BAB II
LANDASAN TEORI

2.1. Konsep Dasar Sistem

Terdapat 2 kelompok pendekatan di dalam mendefinisikan sistem yaitu pendekatan yang menekankan kepada prosedur dan pendekatan yang menekankan pada elemennya. Pendekatan sistem lebih menekankan pada prosedurnya dan mendefinisikan sistem sebagai berikut “bertujuan untuk mendeskripsikan batasan dan ruang lingkup aplikasi basis data serta sudut pandang user yang utama” Indrajani (2011:49).

2.1.1. Pengertian Sistem

Dalam kehidupan sehari-hari orang sering menyamakan makna istilah sistem dengan cara. Istilah sistem dari bahasa yunani yaitu sistem yang berarti penempatan atau mengatur.

Pengertian sistem menurut Mulyadi (2008:2) menjelaskan bahwa “Sistem adalah sekelompok unsur yang erat berhubungan satu dengan yang lainnya dan berfungsi bersama untuk mencapai suatu tujuan tertentu”.

Pengertian sistem Mulyanto (2009:1) secara umum adalah “Kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu sebagai satu kesatuan”.

Menurut Rochaety (2011:2) “Mendefinisikan sistem sebagai suatu kesatuan yang terdiri dari dua atau lebih komponen atau subsistem yang berinteraksi untuk mencapai suatu tujuan”.

Dari beberapa pendapat maka dapat diambil kesimpulan bahwa sistem pada dasarnya merupakan sekelompok komponen-komponen atau unsur-unsur yang saling berhubungan satu dengan yang lainnya dan berfungsi bersama-sama untuk mencapai tujuan tertentu.

2.1.2. Karakteristik Sistem

Menurut Sutabri (2012:13) Model umum sebuah sistem terdiri dari input, proses dan output. Hal ini merupakan konsep sebuah sistem dapat mempunyai konsep sebuah sistem yang sangat sederhana mengikuti sebuah sistem dapat mempunyai beberapa masukan dan keluaran sekaligus sekaligus. Selain itu sebuah sistem juga memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Agar system dapat dikatakan system yang baik maka system tersebut memiliki karakteristik, yaitu :

1. Mempunyai Komponen *(components)*

Komponen sistem adalah segala sesuatu yang menjadi bagian penyusun sistem. Komponen dapat berupa benda nyata ataupun abstrak. Komponen sistem disebut juga sebagai sub sistem, dapat berupa orang, benda, hal atau kejadian yang terlibat dalam sistem.

2. Mempunyai batas *(boundary)*

Batas sistem diperlukan untuk membedakan satu sistem dengan sistem yang lain. Tanpa adanya batas sistem, maka sangat sulit untuk menjelaskan suatu sistem. Batas sistem akan memberikan *scope* tinjauan terhadap sistem.

3. Mempunyai lingkungan *(environments)*

Lingkungan sistem adalah segala sesuatu yang berada di luar sistem. Lingkungan sistem dapat menguntungkan ataupun merugikan. Umumnya, lingkungan yang menguntungkan akan selalu dipertahankan untuk menjaga keberlangsungan sistem. Sedangkan lingkungan sistem yang merugikan akan diupayakan agar mempunyai pengaruh seminimal mungkin, bahkan jika mungkin ditiadakan

4. Mempunyai penghubung/antar muka *(interface)* antar komponen

Penghubung/antar muka *(interface)*

Merupakan komponen sistem, yaitu segala sesuatu yang bertugas menjembatani hubungan antar komponen dalam sistem. Penghubung/antar muka merupakan sarana yang memungkinkan setiap komponen saling berinteraksi dan berkomunikasi dalam rangka menjalankan fungsi masing-masing komponen.

5. Mempunyai masukan *(input)*

Masukan merupakan komponen sistem, yaitu segala sesuatu yang perlu dimasukkan kedalam sistem sebagai bahan yang akan diolah lebih lanjut untuk menghasilkan keluaran yang berguna.

6. Mempunyai pengolahan *(processing)*

Pengolahan merupakan komponen sistem yang mempunyai peran utama mengolah masukan agar menghasilkan keluaran yang berguna bagi para pemakainya. Pengolahan dapat berupa program aplikasi komputer yang kemudian dikembangkan untuk keperluan khusus.Dimana program aplikasi tersebut mampu menerima

masukan, mengolah masukan, dan menampilkan hasil olahan sesuai dengan kebutuhan para pemakai.

