

BAB II

TINJAUAN PUSTAKA

II.1. Pengertian Sistem

Sistem merupakan kumpulan dari unsur atau elemen-elemen yang saling terkait/berinteraksi dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

1. Syarat-Syarat Sistem

- a. Sistem harus dibentuk untuk menyelesaikan tujuan.
- b. Elemen sistem harus mempunyai rencana yang ditetapkan.
- c. Adanya hubungan di antara elemen sistem
- d. Unsur dasar dari proses (arus informasi, energi dan material) lebih penting daripada elemen sistem.
- e. Tujuan organisasi lebih penting dari pada tujuan elemen.

2. Karakteristik Sistem

a. Komponen (*Component*)

Suatu sistem terdiri dari jumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu

dan memengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai proses sistem yang lebih besar yang disebut *supra sistem*.

b. Batas Sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan, karena dengan batas sistem ini fungsi dan tugas dari subsistem yang satu dengan yang lain berbeda tetapi tetapsaling berinteraksi. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

c. Lingkungan Luar sistem (*Environment*)

Environment merupakan segala sesuatu yang berada di luar batas sistem yang memengaruhi operasi dari suatu sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan atau merugikan. Lingkungan luar yang menguntungkan harus dipelihara dan dijaga agar tidak hilang pengaruhnya, sedangkan lingkungan luar yang merugikan harus dimusnahkan atau dikendalikan agar tidak mengganggu operasi sistem.

d. Penghubung Sistem (*interface*)

Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya untuk membentuk satu kesatuan sehingga sumber-sumber daya mengalir dari subsistem yang satu ke subsistem yang lainnya. Dengan kata lain, *output* dari suatu subsistem akan menjadi *input* dari subsistem yang lainnya.

e. Masukan Sistem (*Input*)

Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*Maintenance Input*) adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Masukan sinyal (*Signal Input*) adalah energi yang diproses untuk didapatkan keluaran.

f. Keluaran Sistem (*Output*)

Merupakan hasil dari energi yang diolah oleh sistem, meliputi *output* yang berguna contohnya informasi yang dikeluarkan oleh komputer. Dan *output* yang tidak berguna dikenal sebagai sisa pembuangan, contohnya panas yang dikeluarkan komputer.

g. Pengolah Sistem (*Process*)

Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan contoh CPU pada komputer.

h. Tujuan Sistem (*Goal*)

Setiap sistem mempunyai tujuan ataupun sasaran yang memengaruhi *input* yang dibutuhkan dan *output* yang dihasilkan. Dengan kata lain, suatu sistem itu mengenai sasaran atau tujuannya. Jika sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.

3. Klasifikasi Sistem

a. Sistem Abstrak (*Abstrak system*)

Sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik.

b. Sistem Fisik (*Physical System*)

Merupakan sistem yang ada secara fisik sehingga setiap makhluk dapat melihatnya.

c. Sistem Alamiah (*Natural System*)

Sistem yang terjadi melalui proses alam, dalam artian tidak dibuat oleh manusia.

d. Sistem Buatan Manusia (*Human Made System*)

Sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan interaksi manusia dengan mesin disebut *human machine system*, contohnya sistem informasi.

e. Sistem Tertentu (*Deterministic System*)

Beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi bagian-bagiannya dapat dideteksi secara pasti sehingga keluaran dari sistem dapat diramalkan. Contohnya sistem komputer

f. Sistem Tak Tentu (*Probabilistic System*)

Sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas. Contohnya sistem manusia.

g. Sistem Tertutup (*closed system*)

Sistem yang tidak berubangan dan tidak terpengaruh dengan sistem luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luarnya. Secara teori sistem tersebut ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah *relatively closed system* (secara relatif tertutup, tidak benar-benar tertutup).

h. Sisten Terbuka (*Open System*)

Sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Lebih spesifik dikenal dengan sistem terotomasi, yang merupakan bagian dari sistem buatan manusia dan berinteraksi dengan kontrol oleh satu atau lebih komputer. Sebagai bagian dari sistem yang digunakan dalam masyarakat modern. (Asbon Hendra; 2012 : 157-162).

II.1.1. Pengertian Informasi

Informasi merupakan data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat.

1. Kuantitas Informasi

Merupakan suatu ukuran informasi, tergantung representasi atau pelambangan. Untuk merepresentasikan biner, satuannya bit, byte, word, dan lain-lain.

