

BAB II

TINJAUAN PUSTAKA

II.1. Pengertian Sistem

Sistem merupakan kumpulan dari beberapa unsur atau elemen – elemen yang saling berkaitan/ berinteraksi dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu (Asbon Hendra; 2012).

Contoh :

- Sistem komputer, terdiri dari software, hardware, dan brainware.
 - Sistem Geografis dan lain – lain.
1. Menurut Jerry FithGerald, sistem adalah suatu jaringan kerja dari prosedur – prosedur yang saling berhubungan dan berkumpul bersama – sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.
 2. Menurut Ludwig Von Bartalanfy, sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi diantara unsur – unsur tersebut dengan lingkungan.
 3. Menurut Anatol Raporot, sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain.
 4. Menurut L. Ackof, sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian – bagian dalam keadaan saling tergantung satu sama lainnya.

II.1.1. Syarat – syarat Sistem

1. Sistem harus dibentuk untuk menyelesaikan tujuan.
2. Elemen sistem harus mempunyai rencana yang ditetapkan.
3. Adanya hubungan diantara elemen sistem.
4. Unsur dasar dari proses (arus informasi, energy, dan material) lebih penting daripada elemen sistem.
5. Tujuan organisasi lebih penting dari pada tujuan elemen.

II.1.2. Karakteristik Sistem

1. Komponen (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerjasama membentuk suatu kesatuan. Komponen – komponen sistem dapat berupa suatu subsistem atau bagian – bagian dari sistem. Setiap sistem, tidak peduli betapa pun kecilnya, selalu mengandung komponen – komponen atau subsistem – subsistem. Setiap subsistem mempunyai sifat – sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut supra sistem. Sebagai contoh, suatu perusahaan dapat disebut dengan suatu sistem dan industri yang merupakan sistem yang lebih besar dapat disebut sebagai subsistem. Demikian juga bila perusahaan dipandang sebagai suatu sistem, maka sistem akuntansi adalah subsistemnya.

2. Batas Sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya 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 lainnya berbeda tetapi tetap saling berinteraksi. Batas suatu sistem menunjukkan ruang lingkup (scope) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environment*)

Environment merupakan segala sesuatu diluar 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.

4. 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 lain.

5. 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. Sebagai contoh, di dalam sistem komputer, program adalah maintenance input yang digunakan untuk mengoperasikan komputernya dan data adalah signal input untuk diolah menjadi informasi.

6. 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 oleh komputer.

7. Pengolah Sistem (*Process*)

Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan. Contoh CPU pada komputer, bagian produksi yang mengubah bahan baku menjadi barang jadi, serta bagian akuntansi yang mengolah data transaksi menjadi laporan keuangan.

8. Tujuan Sistem (*Goal*)

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

II.1.3. Klasifikasi Sistem

1. Sistem Abstrak (*Abstrac System*)

Sistem yang berupa pemikiran atau ide – ide yang tidak tampak secara fisik. Sebagai contoh, sistem teologia yang merupakan suatu sistem yang menggambarkan hubungan tuhan dengan manusia.

2. Sistem Fisik (*Physical System*)

Merupaka sistem yang ada secara fisik sehingga setiap makhluk dapat melihatnya. Contohnya, sistem komputer, sistem akuntansi, sistem produksi, dan lain – lain.

3. Sistem Alamiah (*Natural System*)

Sistem yang terjadi melalui proses alam, dalam artian tidak dibuat oleh manusia, seperti sistem tata surya, sistem galaksi, sistem reproduksi, dan lain – lain.

4. Sistem Buatan Manusia (*Human Mode System*)

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

5. Sistem Tertentu (*Deterministic System*)

Beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi bagian – bagianya dapat dideteksi dengan pasti sehingga keluaran dari sistem dapat diramalkan. Contohnya sistem computer.

6. Sistem Tak Tentu (*Probabilistic System*)

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

7. Sistem Tertutup (*Closed System*)

Sistem yang tidak berhubungan 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 relative tertutup, tidak benar – benar tertutup).

