

## **BAB II**

### **LANDASAN TEORI**

#### **II.1. Sistem**

##### **II.1.1. Konsep Dasar Sistem**

Terdapat dua kelompok pendekatan di dalam pendefinisian sistem, yaitu sekelompok yang menekankan pada prosedur dan kelompok yang menekankan pada elemen atau komponennya. Pendekatan yang menekankan pada prosedur dan mendefinisikan sistem sebagai suatu jaringan kerja prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponen mendefinisikan sistem sebagai kumpulan elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. kedua kelompok definisi ini adalah benar dan tidak bertentangan. Yang berbeda adalah cara pendekatannya. (Tata Sutabri : 2012 ; 2).

##### **II.1.2. Pengertian Sistem**

Menurut (Kusrini : 2007 ; 11) Sistem merupakan kumpulan elemen yang saling berkaitan yang bertanggung jawab memproses masukan (*input*) sehingga menghasilkan keluaran (*output*).

Mempelajari suatu sistem akan lebih mengena bila mengetahui terlebih dahulu apakah sistem itu. Pengertian tentang sistem pertama kali dapat diperoleh dari definisi sistem itu sendiri. Jika kita perhatikan dengan seksama, diri kita juga

terdiri dari berbagai sistem yang berfungsi untuk mengantar kita pada tujuan hidup kita. (Tata Sutabri : 2012 ; 4).

### **II.1.3. Data**

Menurut (Kusrini : 2007 ; 3) Data merupakan representasi dari fakta atau gambaran mengenai suatu objek atau kejadian, ambil contoh fakta mengenai biodata mahasiswa yang meliputi nama, alamat, jenis kelamin, agama yang dianut, dan lain-lain. Contoh lain dari fakta mengenai kejadian /transaksi dalam sebuah perusahaan dagangan adalah seperti transaksi penjualan yang meliputi waktu transaksi, pelaku transaksinya (pelanggan, kasir), barang yang ditransaksikan, serta jumlah dan harganya. Data dinyatakan dengan nilai yang berbentuk angka, deretan karakter, atau simbol.

### **II.1.4. Informasi**

Informasi merupakan hasil olahan data, dimana data tersebut sudah diproses dan diinterpretasikan menjadi sesuatu yang bermakna untuk pengambilan keputusan, informasi juga diartikan sebagai himpunan dari data yang relevan dengan satu atau beberapa orang dalam suatu waktu.

Suatu informasi berguna bagi pembuatan keputusan karena informasi bisa menurunkan ketidakpastian (meningkatkan pengetahuan) tentang hal yang sedang dipikirkan. Makna dari sebuah informasi tentu berbeda-beda antara seorang dengan lainnya, tergantung pada tingkat kepentingannya, misalnya informasi daftar pelanggan yang potensial akan sangat dibutuhkan oleh bagian marketing di

suatu perusahaan guna meningkatkan penjualan produk, tetapi barangkali tidak akan menjadi perhatian dibagian personalia.

Kegunaan informasi bagi seseorang juga sangat tergantung pada waktu, pada suatu waktu tertentu informasi tersebut mungkin sangat diperlukan dilain hari, mungkin saja hal tersebut sudah tidak berguna sama sekali. Contohnya, informasi perbandingan harga barang akan sangat dibutuhkan oleh seseorang yang akan membeli barang tersebut. Namun saat ini dia sedang tidak mempertimbangkan untuk membeli barang tersebut. Informasi tersebut menjadi kurang bermakna Menurut (Kusrini : 2007 ; 4).

#### **II.1.5. Kualitas Informasi**

Menurut (Kusrini : 2007 ; 4-5) Agar bisa menyediakan keluaran yang berguna untuk membantu manager atau para pengambil keputusan, sebuah sistem informasi harus mampu mengumpulkan data dan mentransformasikan data tersebut kedalam informasi yang memiliki kualitas-kualitas tersebut. Berikut karakteristik informasi yang berkualitas :

- 1. Relevan.** Informasi yang disajikan sebaiknya terkait dengan keputusan yang akan diambil oleh pengguna informasi tersebut. Misalnya, seorang manager yang akan memberikan kredit kepada pelanggan bisa melihat laporan keuangan pelanggan tersebut karena laporan tersebut terkait dengan keputusan yang akan dibuat, yaitu memberikan atau tidak memberikan kredit kepada pelanggan tersebut.

