

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Konsep Dasar Sistem**

Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu. Teori sistem melahirkan konsep-konsep futuristik. Salah satu konsep yang terkenal adalah konsep sibernetika (*cybernetics*). Konsep kajian ilmiah ini terutama berkaitan dengan upaya menerapkan berbagai disiplin ilmu, yaitu ilmu perilaku, fisika, biologi dan teknik. Oleh karena itu, sibernetika biasanya berkaitan dengan usaha-usaha otomasi tugas-tugas yang dilakukan oleh manusia sehingga studi tentang robotika, kecerdasan buatan (*Artificial Intelligence*), dan lain adalah masukan (*input*), pengolahan (*processing*), dan keluaran (*output*).

Konsep lain yang terkandung di dalam definisi tentang sistem adalah konsep sinergi. Konsep ini mengandaikan bahwa di dalam suatu sistem. *Output* dari suatu organisasi diharapkan lebih besar dari pada *output* individual atau *output* masing-masing bagian.

Sebuah *system* terdiri atas bagian-bagian atau komponen yang terpadu untuk satu tujuan. Model dasar dari bentuk sistem ini adalah adanya masukan, pengolahan, dan keluaran. Akan tetapi, sistem ini dapat dikembangkan hingga menyetakan media penyimpanan. Sistem dapat terbuka dan tertutup akan tetapi sistem informasi biasanya adalah sistem terbuka. Artinya, sistem tersebut dapat

menerima beberapa masukan dari lingkungan luarnya. (Tata Sutabri ; 2012:10-11)

### **II.1.1 Karakteristik Sistem**

Adapun karakteristik yang mencirikan suatu sistem, yaitu :

#### 1. Komponen sistem (*component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen- komponen sistem tersebut dapat berupa suatu bentuk subsistem.

#### 2. Batasan sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

#### 3. Lingkungan luar sistem

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat merugikan sistem tersebut.

#### 4. Penghubung sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem (*Interface*).

#### 5. Masukan sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem,

yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi input bagi subsistem yang lain.

7. Pengolahan sistem (*Procces*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

8. Sasaran sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan. (Tata Sutabri; 2012:20-21)

### **II.1.2 Konsep Dasar Informasi**

Informasi adalah data yang telah diklasifikasikan atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan. Keputusan dapat berkisar dari keputusan berulang sederhana sampai

keputusan strategis jangka panjang. (Tata Sutabri ( 2012:29)

Fungsi utama informasi adalah menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi. Informasi yang disampaikan kepada pemakai mungkin merupakan hasil dari data yang dimasukkan ke dalam pengolahan. Akan tetapi dalam kebanyakan pengambilan keputusan yang kompleks, informasi hanya dapat menambah kemungkinan kepastian atau mengurangi bermacam-macam pilihan. (Tata Sutabri 2012:31).

### **II.1.3 Konsep Dasar Sistem Informasi**

Sistem informasi bukan merupakan hal yang baru. Yang baru adalah komputersasinya. Sebelum ada komputer, teknik penyaluran informasi yang memungkinkan manajer merencanakan serta mengendalikan operasi.

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat *manajerial* dengan kegiatan startegi dari suatu orgnaisai untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. (Tata Sutabri; 2012:46)

### **II.1.4 Komponen sistem Informasi**

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan ( *building block*) yang terdiri dari :

1. Blok masukan (*input block*)

Input mewakili data yang masuk kedalam sistem informasi.

2. Blok model (*model block*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (*output block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok teknologi (*technology block*)

Teknologi merupakan “*tool box*” dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

5. Blok basis data

Basis data (*database*) merupakan kumpulan data yang saling berkaitan berhubungan satu sama lain, tersimpan diperangkat keras computer dan menggunakan perangkat lunak untuk memanipulasinya.