7. Mempunyai keluaran *(output)*

Keluaran merupakan komponen sistem yang berupa berbagai macam bentuk keluaran yang dihasilkan oleh komponen pengolahan.

8. Mempunyai sasaran *(objectives)* dan tujuan *(goal)*

Setiap komponen dalam sistem perlu dijaga agar saling bekerja sama dengan harapan agar mampu mencapai sasaran dan tujuan sistem. Sasaran berbeda dengan tujuan. Sasaran sistem adalah apa yang ingin dicapai oleh sistem untuk jangka waktu yang relatif pendek. Sedangkan tujuan merupakan kondisi/ hasil akhir yang ingin dicapai oleh sistem untuk jangka waktu yang panjang. Dalam hal ini, tahapan merupakan hasil pada setiap tahapan tertentu yang mendukung upaya pencapaian tujuan.

9. Mempunyai Kendali *(control)*

Bagian kendali mempunyai peranan utama dalam menjaga atas proses dalam sistem dapat berlangsung secara normal sesuai dengan batasan yang telah ditetapkan sebelumnya. Kendali dapat berupa validasi masukan, validasi proses, maupun validasi keluaran yang dapat dirancang dandikembangkan secara terprogram.

10. Mempunyai umpan balik *(feed back)*

Umpan balik diperlukan oleh bagian kendali *(control)* sistem untuk mengecek terjadinya penyimpangan proses dalam sistem dan mengembalikannya ke dalam kondisi normal.

2.1.3. Klasifikasi Sistem

Menurut Mulyanto (2009:8) mengemukakan bahwa ”Sistem dapat diklasifikasikan dari beberapa sudut pandang di antaranya sebagai berikut” :

1. Sistem Abstrak *(Abstract System)* dan Sistem Fisik *(Physical System)*

   Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologi yaitu sistem yang berupa pemikiran hubungan antara manusia dengan Tuhan Y.M.E, sedangkan sistem fisik merupakan sistem yang ada secara fisik, contohnya sistem produksi dan sistem komputer.

2. Sistem Alamiah *(Natural System)* dan Sistem Buatan Manusia *(Human Made System)*

   Sistem Alamiah adalah sistem yang terjadi melalui proses alami, tidak dibuat oleh manusia. Contohnya sistem perputaran bumi dan pergantian musim. Sedangkan sistem pembuatan manusia adalah sistem yang di rancang dan dibuat oleh manusia. Contohnya sistem komputerisasi penjualan tunai atau kredit.

3. Sistem Tertentu *(Deterministic System)* Dan Sistem Tak Tertentu *(Probabilistic System)*

   Sistem yang operasinya dapat di prediksi terlebih dahulu. Contohnya sistem komputer yang cara kerjanya dapat dipastikan berdasarkan program-program yang dijalankan. Sedangkan sistem tak tertentu adalah sistem yang kondisi

masa depannya tidak dapat di prediksikan karena mengandung unsur probabilistic. Contohnya ramalan cuaca.

4. Sistem Terbuka *(Open System)* Dan Sistem Tertutup *(Closed System)*

   Sistem terbuka adalah sistem yang berhubungan dan dapat dipengaruhi oleh lingkungan luar. Contohnya sistem administrasi kepegawaian, sedangkan sistem tertutup adalah sistem yang tidak berhubungan dan tidak dapat dipengaruhi dengan lingkungan luar. Contohnya sistem perakitan *assembling* sistem.

2.1.4. Definisi informasi

Menurut Gordon B. Davis dalam Hartono (2013:15), "*information is data has been processed into a form that is meaningful to the recipient and is of real or perceived value in current or prospective decision*". ("Informasi adalah data yang telah diolah menjadi suatu bentuk yang berguna bagi penerimanya dan memiliki nilai bagi penggambilan keputusan saat ini atau dimasa yang akan datang").

Menurut Mulyanto (2009 : 12) dalam bukunya yang berjudul *Sistem Informasi Konsep dan Aplikasi* adalah "Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya, sedangkan data merupakan sumber informasi yang menggambarkan suatu kejadian yang nyata".