2. **Akurat.** Kecocokan antara informasi dengan kejadian-kejadian atau objek-objek yang diwakilinya. Misalnya, laporan inventaris yang tidak akurat menyebutkan bahwa terdapat 15 unit barang di dalam gudang.
3. **Lengkap.** Merupakan derajat sampai seberapa jauh informasi menyertakan kejadian-kejadian atau objek-objek yang berhubungan. Misalnya, penjualan selama satu hari yang seharusnya ada 150 transaksi dilaporkan hanya tercatat sebanyak 145 transaksi.
4. **Tepat Waktu.** Informasi yang tidak tepat waktu akan menjadi informasi yang tidak berguna atau tidak dapat digunakan untuk membantu pengambilan keputusan. Misalnya, informasi jadwal ujian seorang mahasiswa disampaikan setelah kegiatan ujian diselenggarakan. Informasi ini menjadi tidak berguna lagi.
5. **Dapat dipahami.** Hal tersebut terkait dengan bahasa dan cara penyajian informasi agar pengguna lebih mudah mengambil keputusan.
6. **Dapat dibandingkan.** Sebuah informasi yang memungkinkan seorang pemakai untuk mengidentifikasi persamaan dan perbedaan antara dua objek atau kejadian yang mirip. Misalnya, membandingkan laporan pendapatan antara tahun 2006 dan 2007.

#### **II.1.6. Sistem Informasi**

Menurut (Kusrini : 2007 ; 11) suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi,

bersifat manajerial, dan merupakan kegiatan strategi dari suatu organisasi, serta menyediakan laporan-laporan yang diperlukan untuk pihak luar.

Berdasarkan dukungan kepada pemakainnya, sistem informasi dibagi menjadi :

1. Sistem Pemrosesan Transaksi (*Transaksi Processing System*) atau TPS.
2. Sistem Informasi Manajemen (*Manajemen Information System*) atau MIS.
3. Sistem Otomasi Perkantoran (*Office Automation System/OAS*).
4. Sistem Pendukung Keputusan (*Decision Support System*) atau DSS.
5. Sistem Informasi Eksekutif (*Executive Information System*) atau EIS.
6. Sistem Pendukung Kelompok (*Group Support System*) atau GSS.
7. Sistem Pendukung Cerdas (*Inteleigent Support System*) atau ISS.

Mengingat bahwa EIS (*Executive Information System*), *Decision Support System*, dan MIS (*Manajemen Information System*) digunakan untuk pendukung manajemen, maka ketiga system tersebut sering disebut Sistem Pendukung Manajemen (*management support system*) atau MSS.

## **II.2. Sistem Pendukung Keputusan**

Menurut (Kusrini : 2007 ; 15) Sistem Pendukung Keputusan merupakan Sistem Informasi interaktif yang menyediakan informasi pemodelan dan memanipulasi data. Sistem itu digunakan untuk membantu pengambilan

keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat.

Sistem Pendukung Keputusan biasanya dibangun untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang. *Decision Support System* yang seperti itu disebut dengan aplikasi *Decision Support System*). Aplikasi *Decision Support System* digunakan dalam pengambilan keputusan. Aplikasi *Decision Support System* menggunakan CBIS (Computer Based Information Systems) yang fleksibel, interaktif, dan dapat diadaptasi yang dikembangkan untuk mendukung solusi atau masalah manajemen spesifik yang tidak terstruktur.

Aplikasi *Decision Support System* menggunakan data, memberikan antara muka pengguna yang mudah dan dapat digabungkan pengguna yang mudah, dan dapat digabungkan pemikiran pengambilan keputusan.

*Decision Support System* lebih ditujukan untuk mendukung manajemen dalam melakukan pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang kurang jelas.

*Decision Support System* tidak dimaksudkan untuk mengotomatiskan pengambilan keputusan tetapi memberikan perangkat interaktif yang memungkinkan pengambilan keputusan untuk melakukan berbagai analisis menggunakan model-model yang tersedia.

### II.2.1. Karakteristik Sistem Pendukung Keputusan

Menurut (Kusrini : 2007 ; 25) Aplikasi sistem pendukung keputusan bisa berdiri dari beberapa subsistem, yaitu :

1. Subsistem manajemen data

Subsistem manajemen data memasukkan satu database yang berisi data yang relevan untuk suatu situasi dan dikelola oleh perangkat lunak yang disebut sistem manajemen data bisa diinterkoneksi dengan data *warehouse* perusahaan, suatu repository untuk data perusahaan yang relevan dengan pengambilan keputusan.