6. Blok kendali (*control block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi. (Tata Sutabri ( 2012:47).

## II.2 Sistem Informasi Geografis

Istilah “*Geografis*“ merupakan bagian dari spasial (keruangan). Penggunaan kata “*Geografis*” mengandung pengertian suatu persoalan atau hal mengenai (wilayah di permukaan) bumi, baik dua dimensi atau tiga dimensi, dengan demikian istilah ”informasi geografis” mengandung pengertian informasi mengenai tempat- tempat yang terletak di permukaan bumi, pengetahuan mengenai posisi di mana suatu objek terletak di bumi, atau informasi mengenai keterangan-keterangan (atribut) objek penting yang terdapat di permukaan bumi yang posisinya diberikan atau diketahui.

SIG juga merupakan sejenis perangkat lunak, perangkat keras (manusia, prosedur, basis data, dan fasilitas jaringan komunikasi) yang dapat digunakan atau memfasilitasi proses pemasukan, penyimpanan, manipulasi, menampilkan, dan keluaran data. (Eddy Prahasta (2009 : 109-110)

### II.2.I Subsistem SIG

SIG dapat diuraikan menjadi beberapa subsistem yaitu :

1. *Data Input*

Bertugas untuk mengumpulkan, mempersiapkan, dan menyimpan data spasial dan atributnya dari berbagai sumber.

2. *Data Output*

Bertugas untuk menampilkan atau menghasilkan keluaran (termasuk mengekspornya ke format yang dikehendaki) seluruh atau sebagian basis data spasial) baik dalam bentuk *softcopy*, maupun *hardcopy* seperti

halnya tabel, grafik, *report*, peta, dan lain sebagainya.

### 3. *Data Management*

Bertugas mengorganisasikan baik data spasial maupun tabel -tabel atribut terkait ke dalam sebuah sistem basis data sedemikian rupa hingga mudah dipanggil kembali atau di *update* dan di *edit*.

### 4. *Data Manipulation dan Analysis*

Bertugas menentukan informasi-informasi yang dapat dihasilkan oleh SIG. (Eddy Prahasta 2009 : 118).

## II.2.2 Komponen SIG

SIG sebagai sistem terdiri dari beberapa komponen dengan karakteristiknya yaitu :

### 1. Perangkat Keras

Perangkat keras yang sering digunakan untuk aplikasi SIG adalah komputer, *mouse*, monitor (*plus VGA-card grafik*) yang beresolusi tinggi, *digitizer*, *printer*, *plotter*, *receiver GPS* dan *scanner*.

### 2. Perangkat Lunak

SIG bisa merupakan sistem perangkat lunak yang tersusun secara modular di mana sistem basis datanya memegang peranan kunci.

### 3. Data dan Informasi Geografis

SIG dapat mengumpulkan dan menyimpan data atau informasi yang diperlukan baik secara tidak langsung (dengan cara meng-*import*-nya dari format-format perangkat lunak SIG yang lain) maupun secara langsung dengan cara melakukan digitasi data spasialnya (digitasi *on-screen* atau

*head-ups*) diatas tampilan layar monitor atau manual dengan menggunakan (digitizer) dari peta analog dan kemudian memasukkan data atributnya dari tabel-tabel atau laporan dengan menggunakan *keyboard*.

#### 4. *Manajemen*

Suatu proyek SIG akan berhasil jika dikelola dengan baik dan dikerjakan oleh orang-orang memiliki keahlian (kesesuaian dengan *job-description* yang bersangkutan) yang tepat pada semua tingkatan. (Eddy Prahasta (2009 : 120-121)

### **II.3 ArcView**

*Arcview* merupakan salah satu perangkat lunak (*tool*) SIG dan pemetaan yang dikembangkan oleh *ESRI* ( *Environmental Systems Research instite* ). Perangkat lunak ini memberikan fasilitas teknis yang berkaitan dengan pengolahan data spasial. Kemampuan grafis yang baik dan kemampuan teknis dalam pengolahan data spasial tersebut memberikan kekuatan secara nyata pada *ArcView* untuk melakukan analisis spasial. Kekuatan analisis spasial inilah yang pada akhirnya menjadikan *ArcView* banyak diterapkan dalam berbagai pekerjaan seperti analisis pemasaran, perencanaan wilayah dan tata ruang, sistem informasi persil, pengendalian dampak lingkungan bahkan untuk keperluan militer. (Eko Budiyanto (2010 : 177)