8. Sistem Terbuka (*Open System*)

Sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Lebih spesifik dikenal juga yang disebut dengan sistem terotomasi, yang merupakan bagian dari sistem buatan manusia dan berinteraksi dengan kontrol oleh satu atau lebih computer sebagai bagian dari sistem yang digunakan dalam masyarakat modern.

a. Sistem terotomasi mempunyai sejumlah komponen, yaitu

- 1) Perangkat keras (CPU, disk, printer, tape).
- 2) Perangkat lunak (sistem operasi, sistem database, program pengontrol komunikasi, program aplikasi).
- 3) Personel (yang mengoperasikan sistem, menyediakan masukan, mengonsumsi keluaran dan melakukan aktivitas manual yang mendukung sistem).
- 4) Data (yang harus tersimpan dalam sistem selama jangka waktu tertentu).
- 5) Prosedur (intruksi dan kebijakan untuk mengoperasikan sistem).

b. Sistem terotomasi terbagi dalam sejumlah kategori, yaitu

1) *Online Systems*

Sistem online adalah sistem yang menerima langsung input pada area dimana input tersebut direkam dan menghasilkan output yang dapat berupa hasil komputasi pada area dimana mereka dibutuhkan. Area sendiri dapat dipisah – pisah dalam skala, misalnya ratusan kilometer. Biasanya digunakan bagi reservasi angkutan udara, reservasi kereta api, perbankan, dan lain sebagainya.

2) *Real-time Systems*

Sistem real-time adalah mekanisme pengontrolan, perekaman data, dan pemrosesan yang sangat cepat sehingga output yang dihasilkan dapat diterima dalam waktu yang relatif sama. Perbedaan dengan sistem online adalah satuan waktu yang digunakan real-time biasanya seperseratus atau seperseribu detik sedangkan online masih dalam skala detik atau bahkan kadang beberapa menit. Perbedaan lainnya, online biasanya hanya berinteraksi dengan pemakai, sedangkan real-time berinteraksi langsung dengan pemakai dan lingkungan yang dipetakan.

3) *Decision Support System and Strategic Planning System*

Sistem ini memproses transaksi organisasi secara harian dan membantu para manajer mengambil keputusan, mengevaluasi, dan menganalisa tujuan organisasi. Biasanya digunakan untuk sistem penggajian, sistem pemesanan, sistem akuntansi, dan sistem produksi. Biasanya berbentuk paket statistik, paket pemasaran, dan lain – lain. Sistem ini tidak hanya merekam dan menampilkan data, tetapi juga fungsi –

fungsi matematik, data analisa statistik, dan menampilkan informasi dalam bentuk grafik (tabel, chart) sebagaimana laporan konvensional.

4) *Knowledge-based System*

Program komputer ini dibuat mendekati kemampuan dan pengetahuan seorang pakar. Umumnya menggunakan perangkat keras dan perangkat lunak khusus seperti LISP dan PROLOG.

Berdasarkan prinsip dasar secara umum, sistem terbagi dalam :

1. Sistem Terspesialisasi, adalah sistem yang sulit diterapkan pada lingkungan yang berbeda, misalnya sistem biologi : ikan yang dipindahkan ke darat.
2. Sistem Besar, adalah sistem yang sebagian besar sumber dayanya berfungsi melakukan perawatan harian. Misalnya dinosaurus sebagai sistem biologi menghabiskan sebagian besar masa hidupnya dengan makan.
3. Sistem Sebagai Bagian dari Sistem Lain, sistem selalu merupakan bagian dari sistem yang lebih besar, dan dapat terbagi menjadi sistem yang lebih kecil.
4. Sistem Berkembang, walaupun tidak berlaku bagi semua sistem, hampir semua sistem selalu berkembang.

II.2. Informasi

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

suatu proses transformasi data menjadi suatu informasi = input – proses – ouput
(Asbon Hendra;2012).

II.2.1. Kualitas Informasi

Kualitas informas tergantung dari tiga hal ,yaitu informasi harus

1. Akurat, berarti informasi harus bebas dari kesalahan – kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.
2. Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat.
3. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya.
Relenvansi informasi untuk tiap – tipa orang satu dengan yang lainnya berbeda.

II.2.2. Nilai Informasi

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*.

