.

##### **2.1.2. Karakteristik Sistem Pendukung keputusan**

Struktur karakteristik dan kapabilitas kunci yang diharapkan ada pada SPK menurut Turban, et.al, (2005) ditunjukkan pada gambar 2.1 berikut :



Gambar 2.1 Karakteristik dan Kapabilitas Kunci dari SPK (Turban, et.al, 2005)

### 2.1.3. Tujuan Sistem Pendukung Keputusan

Tujuan dari SPK menurut (Turban, et.al, 2005) adalah :

- 1) Membantu manajer dalam pengambilan keputusan atas masalah semiterstruktur.
- 2) Memberikan dukungan atas pertimbangan manajer dan bukannya dimaksudkan untuk menggantikan fungsi manajer.
- 3) Meningkatkan efektivitas keputusan yang diambil manajer lebih dari ada perbaikan efisiensinya.
- 4) Meningkatkan kecepatan komputasi. Komputer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang rendah.
- 5) Peningkatan produktivitas. Produktivitas bisa ditingkatkan menggunakan peralatan optimalisasi yang menentukan cara terbaik untuk menjalankan sebuah bisnis.

- 6) Memberi dukungan kualitas. Komputer bisa meningkatkan kualitas keputusan yang dibuat. Dengan komputer, para pengambil keputusan bisa melakukan simulasi yang kompleks, memeriksa banyak skenario yang memungkinkan, dan menilai berbagai pengaruh secara cepat dan ekonomis.
- 7) Meningkatkan daya saing. Teknologi pengambilan keputusan bisa menciptakan pemberdayaan yang signifikan dengan cara memperbolehkan seseorang untuk membuat keputusan yang baik secara cepat, bahkan jika mereka memiliki pengetahuan yang kurang.
- 8) Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan.

#### **2.1.4. Komponen Sistem Pendukung Keputusan**

Menurut Turban, et.al, (2005) dalam penerapan SPK ada beberapa komponen subsistem yang digunakan yakni:

- 1) Subsistem manajemen data

Subsistem manajemen data merupakan subsistem yang memasukan satu database yang berisi data yang relevan dan dikelola oleh perangkat lunak, yang disebut dengan Database Management System (DBMS). Dapat dikoneksikan dengan data warehouse perusahaan yang relevan untuk pengambilan keputusan.

- 2) Subsistem Manajemen Model

Subsistem manajemen model merupakan paket perangkat lunak yang memasukan model keuangan, statistik, ilmu manajemen atau model kuantitatif lainnya yang memberikan kapabilitas analitik dan manajemen

perangkat lunak yang tepat. Perangkat lunak ini sering disebut sebagai sistem manajemen basis model (MBMS). Dapat dikoneksikan ke penyimpanan eksternal yang ada pada model.

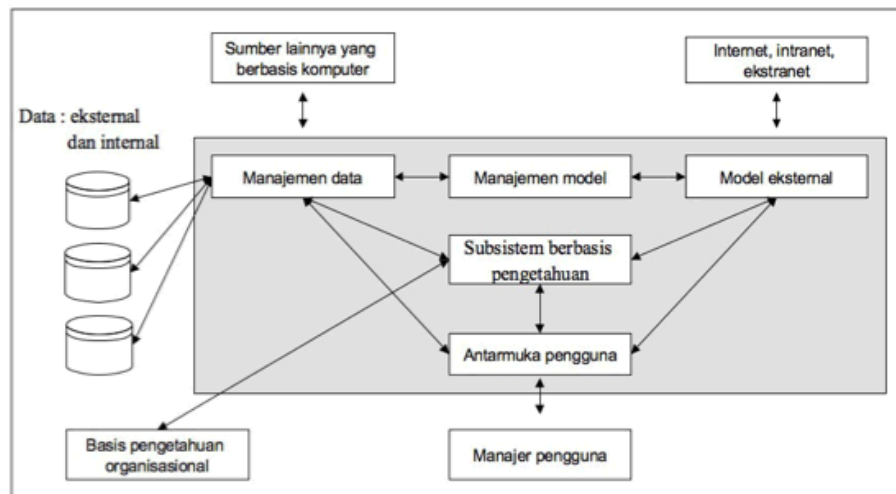
### 3) Subsistem Antarmuka Pengguna

Subsistem Antarmuka Pengguna merupakan subsistem yang dipakai oleh pengguna untuk berkomunikasi dan memerintahkan SPK untuk menjalankan fungsinya atau interaksi antara sistem komputer dengan pembuat keputusan. Pengguna adalah bagian yang dipertimbangkan dari sistem.