### **II.4 PHP**

*PHP* adalah bahasa pemrograman untuk dijalankan melalui halaman *web*, umumnya digunakan untuk mengolah informasi di internet. Sedangkan dalam

pengertian lain *PHP* adalah singkatan dari *PHP Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source* atau gratis. *PHP* merupakan *script* yang menyatu dengan *HTML* dan berada pada *server* (*server side HTML embedded scripting*). (Rulianto Kurniawan;2010:2)

Beberapa hal penting dalam penulisan *PHP* yaitu :

1. Penulisan *PHP* dimulai dengan `<? Dan diakhiri dengan ?>`
2. Setiap perintah pada *PHP* diakhiri dengan titik koma ( ; )
3. Variabel *PHP* ditulis dengan tanda dolar ( \$ ) dan kemudian diikuti nama variabel tanpa spasi. (Dadan sutisna (2008)

*PHP* mempunyai beberapa kekurangan. Namun untuk masalah kekurangan, *PHP* dapat dibilang bahasa pemrograman yang jauh dari kekurangan. Adapaun kekurangan tersebut diantaranya adalah :

1. Permasalahan yang sering terjadi pada *register\_globals*
2. Tidak mengenal *package*.
3. Jika tidak di*encoding*, maka kode *PHP* dapat dibaca semua orang.
4. Tidak memiliki sistem pemrograman berorientasi objek yang sesungguhnya.
5. *PHP* memiliki kelemahan *security*. ( Rulianto Kurniawan;2010:5)

## **II.5. Basis Data**

*Database* atau basis data adalah sekumpulan data yang memiliki hubungan secara logika dan diatur dengan susunan tertentu serta disimpan dalam media penyimpanan komputer. Data itu sendiri adalah representasi dari semua fakta

yang ada pada dunia nyata. *Database* sering digunakan untuk melakukan proses terhadap data-data tersebut untuk menghasilkan informasi tertentu. misalnya dari data nama siswa yang berulang tahun pada hari ini. Tentu saja informasi tersebut akan anda dapatkan dari *software*/pemroses database dengan cara anda memberikan perintah dalam bahasa tertentu yaitu *SQL(Structured Query Language)*.

Pada era kemajuan teknologi seperti sekarang ini, nilai informasi sangatlah penting, terlebih bagi kemajuan perusahaan. Oleh karena itu penggunaan dan penguasaan database sangat penting. Dalam database ada sebutan-sebutan untuk satuan data yaitu:

1. Karakter, ini adalah satuan data terkecil. data terdiri atau susunan karakter yang pada akhirnya mewakili data yang memiliki arti dari sebuah fakta.
2. Field, adalah kumpulan dari karakter yang mewakili fakta tertentu misalnya seperti jenis bibit, satuan, dan lain-lain. Dalam dunia perancangan database, field juga disebut atribut. Bila dipandang dari sudut pemrograman berorientasi obyek maka *name* dan properti *type*. Properti *name* atau nama adalah properti dari field yang berisi field yang mewakili data sejenis yang disimpannya. Sedangkan properti *type* adalah properti yang mengatur tipe data dari data yang akan ditampungnya. Misalnya nama fieldnya adalah jenis bibit maka tipe datanya adalah *char*, bila nama fieldnya adalah tanggal lahir maka tipe datanya adalah *date*. Field dilihat seperti kolom.

3. Record, adalah kumpulan dari field. Pada *record* anda dapat menemukan banyak sekali informasi penting dengan cara mengombinasikan field-field yang ada.
4. Tabel, adalah sekumpulan dari *record-record* yang memiliki kesamaan entity dalam dunia nyata. Kumpulan dari tabel adalah database, wujud fisik sebuah database dalam komputer adalah sebuah file yang didalamnya terdapat berbagai tingkatan data yang telah disebutkan di atas.
5. File, adalah bentuk fisik dari penyimpanan data. File database berisi semua data yang telah disusun dan diorganisasikan sedemikian rupa sehingga memudahkan pemberian informasi. (Wahana Komputer : 2010 : 24).

