

BAB II

TINJAUAN PUSTAKA

II.1. Konsep Dasar Sistem

Sistem sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan himpunan dari suatu 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 (*cybernetic*). 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 otomatis tugas-tugas yang dilakukan oleh manusia sehingga studi tentang robotika, kecerdasan buatan (*Artifical Intelligence*), dan lain adalah masukan (*input*), pengolahan (*processing*), dan keluaran (*output*).

Konsep lain yang terkandung didalam definisi tentang sistem adalah konsep sinergi, konsep ini mengandalkan 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 sistem ini adalah adalah 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 Sutarbi;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 terinteraksi, artinya saling bekerja sama 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 di pandang 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 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 kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan signal (*signal input*).

6. Keluaran sistem (*output*)

Hasil energi yang diolah dan di klasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistm informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini digunakan sebagai masukan untuk pengambila keputusan atau hal-hal lain menjadi input sebagai 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 oprasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan. (Tata Sutarbi;2012:20-21)

II.1.2. Konsep Dasar Informasi

Informasi adalah data yang di klasifikasikan atau diinterpretasikan 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 Sutarbi;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 Sutarbi;2012:31)

III.1.3. Konsep Dasar Sistem Informasi

Sistem informasi bukan merupakan hal yang baru, yang baru adalah komputerisasinya. 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 *majerial* dengan kegiatan strategis dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang di perlukan. (Tata Sutabi;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. Block model (*model block*)

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

3. Block keluaran (*output block*)

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

4. Block 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 mengendalikan dari sistem secara keseluruhan.

5. Block basis data

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

6. Block kendali (*control block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat di cegah

ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi. (Tata Sutarbi;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, 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, fasilitas jaringan komunikasi) yang dapat digunakan atau memfasilitasi proses pemasukan, penyimpanan, manipulasi, menampilkan, dan keluaran data. (Eddy Prahasta;2009:109-110)

II.2.1. 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 sehingga mudah di panggil 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 karakteristik yaitu :

1. Perangkat Keras

Perangkat keras yang sering digunakan untuk aplikasi SIG adalah komputer, *mouse*, monitor (*plus VGA-card grafic*) yang beresolusi tinggi, *digitizer*, *printer*, *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-*importnya* 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 yang 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 System Research Institute*). 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 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).

Arcview mengorganisasikan sistem perangkat lunaknya dalam beberapa komponen penting.

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 *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 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 dibidang bahasa pemrograman yang jauh dari kekurangan. Adapun kekurangan tersebut diantaranya adalah :

1. Permasalahan yang sering terjadi pada *register_globals*.
2. Tidak mengenal *packages*.
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. Database MySQL

Database digunakan untuk penyimpanan data. Pada pemanggilan data *MySQL* melalui *PHP*, kemudian hasilnya dikirim ke komputer klien untuk di tampilkan pada *browser*. Pada data *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.6. Basis Data

Basis data adalah kumpulan data yang saling berhubungan yang disimpan atau diorganisasi secara bersama, dalam bentuk sedemikian rupa, dan tanpa redundansi (pengulangan) tidak perlu supaya dapat dimanfaatkan kembali dengan cepat dan mudah untuk memenuhi berbagai kebutuhan. (Ema Utami, Anggit Dwi Hartono;2012:3)

Sistem basis data dapat terbagi dalam beberapa komponen, yaitu:

1. Data

Merupakan informasi yang tersimpan dalam suatu struktur tertentu yang terintegrasi.

2. *Hardware*

Merupakan keras berupa komputer dengan media penyimpanan sekunder yang digunakan untuk menyimpan data karena pada umumnya basis data memiliki ukuran yang besar.

3. Sistem operasi

Program yang mengaktifkan/ memfungsikan sistem komputer, mengendalikan seluruh sumber daya dalam komputer dan melakukan operasi-operasi dasar dalam komputer yang meliputi operasi *input output (IO)*, pengolahan *file* dan sebagainya.

4. Basis Data

Basis data sebagai inti dari basis data. Basis data menyimpan data serta struktur sistem basis data baik untuk entitas maupun objek-objeknya secara detail.