2.1.5. Definisi sistem informasi

Informasi merupakan hal yang sangat penting bagi manajemen dalam pengambilan keputusan. Informasi diperoleh dari sistem informasi (*information systems*) atau disebut juga dengan *processing system* atau *information processing*

system atau *information generating system*. Sistem informasi adalah suatu sistem didalam 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.

Menurut Laudon (2010:16) “Sistem informasi adalah komponen-komponen yang saling berkaitan yang bekerja bersama-sama untuk mengumpulkan, mengolah, menyimpan dan menampilkan informasi untuk mendukung pengambilan keputusan, pengaturan, analisa, dan visualisasi pada sebuah organisasi”.

Menurut Alter dalam Mulyanto (2009:29),dalam bukunya yang berjudul *Sistem Informasi Konsep dan Aplikasi*, sistem informasi adalah “Kombinasi antar prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi”.

Menurut Bodnar dan Hopwood dalam Mulyanto (2009:29), sistem informasi adalah “Kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna”.

2.1.6. Komponen Sistem Informasi

Menurut Yakub (2012:20) sistem informasi merupakan sebuah susunan yang terdiri beberapa komponen atau elemen. Komponen sistem informasi disebut dengan istilah blok *bangunan (building block)*. Komponen sistem intermasi tersebut terdari dari blok masukan *(input block)*, blok model (*model block*), blok keluaran *(output block)*, blok teknologi *(technology block)*, dan basis data *(database block)*.

a. Blok masukan (*input block*), input memiliki data yang masuk kedalam sistem informasi, juga metode-metode untuk menangkap data yang dimasukan.
b. Blok model (*model block*), blok ini terdiri dari kombinasi prosedur logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan dibasis data.
c. Blok keluaran (*output block*), produk dari sistem informasi adalah keluaran yang merupakan informasi berkualitas dan dokumentasi yang berguina untuk semua tingkatan manajemen serta semua pemakai sistem.
d. Blok teknologi (*technologi block*), blok teknologi digunakan untuk menerima input, menyimpan, mengakses data, menghasilkan dan mengirim keluaran dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu; teknisi(*brainware*), perangkat lunak (*software*) dan perangkat keras*(hardware)*.
e. Basis data *(database block)*, basis data merupakan kumpulan data yang saling berhubungan satu dengan lainnya, tersimpan diperangkat keras komputer dan digunakan perangkat lunak *(software)* untuk memanipulasinya.

2.1.7. Basis data (*database*)

Database adalah kumpulan data yang terdiri dari satu atau lebih tabel yang terintegritas satu sama lain, dimana pengguna (*user*) diberi wewenang untuk dapat mengakses (mengubah, menghapus, menganalisis, menambah, memperbaiki) data dalam tabel-tabel tersebut. Tabel terdiri dari baris yang disebut dengan *record* dan kolom yang disebut *field*.

Menurut Simarmata & Iman Paryudi (2010:1) mengutip dari Silberschatz, dkk mengemukakan bahwa :“Basis data sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan. Sistem manajemen basisdata (DBMS) adalah kumpulan data yang saling berhubungan dan kumpulan program untuk mengakses data. Tujuan utama sistem manajemen basisdata adalah menyediakan cara menyimpan dan mengambil informasi basisdata secara mudah dan efisien”.

2.1.8. Model pengembangan perangkat lunak

Menurut jogiyanto (2010:59) Metode pengembangan *waterfall* (*Sequential Linier*) adalah sebuah metode pengembangan *software* yang bersifat sequential. Metode ini dikenalkan oleh Royce pada tahun 1970 dan pada saat itu disebut sebagai siklus klasik dan sekarang ini lebih dikenal dengan *sequential linier.* Selain itu model merupakan model yang paling banyak dipakai oleh para pengembang *software*. Inti dari metode *waterfall* adalah pengerjaan dari suatu system dilakukan secara berurutan atau secara *linier*.

Model SDLC air terjun *(waterfall)* sering juga disebut model sekuensial linier *(sequential linear)* atau alur hidup klasik *(classic life cycle).* Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung *(support).*

Menurut Rosa A.S dan M. Shalahudin (2013:29) dalam buku rekayasa perangkat lunak terstruktur dan berorientasi objek, menyatakan bahwa “Model *waterfall* adalah model SDLC yang saling sederhana. Model ini hanya cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah”.

