

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau *variable* yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Teori sistem secara umum pertama kali diuraikan oleh Kenneth Boulding, terutama menekankan pentingnya perhatian terhadap setiap bagian yang membentuk sebuah sistem. Kecenderungan manusia yang mendapat tugas untuk memimpin organisasi adalah dia terlalu memusatkan perhatiannya pada salah satu komponen sistem organisasi. Teori sistem mengatakan bahwa setiap unsur pembentuk organisasi itu penting dan harus mendapat perhatian yang utuh supaya manajer dapat bertindak lebih efektif. Yang dimaksud unsur atau komponen pembentuk organisasi disini bukan hanya bagian bagian yang tampak secara fisik, tetapi juga hal-hal yang mungkin bersifat abstrak atau konseptual, seperti misi, pekerjaan, kegiatan, kelompok informal, dan lain sebagainya. (Tata Sutabri : 2012 : 3).

Adapun karakteristik dari sistem adalah sebagai berikut:

1. Komponen sistem
2. Batasan sistem
3. Lingkungan luar sistem
4. Penghubung sistem
5. Masukan sistem
6. Keluaran sistem

7. Pengolahan sistem
8. Sasaran sistem

II.1.1. Karakteristik Sistem

Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. Komponen Sistem (*Components*).

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan.

2. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem.

4. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber – sumber daya mengalir dari satu subsistem ke subsistem lain.

5. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan *signal (signal input)*.

6. Keluaran Sistem (*output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

Keluaran ini merupakan masukan bagi subsistem yang lain.

7. Pengolah sistem (*Proses*)

Sudah sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi kelauran.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. 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 : 13).

II.1.2. Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya :

1. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide – ide yang tampak secara fisik, misalnya komputer, sistem produksim sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

2. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, sedangkan sistem buatan manusia merupakan sistem yang melibatkan interaksi manusia dengan mesin, yang disebut *human machine system*.

3. Sistem deterministik dan sistem probabilistik

Sistem yang beroperasi dengan tingkah yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program – program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistik.

4. Sistem terbuka dan Sistem tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan lainnya. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. (Tata Sutabri : 2012 : 15).

II.1.3. Informasi

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi dan lain sebagainya. (Tata Sutabri : 2012 : 22).

Kualitas informasi tergantung dari 3 hal yaitu :

1. Akurat (*Accurate*)

Informasi harus bebas dari kesalahan – kesalahan dan tidak menyesatkan.

2. Tepat waktu (*Timeline*)

Informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi karena informasi merupakan landasan dalam pengambilan keputusan.

3. Relevan (*Relevance*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk orang satu dengan yang lain berbeda, misalnya informasi sebab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan dan akan lebih relevan bila ditunjukkan ahli teknik perusahaan. (Tata Sutabri : 2012 : 33).

II.1.4. Sistem Informasi

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu

organisasi untuk dapat menyediakan kepada pihak luar dengan laporan-laporan yang diperlukan (Tata Sutabri : 2012 : 38). Sistem informasi terdiri dari komponen – komponen yang disebut blok bangunan (*building block*) yaitu :

1. Blok masukan (*input block*)

Input mewakili data yang masuk kedalam sistem informasi, Input disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen – dokumen dasar.

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 di inginkan.

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, teknologi terdiri dari 3 bagian utama, yaitu teknisi (*brainware*), Perangkat lunak (*software*), perangkat keras (*hardware*).

5. Blok basis data (*database block*)

Basis data atau *database* merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer an menggunakan perangkat lunak untuk memanipulasinya.

6. Blok kendali (*control block*)

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air ,debu, kecurangan – kecurangan, kegagalan – kegagalan. (Tata Sutabri : 2012 : 39-40).

II.2. Sistem Pakar

Pakar adalah seseorang yang memiliki kemampuan khusus terhadap suatu permasalahan, misalnya dokter, petani, ahli permesinan dan lain-lain. Ciri-ciri pakar adalah sebagai berikut :

1. Dapat mengenali dan merumuskan masalah
2. Menyelesaikan masalah dengan cepat dan tepat
3. Belajar dari pengalaman
4. Restrukturisasi pengetahuan
5. Menentukan relevansi.