2. Kualitas Informasi

Adalah bias terhadap *error* karena kesalahan cara pengukuran dan pengumpulan, kegagalan mengikutim prosedur atau koreksi data, kesalahan file histori/master, kesalahan prosedur pemrosesan, serta ketidak fungsian sistem. Kualitas informasi tergantung dari tiga hal, yaitu informasi arus:

- a. Akurat, berarti informasi harus terbebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.
- b. Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat.

c. Relevansi, berarti informasi tersebut mempunyai manfaat untuk pemakainya.

Relevansi informasi untuk tiap-tiap orang satu dengan lainnya berbeda.

3. Umur Informasi

Kapan atau sampai kapan sebuah informasi memiliki nilai/arti bagi penggunaannya. Ada *condition informasi* (mengacu pada titik waktu tertentu) dan *operating information* (menyatakan perubahan pada suatu *range* waktu)

4. Nilai informasi

Ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Pengukuran nilai informasi biasanya dihubungkan dengan analisis *cost effectiveness* atau *cost benefit*. (Asbon Hendra; 2012 : 167-168).

II.1.2. Sistem Informasi

Menurut Asbon Hendra (2012 : 168-172) sistem informasi merupakan sekumpulan prosedur manual atau terkomputerisasi yang mengumpulkan/mengambil, mengolah, menyimpan, dan menyebar informasi dalam mendukung pengambilan dan kendali keputusan.

Sistem informasi dapat diklasifikasikan sebagai sistem informasi formal dan informal. Sistem informasi formal adalah sistem informasi yang memiliki prosedur dan kebijaksanaan tertulis dalam dokumen. Sedangkan sistem informasi informal adalah sistem informasi yang memiliki prosedur dan kebijaksanaan tidak tertulis dalam dokumen.

1. Komponen Fisik Sistem Informasi

- a. Perangkat keras komputer : CPU, *storage*, perangkat *input/output*, terminal untuk interaksi, serta media komunikasi data.
- b. Perangkat lunak komputer : perangkat lunak sistem (sistem operasi dan *utility*-nya), perangkat lunak umum aplikasi (bahasa pemrograman), dan perangkat lunak aplikasi (aplikasi akuntansi).
- c. Basis data : penyimpanan data pada media penyimpanan komputer.
- d. Prosedur : langkah-langkah penggunaan sistem.
- e. Personel untuk pengolahan aplikasi (SDM)
- f. *Clerical* personel (untuk menangani transaksi dan pemrosesan data dan melakukan inquiry = operator)
- g. *First level manager*, untuk mengelola pemrosesan data didukung dengan perencanaan, penjadwalan, identifikasi situasi *out-of-control*, dan pengambilan keputusan level menengah kebawah.
- h. *Staff specialist*, digunakan untuk analisis perencanaan dan pelaporan.
- i. Management, untuk pembuatan laporan berkala, permintaan khusus, analisis khusus, laporan khusus, pendukung identifikasi masalah dan peluang.

2. Hubungan Pengolah dengan Sistem Informasi

Personel sebagai pengolah informasi merupakan salah satu komponen penting dalam sistem informasi. Oleh karena itu, hubungan antara sistem informasi dengan pengelolanya sangat erat. Sistem informasi yang dibutuhkan sangat tergantung dari kebutuhan pengelolanya. Pengelola sistem informasi terorganisasi

dalam suatu struktur manajemen. Jadi, bentuk atau jenis informasi yang diperlukan sesuai dengan level manajemennya.

- a. Manajemen level atas : untuk perencanaan strategi, kebijakan, dan pengambilan keputusan.
- b. Manajemen level menengah : untuk perencanaan taktis.
- c. Manajemen level bawah : untuk perencanaan dan pengawasan
- d. Operator : untuk pemrosesan transaksi dan merespons permintaan.

II.2. Sistem Informasi Geografis

Sistem informasi geografis merupakan suatu sistem yang menekankan pada unsur informasi geogras. Istilah informasi geografis mengandung pengertian informasi mengenai tempat-tempat yang terletak di permukaan bumi, pengetahuan mengenai posisi di mana suatu objek terletak di permukaan bumi, atau informasi mengenai keterangan-keterangan (atribut) objek penting yang terdapat di permukaan bumi yang posisinya diberikan atau diketahui.

Sistem informasi geografis juga merupakan sejenis perangkat lunak, perangkat keras (manusia, prosedur, basis data, dan fasilitas jaringan komunikasi) yang dapat digunakan untuk memfasilitasi proses pemasukan, penyimpanan, manipulasi, menampilkan, dan keluaran data/informasi geografis brikut atribut-atribut terkait. (Eddy Prahasta; 2009 : 109-110).

