

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Mempelajari suatu sistem akan lebih mengena bila mengetahui terlebih dahulu apakah sistem itu. Pengertian tentang sistem pertama kali dapat diperoleh dari definisi sistem itu sendiri. Jika kita perhatikan dengan seksama, diri kita juga terdiri dari berbagai sistem yang berfungsi untuk mengantar kita kepada tujuan hidup kita. Sudah banyak ahli yang mengungkapkan berbagai sistem bekerja dalam diri kita, misalnya sistem kekebalan tubuh untuk menghadapi penyakit.. (Tata Sutabri : 2012 : 4).

Pendekatan sistem memberikan banyak manfaat dalam memahami lingkungan. Pendekatan sistem berusaha menjelaskan sesuatu yang dipandang dari sudut sistem serta berusaha menemukan struktur unsur yang membentuk sistem tersebut. Dengan memahami struktur sistem dan proses sistem, seseorang akan dapat menjelaskan mengapa tujuan suatu sistem tidak tercapai. Orang yang ahli pada dasarnya selalu mendekati masalah yang dijumpainya berdasarkan pendekatan sistem. (Tata Sutabri : 2012 : 5).

Dari uraian diatas dapat diambil kesimpulan bahwa suatu sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Dari definisi ini dapat dirinci lebih lanjut pengertian sistem secara umu, yaitu sebagai berikut :

1. Setiap sistem terdiri dari berbagai unsur. Sistem pernapasan kita terdiri dari suatu kelompok unsur yaitu hidung, saluran pernapasan, paru-paru dan darah.

Unsur-unsur suatu sistem terdiri dari subsistem yang lebih kecil, yang terdiri pula dari kelompok-kelompok unsur yang membentuk subsistem tersebut.

2. Unsur-unsur tersebut merupakan bagian yang tak terpisahkan dari sistem yang bersangkutan.
3. Unsur-unsur didalam sistem tersebut bekerja sama untuk mencapai tujuan sistem. Setiap sistem mempunyai tujuan tertentu.
4. Suatu sistem merupakan bagian sistem lain yang lebih besar. Sistem pernapasan kita merupakan bagian dari sistem metabolisme tubuh.

Dari uraian diatas maka timbul pertanyaan, “untuk apa suatu sistem diciptakan?”. Suatu sistem dibuat untuk menangani sesuatu yang berulang kali atau yang secara rutin terjadi. Suatu sistem dapat dirumuskan sebagai setiap kumpulan komponen atau subsistem yang dirancang untuk mencapai suatu tujuan. (Tata Sutabri : 2012 : 6).

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 (*Environtment*)

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 produksi 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 Pendukung Keputusan

Keputusan merupakan kegiatan memilih suatu strategi atau tindakan dalam pemecahan masalah tersebut. Tindakan memilih strategi atau aksi yang diyakini manajer akan memberikan solusi terbaik atas sesuatu itu disebut pengambilan keputusan. Tujuan dari keputusan adalah untuk mencapai target atau aksi tertentu yang harus dilakukan. (Kusrini M.Kom :2007 :7).

DSS merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat. Aplikasi DSS menggunakan data, memberikan antarmuka pengguna yang mudah, dan dapat menggabungkan pemikiran pengambil keputusan. DSS lebih ditujukan untuk mendukung manajemen dalam melakukan

pekerjaan yang bersifat analitis dalam situasi yang kurang terstruktur dan dengan kriteria yang kurang jelas. (Kusrini M.Kom :2007 :16).

Sistem pendukung keputusan (*decision support system/DSS*) dapat didefinisikan sebagai suatu sistem informasi yang membantu mengidentifikasi kesempatan membuat keputusan atau menyediakan informasi untuk membantu pembuatan keputusan.

Menurut Turban, sistem pendukung keputusan dapat dibagi menjadi beberapa subsistem :

1. Subsistem Manajemen Data

Subsistem manajemen data memasukkan satu database yang berisi data yang relevan untuk situasi dan dikelola oleh perangkat lunak yang disebut ststem manajemen basis data (DBMS).

2. Subsistem Manajemen Model

Subsistem ini terdiri atas Basis Model (*Model base*), Sistem Manajemen Basis Model (*Model Base Management System*), Bahasa Pemodelan, Direktori Model, Eksekusi Model, Integrasi dan Prosesor Perintah.