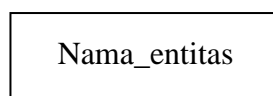
### **II.5.1 Entity Relational Diagram (ERD)**

ERD merupakan suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. ERD juga merupakan gambaran yang menghubungkan antara objek satu dengan objek yang lain dalam dunia nyata. Bisa dikatakan bahwa bahan yang digunakan untuk membuat ERD adalah dari objek di dunia nyata. Secara umum ERD terdiri dari 3 komponen, yaitu :

1. Entitas (*Entity*)

Merupakan suatu “objek nyata” yang mampu dibedakan dengan objek yang lain. Objek tersebut dapat berupa orang benda ataupun hal yang lainnya.

Penggambaran entitas dalam ERD seperti pada gambar II.1

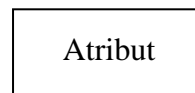


**Gambar II.1. Entitas**

*Sumber : Ema Utami dan Anggit Dwi Hartanto (2012 : 19)*

2. Atribut (*Attribute*)

Merupakan semua informasi yang berkaitan dengan entitas. Di dalam dunia pemrograman, atribut adalah properti dari suatu objek. Penggambaran atribut dalam ERD seperti pada gambar II.2

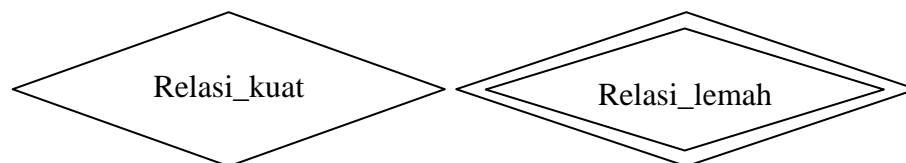


**Gambar II.2. Atribut**

*Sumber : Ema Utami dan Anggit Dwi Hartanto (2012 : 20)*

3. Relasi (*Relationship*)

Belah ketupat merupakan penggambaran hubungan (relasi) antar entitas atau sering disebut kerelasiaan. Ada dua macam penggambaran relasi, yakni relasi kuat dan relasi lemah. Relasi kuat biasanya untuk menghubungkan antar entitas kuat, sedangkan relasi lemah untuk menghubungkan antar entitas kuat dengan entitas lemah. Penggambaran kerelasiaan seperti gambar II.3



**Gambar II.3. Kerelasiaan**

*Sumber : Ema Utami dan Anggit Dwi Hartanto (2012 : 24)*

## II.5.2 Derajat Kardinalitas

Merupakan penjelasan dari tingkat hubungan antar entitas. Ukuran derajat

kardinalitas dibagi menjadi tiga macam, yaitu :

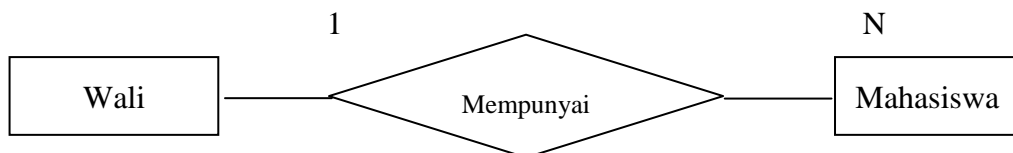
1. 1-1 (*one-to-one*), misalnya seorang ketua jurusan hanya memimpin satu jurusan, begitu juga sebaliknya satu jurusan hanya dipimpin seorang ketua jurusan.



**Gambar II.4. 1-1 (*one-to-one*)**

*Sumber : Ema Utami dan Anggit Dwi Hartanto (2012 : 24)*

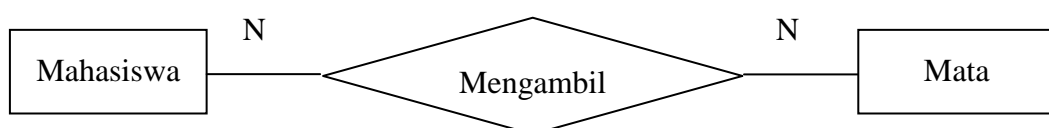
2. 1-N (*one-to-many*) atau N-1 (*many-to-one*), misalnya seorang mahasiswa hanya mempunyai seorang wali, tetapi seorang wali bisa menjadi wali banyak mahasiswa.