5. *Database Management System (DBMS)*

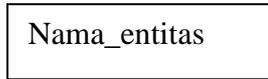
Merupakan perangkat lunak yang digunakan untuk melakukan pengolahan basis data.

II.6.1. *Entity Relational Diagram (ERD)*

ERD merupakan suatu diagram yang menggambarkan desain konseptual dari model konseptual basis data relasional. ERD juga merupakan gambaran yang menghubungkan antara 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. Penggunaan entitas dalam ERD seperti pada gambar II.1.

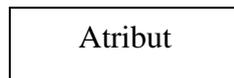


Gambar II.1. Entitas

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

2. Atribut (*Attribute*)

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

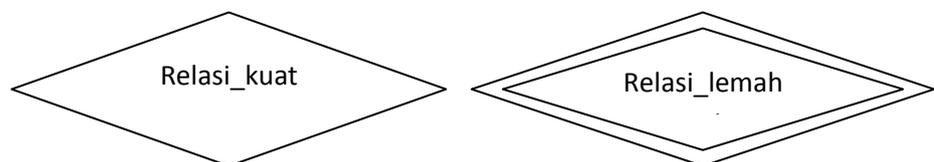


Gambar II.2. Atribut

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

3. Relasi (*Relationship*)

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



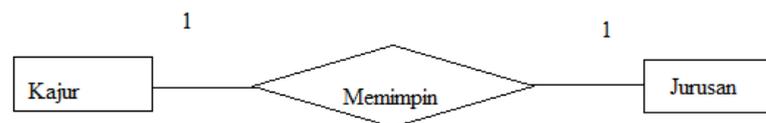
Gambar II.3. Kerelasian

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

II.6.2. Derajat kardinalitas

Merupakan 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. Penggambaran one-to-one seperti pada gambar II.4



Gambar II.4. one-to-one

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

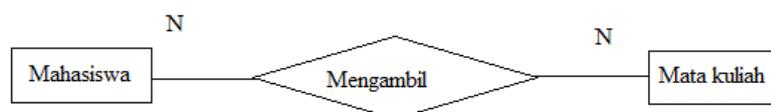
2. 1-N (*one to many*) atau N-1 (*many to one*), misalnya seorang mahasiswa hanya mempunyai seorang wali, tetapi seorang wali bisa menjadi wali banyak mahasiswa. Penggambaran 1-N (*one to many*) seperti pada gambar II.5



Gambar II.5 1-N (*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 mahasiswa. Seperti pada gambar II.6 ini



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

II.6.3. Normalisasi

Merupakan proses pengelompokan elemen dan 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 Normalisasi yaitu:

1. Bentuk Normalisasi Pertama (1NF)

Suatu tabel dikatakan dalam bentuk normal pertama apabila :

- a. Tidak ada data yang terduplikasi 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.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. *Unified Modelling Language* (UML)

II.8.1. Pengertian UML

UML singkatan dari *Unified Modelling Language* yang berarti bahasa permodelan standart. UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

Blok pembangun utama *UML* adalah diagram. Beberapa ada diagram yang rinci (jenis *timing diagram*) dan lainnya ada yang bersifat umum (misalnya diagram kelas). Intinya, *UML* merupakan alat komunikasi yang konsisten dalam *mensupport* para pengembang sistem saat ini. (Prabowo Pudjo Widodo dan Herlawati; 2011:6-7)

II.8.2. Diagram-Diagram UML

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

1. *Class Diagram*

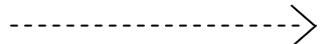
Bersifat statis. Menggambarkan struktur object sistem. Diagram ini menunjukkan *class object* yang menyusun sistem dan juga hubungan antara *class object* tersebut. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