II.3. Sistem Informasi

Menurut Robert A. Leitch, sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung

operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan – laporan yang diperlukan (Asbon Hendra; 2012)

Sistem informasi adalah suatu sistem terintegrasi yang mampu menyediakan informasi yang bermanfaat bagi penggunanya. Sekumpulan prosedur manual atau terkomputerisasi yang mengumpulkan/ mengambil, mengelolah, menyimpan, dan menyebarkan informasi dalam mendukung pengambilan dan kendali keputusan.

II.4. Geografis

Geografi adalah ilmu tentang lokasi serta persamaan dan perbedaan (variasi) keruangan atau fenomena fisik dan manusia diatas permukaan bumi. Kata geografi berasal dari Bahasa Yunani yaitu ‘ge’ (bumi) dan ‘graphein’ (menulis / menjelaskan). Geografi tidak hanya menjawab apa dan dimana diatas muka bumi, tapi juga mengapa disitu dan ditempat lainnya, kadang diartikan dengan “ lokasi pada ruang”, geografi mempelajari hal ini, baik yang disebabkan oleh alam atau manusia, juga mempelajari akibat yang disebabkan dari perbedaan yang terjadi, istilah ini digunakan karena GIS dibangun berdasarkan pada geografi atau spasial.

Object ini mengarah pada spesifikasi lokasi dalam suatu space. Objek bisa berupa fisik, budaya atau ekonomi alamiah. Penampakan tersebut ditampilkan pada suatu peta untuk memberikan gambaran yang representatif dari spasial suatu objek sesuai dengan kenyataannya dibumi. Simbol, warna dan gaya garis digunakan untuk

mewakili setiap spasial yang berbeda pada peta dua dimensi. (Eko Budiarto; 2009 :

1)

Sistem Informasi Berbasis Pemetaan dan Geografis adalah sebuah alat bantu manajemen berupa informasi berbantuan komputer berkait erat dengan sistem pemetaan dan analisis terhadap segala sesuatu serta peristiwa-peristiwa yang terjadi dipermukaan bumi. Teknologi GIS mengintegrasikan operasi pengolahan data berbasis database yang digunakan saat ini pengambilan data berdasarkan kebutuhan, serta analisa statistik dengan menggunakan visualisasi yang khas berbagai keuntungan yang mampu ditawarkan melalui analisis geografi melalui gambar-gambar petanya. (Eko Budiarto; 2009 : 6)

Dengan demikian, SIG diharapkan mampu memberikan kemudahan-kemudahan seperti :

- a. Penanganan data geospasial menjadi lebih baik dalam format baku.
- b. Revisi dan pemutakhiran data menjadi lebih mudah.
- c. Data geospasial dan informasi menjadi lebih mudah dicari, dianalisis dan direpresentasikan.
- d. Menjadi produk yang mempunyai nilai tambah.
- e. Kemampuan menukar data geospasial.
- f. Penghematan waktu dan biaya.
- g. Keputusan yang diambil menjadi lebih baik.

II.5. Sistem Informasi Geografis

Geographic Information System (GIS), merupakan suatu sistem (berbasis komputer) yang digunakan untuk menyimpan dan memanipulasi informasi-informasi geografis. SIG dirancang untuk mengumpulkan, menyimpan, dan menganalisis objek-objek dan fenomena-fenomena dimana lokasi geografis merupakan karakteristik yang penting atau kritis untuk dianalisis. SIG merupakan sistem komputer yang memiliki empat kemampuan berikut dalam menangani data yang bereferensi geografis :

- a. Masukan
- b. Keluaran
- c. Manajemen data (penyimpanan dan pemanggilan data)
- d. Analisis dan manipulasi data (Sumber: Eddy Prahasta: 2009: 1)

II.6. ArcView

ArcView adalah software Sistem Informasi Geografis (SIG). Software SIG mempunyai kemampuan untuk menampilkan , memanipulasi dan merubah data SIG. Saat kita bekerja menggunakan SIG, kita bukan hanya harus mempelajari tentang software-nya tetapi juga datanya. (Eko Budiyanto: 2009: 1)

Data SIG mempunyai dua komponen, yaitu komponen spatial atau geografis dan komponen atribut atau tabel. Data spatial menampilkan lokasi geografis dari suatu features. Pada umumnya feature tersebut di tampilkan dalam bentuk titik

(point), garis (line), polygon (polygone). Sebuah feature titik mewakili sebuah lokasi seperti lokasi kantor pemerintahan dan sekolah. Feature garis yang juga sering disebut dengan arc/segmen mewakili fitur seperti lintasan atau sungai. Feature poligon mewakili fitur area seperti persil tanah, danau, atau penggunaan tanah.