3. Subsistem Antarmuka Pengguna

Pengguna berkomunikasi dengan dan memerintahkan sistem pendukung keputusan melalui subsistem ini. Pengguna adalah bagian yang dipertimbangkan dari sistem. Para peneliti menegaskan bahwa beberapa kontribusi unik dari sistem ini berasal dari interaksi yang intensif antara komputer dan pembuat keputusan.

4. Subsistem Manajemen Berbasis Pengetahuan

Subsistem ini dapat mendukung semua subsistem lain atau bertindak sebagai suatu komponen independen. Ia memberikan intelegensi untuk memperbesar pengetahuan si pengambil keputusan. Subsistem ini dapat diinterkoneksi dengan repositori pengetahuan perusahaan, yang kadang-kadang disebut basis pengetahuan organisasional. Pengetahuan dapat disediakan via *server Web*. (Jurnal Winnie Septiani:57).

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, PostgreSQL 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

Database *MySQL* dapat dibuat dengan menggunakan tampilan jendela phpmyadmin atau menggunakan sebuah script PHP. Penulis menyarankan untuk membuat dan menyiapkan database dengan menggunakan jendela phpmyadmin karena akan lebih mudah langkah-langkah penggunaannya. Penyimpanan data yang fleksibel dan cepat aksesnya sangat dibutuhkan dalam sebuah website yang interaktif dan dinamis. Database sendiri berfungsi sebagai penampung data yang anda inputkan melalui form website. Selain itu dapat juga dapat dibalik dengan menampilkan data yang tersimpan dalam database ke dalam halaman website. Jenis database yang sangat 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. (Madcoms : 2011: 260)

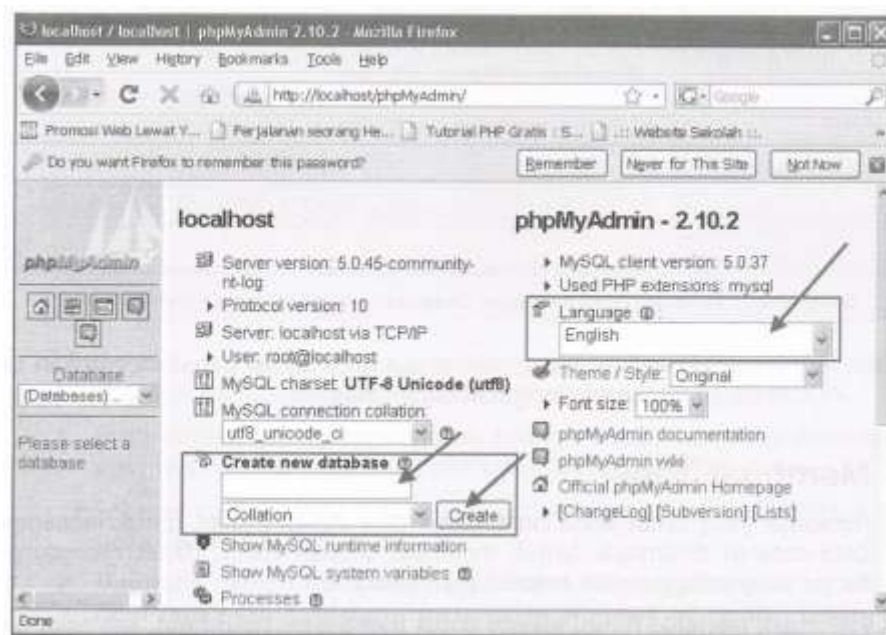
Untuk membuat database baru dengan jendela phpmyadmin ikuti langkah-langkah dibawah ini :

1. Buka jendela phpmyadmin pada jendela browser, masukkan alamat <http://localhost/phpmyadmin>, ikuti dengan menekan tombol enter.
2. Langkah selanjutnya anda diminta untuk memasukkan user dan password agar dapat mengakses database. User dan password yang anda masukkan harus sama dengan user dan password yang anda masukkan saat melakukan

langkah konfigurasi Mysql. User yang sering digunakan adalah root, sedangkan password dapat diisi dengan bebas.

Jika pada saat melakukan konfigurasi Mysql anda tidak mengisi password, maka anda tidak perlu mengisi argumen password saat melakukan koneksi database. Pengosongan password sangat tidak disarankan, karena akan memudahkan file database anda terinfeksi virus dan dirusak oleh spammer atau hacker.