2. Subsistem manajemen modal

Merupakan paket perangkat lunak yang memasukkan model keuangan, statistik, ilmu manajemen, atau model kuantitatif lain yang memberikan kapabilitas analitik dan manajemen perangkat lunak yang tepat. Bahasa-bahasa pemodelan untuk membangun model-model kustom jug dimasukkan. Perangkat lunak itu sering disebut sistem manajemen basis model (MBMS), komponen tersebut bisa dikoneksikan ke penyimpanan korporat atau eksternal yang ada pada model.

3. Subsistem antarmuka pengguna

Pengguna berkomunikasi dengan dan memerintahkan sistem pendukung keputusan melalui subsistem tersebut. Pengguna adalah bagian yang dipertimbangkan dari sistem. Para peneliti menegaskan bahwa beberapa kontribusi unik dari sistem pendukung keputusan

berasal dari interaksi yang interaktif antara computer dan pembuat keputusan.

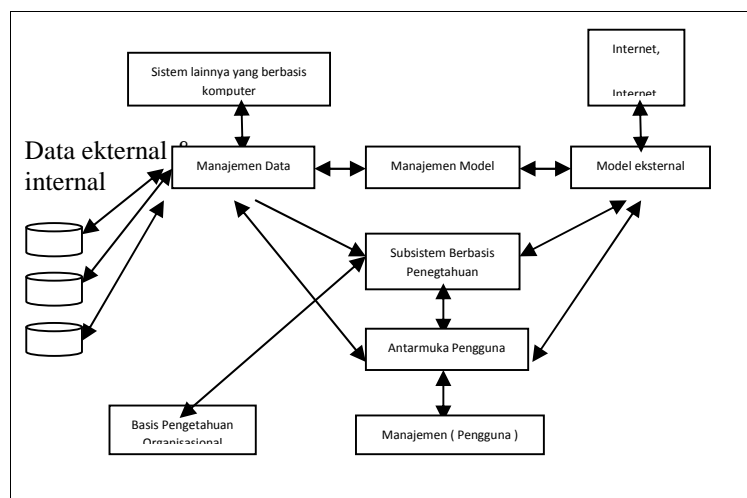
#### 4. Subsistem manajemen berbasis-pengatahuan

Subsistem tersebut mendukung semua subsistem lain atau bertindak langsung sebagai suatu komponen independent dan bersifat opsional.

Selain memberikan intelegensi untuk memperbesar pengetahuan si pengambil keputusan, subsistem tersebut bisa diinterkoneksi dengan repository pengetahuan perusahaan (bagian dari sistem manajemen pengetahuan), yang kadang-kadang disebut basis pengetahuan organisasional.

Berdasarkan defenisi, sistem pendukung keputusan harus mencakup tiga komponen utama dari DBMS, MBMS, dan antarmuka pengguna. Subsistem manajemen berbasis pengetahuan adalah opsional, tetapi bisa memberikan banyak manfaat karena memberikan intelegensi bagi ketiga komponen utama tersebut, seperti pada semua sistem pendukung keputusan. Komponen-komponen tersebut membentuk sistem aplikasi sistem pendukung keputusan yang bisa dikoneksikan ke intranet perusahaan, ekstranet, atau internet. Arsitektur dari sistem pendukung keputusan ditunjukkan dalam tabel II.1 berikut.

Tabel II.1 Arsitektur DSS



(Sumber : Kusri : 2007 ; 26)

### II.3. Metode Profile Matching

*Profile Matching* adalah sebuah mekanisme pengambilan keputusan dengan mengasumsikan bahwa terdapat tingkat variabel prediktor yang ideal yang harus dimiliki oleh pelamar, bukannya tingkat minimal yang harus dipenuhi dan dilewati. Dalam pencocokan profil, dilakukan identifikasi terhadap kelompok tersebut diukur menggunakan beberapa kriteria penilaian. Jika pelaku yang baik memperoleh skor yang berbeda dari pelaksana yang buruk atau sebuah karakteristik, maka variabel tersebut bermanfaat untuk memilih pelaksanaan yang baik. Begitu beberapa variabel yang membedakan antara pelaksana-pelaksana yang baik dan buruk telah teridentifikasi, profil ideal dari karyawan yang berhasil bisa di buat. (Kusri : 2007 ; 53).