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

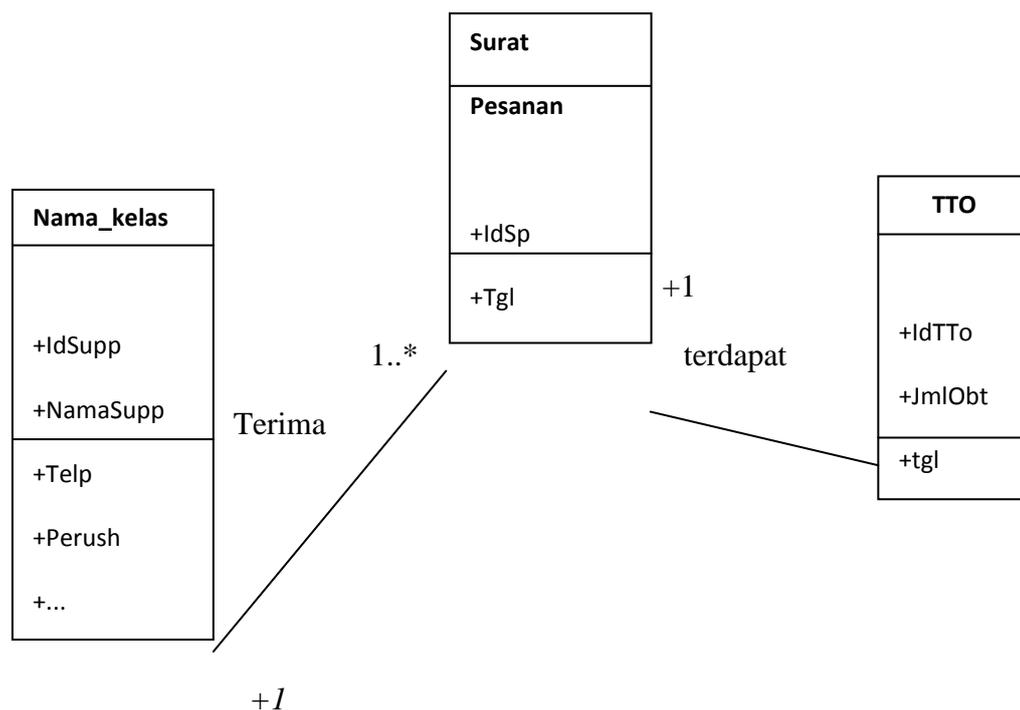
Berikut adalah simbol-simbol yang ada pada diagram kelas :

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 / <i>dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi / <i>aggregation</i>	Relasi antar kelas dengan makna

Sumber : Yuni Sugiarti (2013 : 59)



Gambar II.7. Contoh Class Diagram

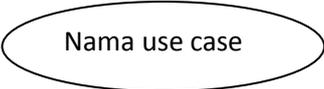
Sumber : Yuni Sugiarti (2013 : 59)

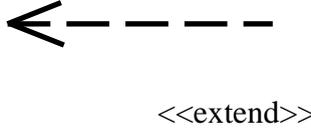
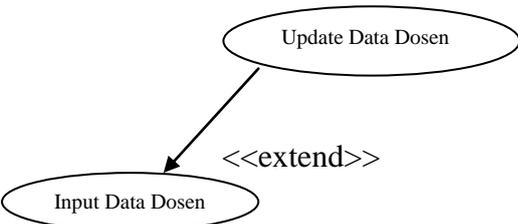
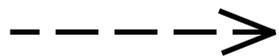
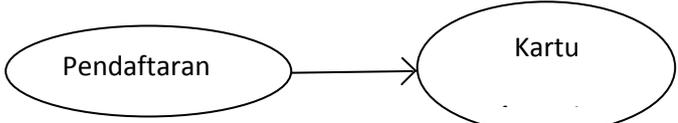
2. Diagram *Use-Case*

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 :

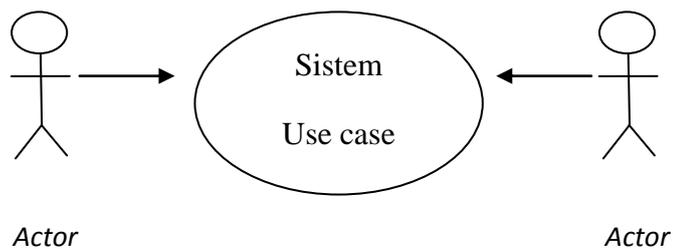
Tabel II.2. Simbol-simbol Use Case

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 	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 aktor 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.