Untuk mendukung suatu sistem informasi geografis, pada prinsipnya terdapat dua jenis, yaitu:

a. Data spasial

Data yang berkaitan dengan aspek keruangan dan merupakan data yang menyajikan lokasi geografis atau gambaran nyata suatu wilayah di permukaan bumi. Umumnya dipresentasikan berupa grafis, peta, atau pun gambar dengan format digital dan disimpan dalam bentuk koordinat x,y (vektor) atau bentuk image (raster) yang memiliki nilai tertentu.

b. Data non-spasial

Disebut juga data atribut, yaitu data yang menerangkan keadaan atau informasi-informasi dari suatu objek (lokasi dan posisi) yang ditunjukkan oleh data spasial. (Eko Budiyanto; 2009 : 9)

II.2.1 Subsystem Sistem Informasi Geografis

Sistem informasi geografis dapat diuraikan menjadi beberapa sub-sistem sebagai berikut :

1. Data Input : sub-sistem ini bertugas untuk mengumpulkan, mempersiapkan, dan menyimpan data spasial dan atributnya dalam berbagai sumber.
2. Data Output : sub-sistem ini bertugas untuk menampilkan atau menghasilkan keluaran seluruh atau sebagian basis data baik dalam bentuk *softcopy* maupun *hardcopy* seperti halnya tabel, grafik, peta dan lain sebagainya.
3. Data Management : sub-sistem ini mengorganisasikan baik data spasial maupun tabel-tabel atribut terkait ke dalam sebuah sistem basis data sedemikian rupa sehingga mudah dipanggil kembali atau di *retrieve*, di *update*, dan di *edit*.

4. Data Manipulasi & Analysis : sub-sistem ini menentukan informasi-informasi yang dapat di hasilakn oleh SIG. (Eddy Prahasta; 2009 : 118).

II.2.2 Komponen Sistem Informasi Geografis

Sistem informasi geografis sebagai sistem terdiri dari beberapa komponen sebagai berikut :

1. Perangkat Keras. Pada saat ini sistem informasi geografis sudah tersedia bagi berbagai *platform* perangkat keras, mulai dari kelas *PC desktop*, *workstations* hingga *multi-user* yang bahkan dapat digunakan oleh banyak orang secara bersamaan dalam jaringan komputer yang luas, tersebar, berkemampuan tinggi, memiliki ruang penyimpanan yang besar, dan mempunyai kapasitas memori (RAM) yang besar.
2. Perangkat Lunak. Sistem informasi geografis juga merupakan sistem perangkat lunak yang tersusun secara modular di mana sistem basis datanya memegang peranan kunci.
3. Data & informasi geografis. Sistem informasi geografis dapat mengumpulkan dan menyimpan data atau informasi yang diperlukan baik secara tidak langsung maupun secara langsung dengan melakukan digitasi data spasialnya dari data 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 oarang-orang memiliki keahlian yang tepat pada semua tingkat (Eddy Prahasta; 2009 : 120-121).

II.3. Basis Data

Menurut Ema Utami dan Anggit Dwi Hartanto (2012 : 3) basis adalah markas, gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, peristiwa, konsep, keadaan yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, atau kombinasinya.

Basis data adalah kumpulan data yang saling berhubungan yang disimpan/diorganisasi secara bersama, dalam bentuk demikian rupa, dan tanpa redundansi (pengulangan) yang tidak perlu supaya dapat dimanfaatkan kembali dengan cepat dan mudah untuk memenuhi berbagai kebutuhan.

II.3.1 Database Management System (DBMS)

DBMS merupakan perangkat lunak yang dirancang untuk dapat melakukan utilisasi dan mengelola koleksi data dalam jumlah besar. DBMS juga dirancang untuk dapat melakukan manipulasi data secara lebih mudah. Penyimpanan data oleh DBMS disesuaikan dengan bentuk model datanya, beberapa contoh DBMS adalah PostgreSQL, MySQL, DB2, Oracle, SQL Server, dan lain-lain. (Ema Utami dan Anggit Dwi Hartanto; 2012 : 5).

Secara umum menurut Ema Utami dan Anggit Dwi Hartanto (2012 : 9-24) terdapat tiga model data yang digunakan untuk memodelkan penyimpanan data dalam DBMS, yakni:

1. Model Hierarki

Merupakan model data yang dibangun secara hierarki mirip dengan *sytemfile* di LINUX, dimana terdapat *Root* sebagai titik puncaknya. Dalam model ini setiap data hanya akan memiliki sebuah penghubung untuk data dibawahnya.

2. Model Jaringan

Model data jaringan menggunakan pointer untuk menghubungkan data satu dengan data lain. Jadi, dalam model data jaringan satu data dapat memiliki banyak penghubung dengan data yang lain.