### 4) Subsistem Manajemen Knowledge atau Manajemen Berbasis Pengetahuan.

Subsistem ini dapat mendukung subsistem lain dan bertindak sebagai suatu komponen independen. Subsistem ini dapat memberikan intelegensi untuk memperbesar pengetahuan si pengambil keputusan.

Skematik dari SPK dan komponennya ditunjukkan pada gambar 2.2 berikut :



Gambar 2.2 Skematik dari SPK dan komponennya (Turban, et.al, 2005)

## **2.2. Pinjaman**

Pemberian pinjaman merupakan jasa atau bisnis yang beresiko, karena kemungkinan pinjaman yang diberikan tidak dapat ditagih/macet. Sehubungan dengan hal tersebut sudah menjadi keharusan bagi koperasi hanya memberikan pinjaman kepada calon peminjam yang layak dengan menseleksi setiap usulan pinjaman.

Pinjam adalah penyedia uang atau yang disamakan dengan berdasarkan persetujuan atau kesepakatan pinjam antara pihak penyedia dengan pihak lain yang mewajibkan pihak peminjam untuk melunasi hutangnya setelah jangka waktu tertentu dengan jumlah bunga. (Suharjono, 2003).

## **2.3. Penelitian Terdahulu**

Penelitian yang pernah dilakukan tentang “Perancangan dan Pembuatan Sistem Pendukung Penyeleksian Pemberian Pinjaman bagi Calon Peminjam Menggunakan Metode TOPSIS (Studi Kasus : PT Bank NTT)”. Kriteria yang dipakai adalah kemampuan membayar kembali, nilai agunan, jangka waktu, dan status rumah. Nilai yang digunakan pada tiap kriteria merupakan nilai yang dikategorikan menjadi empat kategori, yaitu sangat baik, baik, cukup, dan buruk. Metode TOPSIS akan melakukan perhitungan dan perbandingan setiap calon peminjam berdasarkan jarak nilainya dengan nilai terbaik dan nilai terburuk setiap kriteria melalui perbandingan berpasangan antar calon peminjam pada kriteria yang sama. Metode TOPSIS dalam perhitungan akan melakukan perkalian antar matriks perbandingan berpasangan antar calon peminjam yang sama dengan bobot keputusan masing-masing kriteria. Penggunaan sistem pendukung keputusan ini, diharapkan akan dapat membantu dalam menentukan calon peminjam yang layak untuk memperoleh pinjaman. Hasil yang diperoleh dari

penelitian ini berupa urutan alternatif saran atau pilihan calon peminjam yang layak memperoleh pinjaman. Hasil tersebut dapat memberikan pertimbangan penilaian kepada pihak bank untuk menentukan solusi terbaik dalam menentukan calon peminjam yang layak memperoleh pinjaman (Djami, 2011).

#### **2.4. Metode TOPSIS**

TOPSIS diperkenalkan pertama kali oleh Yoon dan Hwang pada tahun 1981 untuk digunakan sebagai salah satu metode dalam memecahkan masalah multikriteria (Sachdeva, 2009). TOPSIS memberikan sebuah solusi dari sejumlah alternatif yang mungkin dengan cara membandingkan setiap alternatif dengan alternatif terbaik dan alternatif terburuk yang ada diantara alternatif-alternatif masalah. Metode ini menggunakan jarak untuk melakukan perbandingan tersebut. TOPSIS telah digunakan dalam banyak aplikasi termasuk keputusan investasi keuangan, perbandingan performansi dari perusahaan, perbandingan performansi dalam suatu industri khusus, pemilihan sistem operasi, evaluasi pelanggan, dan perancangan robot.

Yoon dan Hwang mengembangkan metode TOPSIS berdasarkan intuisi yaitu alternatif pilihan merupakan alternatif yang mempunyai jarak terkecil dari solusi ideal positif dan jarak terbesar dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak *Euclidean* (Sachdeva, 2009). Namun, alternatif yang mempunyai jarak terkecil dari solusi ideal positif, tidak harus mempunyai jarak terbesar dari solusi ideal negatif. Maka dari itu, TOPSIS mempertimbangkan keduanya, jarak terhadap solusi ideal positif dan jarak terhadap solusi ideal negatif secara bersamaan. Solusi optimal dalam metode

TOPSIS didapat dengan menentukan kedekatan relatif suatu alternatif terhadap solusi ideal positif. TOPSIS akan merangking alternatif berdasarkan prioritas nilai kedekatan relatif suatu alternatif terhadap solusi ideal positif. Alternatif-alternatif yang telah dirangking kemudian dijadikan sebagai referensi bagi pengambil keputusan untuk memilih solusi terbaik yang diinginkan.