<p>Extend</p>  <p><<extend>></p>	<p>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.</p> 
<p>Include</p> <p><<include>></p> 	<p>Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini. Ada dua sudut pandang yang cukup besar mengenai include di use case, include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan, contoh :</p> 

Sumber : Yuni Sugiarti (2013 : 42)

Berikut contoh gambar use diagram :

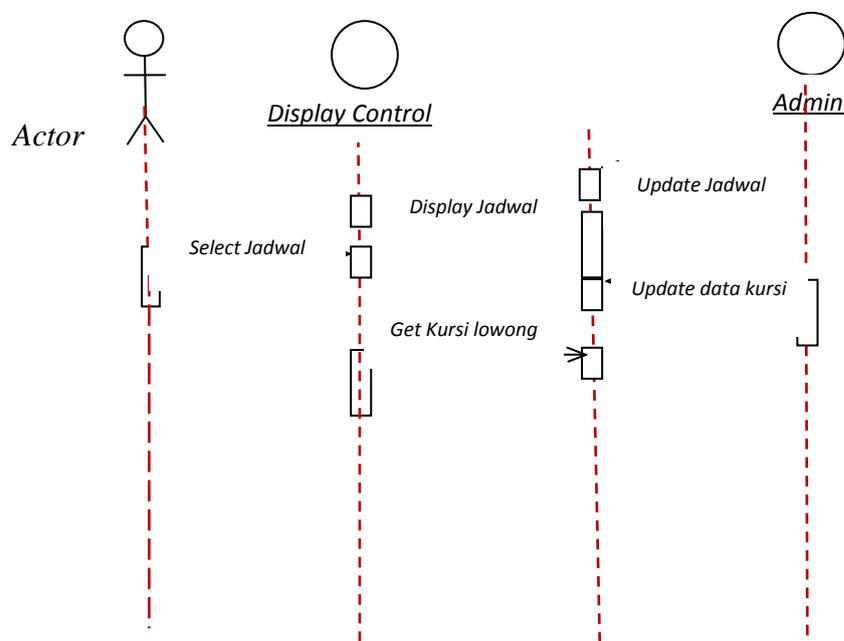


Gambar II.8. Use Case Model

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.



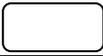
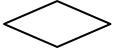
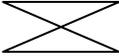
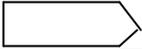
Gambar II.9. Display Sequence Diagram

Sumber : Yuni Sugiarti (2013 : 59)

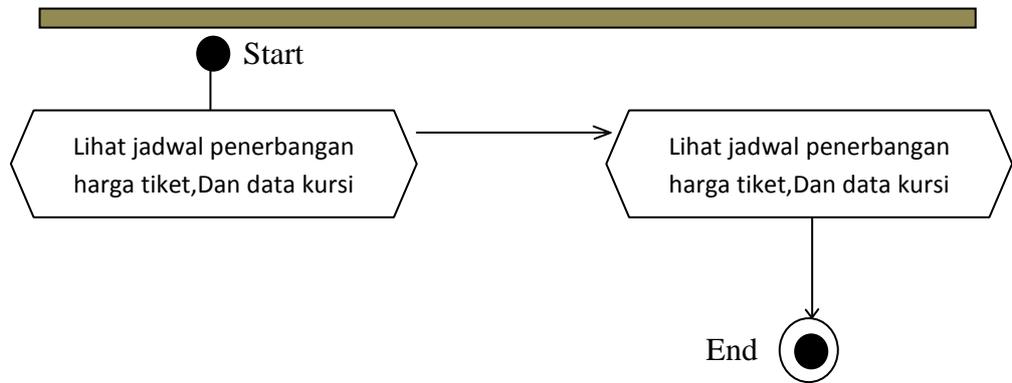
4. Diagram Aktifitas (*Activity Diagram*)

Bersifat dinamis. Diagram 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.3. 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.10. Contoh Activity Diagram

Sumber : Yuni Sugiarti (2013 : 78)