3. Model Relasional

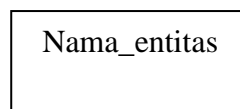
Model basis data relasional dapat dikatakan sebagai kumpulan satu atau lebih relasi dimana setiap relasi merupakan koleksi dari data yang disajikan dalam bentuk tabel yang terdiri dari baris dalam kolom. Relasi merupakan konstruksi diskripsi data yang utama dalam model relasioanal. Relasi dapat dibayangkan sebagai sebuah kumpulan *record* dimana setiap *record* mempunyai jumlah *field* yang sama serta domain yang bersesuaian sama. Desain basis data relasional secara umum dapat dibagi dalam empat langkah :

a. Entity relational diagram (ERD)

ERD adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasioanal. ERD juga merupakan gambaran yang menghubungkan antara objek satu dengan objek yang lain dalam dunia nyata. Sebagai contoh, jika membuat ERD dari sistem akademik suatu perguruan tinggi, maka sebagai objek ERD yaitu : mahasiswa, dosen ruang kelas, mata kuliah dan lain sebagainya.

b. Entitas

Entitas merupakan suatu “objek nyata” yang mampu dibedakan dengan objek lain. Agar mudah menentukan suatu entitas maka bisa dianalogkan bahwa entitas berupa peran, kejadian, abstrak dan fisik. Penggambaran entitas dalam ERD seperti pada gambar II.1.



Gambar II.1. Entitas

Sumber : Ema Utami, Anggi Dwi Hartanto (2012 : 19)

c. Atribut

Atribut merupakan semua informasi yang berkaitan dengan entitas. Sebagai contoh, jika entitas adalah manusia maka atributnya adalah rambut, mata, hidung dan lain-lain. Atribut digambarkan dengan suatu lingkaran dengan nama atribut ditulis ditengahnya seperti gambar II.2.



Gambar II.2. Atribut

Sumber : Ema Utami, Anggi Dwi Hartanto (2012 : 20)

d. Relasi

Belah ketupat merupakan gambaran hubungan (relasi) antar entitas atau sering disebut kerelasiaan. Ada tiga relasi menurut derajatnya, yaitu *unary*, relasi yang menghubungkan entitas; *binary*, relasi yang menghubungkan dua entitas; *tenary*, relasi yang menghubungkan lebih dari dua entitas. Penggambaran kerelasiaan seperti pada gambar II.3.



Gambar II.3. Relasi

Sumber : Ema Utami, Anggi Dwi Hartanto (2012 : 20)

II.3.2. Normalisasi

Normalisasi merupakan proses pengelompokan elemen data dari tabel yang menunjukkan entitas sekaligus relasinya. Tujuan utama normalisasi adalah untuk mencegah terjadinya :

1. *Insertion anomaly* yaitu kesalahan penambahan data kedalam basis data.
2. *Delete Anomaly* yaitu kesalahan dalam menghapus data yang ada didalam basis data.
3. *Update Anomaly* yaitu kesalahan dalam mengubah data, baik dalam hal penambahan, penghapusan atau keduanya.

II.3.3 Bentuk-bentuk Normalisasi

1. Bentuk Normalisasi Pertama (1NF)

Suatu tabel 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 dan 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. Dengan kata lain, 2NF bertujuan untuk menghilangkan ketergantungan parsial.

3. Bentuk Normalisasi Ketiga (3NF)

Tabel dalam keadaan 3NF apabila tabel dalam keadaan 2NF dan dalam tabel tersebut tidak ada ketergantungan transitif. Artinya semua *field* dapat menjadi atribut biasa pada suatu relasi tetapi menjadi kunci pada relasi lainnya. Setiap atribut yang bukan kunci haruslah bergantung 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. Pada tahap BCNF, kita harus menghilangkan kunci kandidat yang bukan determinan.

5. Bentuk Normalisasi Keempat (4NF)

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

6. Bentuk Normalisasi Kelima (5NF)

Tabel dalam keadaan 4NF dan setiap 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 dalam domain. ((Ema Utami dan Anggit Dwi Hartanto; 2012 : 41-42).