Metode ini banyak digunakan untuk menyelesaikan pengambilan keputusan secara praktis. Hal ini disebabkan metode Topsis memiliki keunggulan yaitu konsepnya sederhana dan mudah dipahami, komputasinya efisien, dan memiliki kemampuan mengukur kinerja relatif dari alternatif-alternatif keputusan. Topsis memiliki karakteristik yaitu bahwa alternatif yang dipilih harus memiliki jarak geometris terpendek dari solusi ideal positif dan jarak geometris terpanjang dari solusi ideal negatif. Ini adalah agregasi kompensasi yang membandingkan satu set alternatif dengan mengidentifikasi bobot untuk setiap kriteria, normalisasi bobot untuk setiap kriteria dan menghitung jarak geometris antara masing-masing alternatif dan alternatif yang ideal, yang merupakan nilai terbaik di setiap kriteria.

Langkah-langkah dari metode dimulai dengan membangun sebuah matriks keputusan.

$$D = \begin{matrix} & x_{11} & \cdots & x_{1n} \\ & \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} & \end{matrix} \dots\dots\dots(1)$$

Dimana:

$D$  =matriks

$m$  = alternatif

$n$  = kriteria

$x_{ij}$  = alternatif ke  $i$  dan kriteria ke  $j$

**a. Menghitung Matriks Ternormalisasi**

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

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}} \dots\dots\dots(2)$$

Untuk  $i=1,2,3,\dots,m$ ;

$j =1,2,3,\dots,n$

**b. Menghitung matriks ternormalisasi terbobot**

Diberikan bobot  $W=(w_1,w_2,\dots,w_n)$ , sehingga *weighted normalized matrix*  $V$  dapat dihasilkan sebagai berikut:

$$V_{ij} = [W_{ij} \cdot r_{ij}] \dots\dots\dots(3)$$

Dengan  $i =1,2,3,\dots,m$

$j =1,2,3,\dots,n$

**c. Menentukan Jarak Antara Nilai Setiap Alternatif Dengan Matriks Solusi Ideal Positif dan Matriks Solusi Ideal Negatif**

Solusi ideal positif dinotasikan dengan  $A^+$  dan solusi ideal negatif dinotasikan dengan  $A^-$  sebagai berikut :

Menentukan Solusi Ideal (+) dan Solusi Ideal (-)

$$A^+ = \left\{ \left( \max v_{ij} \mid j \in J \right) \left( \min v_{ij} \mid j \in J' \right), i = 1, 2, 3, \dots, m \right\} = \{v_1^+, v_2^+, \dots, v_m^+\}$$

$$A^- = \left\{ \left( \max v_{ij} \mid j \in J \right) \left( \min v_{ij} \mid j \in J' \right), i = 1, 2, 3, \dots, m \right\} = \{v_1^-, v_2^-, \dots, v_m^-\}$$

\dots\dots\dots(4)



Dimana :

$v_{ij}$  = elemen matriks  $V$  baris ke- $i$  dan kolom ke- $j$

$J = \{j=1,2,3,\dots,n \text{ dan } j \text{ berhubung dengan } \textit{benefit criteria}\}$

$J' = \{j=1,2,3,\dots,n \text{ dan } j \text{ berhubung dengan } \textit{cost criteria}\}$

**d. 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:

*Separation measure* untuk solusi ideal positif :

$$D_i^+ = \sum_{j=1}^n (v_{ij} - v_j^+)^2 \dots\dots\dots(5)$$

dengan  $i = 1,2,3,\dots,n$

*Separation measure* untuk solusi ideal negatif :

$$D_i^- = \sum_{j=1}^n (v_{ij} - v_j^-)^2 \dots\dots\dots(6)$$

dengan  $i = 1,2,3,\dots,n$

**e. Menghitung Nilai Preferensi Untuk Setiap Alternatif**

Kedekatan relatif dari alternatif  $D^+$  dengan solusi ideal  $D^-$  direpresentasikan dengan:

$$V_i = \frac{D_i^-}{D_i^- + D_i^+}, \dots\dots\dots(7)$$

dengan  $0 < V_i < 1$  dan  $i = 1,2,3,\dots,m$

#### f. Mengurutkan Pilihan

Alternatif dapat dirangking berdasarkan urutan  $V_i$ . Maka dari itu, alternatif terbaik adalah salah satu yang berjarak terpendek terhadap solusi ideal positif dan berjarak terjauh dengan solusi ideal negatif.