3. Tampilan halaman awal phpmyadmin adalah sebagai berikut :

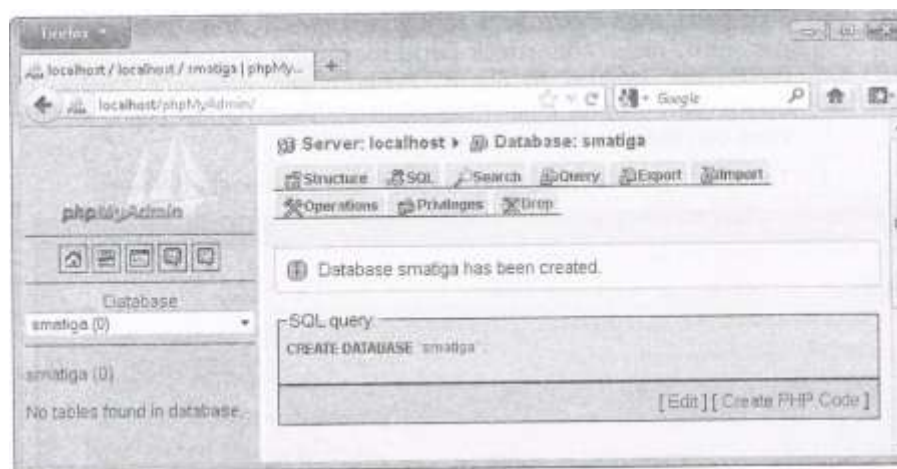


Gambar II.1. Tampilan jendela phpmyadmin

Sumber : (Madcoms : 2011: 263)

4. Pilih bahasa yang digunakan pada bagian language, sebagai contoh tetap menggunakan pilihan english. Ketikkan nama database dalam kotak create new database, sebagai contoh anda ketikkan smatiga, kemudian klik tombol create.

5. Setelah database berhasil dibuat, maka pada kotak database di bagian sebelah kiri dapat anda pilih nama database tersebut untuk mengolahnya. Sebagai contoh membuat tabel-tabel dan pengolahannya.
6. Setelah database terbentuk, maka anda sudah dapat mengisi tabel di dalamnya serta mengolah data-datanya. (Madcoms : 2011: 262-264)



Gambar II.2. Tampilan Hasil Membuat Database Baru Pada Jendela Phpmyadmin

Sumber : (Madcoms : 2011: 264)

II.2.3. Metode *Analitycal Hierarchy Process* (AHP)

AHP merupakan suatu model pendukung keputusan yang dikembangkan oleh Thomas L. Saaty. Model pendukung keputusan ini akan menguraikan masalah multi faktor atau multi kriteria yang kompleks menjadi suatu hirarki, menurut Saaty (1993), hirarki didefinisikan sebagai suatu representasi dari sebuah permasalahan yang kompleks dalam suatu struktur multi level dimana level pertama adalah tujuan, yang diikuti level faktor, kriteria, sub kriteria, dan seterusnya ke bawah hingga level terakhir dari alternatif. Dengan hirarki, suatu

masalah yang kompleks dapat diuraikan ke dalam kelompok-kelompoknya yang kemudian diatur menjadi suatu bentuk hirarki sehingga permasalahan akan tampak lebih terstruktur dan sistematis. AHP sering digunakan sebagai metode pemecahan masalah dibanding dengan metode yang lain karena alasan-alasan sebagai berikut :

1. Struktur yang berhirarki, sebagai konsekuensi dari kriteria yang dipilih, sampai pada subkriteria yang paling dalam.
2. Memperhitungkan validitas sampai dengan batas toleransi inkonsistensi berbagai kriteria dan alternatif yang dipilih oleh pengambil keputusan.
3. Memperhitungkan daya tahan *output* analisis sensitivitas pengambilan keputusan. (Syaifullah:2010:1)

Dalam menyelesaikan permasalahan dengan AHP ada beberapa prinsip yang harus dipahami, diantaranya adalah :

1. Membuat hierarki

Sistem yang kompleks bisa dipahami dengan memecahnya menjadi elemen-elemen pendukung, menyusun elemen secara hierarki dan menggabungkannya atau mensistesisnya.

2. Penilaian kriteria dan alternatif

Kriteria dan alternatif dilakukan dengan perbandingan berpasangan.