**Gambar II.5. 1-1 (*one-to-many*)**

*Sumber : Ema Utami dan Anggit Dwi Hartanto (2012 : 25)*

3. N-N (*many-to-many*), misalnya seorang mahasiswa bisa mengambil banyak mata kuliah, begitu juga sebaliknya satu mata kuliah bisa diambil oleh banyak a.



**Gambar II.6. 1-1 (many-to-many)**  
*Sumber : Ema Utami dan Anggit Dwi Hartanto (2012 : 25)*

### II.5.3 Normalisasi

Merupakan proses pengelompokan elemen data menjadi tabel yang menunjukkan entitas sekaligus relasinya. Tujuan dari normalisasi adalah mengurangi kemungkinan terjadinya anomali yang terjadi dalam basis data. (Ema Utami dan Anggit Dwi Hartanto (2012 : 40).

Adapun bentuk-bentuk normalisasi, yaitu :

1. Bentuk Normalisasi Pertama (1NF)

Suatu tael dikatakan dalam bentuk normal pertama apabila :

- a. Tidak ada baris data yang terduplikat atau berulang dalam tabel.
- b. Setiap sel memiliki nilai tunggal, artinya tidak ada perulangan *group* atau *array*.
- c. Data dalam kolom (atribut atau *field*) memiliki tipe data yang sejenis.

2. Bentuk Normalisasi Kedua (2NF)

Tabel dalam keadaan 2NF apabila tabel sudah dalam keadaan 1NF dan semua atribut yang bukan kunci bergantung pada semua kunci dalam tabel. 2NF bertujuan untuk menghilangkan ketergantungan parsial.

3. Bentuk Normalisasi Ketiga (3NF)

Tabel dalam keadaan 2NF apabila tabel dalam keadaan 2NF dan dalam tabel tersebut tidak ada ketergantungan transitif. Artinya sebuah field dapat menjadi atribut biasa pada suatu relasi tetapi menjadi kunci pada relasi lain. Setiap atribut yang bukan kunci haruslah bergantung hanya pada *primary key*.

4. Bentuk Normalisasi *Boyce-Codd* (BCNF)

Tabel dalam keadaan 3NF dan setiap determinan merupakan kunci kandidat. Determinan adalah suatu atribut/*field* atau gabungan atribut dimana beberapa atribut lain bergantung pada atribut tersebut.

5. Bentuk Normalisasi Keempat (4NF)

Tabel dalam keadaan BCNF dan tidak ada ketergantungan *multi value*.

6. Bentuk Normalisasi Kelima (5NF)

Tabel dalam keadaan 4NF dan setia ketergantungan *join* dalam tabel merupakan akibat dari kunci kandidat tabel.

7. Bentuk Normalisasi *Domain-Key* (DKNF)

Tabel dikatakan dalam keadaan DKNF jika setiap *constraint* tabel merupakan akibat dari definisi kunci-kunci dan domain.

## II.6 Database MySQL

*Database* digunakan untuk penyimpanan data. Pada pemanggilan data pada *MySQL* melalui *PHP*, kemudian hasilnya dikirim ke komputer klien untuk ditampilkan pada *browser*. Data pada *MySQL* dapat dipanggil, dihapus, atau ditambah melalui *query*. *Database* pada *MySQL* terdiri dari tabel-tabel. Setiap tabel mempunyai kolom, baris, serta *record* untuk menyimpan data. (Dadan Sutisna 2008).

## **II.7 Dreamweaver**

*Dreamweaver* adalah perangkat lunak terkemuka untuk *desain web* yang menyediakan kemampuan visual yang intuitif termasuk pada tingkat kode, yang dapat digunakan untuk membuat dan mengedit *website* HTML serta aplikasi *mobile* seperti *smartphone*, *tablet*, dan perangkat lainnya.