II.7. UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah sebuah "bahasa" yang telah menjadi standar dalam industry untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. (Yuni Sugiarti, S.T.M.Kom; 2013 : 34). Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan syntax/semantic. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML syntax mendefinisikan bagaimana bentuk – bentuk tersebut dapat dikombinasikan.

Unified Modeling Language biasa digunakan untuk :

- a. Menggambarkan batasan sistem dan fungsi – fungsi sistem secara umum, di buat dengan use case dan actor.
- b. Menggambarkan kegiatan atau proses bisnis yang di laksanakan secara umum, di buat dengan interaction diagrams.
- c. Menggambarkan representasi struktur static sebuah sistem dalam bentuk class diagram.

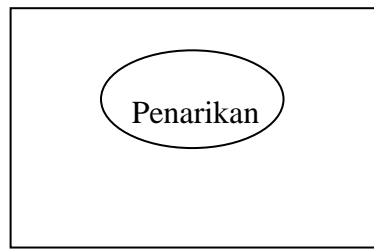
- d. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan state transition diagrams.
- e. Menyatakan arsitektur implementasi fisik menggunakan component and development diagrams.
- f. Menyampaikan atau memperluas functionality dengan stereotypes.

II.7.1. Notasi Dasar UML

II.7.1.1. Use Case Diagram

Use case adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara user sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dengan sistem disebut *scenario*. Setiap *scenario* mendeskripsikan urutan kejadian. Setiap urutan diinisialisasi oleh orang, sistem yang lain, perangkat keras atau urutan waktu. Dengan demikian secara singkat bisa dikatakan *use case* adalah serangkaian *scenario* yang digabungkan bersama – sama oleh tujuan umum pengguna.

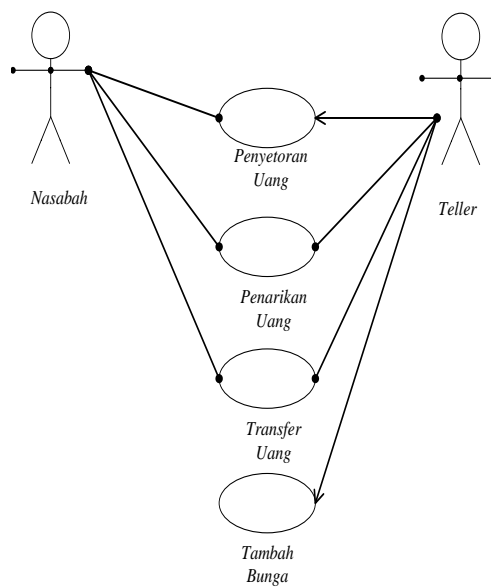
Menurut (Pilone, 2005: bab 7.1) *use case* menggambarkan fungsi tertentu dalam suatu sistem berupa komponen, kejadian atau kelas. Sedangkan (Whitten, 2004: 258) mengartikan *use case* sebagai urutan langkah-langkah yang secara tindakan saling terkait (Skenario), baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal. *Use case digambarkan dalam bentuk ellips/oval.*



Gambar II.1. Simbol Use Case

Sumber : (Prabowo Pudjo Widodo dan Herlawati, 2011)

Ilustrasi actor, *use case* dan *system* ditunjukkan pada gambar II.2



Gambar II.2. Use Case Diagram

Sumber : (Prabowo Pudjo Widodo dan Herlawati, 2011)

Untuk mengidentifikasi *actor*, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. *Actor* adalah

abstraction dari orang dan sistem yang lain mengidentifikasi fungsi dari target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa *actor* berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atau *use case*.

Use case adalah abstraksi dari interaksi antara sistem dan *actor*. Oleh karena itu sangat penting untuk memilih abstraksi yang cocok. *Use case* dibuat berdasarkan keperluan *actor*. *Use case* harus merubah ‘apa’ yang dikerjakan *software* aplikasi, bukan ‘bagaimana’ *software* aplikasi mengerjakannya. Setiap *use case* harus diberi nama yang menyatakan apa hal yang dicapai dari hasil interaksinya dengan *actor*. Namun *use case* boleh terdiri dari beberapa kata dan tidak boleh ada *use case* yang memiliki nama yang sama.