3. *Synthesis of priority*

Untuk setiap kriteria dan alternatif perlu dilakukan perbandingan berpasangan (*pairwise comparisons*). Nilai-nilai perbandingan relatif dari seluruh alternatif

kriteria bisa disesuaikan dengan judgement yang telah ditentukan untuk menghasilkan bobot dan prioritas.

4. *Logical consistency* (Konsistensi logis)

Konsistensi memiliki dua makna. Pertama, objek-objek yang serupa bisa dikelompokkan sesuai dengan keseragaman dan relevansi. Kedua, menyangkut tingkat hubungan antar objek yang didasarkan pada kriteria tertentu.

Pada dasarnya prosedur atau langkah-langkah dalam metode AHP meliputi :

1. Mendefinisikan masalah dan menentukan solusi yang diinginkan, lalu menyusun hierarki dari permasalahan yang dihadapi.
2. Menentukan prioritas elemen
3. Sintesis

Pertimbangan-pertimbangan terhadap perbandingan berpasangan disintesis untuk memperoleh keseluruhan prioritas.

4. Mengukur konsistensi
5. Hitung *Consistency Index I* (CI) dengan rumus :

$$CI = (\lambda_{maks} - n) / n$$

Dimana n = banyaknya elemen

6. Hitung rasio konsistensi / *consistency ratio* (CR) dengan rumus :

$$CR = CI / RC$$

Dimana CR = Consistency Ratio

CI = Consistency Index

IR = Indeks Random Consistency

7. Memeriksa konsistensi hierarki. Jika nilainya lebih dari 10%, maka penilaian data *judgement* harus diperbaiki. Namun jika rasio konsistensi (CI/IR) kurang atau sama dengan 0,1, maka hasil perhitungan bisa dinyatakan benar. (Kusrini,M.Kom:2007:133-136).

Daftar indeks random konsistensi (IR) dapat dilihat pada tabel II.1 :

Tabel II.1 Daftar indeks random konsistensi

Ukuran Matriks	Nilai IR
1,2	0,00
3	0,58
4	0,90
5	1,12
6	1,24
7	1,32
8	1,41
9	1,45
10	1,49
11	1,51
12	1,48
13	1,56
14	1,57
15	1,59

Sumber : (Kusrini,M.Kom:2007:136)

II.3. Alat Bantu Pengembangan Sistem

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

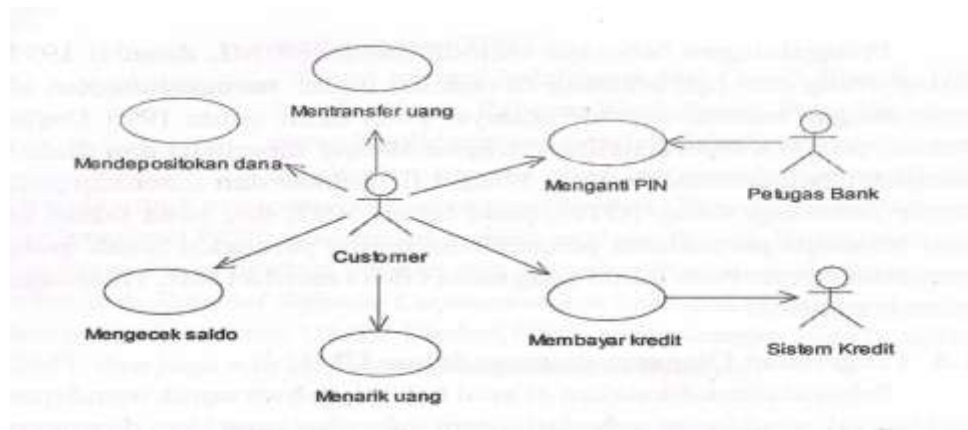
II.3.1 UML (Unified Modelling Language)

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).