Rekayasa perangkat lunak *(Software Enginering)* merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomis yang dipercaya dan bekerja secara efisien menggunakan mesin. Perangkat lunak banyak dibuat dan pada akhirnya sering tidak digunakan karena tidak memenuhi kebutuhan pelanggan atau bahkan karena masalah non-teknis seperti keengganan pemakai perangkat lunak *(user)* untuk mengubah cara kerja dari manual keotomatis atau ketidakmampuan *user* menggunakan komputer.

Model SDLC air terjun *(Waterfall)* sering juga disebut model skuensial linier *(Sequential Linier)* atau alur hidup klasik *(Classic Life Cycle)*. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara *skuensial* atau terurut dimulai dari analisis desain, pengkodean, pengujian, dan tahap pendukung *(Support)*.

Kelebihan yang dimiliki model waterfall menurut shalahudin (2013:27), sebagai berikut:

1. Lebih baik dari pada “semua tangan di *keyboard*, habiskan separuh biaya garsware terbaru, tanpa perencanaan”.
2. Kesalahan interprestasi dapat diketahui lebih awal.
3. Berbagai kesalahan dapat diketahui lebih awal, seperti: pengeluaran yang berlebih, dan kelayakan yang rendah.

Tahapan-tahapan dalam model *Waterfall* shalahudin (2013:29), diantaranya:

1. Analisis Kebutuhan Perangkat Lunak

   Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasi kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh pengguna.

2. Desain

Desain perangkat lunak adalah proses *multi* langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentraslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan representasi desain agar dapat diimplementasika menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentaikan.

3. Pembuatan Kode Program

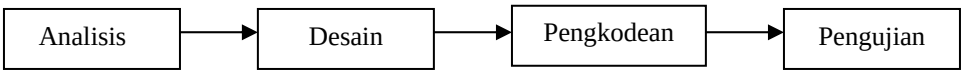
Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan *(eror)* dan memastikan keluaran yang dihasilkan sesuai yang diinginkan.

5. Pendukung *(Support)* atau Pemeliharan *(Maintance)*

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.



Sumber: Shalahudin (2013:29)

Gambar II.1

Ilustrasi model *waterfall*

2.2. Teori pendukung

Teori pendukung merupakan alat yang digunakan untuk menggambar logika model dari suatu sistem dengan menggunakan simbol-simbol, lambang-lambang, ataupun diagram-diagram yang menunjukkan secara tepat arti dan fungsinya. Fungsi dari peralatan pendukung *(Tool System)* adalah untuk menjelaskan kepada user bagaimana fungsi dari sistem informasi dapat bekerja dengan suatu bentuk *logika model* dan *phsycal model.*

2.2.1. Diagram Alir Data (*Data Flow Diagram*)

1. Konsep Dasar

Menurut Kendall dan kendall (2015:263) Diagram Alir Data (DAD) menggambarkan, “Pandangan sejauh mungkin mengenai masukan, proses, dan keluaran sistem, yang berhubungan dengan masukan, proses, dan keluaran dari Tahapan Pembuatan Diagram Alir Data (DAD).

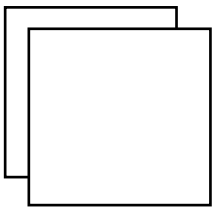
2. Simbol Data Flow Diagram

Menurut Kendall dan kendall (2015:263) simbol data flow diagram adalah sebagai berikut :

a. Kesatuan Luar *(External Entity)*

Digunakan untuk menggambarkan kesatuan diluar lingkungan system dapat berupa orang, organisasi atau system lainnya.

Simbol :



Sumber: Kendall dan kendall (2015:263)

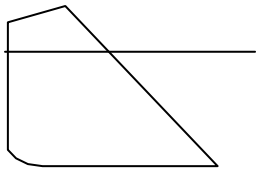
Gambar II.2.

Gambar External Entity

b. Proses *(Process)*

Digunakan untuk menggambarkan kegiatan yang dilakukan oleh orang, mesin atau komputer.

Simbol :



Sumber: Kendall dan kendall (2015:263)

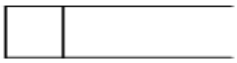
Gambar II.3.

Gambar Proses

c. Data *Source*

Digunakan untuk menggambarkan media penyimpanan data berupa file atau database, arsip, table, agenda atau buku.