Jenis-jenis pengetahuan yang dimiliki dalam kepakaran adalah sebagai berikut :

1. Teori-teori dari permasalahan
2. Aturan dan prosedur yang mengacu pada area permasalahan
3. Aturan yang harus dikerjakan pada situasi yang terjadi

4. Strategi global untuk menyelesaikan berbagai jenis masalah
5. *Meta-knowledge* (pengetahuan tentang pengetahuan)
6. Fakta-fakta. (Nita Merlina, M.Kom:2012:1)

Sistem pakar (*expert system*) merupakan solusi AI bagi masalah pemrograman pintar. Profesor Edward Feigenbaum dari Stanford University yang merupakan pionir dalam teknologi sistem pakar mendefinisikan sistem pakar sebagai sebuah program komputer pintar yang memanfaatkan pengetahuan dan prosedur inferensi untuk memecahkan masalah yang cukup sulit hingga membutuhkan keahlian khusus dari manusia. Dengan kata lain, sistem pakar adalah sistem komputer yang ditujukan untuk meniru semua aspek kemampuan pengambilan keputusan seorang pakar. Sistem pakar memanfaatkan secara maksimal pengetahuan khusus selayaknya seorang pakar untuk memecahkan masalah. (Rika rosnely:2012:2).

II.2.1. PHP

PHP merupakan singkatan dari *Hypertext Preprocessor*. Pada awalnya PHP merupakan kependekan dari *Personal Home Page* (situs personal) dan PHP itu sendiri pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995 dan pada saat itu PHP masih bernama FI (*Form Interpreter*) yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data *form* dari *web*. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum. 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 spesifik dan mudah dimengerti. PHP dibuat untuk membuat tampilan

web menjadi lebih dinamis, dengan PHP anda bisa membuat atau menjalankan beberapa *file* dalam satu *file* dengan cara di-*include* atau *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa *database* walaupun dengan kelengkapan yang berbeda, yaitu seperti DBM, Filepro, Informix, Ingres, Interbase, Microsoft Access, MSSQL, MYSQL, Oracle, PostgrSQL dan Sybase. Dari uraian diatas maka dapat diambil 4 *point* utama tentang PHP yaitu :

1. PHP adalah singkatan dari *Hypertext Preprocessor*.
2. PHP adalah bahasa *scripting server side*, artinya dijalankan di *server* kemudian *outputnya* dikirimkan ke *client (browser)*.
3. PHP digunakan untuk membuat aplikasi *web*.
4. PHP mendukung banyak *database* (MYSQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, dll).

Cara kerja PHP yaitu pertama *client web browser* atau pengguna memakai komputer atau laptop kemudian pengguna menjalankan *file* PHP itu di *web browser*, kemudian *file* PHP itu dikirim ke *web server*. *Web server* mengirimkannya lagi ke *engine* PHP atau mesin PHP dan di dalam mesin PHP itu diproses dan setelah diproses oleh mesin PHP maka akan berbentuk *file* HTML, dan oleh *file* HTML ini akan dikirimkan ke *web server* dan *web server* akan memberikan ke pengguna. *Software-software* yang harus dipersiapkan untuk menjalankan PHP adalah :

1. Program PHP
2. *Web server apache*
3. *Database Mysql*

4. Editor (contoh: adobe dreamweaver, Editplus, Notepad, PHP editor dan lain-lain)
5. Mysql editor (contoh: phpmyadmin, mysql front, dan lain-lain)
6. *Web browser* (contoh: mozilla firefox, internet explorer, dan lain-lain)

Untuk program PHP, apache, mysql, phpmyadmin, banyak distribusi program yang sudah dipaket yang berisi ketiga program diatas. Artinya dengan satu kali instal, maka otomatis akan terinstal ketiga program diatas, misalnya PHPtriad, Appserv, Vertigo, Xampp, Wamp dan banyak lagi yang lainnya. (Madcoms :2011:228-229).

II.2.2. *MySQL*

MySQL adalah salah satu jenis *database server* yang sangat terkenal. Kepopulerannya disebabkan *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses *databasenya*. Selain itu, ia bersifat *Open Source* (Anda tidak perlu membayar untuk menggunakannya).

Database digunakan untuk penyimpanan data, demikian pula dengan *MYSQL*. Kita akan memanggil data pada *MYSQL* melalui *PHP*, kemudian hasilnya dikirim kekomputer *klien* untuk ditampilkan pada *browser*. Data pada *MYSQL* dapat dipanggil, dihapus atau di tambah melalui *query* .(Dadan Sutisna : 2008 : 46)

Langkah untuk menjalankan *webservice local* adalah :