### 2.5. Cost Benefit Analisis

Pengertian Cost Benefit Analisis menurut pendapat para ahli antara lain:

- a) Menurut Mare J. Schniederjans, Jamie L. Hamaker, Ashlyn M. Schiederjans (2004), *Cost Benefit Analysis* adalah suatu teknik untuk menganalisis biaya dan manfaat yang melibatkan estimasi dan mengevaluasi dari manfaat yang terkait dengan alternatif tindakan yang akan dilakukan. Teknik ini membandingkan nilai manfaat kini dengan investasi dari biaya investasi yang sama sebagai alat bantu dalam pengambilan keputusan.
- b) Menurut Keen (2003), mendefinisikan *Cost Benefit Analysis* sebagai analisis yang menjabarkan alasan bisnis, kenapa atau kenapa tidak pilih spesifik suatu investasi harus dipilih.
- c) Menurut Siegel dan Shimp (1994), *Cost Benefit Analysis* adalah cara untuk menentukan apakah hasil yang menguntungkan dari sebuah alternatif, akan cukup untuk dijadikan alasan dalam menentukan biaya pengambilan alternatif. Analisa ini telah dipakai secara luas dalam hubungannya dengan proyek pengeluaran modal.

Berdasarkan pengertian *Cost benefit analysis* yang disampaikan para ahli di atas, dapat disimpulkan bahwa *cost benefit analysis* memiliki pengertian suatu analisis sistematis yang berupa perbandingan antara manfaat dan biaya yang

dikeluarkan dalam menyelenggarakan kegiatan atau proyek. Tujuan akhir yang ingin dicapai adalah secara akurat membandingkan kedua nilai, manakah yang lebih besar. Selanjutnya dari hasil perbandingan ini, pengambil keputusan dapat mempertimbangkan untuk melanjutkan suatu rencana atau tidak dari sebuah aktivitas, produk atau proyek, atau dalam konteks evaluasi atas sesuatu yang telah berjalan, adalah menentukan keberlanjutannya.

Kriteria *benefit* adalah kriteria dimana pengambil keputusan menginginkan nilai maksimum diantara seluruh nilai alternatif. Kriteria *cost* adalah kriteria dimana pengambil keputusan menginginkan nilai minimum diantara seluruh nilai alternatif.

## 2.6. MySQL

MySQL merupakan suatu database. MySQL dapat juga dikatakan sebagai database yang sangat cocok bila dipadukan dengan PHP. Secara umum, database berfungsi sebagai tempat atau wadah untuk menyimpan. Mengklarifikasikan data secara profesional. MySQL bekerja menggunakan *SQL Language (Structure Query Language)*. Itu dapat diartikan bahwa MySQL merupakan standar penggunaan database di dunia untuk pengolahan data.

MySQL termasuk jenis RDBMS (*Relational Database Management System*). Sedangkan RDBMS sendiri akan lebih banyak mengenal istilah seperti tabel, baris, dan kolom digunakan dalam perintah-perintah di MySQL. MySQL merupakan sebuah basis data yang mengandung satu atau sejumlah tabel. Tabel terdiri dari atas sejumlah baris dan setiap baris mengandung satu atau sejumlah kolom.

Pada umumnya, perintah yang paling sering digunakan dalam MySQL adalah *Select* (mengambil), *insert* (menambah), *update* (mengubah), dan *delete* (menghapus). Selain itu, SQL juga menyediakan perintah untuk membuat *database*, *field*, ataupun *index* guna menambah atau menghapus data (Kustiyaningsih, 2010).

## 2.7. PHP

PHP (atau resminya PHP: *Hypertext Preprocessor*) adalah skrip bersifat *server-side* yang ditambahkan ke dalam HTML. PHP sendiri merupakan singkatan dari *Personal Home Page Tools*. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu halaman web tidak lagi bersifat statis, namun bersifat dinamis. Sifat *server-side* berarti pengerjaan skrip akan dilakukan di *server*, baru kemudian hasilnya dikirimkan ke *browser* (Kurniawan, 2001).