II.7.1.2 Class Diagram


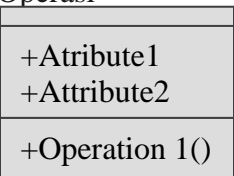




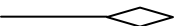
Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan di buat untuk membangun sistem. Kelas memiliki apa yang di sebut atribut dan metode atau operasi.

1. Atribut merupakan variabel- variabel yang di miliki oleh suatu kelas.
2. Atribut mendeskripsikan properti dengan sebaris teks di dalam kotak kelas tersebut.
3. Operasi atau metode adalah fungsi – fungsi yang di miliki oleh suatu kelas.

Diagram kelas mendeskripsikan jenis – jenis objek dalam sistem dan berbagai hubungan statis yang terdapat di antara mereka. Diagram kelas juga menunjukkan

properti dan operasi sebuah kelas dan batasan – batasan yang terdapat dalam hubungan – hubungan objek tersebut. Berikut ini ditunjukkan pada Tabel II.1 :

Tabel II.1 Simbol – symbol yang ada pada diagram kelas.

Simbol	Deskripsi
Package 	Package merupakan sebuah bungkusan dari satu atau lebih kelas
Operasi 	Kelas pada struktur sistem
Antarmuka / interface 	Sama dengan konsep interface dalam pemrograman berorientasi objek
Asosiasi 1 _____ 1..*	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
Asosiasi berarah/directed asosiasi 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
Generalisasi 	Relasi antar kelas dengan makna generalisasi – spesialisasi (umum khusus).
Kebergantungan / defedency 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi 	Relasi antar kelas dengan makna semua bagian (whole-part)

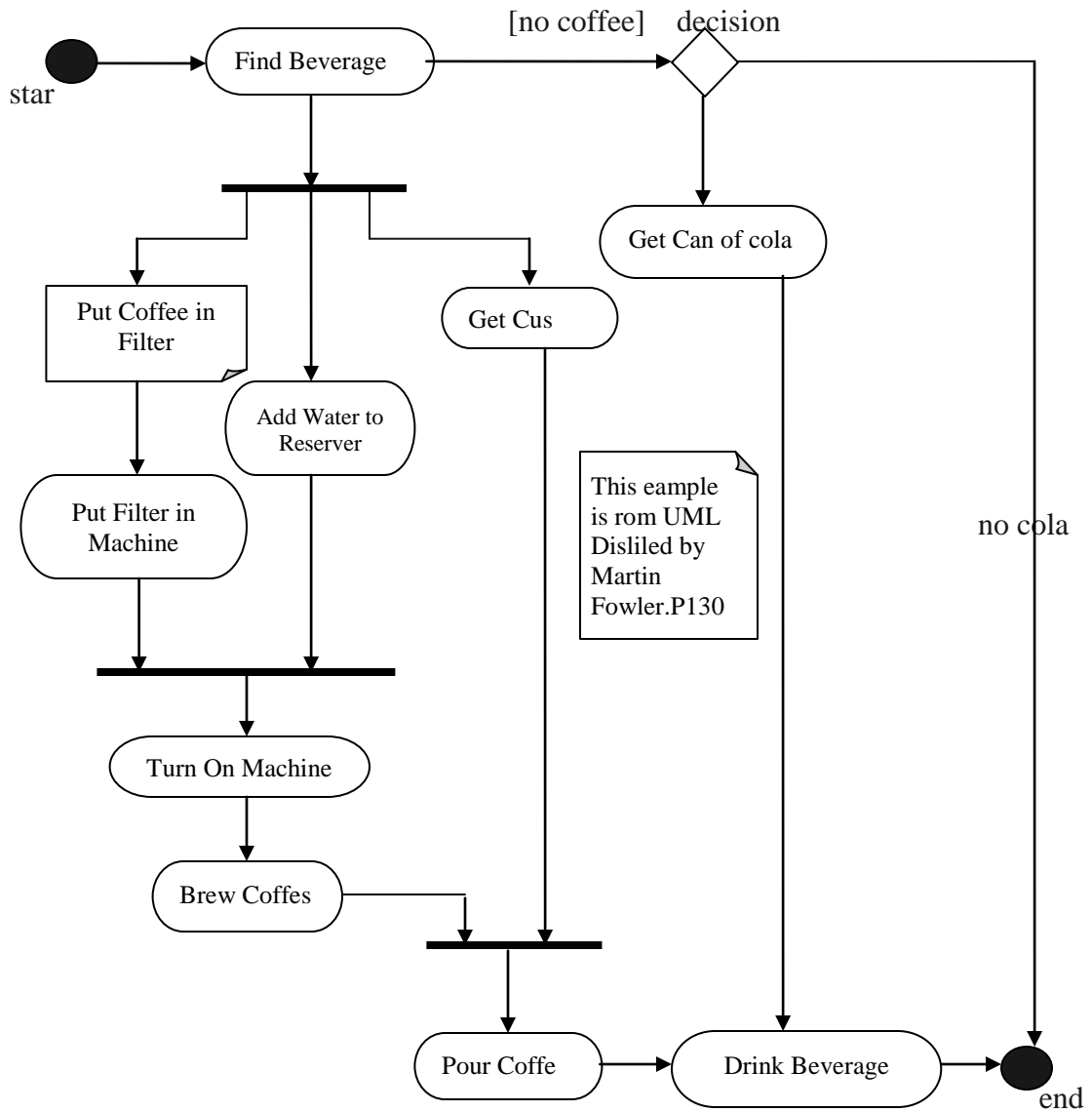
Sumber : (Yuni Sugiarti, S.T.M.Kom ; 2013 :59)