1. Buka program *browser*, misalnya ***Mozilla Firefox*** atau ***Internet Explorer***.

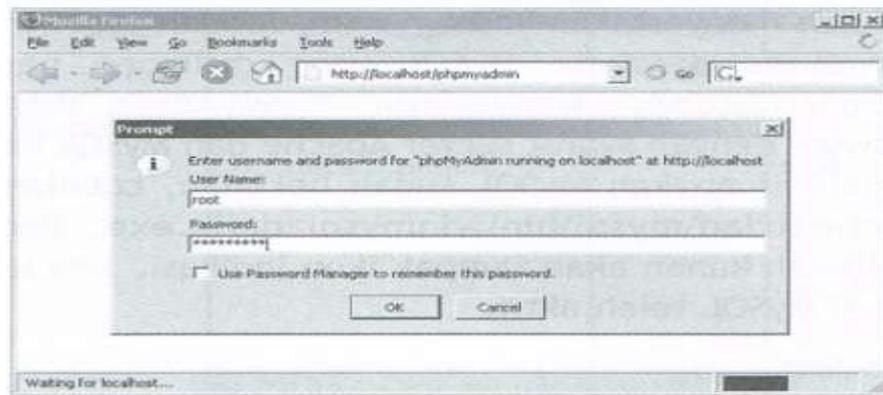
2. Dalam jendela *browser*, ketik *localhost* dibagian *Address*, kemudian tekan *Enter*. Jika proses instalasi benar maka akan tampil halaman *localhost* dapat dilihat pada gambar II.1.



Gambar II.1. Halaman Awal Localhost

Sumber : (Dadan Sutisna : 2008 : 57)

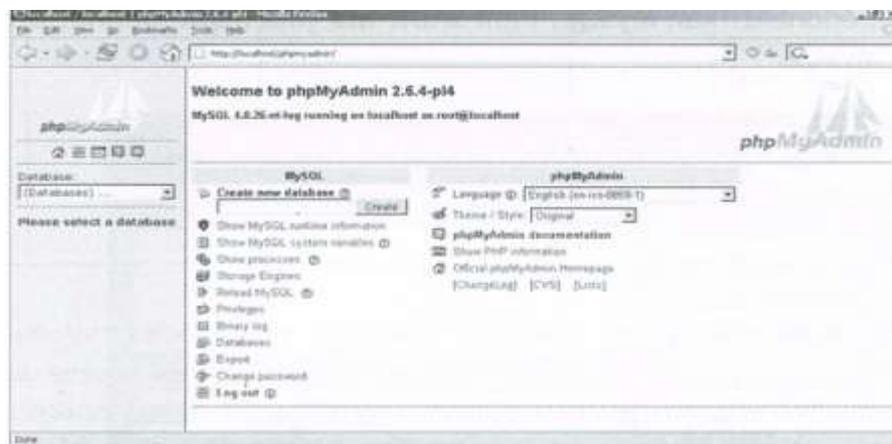
3. Gambar II.1 juga menampilkan informasi 4 komponen pendukung *webservice* yang telah terinstal pada komputer sekaligus menampilkan versi yang digunakan.
4. Berikutnya program akan meminta memasukkan *user* dan *password* dari *server MySQL*. Ketik *user* nama *root* pada kotak *User Name*, dan masukkan *password*-nya pada kotak *Password* sesuai dengan *password* pada saat penginstalan. Tampilan konfirmasi permintaan *username* dan *password* *server MySQL* dapat dilihat pada gambar II.2.



Gambar II.2. Konfirmasi Permintaan Username dan Password Server MySQL

Sumber : (Dadan Sutisna : 2008 : 58)

Jika *username* dan *password* yang dimasukkan benar, maka tampil halaman *phpMyAdmin* tampak pada gambar II.3.



Gambar II.3. Tampilan Halaman PhpMyAdmin

Sumber : (Dadan Sutisna : 2008 : 58)

II.2.3. Metode *Forward Chaining*

Forward chaining merupakan pencocokan fakta atau pernyataan dimulai dari bagian sebelah kiri (IF dulu). Dengan kata lain, penalaran dimulai dari fakta dahulu untuk menguji kebenaran hipotesis. (Rika Rosnely:2012:57).

Pada saat melakukan pengujian untuk menentukan jenis buah durian maka diambil beberapa data dari penelitian terhadap pedagang buah durian. Kemudian dari data yang ada tersebut maka akhirnya akan didapat sebuah hasil yang menyatakan jenis buah durian.

Pengertian lain dari metode *forward chaining* adalah metode pencarian atau penarikan kesimpulan yang berdasarkan pada data atau fakta yang ada lalu bergerak maju melalui premis-premis untuk menuju ke kesimpulan. *Forward chaining* melakukan pencarian dari suatu masalah kepada solusinya. Forward chaining adalah pendekatan data driven yang dimulai dari informasi yang tersedia atau dari ide dasar, kemudian mencoba menarik kesimpulan.

Data aturan kesimpulan :

A = 1 IF A = 1 AND B = 2

B = 2 THEN C = 3 C = 3

Contoh :

IF akar tanaman rusak.

AND terdapat telur-telur ulat pada rerumputan.