Keunggulan dari sifatnya yang *server-side* tersebut antara lain :

- a) Tidak diperlukan kompatibilitas *browser* atau harus menggunakan *browser* tertentu, karena *server*-lah yang akan mengerjakan skrip PHP. Hasil yang dikirimkan kembali ke *browser* umumnya bersifat teks atau gambar saja sehingga pasti dikenal oleh *browser* apapun.
- b) Dapat memanfaatkan sumber-sumber aplikasi yang dimiliki oleh *server*, misalnya koneksi ke *database*.
- c) Skrip tidak dapat “diintip” dengan menggunakan fasilitas *view HTML source*.

## 2.8. HTML

HTML kependekan dari *HyperText Markup Language*. Dokumen HTML adalah file teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *web page*. Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*. Dokumen ini umumnya berisi informasi atau interface aplikasi di dalam Internet. Ada dua cara untuk membuat sebuah *web page*, dengan HTML editor atau dengan editor teks biasa (misalnya notepad) (Sidik dan Pohan, 2009).

Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML (*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium* (W3C). Versi terakhir dari HTML adalah HTML 5.01, meskipun saat ini telah berkembang XHTML yang merupakan pengembangan dari HTML.

HTML berupa kode-kode tag yang menginstruksikan browser untuk menghasilkan tampilan sesuai dengan yang diinginkan. Sebuah file yang merupakan file HTML dapat dibuka dengan menggunakan *web browser* seperti *Mozilla Firefox* atau *Microsoft Internet Explorer*. HTML juga dapat dikenali oleh aplikasi pembuka email ataupun dari PDA dan program lain yang memiliki kemampuan *browser*.

## 2.9. CSS

CSS merupakan singkatan dari *Cascading Style Sheet*. Kegunaannya adalah untuk mengatur tampilan dokumen HTML, contohnya seperti pengaturan jarak antar baris, teks, warna dan format border bahkan penampilan file gambar. CSS dikembangkan oleh W3C, organisasi yang mengembangkan teknologi internet. Tujuannya tak lain untuk mempermudah proses penataan web. Perlu diingat, CSS hanyalah berupa kumpulan script yang tujuannya bukan untuk menggantikan HTML, melainkan pelengkap agar dokumen HTML bisa tampil lebih cantik dan dinamis.

Sejak ditemukannya CSS pada awal dekade sembilan puluhan, CSS terus dikembangkan dan diserap oleh *web developer*. Hingga sekarang telah mencapai versi ke-3. Kode CSS bersifat lintas *platform*, yang berarti script ini dapat dibaca oleh berbagai macam sistem operasi dan browser. Hanya saja browser seperti Internet Explorer, seringkali salah mengartikan script CSS yang menyebabkan ketidaksempurnaannya tampilan dokumen HTML. Script CSS perlu dioptimalkan agar tampil maksimal pada *browser internet explorer* (Jayan, 2010).

## 2.10. UML

*Unified Modeling Language* (UML) merupakan alat merancang perangkat lunak, sarana komunikasi antara perangkat lunak dengan, menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem, mendokumentasikan sistem yang ada, proses-proses dan organisasinya (Herlawati, 2011).

Salah satu cara untuk mengatur diagram UML adalah dengan menggunakan *view*. *View* adalah kumpulan diagram yang menggambarkan aspek yang sama dari proyek. *View* mempunyai 3 pelengkap, yaitu *Static View*, *Dynamic View*, dan *Functional View*.

#### **2.10.1. *Static View***

*Static View* termasuk diagram yang memberikan gambaran dari unsur-unsur dari sistem tetapi tidak memberitahu bagaimana elemen akan berperilaku. Hal ini sangat mirip *Blueprint*. *Blueprint* itu komprehensif, tetapi mereka hanya menunjukkan apa yang tetap diam, maka disebut *Static View*. *Static View* dibentuk oleh dua diagram, yaitu *Class Diagram* dan *Object Diagram*.

#### **2.10.2. *Dynamic View***

Pada *Dynamic View* meliputi diagram yang mengungkapkan bagaimana benda berinteraksi dengan satu sama lain dalam respon terhadap lingkungan. Ini termasuk *Sequence Diagram* dan *Collaboration Diagram*, yang kolektif disebut sebagai diagram interaksi. Mereka secara khusus dirancang untuk menjelaskan bagaimana benda berbicara satu sama lain. Ini juga mencakup *Statechart Diagram*, yang menunjukkan bagaimana dan mengapa perubahan objek dari waktu ke waktu dalam menanggapi lingkungan.