II.7.1.3 Activity Diagram (Aktivitas)

Diagram aktivitas atau *activity diagram* menggambarkan workflow (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis.

Activity diagram merupakan state diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (*internal processing*). Oleh karena itu activity diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.



Gambar II.3 Activity Diagram Tanpa Swimlane

Sumber : Analisis & Perancangan UML Generated VB.6 (Yuni Sugiarti, S.T.M.Kom ; 2013 : 76)

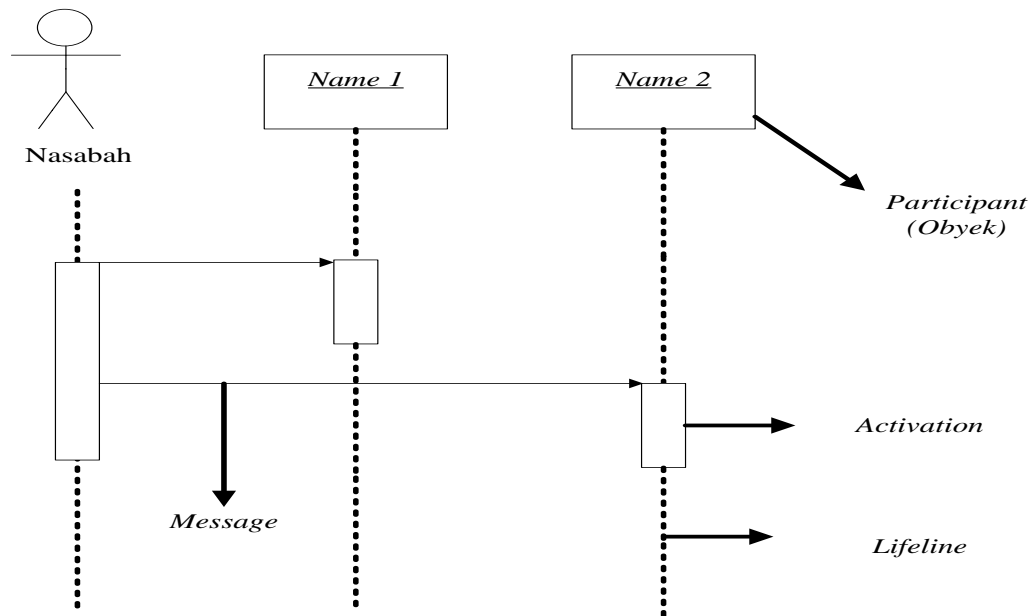
II.7.1.4 Sequence Diagram

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

Diagram sequence memiliki ciri yang berbeda dengan diagram interaksi pada diagram kolaborasi sebagai berikut :