THEN terserang hama ulat grayak. (Nita Merlina, M.Kom:2012:22)

II.3. Alat Bantu Pengembangan Sistem

Alat bantu pengembangan sistem yang digunakan dalam penulisan skripsi ini adalah sebagai berikut :

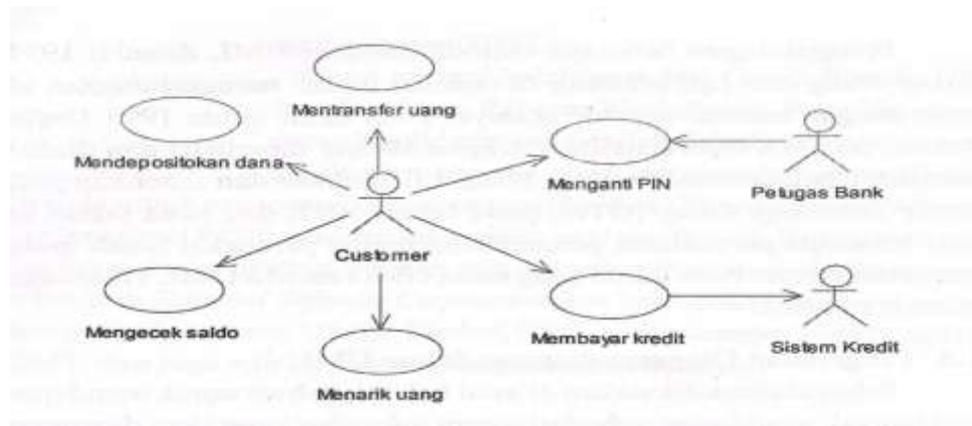
II.3.1 UML (Unified Modelling Language)

Unified Modelling Language (UML) merupakan alat komunikasi yang konsisten dalam mensupport para pengembang sistem saat ini. Notasi UML dibuat sebagai kolaborasi dari Grady Booch, DR.James Rumbough, Ivar Jacobson, Rebecca Wirfs Brock, Peter Yourdon dan lainnya. Jacobson menulis tentang pendefinisian persyaratan-persyaratan sistem yang disebut *use case*, dan juga mengembangkan sebuah metode untuk perancangan sistem yang disebut *Object Oriented Software Enginnering (OOSE)* yang berfokus pada analisis. Booch, Rumbough, dan Jacobson biasa disebut dengan tiga sekawan.

II.3.2 Diagram – diagram Pada Metode UML

1. Use Case Diagram

Diagram *use case* atau *use case* diagram menyajikan interaksi antara *use case* dan aktor. Dimana aktor dapat berupa orang, peralatan atau sistem lain yang berinteraksi dengan sistem yang dibangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai. (Sholiq ; 2006 :7).

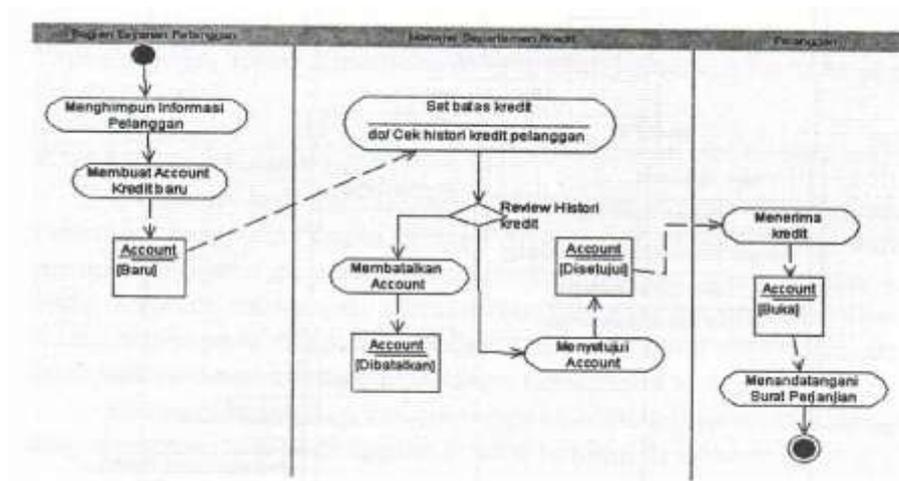


Gambar II.4. Use Case Diagram

Sumber : (Sholih ; 2006 :8)

2. Activity diagram

Diagram aktifitas atau *Activity diagram* menggambarkan aliran fungsional sistem. Pada tahap pemodelan bisnis, diagram aktifitas dapat digunakan untuk menggambarkan aliran kejadian (*flow of events*) dalam *use case*. (Sholih ; 2006 :8). Berikut gambar dari sederhana dari *activity diagram*.