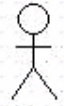
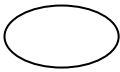

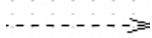
#### **2.10.3. *Functional View***

*Functional View* terbentuk oleh *Use Case Diagram* dan *Activity Diagram*.

##### **a. *Use Case Diagram***

*Use case* diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Berikut simbol-simbol yang ada pada diagram *use case* :

Tabel 2.1 Simbol *Use Case Diagram*










Simbol	Keterangan
	<i>Actor</i> : Peran yang dimainkan oleh seseorang, sistem atau perangkat yang memiliki saham dalam keberhasilan operasi dari sistem.
	<i>Use Case</i> : Mengungkapkan tujuan bahwa sistem harus dicapai.
	<i>Association</i> : mengidentifikasi interaksi antara aktor dan <i>Use Case</i>
	<i>Dependency</i> : Mengidentifikasi hubungan komunikasi antara dua <i>Use Case</i>

Sumber : Shalahuddin dan Rosa, 2011

### b. *Activity Diagram*

Diagram *aktivitas* atau *activity diagram* menggambarkan *workflow* atau aktivitas dari sebuah sistem atau proses bisnis. Berikut simbol-simbol yang ada pada diagram aktivitas :

Tabel 2.2 Simbol *Activity Diagram*

Simbol	Keterangan
	Titik Awal
	Titik Akhir
	Activity
	Pilihan untuk pengambilan keputusan
	Fork menunjukkan kekuatan yang dilakukan secara parallel
	Rake menunjukkan adanya dekomposisi
	Tanda Waktu
	Tanda Penerimaan
	Aliran Akhir (Flow Final)

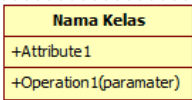



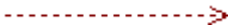

Sumber :Shalahuddin dan Rosa , 2011



### c. Class Diagram

Kelas Diagram terdiri dari tiga kompartemen (ruang persegi panjang) yang mengandung informasi yang berbeda diperlukan untuk menjelaskan sifat-sifat satu jenis objek. Berikut simbol-simbol yang digunakan dalam *class diagram*:

Tabel 2.3 Simbol *Class Diagram* (Shalahuddin dan Rosa, 2011)




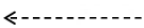


Simbol	Deskripsi
Kelas 	Kelas pada struktur system
Paket/package 	Paket/package merupakan sebuah bungkusan dari satu atau lebih kelas (kumpulan kelas)
Asosiasi 	Asosiasi merupakan hubungan antar kelas dengan makna umum, asosiasi biasanya jua disertai dengan <i>multiplicity</i>
Generalisasi 	Generalisasi merupakan hubungan generalisasi dan spesialisasi (umum khusus) antara dua kelas dimana fungsi yang satu adalah fungsi yang lebih umum dari fungsi yang lainnya
Dependency 	Dependency merupakan hubungan antarkelas yang saling bergantung, membutuhkan satu sama lain.
Agregasi 	Agregasi merupakan hubungan antar kelas dimana satu kelas merupakan semua bagian dari kelas-kelas yang lain.

Sumber :Shalahuddin dan Rosa, 2011

#### d. *Sequence Diagram*

*Sequence Diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Berikut adalah simbol-simbol yang ada pada *sequence diagram*:

Tabel 2.4 Simbol *Sequence Diagram* (Shalahuddin dan Rosa, 2011).

<b>Nama</b>	<b>Simbol</b>	<b>Keterangan</b>
Objek dan kelas		Objects mewakili peserta
<i>Lifeline</i>		<i>Lifeline</i> merupakan siklus hidup sebuah pesan/message berdasarkan waktu
<i>Message</i>		Pesan/ <i>message</i> merupakan komunikasi antara objek yang satu dengan objek yang lainnya
<i>Return message</i>		<i>Return message</i> merupakan balasan/hasil yang berisi nilai dari sebuah objek yang meminta (mengirim pesan)
<i>Self message</i>		<i>Self message</i> merupakan pesan dari sebuah objek kepada objek itu sendiri untuk melakukan suatu aksi
<i>Return self message</i>		Balasan/hasil dari <i>self message</i> yang berisi suatu nilai kepada objek itu sendiri

Sumber : Shalahuddin dan Rosa, 2011