1. Pada diagram sequence terdapat garis hidup objek. Garis hidup objek adalah garis vertical yang mencerminkan eksistensi sebuah objek sepanjang periode waktu. Sebagian besar objek – objek yang tercakup dalam diagram interaksi akan eksis sepanjang durasi tertentu dari interaksi, sehingga objek – objek itu diletakkan dibagian atas diagram dengan garis hidup tergambar dari atas hingga bagian bawah diagram. Suatu objek lain dapat saja diciptakan, dalam hal ini garis hidup dimulai saat pesan destroy, jika kasus ini terjadi, maka garis hidupnya juga berakhir.
2. Terdapat focus kendali (*Focus Of Control*), berupa empat persegi panjang ramping dan tinggi yang menampilkan aksi suatu objek secara langsung atau sepanjang sub ordinat. Puncak dari empat persegi panjang adalah permulaan aksi, bagian dasar adalah akhir dari suatu aksi. Pada diagram ini mungkin juga

memperhatikan penyaringan (*nesting*) dan focus kendali yang disebabkan oleh proses rekursif dengan menumpuk focus kendali yang lain pada induknya. Berikut simbol – simbol yang ada pada sequence diagram



Gambar II.4 Sequence Diagram
Sumber: (Yuni Sugiarti, S.T,M.Kom)

II.8. PHP (*Hypertext Preprocessor*)

PHP adalah bahasa pemrograman (*script*) yang digunakan untuk membuat halaman *web* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru atau *up to date*. Semua script PHP dieksekusi pada server dimana script tersebut dijalankan. (Rulianto Kurniawan; 2010 : 3).

II.8.1 Kelebihan PHP dari Bahasa Pemrograman Web

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. *Web Server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai *apache*, *IIS* hingga *xitami* dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (*Linux*, *Unix Macintosh*, *Windows*) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah system. (Alan Nur Aditya ; 2011 : 2)

II.8.2 Fungsi Dalam PHP

Fungsi merupakan hal yang paling penting dalam membuat aplikasi *web*. Dengan membagi kode-kode yang ada ke dalam fungsi-fungsi akan memudahkan kita apabila kita menggunakan kembali kode tersebut. Atau apabila kita ingin membuat *website* dengan fitur yang sama dengan *website* yang pernah kita buat maka kita cukup menggunakan fungsi-fungsi yang pernah kita buat.

Hal ini akan sangat menghemat waktu dan mempercepat proses pembuatan *website*. Fungsi pada PHP *syntax*-nya adalah *function namafungsi()* dimana

namafungsi merupakan nama fungsi tersebut dan bisa kita ganti sesuai yang kita inginkan. (Alan Nur Aditya ; 2011 : 54)

II.9. MySQL

MySQL adalah salah satu jenis database server yang sangat terkenal. MySQL termasuk jenis RDBMS (*Relational Database Management System*). MySQL ini mendukung bahasa pemrograman PHP. MySQL juga mempunyai query atau bahasa SQL (*Structured Query Language*) yang simple dan menggunakan *escape character* yang sama dengan PHP. (Rulianto Kurniawan; 2010 : 17)

II.10. ERD (*Entity Relationship Diagram*)

ERD (*Entity Relationship Diagram*) adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas – entitas dan menentukan hubungan antarentitas. Proses memungkinkan analisis menghasilkan struktur basisdata yang baik sehingga data dapat di simpan dan di ambil secara efisien. (Janner Simarmata & Iman Paryudi; 2010 : 67)

II.10.1 Entitas (*Entity*)

Entitas adalah suatu yang nyata atau abstrak di mana kita akan menyimpan data. Ada 4 kelas entitas, yaitu misalnya pegawai, pembayaran, kampus, dan buku. Contoh suatu entitas di sebut instansi, misalnya pegawai Adi, pembayaran Joko, dan lain sebagainya.

II.10.2 Relasi (*Relationship*)

Relasi adalah hubungan alamiah yang terjadi antara satu atau lebih entitas, misalnya proses pembayaran pegawai. Kardinalitas menentukan kejadian suatu entitas untuk satu kejadian pada entitas yang berhubungan. Misalnya, mahasiswa bias mengambil banyak mata kuliah.