II.3.4. Kamus Data

Menurut Rosa.A.S dkk (2011:67-68) kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem

perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan). Kamus data biasanya berisi:

1. Nama - nama dari data
2. Digunakan pada – merupakan proses-proses yang terkait data
3. Deskripsi – merupakan deskripsi data
4. Informasi tambahan – seperti tipe data, nilai data, batas nilai data dan komponen yang membentuk data.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut :

Tabel II.1. Simbol tambahan dalam DFD

Simbol	Keterangan
=	Disusun atau terdiri dari
+	dan
[]	Baik ...atau...
{ } _n	n kali diulang/bernilai banyak
()	Data opsional
...	Batas komentar

Sumber : Rosa A.S, M. Shalahuddin (2011 : 68)

Kamus data pada DFD harus dapat dipetakan dengan hasil perancangan basis data yang dilakukan sebelumnya. Jika ada kamus data yang tidak dapat dipetakan pada tabel hasil perancangan basis data berarti hasil perancangan basis data dengan perancangan DFD masih belum selesai, sehingga harus ada yang diperbaiki perancangan basis datanya, perancangan DFD-nya, atau keduanya.

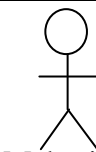
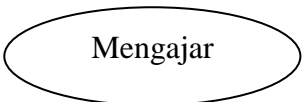
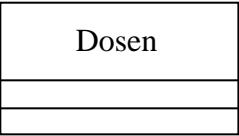
II.4. UML (*Unified Modeling Language*)

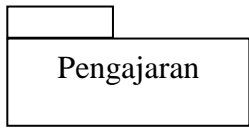
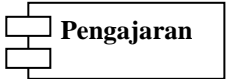


Menurut Adi Nugroho (2010 : 6-7) UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan (*modeling*) sesungguhnya digunakan untuk menyederhanakan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. Adapun tujuan pemodelan yaitu sebagai sarana analisis, pemahaman, visualisasi dan komunikasi antaranggota tim pengembang, serta sebagai sarana dokumentasi.

II.4.1. Pengklasifikasi (*Classifier*)

Pengklasifikasi (*classifier*) pada prinsipnya merupakan konsep diskret dalam model yang memiliki identitas (*identity*), state, perilaku (*behaviour*), serta relasi dengan pengklasifikasi yang lainnya (*relationship*). Jenis pengklasifikasi mencakup di dalamnya *use case*, komponen, simpul (*node*), serta subsistem.

Tabel II.2. Beberapa jenis pengklasifikasi

Pengklasifikasi	Kegunaan	Notasi
<i>Actor</i>	Menggambarkan semua objek di luar sistem (bukan hanya pengguna sistem/perangkat lunak) yang berinteraksi dengan sistem yang dikembangkan.	 Mahasiswa
<i>Use Case</i>	Menggambarkan fungsionalitas yang dimiliki sistem	 Mengajar
Kelas (<i>Class</i>)	Menggambarkan konsep dasar pemodelan sistem.	 Dosen

Subsistem (<i>subsistem</i>)	Menggambarkan paket spesifikasi serta implementasi	
Komponen (<i>Component</i>)	Menggambarkan bagian-bagian fisik sistem/perangkat lunak yang dikembangkan.	
Antarmuka (<i>interface</i>)	Menggambarkan antarmuka pengiriman pesan (<i>message</i>) antarpengklasifikasi	 Antarmuka Pengajar- Mahasiswa
Simpul (<i>node</i>)	Menggambarkan sumber daya komputasional yang digunakan oleh sistem.	

Sumber : Adi Nugroho (2010 : 16)

Kelas sesungguhnya merupakan himpunan dari objek-objek yang memiliki struktur-struktur yang serupa, serta memiliki perilaku dan relasi yang serupa pula. Ada beberapa himpunan objek yang bisa dipertimbangkan untuk menjadi kandidat-kandidat kelas, diantaranya:

1. Kelas Konsep

Konsep adalah ide atau pemahaman yang kita miliki tentang suatu fakta yang ada di dunia nyata. Kelas konsep tidak terukur, tetapi sering digunakan untuk berkomunikasi. Contoh dari kelas konsep adalah kinerja.

2. Kelas Event

Adalah sebuah titik dalam waktu yang harus direkam. Sesuatu yang berhubungan dengan event adalah pertanyaan-pertanyaan seperti kapan, siapa, apa, dimana, bagaimana atau mengapa. Contoh dari kelas event adalah pendaratan, interupsi, permintaan, penekanan serta pemesanan.

3. Kelas Organisasi

Adalah kumpulan orang-orang, sumber daya, fasilitas-fasilitas, atau kelompok yang dimiliki pengguna. Contohnya adalah departemen kuntansi, departemen riset, dan sebagainya.

4. Kelas Manusia

Kelas manusia menampilkan peran-peran (*roles*) yang berbeda dalam hubungannya dengan sistem/perangkat lunak yang kita kembangkan. Contoh kelas manusia adalah karyawan, dosen, direktur, serta manager.