Simbol :



Sumber: Kendall dan kendall (2015:263)

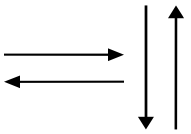
Gambar II.4.

Gambar *Data Source*

d. Arus Data *(Data Flow)*

Digunakan untuk menggambarkan arus data yang dapat berupa Masukan untuk sistem atau hasil dari proses sistem.

Simbol :



Sumber: Kendall dan kendall (2015:263)

Gambar II.5.

Gambar *Data Flow*

3. Tahapan pembuatan diagram alir data dibagi menjadi tiga tingkat konstriksi diagram alir data, yaitu:

a. Diagram Konteks

Tingkatan tertinggi dalam diagram alir data dan hanya memuat satu proses, menunjukan sistem secara keseluruhan, diberi nomor nol dan tidak memuat penyimpanan data.

b. Diagram Nol

Menggambarkan detail diagram konteks, masukan dan keluaran yang ditetapkan dalam konteks tetap konstan dalam semua diagram sub-urutannya dan sudah menunjukan bentuk penyimpanan.

c. Diagram Detail

Dikembangkan untuk menciptakan diagram anak yang lebih mendetail. Proses pada diagram nol yang dikembangkan disebut Proses Induk *(parent process)* dan diagram yang dihasilkan disebut *child diagram.* Apabila proses tersebut tidak dapat dikembangkan maka disebut *process primitive.*

4. Aturan Main Diagram Alir Data (DAD)

Dalam menggunakan diagram alir data ada beberapa aturan main yang harus diperhatikan dalam membuat sistem, yaitu sebagai berikut:

a. Didalam diagram alir data tidak boleh menghubungkan antara eksternal entity dengan eksternal entity lainnya secara langsung.
b. Didalam diagram alir data tidak boleh menghubungkan antara data store dengan data store lainnya secara langsung.

c. Didalam diagram alir data tidak boleh menghubungkan antara data store dengan eksternal entity secara langsung.

d. Setiap proses harus ada data flow yang masuk dan data flow yang keluar.

2.2.2. Kamus Data *(Data dictionary)*

Kamus data (*Data dictionary*) menurut Kendall & Kendall (2013:333) adalah "suatu aplikasi khusus dari jenis kamus-kamus yang digunakan sebagai refrensi kehidupan setiap hari. Kamus data sebagai suatu dokumen, kamus data mengumpulkan dan mengkoordinasi istilah-istilah data tertentu, dan menjelaskan apa arti setiap istilah yang ada".

Kamus data menurut Kristanto (2008:72) yaitu, "Kumpulan elemen – elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap field atau file di dalam sistem."

Bisa dikatakan kamus data merupakan tempat penyimpanan semua struktur dan elemen data yang ada pada sistem. Juga sebagai katalog untuk mengetahui detail data seperti sumber dan tujuan data, deskripsi, bentuk, dan struktur dari data. Kamus data dapat digunakan sebagai alat komunikasi antara analisis sistem tentang data yang mengalir pada sistem.

1. Nama Arus Data

   Karena kamus data dibuat berdasarkan arus data yang mengalir di DAD, maka nama arus data juga harus dicatat dikamus data, sehingga mereka yang membaca DAD akan memerlukan penjelasan lebih lanjut tentang

suatu arus data tertentu di DAD dapat langsung mencairkannya dengan mudah di kamus data.

2. Alias

   Alias perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan yang lainnya.

3. Bentuk Data

   Bentuk dari data yang mengalir dapat berupa dokumen dasrar atau folmulir, laporan tercetak, tampilan dilayar monitor, variable, parameter, dan *field.*

4. Arus Data

   Arus data menunjukan dari mana data mengalir dan kemana data akan menuju. Keterangan arus data ini perlu dicatat dikamus data supaya memudahkan mencari arus data didalam Diagram Alir Data (DAD).

5. Penjelasan

   Bagian penjelasan ini dapat diisi dengan keterangan – keterangan tentang arus data yang terjadi pada kamus data.

6. Periode

   Periode ini menunjukan kapan terjadinya arus data umum. Periode ini dicatat dikamus data karena dapat digunakan untuk mendefinisikan kapan input data harus dimasukan kedalam sistem, kapan proses program harus dilakukan dan kapan laporan – lapoan harus dihasilkan.