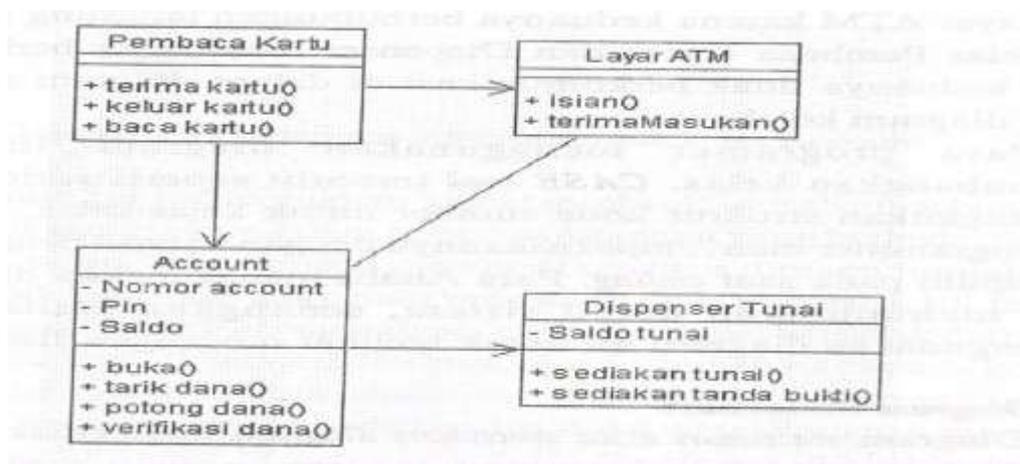


Gambar II.5 Contoh Activity Diagram

Sumber : (Sholih ; 2006 :9)

3. Class Diagram

Diagram kelas atau *class diagram* menunjukkan interaksi antar kelas dalam sistem. Sebagai contoh, nomor *account* milik Arvin adalah sebuah objek dari kelas *account*. Kelas mengandung informasi dan tingkah laku yang berkaitan dengan informasi tersebut. (Sholih ; 2006 :13).

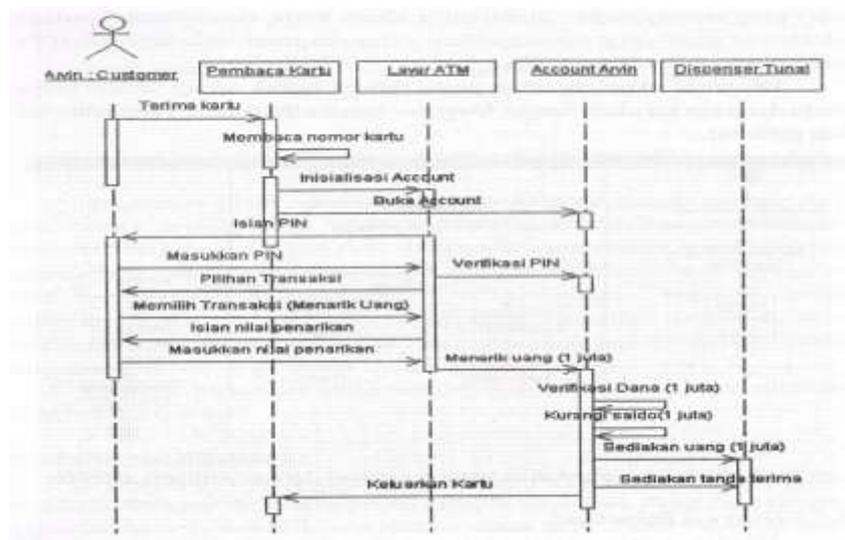


Gambar II.6 Contoh Class Diagram

Sumber : (Sholih ; 2006 :13)

4. Sequence Diagram

Diagram sekuensial atau *Sequence Diagram* digunakan untuk menunjukkan aliran fungsional dalam *use case*. Misalkan dalam *use case* menarik uang, mempunyai beberapa kemungkinan seperti penarikan uang secara normal, percobaan penarikan uang tanpa kecukupan ketersediaan dana, penarikan dengan penggunaan PIN yang salah dan lainnya. Diagram ini menunjukkan aliran proses dalam *use case* menarik uang. (Sholih ; 2006 :10).



Gambar II.7 Contoh Activity Diagram Sederhana

Sumber : (Sholih ; 2006 :10)

II.3.3. Sistem Basis Data

Dalam keseharian, sering terjadi kerancuan makna antara istilah basis data dan sistem basis data, yang semestinya perlu dibedakan. Istilah sistem basis data tentu saja berbeda dengan istilah basis data. Sistem basis data mempunyai lingkup yang lebih luas dibandingkan dengan basis data. Sistem basis data dapat diartikan sebagai sekumpulan basis data dalam suatu sistem yang mungkin tidak ada hubungan satu sama lain, tetapi secara keseluruhan mempunyai hubungan sebagai sebuah sistem dengan di dukung oleh komponen lainnya. Istilah sistem basis data juga dapat didefinisikan sebagai sekumpulan subsistem yang terdiri atas basis data dengan para pemakai yang menggunakan basis data secara bersama-sama, personal-personal yang merancang dan mengelola basis data, teknik-teknik untuk

merancang dan mengelola basis data, serta sistem komputer untuk mendukungnya (Edhy Sutanta:2011:32).