## **II.8 Server**

*Server* adalah sebuah sistem komputer yang menyediakan jenis layanan tertentu dalam sebuah jaringan komputer. *Server* ini di dukung dengan prosesor yang bersifat *scalable* dan RAM yang besar, dan juga dilengkapi dengan sistem operasi khusus, yang disebut sebagai sistem operasi jaringan. *Server* ini juga menjalankan perangkat lunak administrasi yang mengontrol akses terhadap jaringan dan sumber daya yang terdapat didalamnya seperti halnya berkas atau percetakan, dan memberikan akses kepada stasiun kerja anggota jaringan.

## **II.9 Unified Modelling Language (UML)**

Terdapat sembilan jenis diagram UML, namun Penulis akan menjabarkan empat jenis diantaranya :




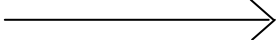
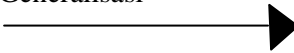
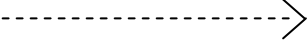
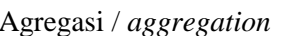
### **1. Class Diagram**

Bersifat statis. Menggambarkan struktur object sistem. Diagram ini menunjukkan *class object* yang menyusun sistem dan juga hubungan antara *class object* tersebut. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

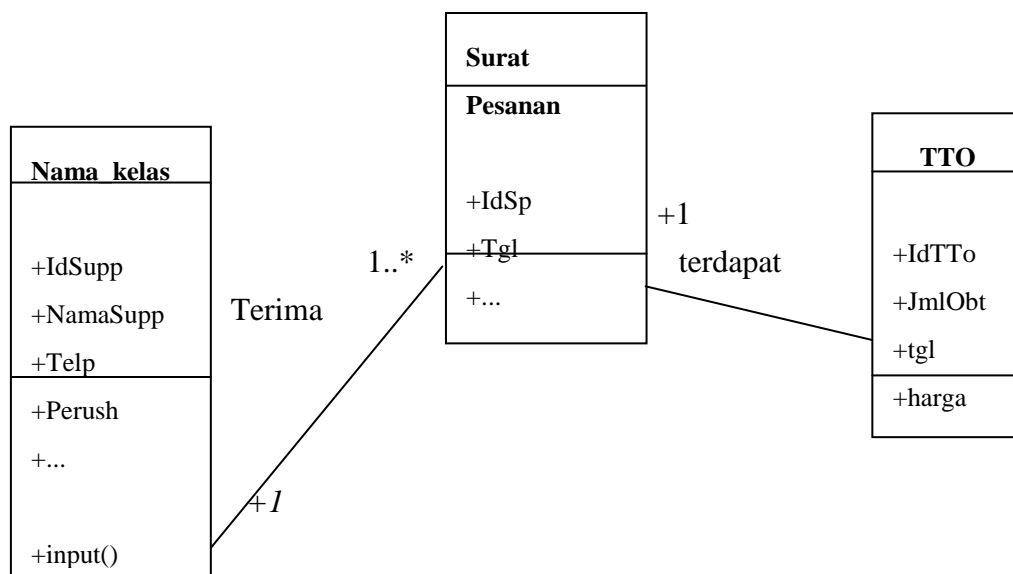
- a. Atribut merupakan *varabel-variabel* yang dimiliki oleh suatu kelas.
- b. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Berikut adalah simbol-simbol yang ada pada diagram kelas :

**Tabel II.1. Simbol-simbol Class Diagram**

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / Interface Nama_interface 	Sama dengan konsep interface dalam pemrograman berorientasi objek
Asosiasi / association 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Asosiasi berarah / directed association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).
Kebergantungan / dependency 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi / aggregation 	Relasi antar kelas dengan makna

Sumber : Yuni Sugiarti (2013 : 59)



**Gambar II.7. Contoh Class Diagram**

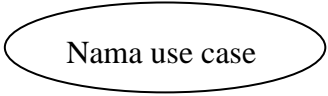
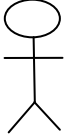