### II.3.1. Langkah-Langkah Metode *Profile Matching*

Menurut (Nina Sherly : 2013 ; 44-45) adapun langkah-langkah dalam penyelesaian metode *Profile Matching* adalah sebagai berikut:

#### 1. Perhitungan *GAP* Kompetensi

Setelah proses pemilihan kandidat, pada proses ini ialah perhitungan pemetaan *gap* kompetensi dimana yang dimaksud dengan *gap* adalah beda antara profil jabatan dengan profil karyawan atau dapat ditunjukkan pada rumus di bawah ini:

$$\mathbf{Gap = Profil Karyawan - Profil Jabatan}$$

#### 2. Perhitungan Pemetaan *GAP*

Kompetensi berdasarkan aspek-aspek untuk pengumpulan *gap-gap* yang terjadi itu sendiri pada tiap aspeknya mempunyai perhitungan yang berbeda-beda. Setelah didapatkan tiap *gap* masing-masing karyawan maka tiap profil karyawan diberi bobot nilai dengan patokan tabel nilai *gap* seperti yang dapat di lihat pada Tabel II.2. di bawah ini:

**Tabel II.2. Keterangan Bobot Nilai *Gap***

No.	Selisih	Bobot Nilai	Keterangan
1	0	5	Tidak ada selisih (kompetensi sesuai dengan yang dibutuhkan).
2	1	4,5	Kompetensi individu kelebihan 1 tingkat/level.
3	-1	4	Kompetensi individu kekurangan 1 tingkat/level.
4	2	3,5	Kompetensi individu kelebihan 2 tingkat/level.
5	-2	3	Kompetensi individu kekurangan 2 tingkat/level.

6	3	2,5	Kompetensi individu kelebihan 3 tingkat/level.
7	-3	2	Kompetensi individu kekurangan 3 tingkat/level.
8	4	1,5	Kompetensi individu kelebihan 4 tingkat/level.
9	-4	1	Kompetensi individu kekurangan 4 tingkat/level.

(Sumber : Nina Sherly ; 2013)

### 3. Perhitungan dan Pengelompokan *Core* dan *Secondary Factor*

Setelah menentukan bobot nilai *gap* untuk setiap kriteria, kemudian tiap aspek kriteria di kelompokkan menjadi 2 (dua) kelompok yaitu kelompok *Core Factor* dan *Secondary Factor*. Untuk perhitungan *Core Factor* dapat dilihat pada rumus di bawah ini:

$$NCF = \frac{\sum NC(i, s, p)}{\sum IC}$$

Keterangan:

NCF : Nilai rata-rata *core factor*

NC(i, s, p) : Jumlah total nilai *core factor*

IC : Jumlah *item core factor*

Sedangkan untuk perhitungan *secondary factor* dapat di lihat pada rumus di bawah ini:

$$NCS = \frac{\sum NS(i, s, p)}{\sum IS}$$

Keterangan:

NSF : Nilai rata-rata *secondary factor*

NS(i, s, p) : Jumlah total nilai *secondary factor*

ICS : Jumlah *item secondary factor*

#### 4. Perhitungan Nilai Total

Dari hasil perhitungan dari tiap aspek di atas kemudian dihitung nilai total berdasarkan presentasi dari *core* dan *secondary factor* yang diperkirakan berpengaruh terhadap kinerja tiap-tiap profil. Contoh perhitungan dapat dilihat pada rumus di bawah ini:

$$N(i, s, p) = (x)\%NCF(i, s, p)(x)\%NSF(i, s, p)$$

Keterangan:

(i,s,p) : (Intelektual, Sikap Kerja, Perilaku)

N(i,s,p) : Nilai rata-rata *core factor*

NCF (i,s,p) : Nilai rata-rata *secondary factor*

(x)% : Nilai persen yang diinputkan

#### 5. Perhitungan Penentuan Hasil Akhir Atau Ranking

Hasil akhir dari proses ini adalah ranking dari kandidat yang diajukan.