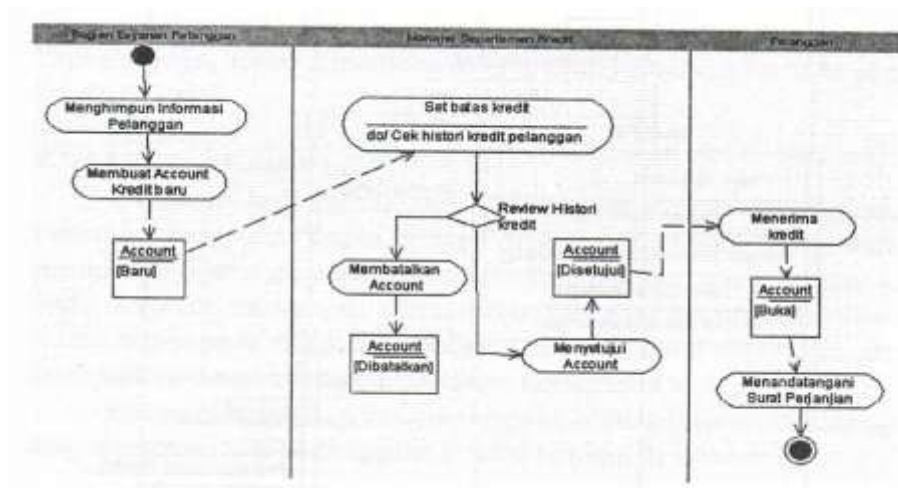


Gambar II.3. Use Case Diagram

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*.

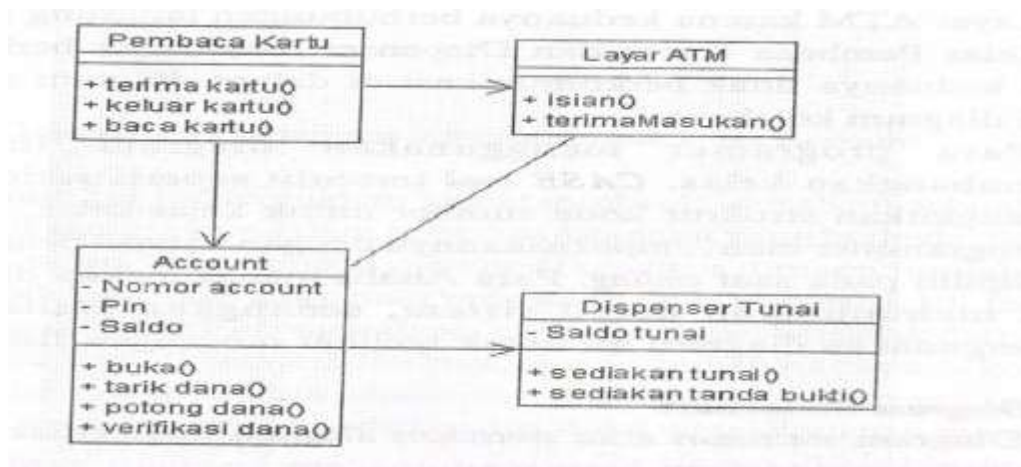


Gambar II.4 Contoh Activity Diagram

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).

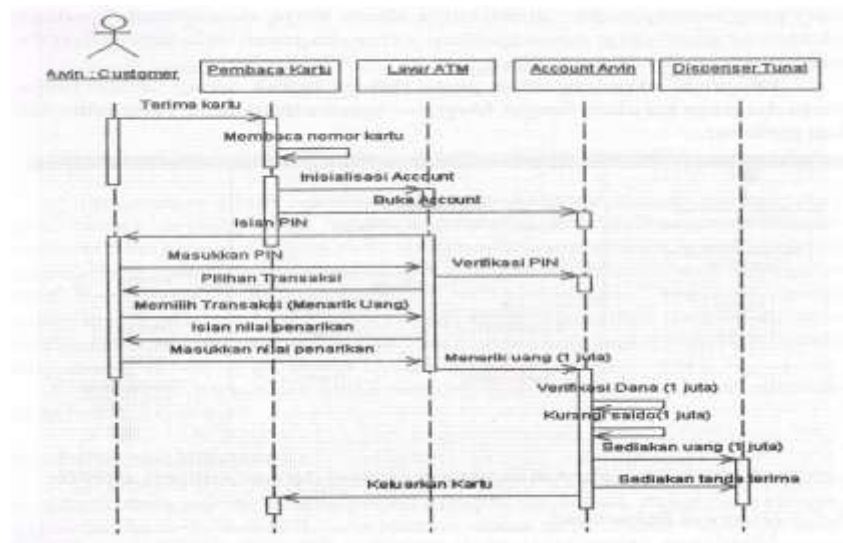


Gambar II.5 Contoh Class Diagram

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.6 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 didukung 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). Secara berturut-turut masing-masing level normal tersebut dibahas berikut ini, dimulai dari bentuk tidak normal :

1. Relasi bentuk tidak normal (*un normalized form/UNF*)

Relasi-relasi yang dirancang tanpa mengindahkan batasan dalam definisi basis data dan karakteristik RDBM menghasilkan relasi UNF. Bentuk ini harus dihindari dalam perancangan relasi dalam basis data.