Dari definisi tersebut dapat disimpulkan bahwa sistem basis data mempunyai beberapa elemen penting yaitu :

1. Basis data sebagai inti dari sistem basis data.
2. Perangkat lunak (*software*) untuk perancangan dan pengelolaan basis data.
3. Perangkat keras (*hardware*) sebagai pendukung operasi pengelolaan data.
4. Manusia (*brainware*) yang mempunyai peran penting dalam sistem tersebut, yaitu sebagai pemakai atau para spesialis informasi yang mempunyai fungsi sebagai perancang atau pengelola.

Perangkat lunak untuk pengelolaan basis data merupakan perangkat lunak yang umumnya mempunyai dua fungsi utama, yaitu untuk mendefinisikan data dalam basis data dan untuk mengakses pengelolaan data dalam basis data tersebut. Secara lebih luas perangkat lunak dapat dikategorikan dalam tiga bagian yaitu :

1. Perangkat lunak sistem operasi (*operating system/OS*), yaitu program yang ditulis untuk mengendalikan dan mengkoordinasi kegiatan dari perangkat keras sistem komputer. Contoh perangkat lunak sistem operasi adalah MS DOS, PC DOS, Windows, Unix, Linux.
2. Perangkat lunak bahasa (*language software*) yaitu program yang digunakan untuk menerjemahkan instruksi-instruksi yang ditulis dalam bahasa pemrograman ke dalam bahasa mesin supaya dapat dimengerti oleh

komputer. Contoh perangkat lunak bahasa adalah Pascal, C++, Foxpro, Foxbase, Visual Foxpro, Delphi, dan masih banyak lagi.

3. Perangkat lunak aplikasi (*application software*) yaitu program yang ditulis dan diterjemahkan oleh language software untuk menyelesaikan aplikasi tertentu. (Edhy Sutanta:2011:33).

II.4. Konsep DataBase

Database adalah kumpulan beragam elemen informasi yang akan digunakan demi tujuan klasifikasi. Terdapat tiga tingkat arsitektur yang terkait dengan *database* dan sistem manajemen *database* yaitu tingkat konseptual, tingkat logika, dan tingkat fisik.

Dalam konsep perancangan *database* terdapat pendekatan yang perlu diperhatikan yaitu sebagai berikut :

II.4.1 Normalisasi

Normalisasi diartikan sebagai suatu teknik yang menstrukturkan data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data. Permasalahan yang dimaksud adalah berkaitan dengan penyimpangan-penyimpangan yang terjadi akibat adanya kerangkapan data dalam relasi dan in-efisiensi pengolahan. (Edhy Sutanta ; 2011 : 174).

Proses normalisasi menghasilkan relasi yang optimal, yaitu :

1. Memiliki struktur *record* yang konsisten secara logik.
2. Memiliki struktur *record* yang mudah untuk dimengerti.
3. Memiliki struktur *record* yang sederhana dalam pemeliharaan.

4. Memiliki struktur *record* yang mudah ditampilkan kembali untuk memenuhi kebutuhan pengguna.
5. Minimalisasi kerangkapan data guna meningkatkan kinerja sistem. (Edhy Sutanta ; 2011 : 175).

Bentuk-bentuk normal *first norm form/1NF*, *second norm form/2NF*, dan *third norm form/3NF* dikemukakan oleh E.F.Codd, sedangkan bentuk normal *Boyce-Codd norm form/BCNF*, dikemukakan oleh R.F Boyce dan E.F. Codd. Bentuk normal BCNF, *forth norm form/4NF*, dan *fifth norm form/5NF* dapat terjadi pada relasi-relasi yang memiliki ketergantungan pada banyak nilai. Bentuk *Domain Key Norm Form/DKNF* dan *restriction union norm form/RUNF* dapat terjadi pada relasi-relasi yang bersifat sangat spesifik sehingga tidak semua relasi memungkinkan untuk mencapai level ini. Umumnya rancangan relasi dalam basis data telah optimal jika telah memenuhi kriteria bentuk 3NF. Level normalisasi ditentukan berdasarkan kriteria bentuk normal, bukan banyaknya langkah menstrukturkan sebuah relasi. (Edhy Sutanta:2011:175).

II.4.2 Kamus Data

Kamus Data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data selain digunakan untuk dokumentasi dan mengurangi redundansi, juga dapat digunakan untuk:

1. Memvalidasi diagram aliran data dalam hal kelengkapan dan keakuratan.

2. Menyediakan suatu titik awal untuk mengembangkan layar dan laporan-laporan.
3. Menentukan muatan data yang disimpan dalam *file-file*.
4. Mengembangkan logika untuk proses-proses diagram aliran data.

Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem. Pada tahap analisis sistem, Kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. Pada tahap perancangan sistem, Kamus data digunakan untuk merancang input, merancang laporan-laporan dan *database*.

Kamus data mendefinisikan elemen data dengan fungsi sebagai berikut:

1. Menjelaskan arti aliran data dan penyimpanan data dalam DFD
2. Mendeskripsikan komposisi paket data yang bergerak melalui aliran (misalnya alamat diuraikan menjadi kota, negara dan kode pos)
3. Mendeskripsikan komposisi penyimpanan data
4. Menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran
5. Mendeskripsikan hubungan detil antar penyimpanan (yang akan menjadi titik perhatian dalam *entity-relationship diagram*). (Jurnal Kamus Data:66)

Tabel II.1 Simbol-simbol dalam kamus data

Sumber : (Jurnal Kamus Data:66)

Simbol	Uraian
=	Terdiri atas,mendefinisikan,diuraikan menjadi,artinya

	Contoh : Nama=Sebutan + Nama1+Nama2 + Gelar1 + Gelar2
+	Dan
()	Optional (pilihan boleh ada atau boleh tidak) Contoh : Alamat = Alamat rumah + (Alamat surat)
{ }	Pengulangan Contoh : Nama1 = {Karakter _valid}
[]	Memilih salah satu dari sejumlah alternatif,seleksi. Contoh : Sebutan =[Bapak Ibu Yang Mulia]
**	Komentar Contoh : *Seminar yang akan diikuti*
	Pemisah sejumlah alternatif pilihan antara simbol []
@	Identifikasi atribut kunci

II.4.3. E-R Diagram (ERD)

ERD merupakan suatu model data yang dikembangkan berdasarkan objek. ERM digambarkan dalam bentuk diagram yang disebut ER (ER

Diagram/ERD). Untuk menggambarkan ERD digunakan simbol-simbol grafis tertentu. Sebuah diagram ERD tersusun atas tiga komponen yaitu entitas, atribut dan kerelasiaan antar entitas. Secara garis besar, entitas merupakan objek dasara yang terlibat dalam sistem. Atribut berperan sebagai penjelas entitas, sedangkan kerelasiaan menunjukkan hubungan yang terjadi di antara dua entitas. (Edhy Sutanta:2011:91-92).

1. Entitas (*Entity*)

Entity menunjukkan objek-objek dasar yang terkait didalam sistem. Objek dasar dapat berupa orang, benda atau hal yang keterangannya perlu disimpan di dalam basis data. Untuk menggambarkan sebuah entitas digunakan aturan sebagai berikut :

- a. Entitas dinyatakan dengan simbol persegi panjang
- b. Nama entitas dituliskan didalam simbol persegi panjang
- c. Nama entitas berupa kata benda tunggal
- d. Nama entitas sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas. (Edhy Sutanta:2011:92).

Sebagai contoh , dalam suatu subsistem pengolahan data akademik yang sederhana , entitas yang terlibat dalam subsistem tersebut dapat meliputi orang-orang sebagaimana ditunjukkan pada gambar berikut :

Objek Dasar	Simbol Entitas
Mahasiswa	Mahasiswa
Dosen	Dosen
Wali Mahasiswa/Orangtua	Wali Mahasiswa

Gambar II.8 Entitas

Sumber : (Edhy Sutanta:2011:94)

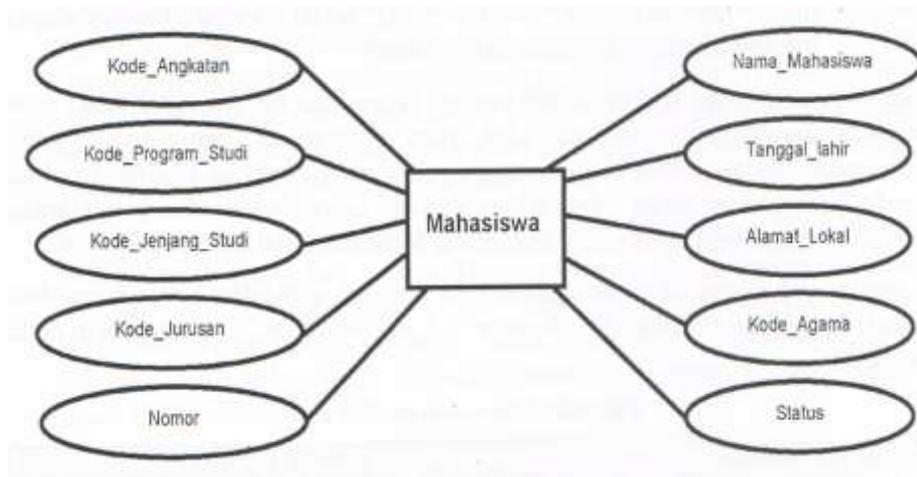
2. Atribut

Atribut sering pula disebut sebagai properti (*property*), merupakan keterangan-keterangan yang terkait pada sebuah entitas yang perlu disimpan dalam basis data. Atribut berfungsi sebagai penjelas pada sebuah entitas.