5. Kelas Tempat

Kelas tempat meliputi lokasi fisik yang hanya dicatat dalam sistem. Misalnya : gedung, toko, kantor, dan sebagainya.

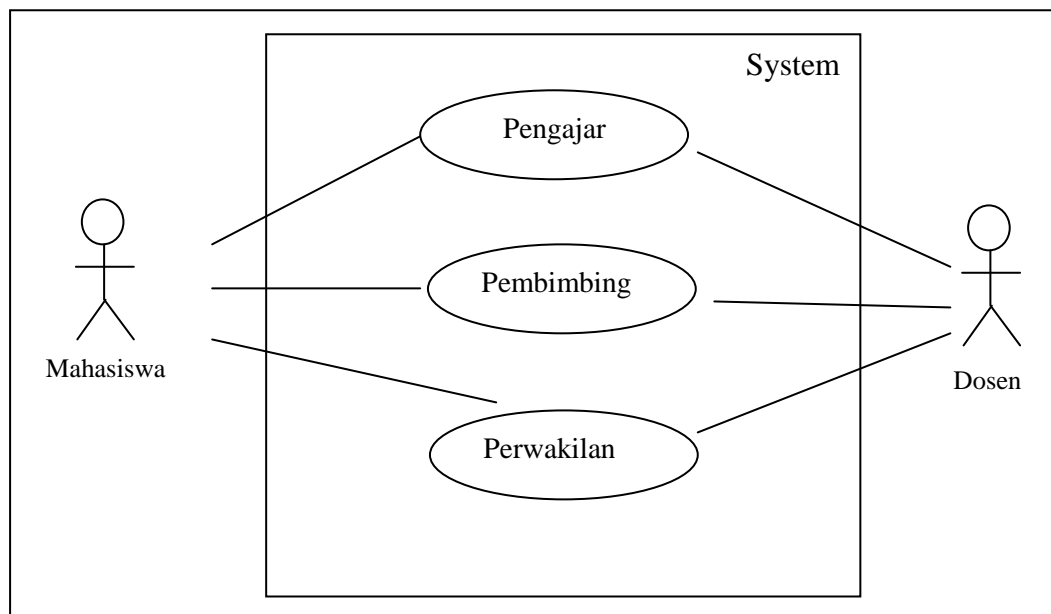
6. Sesuatu yang terukur serta kelas peralatan

Kelas ini meliputi objek fisik serta kelompok objek yang terukur serta peralatan-peralatan yang berinteraksi dengan sistem yang kita kembangkan. Contohnya, mobil adalah sesuatu yang terukur sedangkan sensor tekanan adalah peralatan. (Adi Nugroho; 2010:16-18)

II.4.2. Use Case

Use case merupakan unit koheren dari fungsionalitas sistem/perangkat lunak yang yang atnpak dari luar dan diekspresikan sebagai urutan pesan-pesan yang dipertukarkan unit-unit sistem dengan satu atau lebih *actor* yang ada siluar sistem. Sebuah *actor* pada dasarnya menggambarkan interaksi pengguna-pengguna sistem dengan sistem /perangkat lunak yang sedang dikembangkan. Masing-masing *actor* bisa saja berpartisipasi dalam satu atau lebih *use case* yang

berinteraksi dengan *use case* dan saling bertukar pesan. Jadi kegunaan *use case* sesungguhnya adalah mendefinisikan suatu bagian perilaku sistem yang bersifat koheren tanpa perlu menyingkapkan struktur internal sistem/perangkat lunak yang sedang dikembangkan. Contoh penggambaran diagram *use case* terlihat pada gambar II.4



Gambar II.4. Diagram *Use Case*

Sumber : Adi Nugroho (2010 : 34)

Suatu *Use Case* dapat secara sederhana menggabungkan perilaku *use case* yang lainnya sebagai bagian dari perilakunya sendiri. Hal ini dinamakan sebagai relasi. (Adi Nugroho; 2010:34-36).