2. Relasi bentuk normal pertama (*first norm form/1NF*)

Relasi disebut sebagai 1NF jika memenuhi kriteria sebagai berikut :

- a. Jika seluruh atribut dalam relasi bernilai atomik (*atomic value*)
- b. Jika seluruh atribut dalam relasi bernilai tunggal (*single value*)
- c. Jika relasi tidak memuat set atribut berulang
- d. Jika semua record mempunyai sejumlah atribut yang sama

3. Bentuk normal kedua (*second norm form/2NF*)

Relasi disebut sebagai 2NF jika memenuhi kriteria sebagai berikut :

- a. Jika memenuhi kriteria 1NF
- b. Jika semua atribut nonkunci FD pada PK

4. Bentuk normal ketiga (*third norm form/3NF*)

Suatu relasi disebut 3NF jika memenuhi kriteria sebagai berikut :

- a. Jika memenuhi kriteria 2NF
- b. Jika setiap atribut nonkunci tidak TDF (*non transitive dependency*) terhadap PK.

5. Bentuk normal *Boyce-Cood* (*Boyce-Cood norm form/BCNF*)

Bentuk normal BCNF dikemukakan oleh R.F.Boyce dan E.F.Codd. Suatu relasi disebut sebagai BCNF jika memenuhi kriteria sebagai berikut :

- a. Jika memenuhi kriteria 3NF
- b. Jika semua atribut penentu (determinan) merupakan CK.

6. Bentuk normal keempat (*forth norm form/4NF*)

Relasi disebut sebagai 4NF jika memenuhi kriteria sebagai berikut :

- a. Jika memenuhi kriteria BCNF
- b. Jika setiap atribut di dalamnya tidak mengalami ketergantungan pada banyak nilai. Atau dengan kalimat lain, bahwa semua atribut yang

mengalami ketergantungan pada banyak nilai adalah bergantung secara fungsional.

7. Bentuk normal kelima (*fifth norm form/5NF*)

Suatu relasi memenuhi kriteria 5NF jika kerelasian antardata dalam relasi tersebut tidak dapat direkonstruksi dari struktur relasi yang sederhana.

8. Bentuk normal kunci domain (*domain key norm form/DKNF*)

Suatu relasi disebut sebagai DKNF jika setiap batasan dapat disimpulkan secara sederhana dengan mengetahui sekumpulan nama atribut dan domainnya selama menggunakan sekumpulan atribut pada kuncinya. (Edhy Sutanta:2011:176-179)..

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 redudansi, 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 Muhammad Arhami,M.Kom:2010:7)

Tabel II.2 Simbol-simbol dalam kamus data

Simbol	Uraian
=	Terdiri dari,mendefinisikan,diuraikan menjadi
+	Dan
()	Menunjukkan elemen yang bersifat pilihan (Optional). Elemen-elemen yang bersifat pilihan ini bisa dikosongkan pada layar masukan atau bisa juga dengan memuat spasi atau nol untuk field-field numeric pada

	struktur file.
{ }	Menunjukkan elemen-elemen repetitive juga disebut kelompok berulang atau tabel-tabel. Kemungkinan bisa ada satu atau beberapa elemen berulang didalam kelompok tersebut
[]	Menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada. Tetapi tidak bisa kedua-duanya ada secara bersamaan. Elemen-elemen yang ada didalam tanda kurung ini saling terpisah satu sama lain. (Dengan kata lain, memilih salah satu dari sejumlah alternatif, seleksi)
**	Komentar Contoh : *Seminar yang akan diikuti*
	Pemisah sejumlah alternatif pilihan antara simbol []
@	Identifikasi atribut kunci

Sumber : (Modul Muhammad Arhami:2010:12)

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

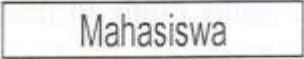
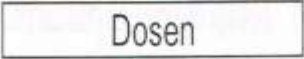
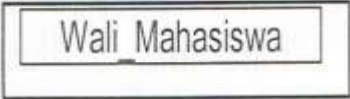
diagram ERD tersusun atas tiga komponen yaitu entitas, atribut dan kerelasiaan antar entitas. Secara garis besar, entitas merupakan objek dasara yang terlibat dalam sistem. Atribut berperan sebagai penjelas entitas, sedangkan kerelasiaan menunjukkan hubungan yang terjadi di antara dua entitas. (Edhy 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	
Dosen	
Wali Mahasiswa/Orangtua	