Adapun perhitungan tersebut dapat dilihat pada rumus di bawah ini:

$$Ha = (x)\%Ni + (x)\%Ns + (x)\%Np$$

Keterangan:

Ha : Hasil akhir

Ni : Nilai Intelektual

Ns : Nilai Sikap

Np : Nilai Perilaku

(x)% : Nilai persen yang diinputkan

#### **II.4. Pengertian Database**

Sebuah basis data adalah sebuah kumpulan data yang saling berhubungan secara logis, dan merupakan sebuah penjelasan dari data tersebut, yang didesain untuk menemukan data yang dibutuhkan oleh sebuah organisasi. Di dalam basis data, semua data diintegrasikan dengan departemen dan pemakai. Basis data tidak hanya memegang data operasional organisasi, tetapi juga penjelasan mengenai data tersebut. Karena alasan tersebut basis data dapat juga dideskripsikan sebagai kumpulan data yang saling terintegrasi. Basis data juga merupakan sekumpulan elemen data terintegrasi yang secara logika saling berhubungan. Basis data mengonsolidasikan berbagai catatan yang terlebih dahulu disimpan dalam file-file terpisah ke dalam satu gabungan umum elemen data yang menyediakan data untuk banyak aplikasi. Elemen data mendeskripsikan entitas-entitas dan hubungan antara entitas-entitas tersebut (Indrajani : 2015 ; 70).

Database adalah kumpulan data yang saling terkait yang diorganisasi untuk memenuhi kebutuhan dan struktur sebuah organisasi serta bisa digunakan oleh lebih dari satu orang dan lebih dari satu aplikasi. (Kusrini : 2007 ; 33).

#### **II.5. Normalisasi**

Normalisasi adalah suatu teknik dengan pendekatan *bottom-up* yang digunakan untuk membantu mengidentifikasi hubungan. Dimulai dari menguji hubungan, yaitu *functional dependencies* antara atribut. Pengertian lainnya adalah suatu teknik yang menghasilkan sekumpulan hubungan dengan sifat-sifat yang diinginkan dan memenuhi kebutuhan pada perusahaan. (Indrajani : 2015 ; 7).

Adapun bentuk-bentuk normalisasi menurut (Indrajani : 2015 ; 9-10) adalah sebagai berikut:

1. Unnormalized Form (UNF)

Merupakan suatu tabel yang berisikan satu atau lebih grup yang berulang. Membuat tabel yang *unnormalized*, yaitu dengan memindahkan data dari sumber informasi.

Contoh: nota penjualan yang disimpan ke dalam format tabel dengan baris dan kolom.

2. First Normal Form (1NF)

Merupakan sebuah relasi di mana setiap baris dan kolom berisikan satu dan hanya satu nilai.

Proses UNF ke 1NF

a. Tentukan satu atau sekumpulan atribut sebagai kunci untuk tabel *unnormalized*.

b. Identifikasikan grup yang berulang dalam tabel *unnormalized* yang berulang untuk kunci atribut.

3. Second Normal Form (2NF)

Berdasarkan pada konsep *full functional dependency*, yaitu A dan B merupakan atribut sebuah relasi. B dikatakan *fully dependent* terhadap A jika B *functional dependent* pada A tetapi tidak pada propersubset dari A. 2NF merupakan sebuah relasi dalam 1NF dan setiap atribut *non-primary-key* bersifat *fully functionally dependent* pada primary key.

1NF ke 2NF

- a. Identifikasikan *primary key* untuk relasi 1NF.
- b. Identifikasikan *functional dependencies* dalam relasi.
- c. Jika terdapat *partial dependencies* terhadap *primary key*, maka hapus dengan menempatkan dalam relasi yang baru bersama salinan determinannya.

4. Third Normal Form (3NF)

Berdasarkan pada konsep *transitive dependency*, yaitu suatu kondisi di mana A, B, dan C merupakan atribut sebuah relasi, maka A – B dan B – C, maka *transitively dependent* pada A melalui B. 3NF adalah sebuah relasi dalam 1NF dan 2NF, di mana tidak terdapat atribut *non primary key* yang bersifat *transitively dependent* pada *primary key*.

2NF ke 3NF

- a. Identifikasi *primary key* dalam relasi 2NF.
- b. Identifikasi *functional dependencies* dalam relasi.
- c. Jika terdapat *transitive dependencies* terhadap *primary key* hapus dengan menempatkannya dalam relasi yang baru bersama dengan salinan determinannya.