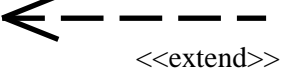
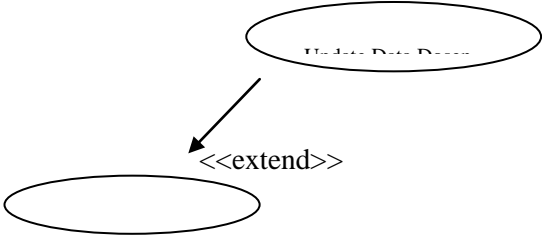
*Sumber : Yuni Sugiarti (2013 : 59)*

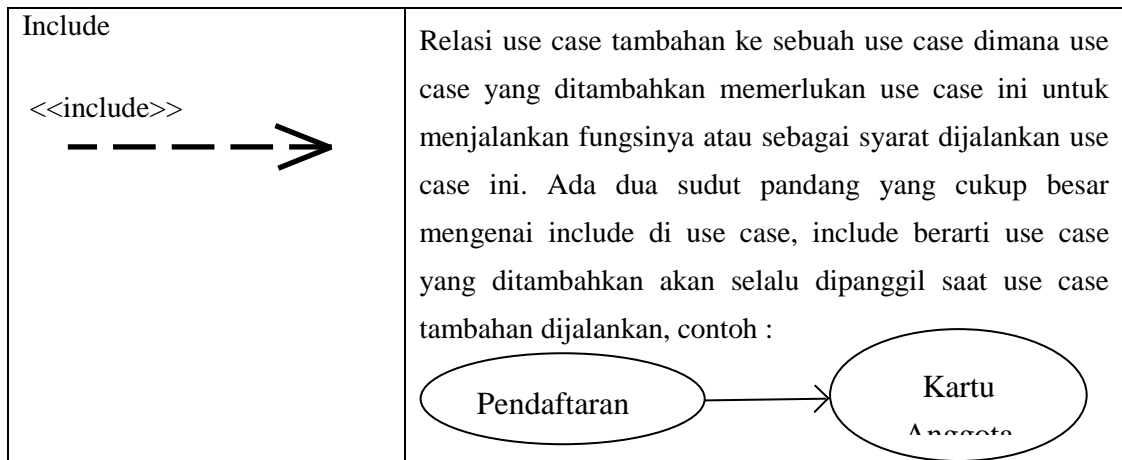
## 2. Diagram Use-Case

*Use case* diagram secara grafis menggambarkan interaksi antara sistem, sistem *eksternal*, dan pengguna. Dengan kata lain *Use Case* diagram secara grafis mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna (*user*) mengharapkan interaksi dengan sistem itu. *Use Case* secara naratif digunakan untuk secara tekstual menggambarkan sekuensi langkah-langkah dari setiap interaksi. Diagram ini memperlihatkan himpunan *use-case* dan aktor-aktor (suatu jenis khusus dari kelas).

Berikut adalah simbol-simbol yang ada pada diagram Use Case :

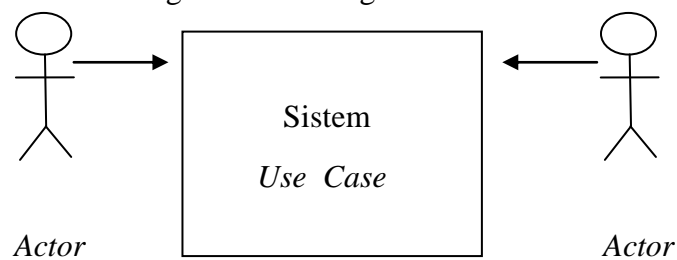
Tabel II.2. Simbol-simbol Use Case

Simbol	Deskripsi
Use Case 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama use case.
Actor   <i>Nama</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
asosiasi / association 	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
Extend 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, arah panah menunjuk pada use case yang dituju.  