Gambar II.7 Entitas

Sumber : (Edhy Sutanta:2011:94)

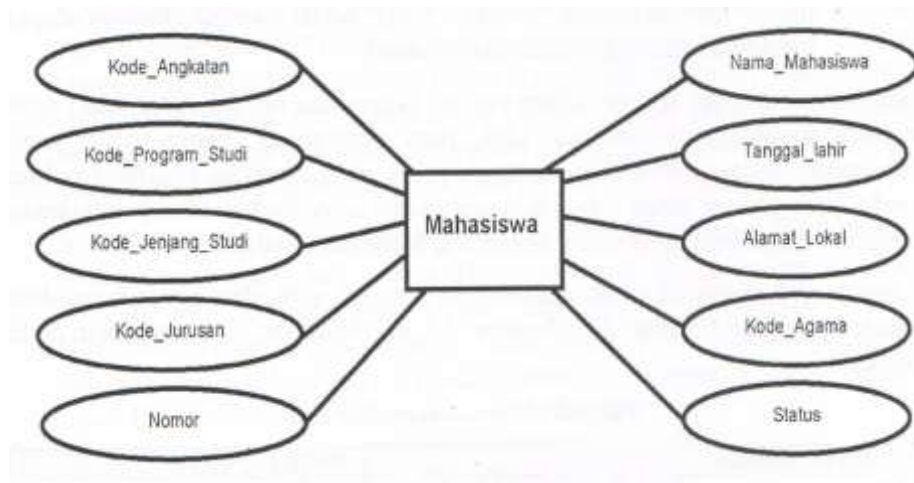
2. Atribut

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 :

- Atribut dinyatakan dengan simbol elips
- Nama atribut dituliskan di dalam simbol elips
- Nama atribut berupa kata benda tunggal
- Nama atribut sedapat mungkin menggunakan nama yang udah dipahami dan dapat menyatakan maknanya dengan jelas.
- 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.8 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

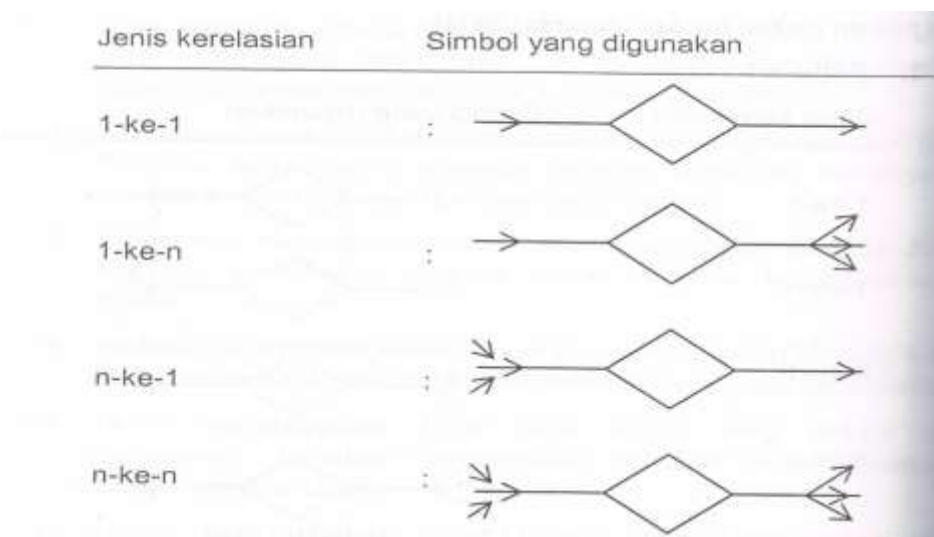
Kerelasian jenis ini terjadi jika kejadian atau transaksi di antara dua entitas yang berhubungan hanya memungkinkan terjadi sebuah kejadian atau transaksi pada kedua entitas.

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

Kerelasian jenis ini terjadi jika kejadian atau transaksi di 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.9 Atribut

Sumber : (Edhy Sutanta:2011:110)