5. Boyce-code Normal Form (BCNF)

Berdasarkan pada *functional dependencies* yang dimasukkan dalam hitungan seluruh *candidate key* dalam suatu relasi. Bagaimanapun BCNF juga memiliki batasan-batasan tambahan disamakan dengan defenisi umum dari 3NF. Suatu relasi dikatakan BCNF, jika dan hanya

jika setiap determinan merupakan *candidate key*. Perbedaan antara 3NF dan BCNF yaitu untuk *functional dependent*  $A - B$ , 3NF memungkinkan *dependency* ini dalam suatu relasi jika adalah atribut *primary key* dan A bukan merupakan *candidate key*. Sedangkan BCNF menetapkan dengan jelas bahwa untuk *dependency* ini agar ditetapkan dalam relasi A, maka A harus merupakan *candidate key*. Setiap relasi dalam BCNF juga merupakan 3NF, tetapi relasi dalam 3NF belum tentu termasuk ke dalam BCNF. Dalam BCNF kesalahan jarang sekali terjadi, Kesalahan dapat terjadi pada relasi yang:

- a. Terdiri atas 2 atau lebih *composite candidate key*.
- b. *Candidate key overlap*, sedikitnya satu atribut.

## II.6. *Unified Modeling Language (UML)*

UML adalah metode pemodelan berorientasi objek. Sebelum UML telah banyak dibuat pemodelan berorientasi objek seperti: Simula -67, Objective C, C++, Eiffel, dan CLOS. Sesuai dengan namanya, *Unified*, pada dasarnya UML merupakan pemodelan yang generik, Sifatnya yang general memudahkan UML untuk digunakan pada berbagai tipe sistem yang dimodelkan.

Pada level atas, cara pandang dalam UML dapat dibagi menjadi tiga area: *structural clasification*, *dynamic behavior*, dan model *management*. Klasifikasi struktural menjelaskan suatu objek dalam sistem dan hubungan antar objek tersebut. Pembahasan di dalamnya menyangkut *class*, *use case*, komponen dan *node*. *Dynamic behavior* menjelaskan perilaku sistem dari waktu ke waktu.

Termasuk dalam *dynamic, behavior view* adalah *state machine view, activity view, dan interaction view*. Model *management* menjelaskan pengorganisasian model secara hierarkis. (Maman Abdurohman, dkk : 2010 ; 93).

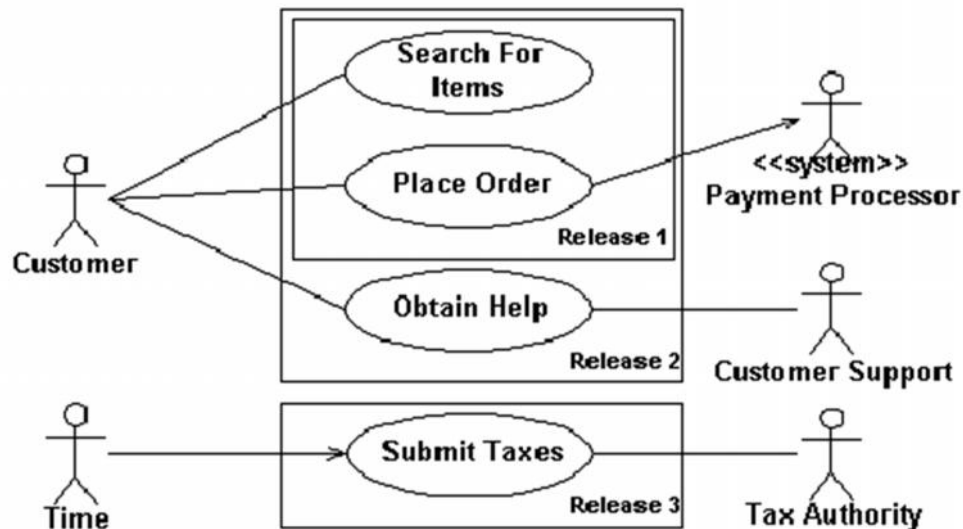
Unified Modelling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML dapat dibuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman. Tetapi karena UML juga menggunakan class dan operation dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek seperti C++, Java, atau VB.Net. (Prastuti Sulistyorini : 2009 ; 23-24).