Untuk menggambarkan atribut digunakan aturan sebagai berikut :

- a. Atribut dinyatakan dengan simbol elips
- b. Nama atribut dituliskan di dalam simbol elips
- c. Nama atribut berupa kata benda tunggal
- d. Nama atribut sedapat mungkin menggunakan nama yang udah dipahami dan dapat menyatakan maknanya dengan jelas.
- e. Atribut dihubungkan dengan entitas yang bersesuaian dengan menggunakan sebuah garis. (Edhy Sutanta:2011:98).

Sebagai contoh penggambaran atribut pada entitas mahasiswa ditunjukkan pada gambar berikut :



Gambar II.9 Atribut

Sumber : (Edhy Sutanta:2011:100)

3. Kerelasian antar entitas / *relationship*

Kerelasian antar entitas mendefinisikan hubungan antara dua buah entitas. Kerelasian adalah kejadian atau transaksi yang terjadi di antara dua buah entitas yang keterangannya perlu disimpan dalam basis data. Kejadian atau transaksi yang tidak perlu disimpan dalam basis data (sekali pun benar-benar terjadi) bukan termasuk kerelasian. Aturan penggambaran kerelasian antar entitas adalah sebagai berikut :

- a. Kerelasian dinyatakan dengan simbol belah ketupat
- b. Nama kerelasian dituliskan didalam simbol belah ketupat
- c. Kerelasian menghubungkan dua entitas
- d. Nama kerelasian berupa kata kerja aktif (diawali dengan awalan me-), tunggal.
- e. Nama kerelasian sedapat mungkin menggunakan nama yang mudah dipahami dan dapat menyatakan maknanya dengan jelas.

Kerelasian antar entitas dikelompokkan dalam tiga jenis yaitu :

a. Kerelasian jenis 1 ke 1 / satu ke satu

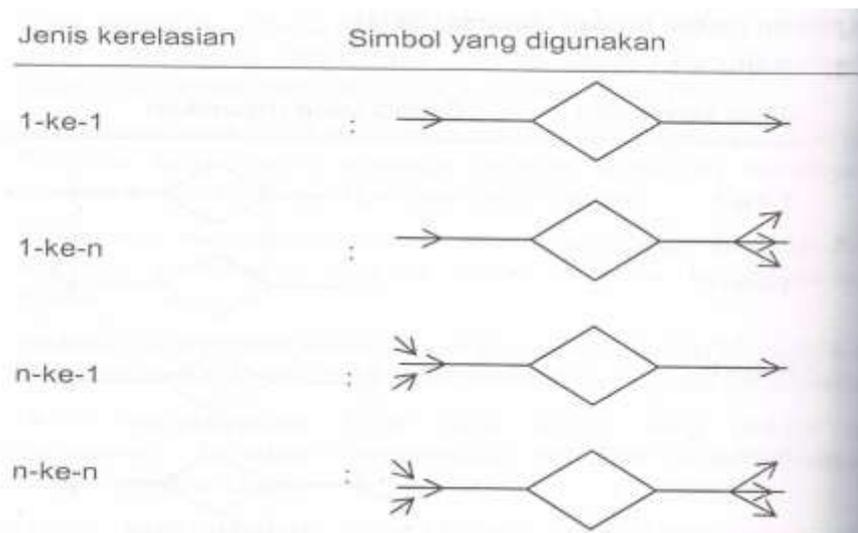
Kerelasian jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan hanya memungkinkan terjadi sebuah kejadian atau transaksi pada kedua entitas.

b. Kerelasian jenis n-ke 1 / banyak ke satu atau 1 ke n / satu ke banyak

Kerelasian jenis ini terjadi jika kejadian atau transaksi dia antara dua entitas yang berhubungan hanya memungkinkan terjadi satu kali dalam entitas pertama dan terjadi lebih dari satu kali kejadian atau transaksi pada entitas kedua.

c. Kerelasian jenis n ke n / banyak ke banyak

Kerelasian jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan memungkinkan terjadi lebih dari satu kali dalam entitas pertama dan entitas kedua. (Edhy Sutanta:2011:101-103).



Gambar II.10 Atribut

Sumber : (Edhy Sutanta:2011:110)