Tabel II.3. Relasi-relasi dalam Use Case

Relasi	Fungsi	Notasi
Asosiasi (<i>association</i>)	Lintasan Komunikasi antara <i>actor</i> dengan <i>Use Case</i>	—————
Extend	Penambahan perilaku ke suatu <i>Use Case</i>	----->
Generalisasi <i>Use Case</i>	Menggambarkan hubungan antara <i>use case</i> yang bersifat umum dengan use case-use case yang bersifat lebih spesifik	—————>
Include	Penambahan perilaku ke suatu <i>use case</i> dasar yang secara eksplisit mendeskripsikan penambahan tersebut	----->

Sumber : Adi Nugroho (2010 : 36)

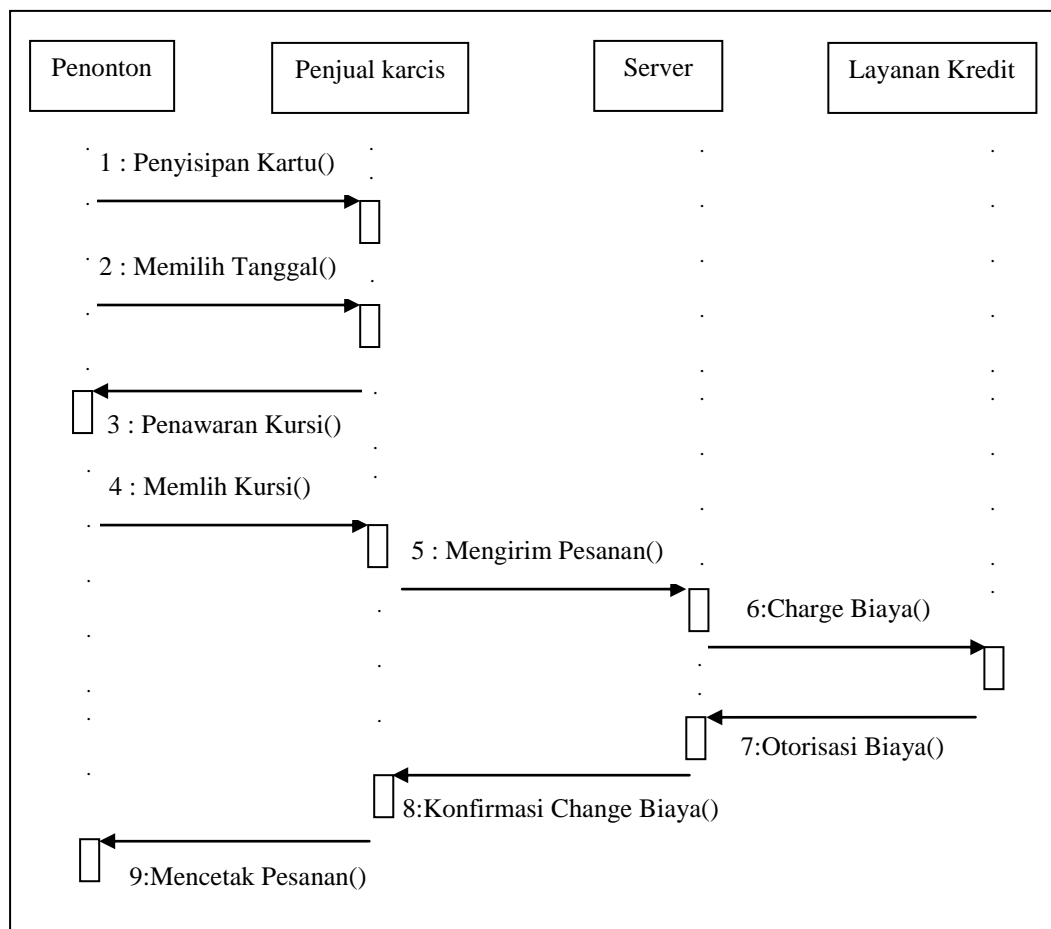
II.4.3. Activity Diagram

Diagram aktivitas sesungguhnya merupakan bentuk khusus dari state machine yang bertujuan memodelkan komputasi-komputasi dan aliran-aliran kerja yang terjadi dalam sistem/perangkat lunak yang sedang dikembangkan. Suatu diagram aktivitas memuat di dalamnya *activity state* di mana suatu *activity state* mempresentasikan suatu aktivitas pernyataan dalam dalam suatu prosedur atau kinerja suatu aktivitas dalam suatu aliran kerja.