Sumber : Yuni Sugiarti (2013 : 42)

Berikut contoh gambar use diagram :

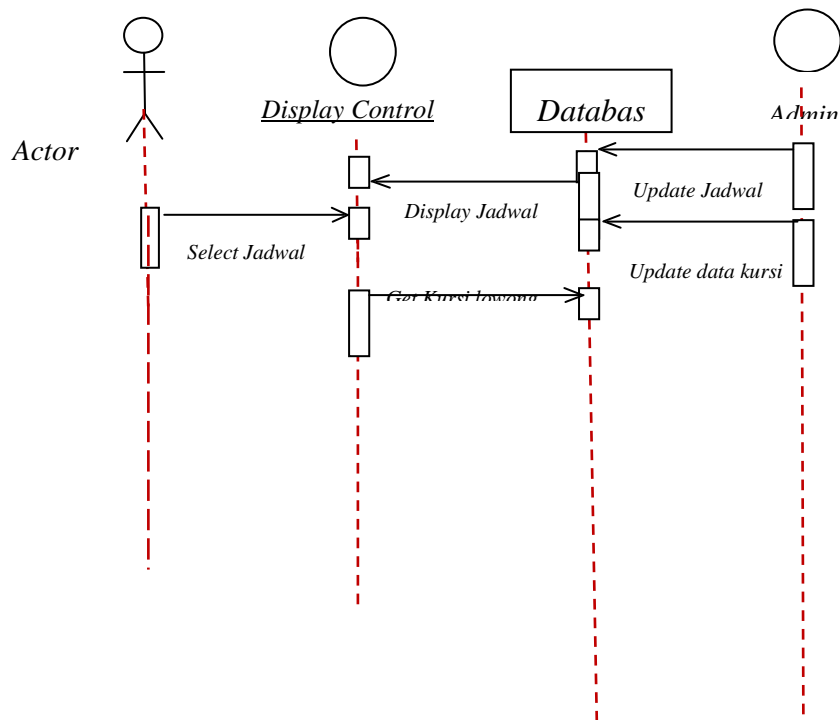


**Gambar II.8. Use Case Model**

Sumber : Yuni Sugiarti (2013 : 56)

### 3. Diagram interaksi dan *sequence* (urutan)

Diagram *sequence* menggambarkan kelakuan/prilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram *sequences* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansi menjadi objek itu.



**Gambar II.9. Display Sequence Diagram**

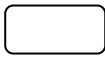
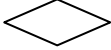


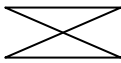
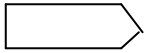
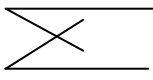

Sumber : Yuni Sugiarti (2013 : 59)

#### 4. Diagram Aktifitas (*Activity Diagram*)

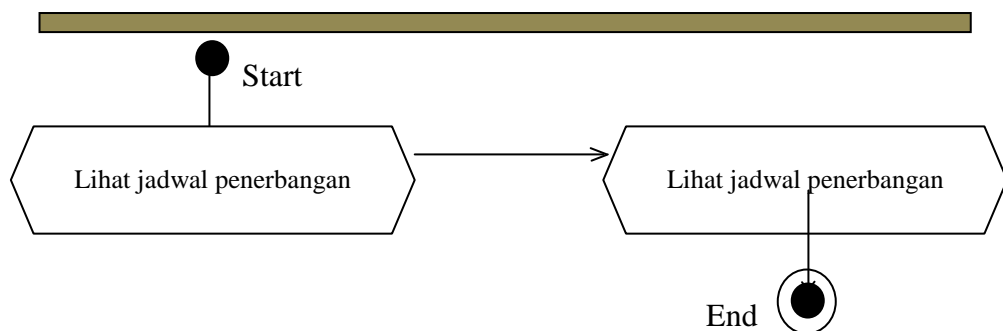
Bersifat dinamis. Diagra aktifitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem. Berikut adalah contoh *Activity diagram*.

**Tabel II.2. Simbol-simbol Activity Diagram**

Simbol	Keterangan
●	Titik awal
○	Titik Akhir

	Activity
	Pilihan untuk pengambilan keputusan
	Fork ; digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	Rake; menunjukkan adanya dekomposisi
	Tanda Waktu
	Tanda Pengiriman
	Tanda Penerimaan
	Aliran akhir (Flow Final)

Berikut contoh dari Activity Diagram :



**Gambar II.1. Contoh Activity Diagram**

Sumber : Yuni Sugiarti (2013 : 78)