II.10.3 Atribut (*Attribute*)

Atribut adalah cirri umum semua atau sebagian besar instansi pada entitas tertentu. Sebutan lain atribut adalah property, elemen data, dan field. Misalnya nama, alamat, nomor pegawai, dan gaji adalah etribut entitas pegawai.

Berikut adalah metodologi ERD yang ada pada alat pemodelan data :

Tabel II.2 Metodologi ERD

1. Menentukan Entitas	Menentukan peran, kejadian, lokasi, hal nyata, dan konsep di mana menggunakan menyimpan data.
2. Menentukan Relasi	Menentukan hubungan antar pasangan entitas menggunakan matriks relasi.
3. Gambar ERD sementara	Entitas di gambarkan dengan kotak dan relasi dengan garis yang menghubungkan entitas.
4. Isi Kardinalitas	Menentukan jumlah kejadian satu entitas untuk sebuah kejadian pada entitas yang berhubungan.
5. Tentukan Kunci Utama	Menentukan atribut yang mengidenrifikasi satu dan hanya satu kejadian masing – masing entitas.
6. Gambar ERD berdasarkan Kunci	Menghilangkan relasi many-to-many dan memasukkan primary dan kunci tamu pada masing – masing entitas.
7. Menentukan Atribut	Menuliskan field – field yang di perlukan oleh sisitem.
8. Pemetaan Atribut	Untuk masing – masing atribut, memasang atribut dengan satu entitas yang sesuai.
9. Gambar ERD dengan Atribut	Mengatur ERD dari langkah 6 dengan menambah entitas atau relasi yang di temukan pada langkah 8.
10. Periksa Hasil	Apakah ERD sudah menggambar sistem yang akan di bangun?

Sumber : (Janner Simarmata & Iman Paryudi; 2010: 68)

II.11. Normalisasi

Normalisasi adalah teknik perancangan yang banyak digunakan sebagai pemandu dalam merancang basisdata relasional. Pada dasarnya, normalisasi adalah proses dua langkah yang meletakkan data dalam bentuk tabulasi dengan menghilangkan kelompok berulang lalu menghilangkan data yang terduplikasi dari tabel relasional (www.utexas.edu). (Janner Simarmata & dkk; 2010 : 77)

II.11.1 Bentuk – Bentuk Normalisasi

Adapun bentuk – bentuk dalam normalisasi adalah sebagai berikut :

1. Bentuk tidak normal
2. Bentuk normal tahap pertama (1st Normal Form)

Dicontohkan dari tabel Kuliah yang memiliki atribut: kode_kul, sks, semester, waktu, tempat dan nama_dos. Tabel tersebut tidak memenuhi normalisasi pertama, karena terdapat atribut waktu yang tergolong ke dalam atribut bernilai banyak. Agar tabel tersebut dapat memenuhi 1NF, maka solusinya adalah mendekomposisikan tabel Kuliah menjadi :

- Tabel Kuliah (kode_kul, nama_kul, sks, semester, nama_dos)
- Tabel Jadwal (kode_kul, waktu, ruang).

3. Bentuk normal tahap kedua (2nd Normal Form)

Misal tabel Nilai terdiri dari atribut kode_kul, nim dan nilai. Jika pada Nilai, misalnya kita tambahkan sebuah atribut yang bersifat redundan, yaitu nama_mhs, maka tabel Nilai ini di anggap melanggar 2NF. Primary key pada Nilai adalah

[kode_kul,nim] Bentuk normal kedua ini di anggap belum memadai karena meninjau sifat ketergantungan atribut terhadap primary key saja.

4. Bentuk normal tahapan ketiga (3rd Normal Form)

Misalnya pada tabel Mahasiswa, atribut alamat_mhs di pecah kedalam alamat_jalan, alamat_kota dan kode-pos. Bentuk ini tidak memenuhi 3NF karena terdapat ketergantungan fungsional baru yang muncul pada tabel tersebut.

5. Bentuk normal tahap keempat dan kelima

6. *Boyce Code Normal Formal* (BCNF) (Kusrini, M.Kom; 2007: 39-43).

II.12. Kamus Data

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. (Janner Simarmata & dkk; 2010 : 67).