Menurut (Indrajani : 2015 ; 45) diagram-diagram yang termasuk di dalam UML, yaitu:

### **1. *Use Case Diagram***

Merupakan suatu diagram yang berisi *use case*, actor, serta *relationship* di antaranya. Use case diagram merupakan titik awal yang baik dalam memahami dan menganalisis kebutuhan apa saja yang diperlukan dari suatu sistem. Jadi, dapat digambarkan dengan detail bagaimana suatu sistem memproses atau melakukan sesuatu, bagaimana cara actor akan menggunakan sistem, serta apa saja yang dapat dilakukan terhadap suatu sistem. Notasi yang digunakan dalam

Use Case adalah persegi panjang yang merupakan system boundary, oval yang merupakan suatu proses, dan gambar orang yang berinteraksi dalam proses tersebut. Adapun mengilustrasikan actor, use case dan boundary seperti Gambar II.1. di bawah ini:

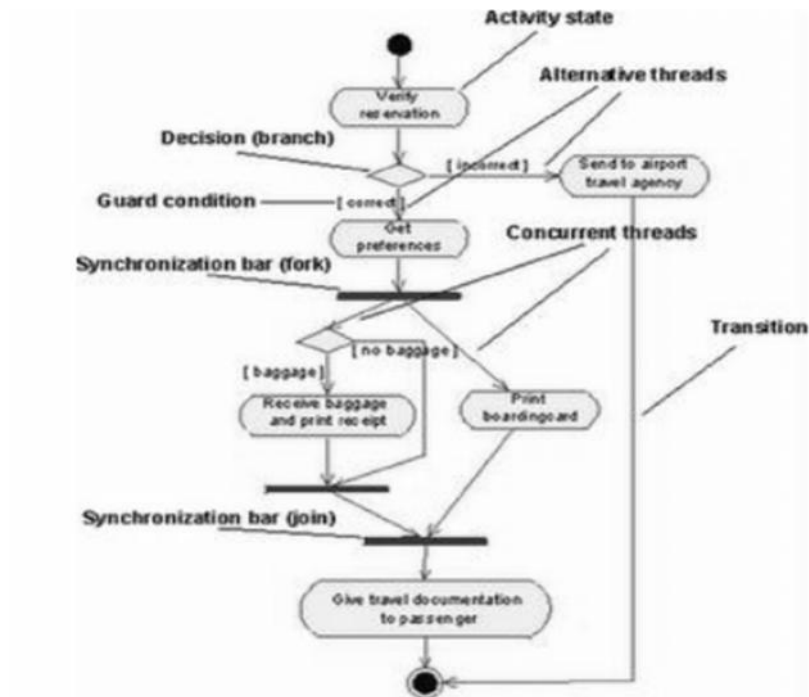


**Gambar II.1. Notasi Use Case Diagram**  
(Sumber : Haviluddin : 2011 ; 4)

## 2. Activity Diagram

Digunakan untuk menganalisis behaviour dengan use case yang lebih kompleks dan menunjukkan interaksi-interaksi di antara mereka satu sama lain. Activity diagram sebenarnya memiliki kesamaan dengan statechart activity diagram banyak digunakan pada saat peralihan antara analisis dan fase desain. Umumnya digunakan untuk menggambarkan sistem interaktif yang “real time”. Diagram ini bersifat optional dalam suatu perancangan sistem. Kumpulan sub-state

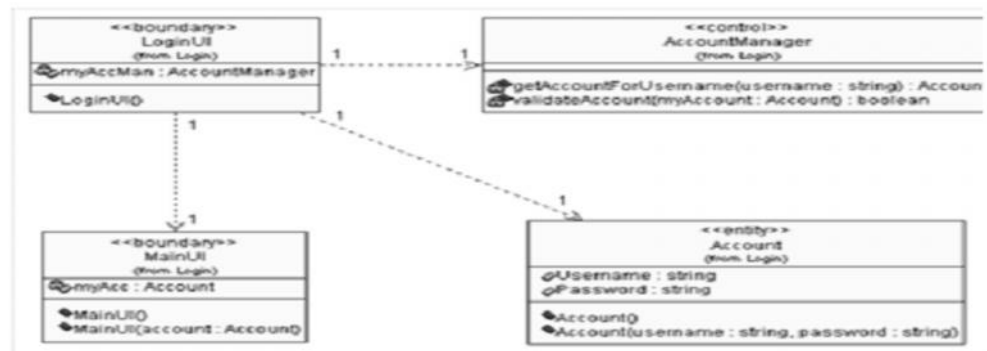
yang dikelompokkan ke dalam sebuah state disebut sebagai composite state.