Suatu diagram aktivitas mungkin di dalamnya juga memuat suatu *action state*, yang mirip dengan *activity state*, kecuali mereka bersifat atomik dan tidak mengizinkan transisi-transisi terjadi selama mereka aktif. Suatu diagram aktivitas pada dasarnya bertindak seperti diagram alir (*flow chart*) biasa kecuali diagram aktivitas memungkinkan terjadinya kendali konkuren sebagai tambahan pada kendali sekuensial biasa. (Adi Nugroho; 2010:62-63)

II.4.4. Sequence Diagram

Menurut Adi Nugroho (2010:46) Squence Diagram merupakan diagram interaksi yang memperlihatkan urutan-urutan pesan dalam kerangka waktu secara jelas, tetapi tidak secara eksplisit memperlihatkan relasi-relasi yang terjadi pada objek. Dan squence diagram sangat bermanfaat untuk memperlihatkan skenario-skenario. Diagram squence terlihat pada gamabar II.5



Gambar II.5. Sequence Diagram

Sumber : Adi Nugroho (2010 : 42)

II.5. PHP (*Hypertext Preprocessor*)

PHP adalah sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaksnya mirip dengan bahasa pemrograman C, Java, ASP dan Perl, ditambah beberapa fungsi PHP yang spesifik dan mudah dimengerti. PHP digunakan untuk membuat tampilan web menjadi lebih dinamis, dengan PHP kita bisa menampilkan atau menjalankan beberapa *file* dalam 1 *file* dengan cara di *include* atau *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa database walaupun dengan kelengkapan yang berbeda-beda, yaitu seperti : DBM, FilePro, Informix, Ingres, IntraBase, Microsoft Access, MSSQL, MySQL, Oracle, PostgreSQL, dan Sybase. (Madcoms Medium; 2013 : 309).

II.6. MySQL

Penyimpanan data yang fleksibel dan cepat aksesnya sangat dibutuhkan dalam sebuah website yang interaktif dan dinamis. Database sendiri berfungsi sebagai penampung data yang diinputkan melalui form website. Selain itu dapat juga dibalik dengan menampilkan data yang tersimpan dalam database ke dalam halaman website. Jenis database yang populer dan digunakan pada banyak website di internet sebagai bank data adalah MySQL. MySQL menggunakan SQL dan bersifat gratis, selain itu MySQL dapat berjalan diberbagai platform, antara lain LINUX, Windows dan sebagainya.

Untuk dapat mengakses data dalam database MySQL, yang harus dilakukan adalah mengkoneksikan terlebih dahulu. Beberapa fungsi yang dapat digunakan untuk melakukan koneksi dengan database adalah :

- a. `Mysql_connect();`
- b. `Mysql_ponnect();`
- c. `Mysql_select_db();`

Koneksi database yang pertama kali dilakukan adalah dengan menggunakan fungsi `mysql_connect()`. (Madcoms Madium; 2013 : 339-340).

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. (Madcoms Madium; 2013 : 1).

Dreamweaver tidak hanya dapat membuat dan mengolah dokument HTML saja, melainkan dapat membuat dan mengolah berbagai dokumen yang berbasis teks yaitu seperti : CFML, ASP, PHP, JavaScript, dan CSS. (Madcoms Madium; 2013 : 56).

Saat bekerja dengan dreamweaver , kita akan mengenal istilah “lokasi” yang digunakan untuk menempatkan dokument isi dari website baik remote atau lokal. Lokasi inilah yang mengorganisir dan mengatur semua dokumen web, antaranya upload dari lokal server web, memeriksa alur link halaman, juga mengatur dan berbagai file dokumen web. Ada 3 lokasi atau folder penyimpanan site dalam dreamweaver:

1. Lokal (*Local Root Folder*), digunakan sebagai lokal penyimpanan site pada komputer lokal. Dapat juga dibuat di dalam server, dan akan selalu melakukan upload apabila penyimpanan dokumen site.
2. Server (*Remote Folder*), digunakan sebagai lokasi penyimpanan file untuk diuji hasil desainnya, cara kerja, ataupun hubungannya (*link*). *Remote Folder* lokasinya berada pada web server yang sedang berjalan.
3. Testing server folder, adalah folder dimana Dreamweaver menguji halaman web yang dinamis. (Madcoms Madium; 2013 : 41-42).

II.8. ArcView

Arcview adalah *software* yang di keluarkan oleh ESRI (*Environmental System Research Institute*). Perangkat lunak ini memberikan fasilitas teknis yang berkaitan dengan pengolahan data spasial. Arcview memiliki kemampuan yang hebat disebabkan oleh adanya dukungan skrip Avenue. Avenue dapat digunakan untuk merombak wajah Arcview sesuai kebutuhan penggunanya. (Eko Budiyanto; 2010:177).

Data atribut pada sistem informasi dapat berupa data atribut tipe Ftab ataupun data atribut tipe Vtab. Data atribut tipe Ftab menjelaskan tampilan data spasial pada view. data atribut ini biasa digunakan untuk proses-proses pemunculan informasi secara interaktif melalui tampilan peta atau citra. Data atribut Vtab biasa digunakan untuk berbagai proses query atau pencarian data serta editing data. (Eko Budiyanto; 2010:181).