**Gambar II.2. Notasi Activity Diagram**  
(Sumber : Haviluddin : 2011 ; 4)

### 3. Class Diagram

Diagram ini digunakan untuk menggambarkan perbedaan yang mendasar antara class-class, hubungan antar-class, dan di mana sub-sistem class tersebut. Pada class diagram terdapat nama class, attributes, operations, serta association (hubungan antar-class).

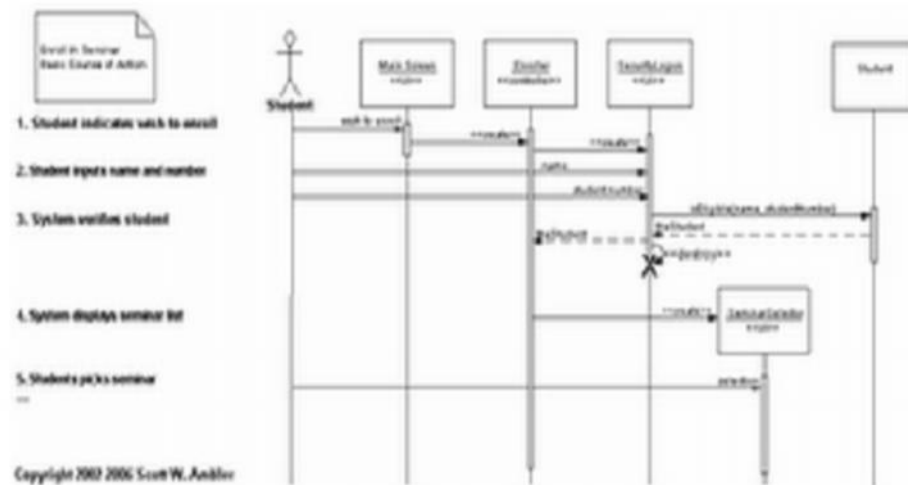


**Gambar II.3. Notasi Class Diagram**

(Sumber : Havaluddin : 2011 ; 3)

#### 4. Sequence Diagram

Merupakan suatu diagram interaksi yang menggambarkan bagaimana objek-objek berpartisipasi dalam bagian interaksi (particular interaction) dan pesan yang ditukar dalam urutan waktu. Sequence diagram merupakan peralatan untuk interaksi berkomunikasi diagram. Sebuah interaksi didesain antara objek atau sistem yang berpartisipasi dalam sebuah kolaborasi. Beberapa konsep dapat diterapkan pada konteks yang bervariasi. Sequence diagram menggambarkan interaksi antara objek secara beraturan sesuai dengan waktu. Sequence diagram dapat digambarkan dalam beberapa level secara detail dan untuk tujuan yang berbeda pada beberapa langkah yang dikembangkan secara *lifecycle*. Adapun simbol-simbol sequence diagram dapat dilihat Gambar II.4. berikut:



**Gambar II.4. Notasi Sequence Diagram**  
(Sumber : Haviluddin : 2011 ; 5)

## II.7. Microsoft Visual Basic 2010

Pada akhir tahun 1999, Teknologi .NET diumumkan. Microsoft memosisikan teknologi tersebut sebagai *platform* untuk membangun XML Web Services. XML Web services memungkinkan aplikasi tipe manapun dan dapat mengambil data yang tersimpan pada server dengan tipe apapun melalui internet.

Visual Basic.NET adalah Visual Basic yang direkayasa kembali untuk digunakan pada *platform* .NET sehingga aplikasi yang dibuat menggunakan Visual Basic .NET dapat berjalan pada sistem komputer apa pun, dan dapat mengambil data dari server dengan tipe apa pun asalkan terinstal .NET Framework. (Priyanto Hidayatullah, 2012 ; 5).

## II.8. *SQL Server 2008*

*SQL (Structured Query Language)* adalah sebuah bahasa yang dipergunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara *de facto* merupakan bahasa standar yang dipergunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya.

*SQL* terdiri dari dua bahasa, yaitu *Data Definition Language Manipulation Language (DDL dan DML)*. Implementasi DDL dan DML sistem manajemen basis data (*SMBD*), namun secara umum implemen bahasa ini memiliki bentuk standar yang ditetapkan oleh *ANSI*. (Adelia, Jimmy Setiawan : 2011 ; 115 ).