7. Volume

   Volume yang perlu dicatat dalam kamus data adalah tentang volume rata–rata dan volume puncak darai kamus data. Volume rata–rata menunjukan banyaknya arus data yang mengalir dalam satu periode tertentu, sedangkan volume puncak menunjukan volume yang terbanyak.

8. Struktur Data

Struktur yang menunjukan arus data yang dicatat pada kamus data yang terdiri dari item – item atau elemen – elemen data.

Kamus data juga mempunyai suatu bentuk untuk mempersingkat arti atau makna dari simbol-simbol yang dijalankan disebut dengan notasi.

Notasi atau simbol yang digunakan dibagi menjadi 2 (dua) macam yaitu :

1. Notasi Tipe Data

Notasi tipe data digunakan untuk membuat spesifikasi format *input* maupun *output* suatu data.

Tabel II.1 Notasi Tipe Data

Notasi	**Keterangan**
X	Setiap karakter
9	Angka numerik
A	Karakter Alfabet
Z	Angka nol tampilkan sebagai spasi kosong
.	Titik sebagai penulisan ribuan
,	Koma sebagai pemecahan pecahan
-	*Hypen* sebagai tanda penghubung
/	*Slash* sebagai tanda pembagi

Sumber : Kristanto (2008:72)

2. Notasi Struktur Data

Notasi diguanakan untuk membuat spesifikasi elemen data sebagai berikut :

Tabel II.2 Notasi Struktur Data

Notasi	Keterangan
=	Terdiri dari
+	And (dan)
( )	Pilihan (Ya atau Tidak)
{ }	Iterasi / pengulangan proses
[ ]	Pilih salah satu pilihan , pemisah pilihan didalam tanda [ ]

Sumber : Kristanto (2008:72)

2.2.3. Key

Key adalah satu gabungan dari beberapa atribut yang dapat membedakan semua basis data (row) dalam secara unik. Key didalam dataField Key atau kunci yang dapat digunakan sebagai berikut :

1. Kunci Kandidat *(Candidate Key)*

   Suatu atribut atau satu set minimal atribut mengidentifikasikan secara unik suatu kejadian yang spesifik dari suatu *entity.*

2. Kunci Primer *(Primary Key)*

   Suatu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasikan secara unik suatu kejadian yang secara spesifik, akan

tetapi juga dapat mewakili setiap kejadian dari suatu *entity*. Setiap kunci kandidat punya peluang menjadi *primary key*, akan tetapi sebaiknya dipilih satu saja yang dapat mewakili entitas yang ada secara menyeluruh.

3. Kunci Alternatif *(Alternate Key)*

Kunci kandidat yang tidak dipakai sebagai *primary key*. Kerapkali kunci alternatif ini dipakai sebagai kunci pengurutan dalam pembuatan laporan.

4. Kunci Tamu *(Foreign Key)*

Satu atribut atau set minimal atribut yagn melengkapi satu hubungan yang menunjukkan keinduknya.

2.2.3. Enterprise Relationship Diagram (ERD)

1. *Enterprise Relationship Diagram* (ERD)

Menurut Sukamto dan Salahuddin (2013:53) “ERD adalah bentuk paling awal dalam melakukan perancangan basis data rasional. Jika menggunakan OODBMS maka perancangan ERD tidak perlu dilakukan”. ERD juga merupakan gambaran yang menghubungkan antara objek satu dengan objek yang lainnya dalam dunia nyata. Bisa dikatakan b ahwa bahan yang akan digunakan untuk membuat ERD adalah dari objek dunia nyata.

a. Komponen ERD

Komponen dalam Entity Relationship Diagram adalah sebagai berikut:

1.) Entitas

Adalah segala sesuatu yang dapat digambarkan oleh data. Entitas juga dapat diartikan sebagai individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari suatu yang lain. Ada dua macam entitas yaitu entitas kuat dan entitas lemah. Entitas kuat merupakan entitas yang tidak memiliki ketergantungan dengan entitas lainnya. Contoh entitas anggota. Sedangkan entitas lemah merupakan entitas yang kemunculannya tergantung pada entitas lain dalam suatu relasi. Entitas digambarkan dengan persegi panjang.

Simbol :

Sumber: Sukamto dan Shalahuddin (2013:53)

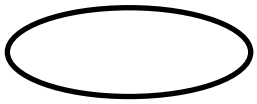
Gambar II.6.

Gambar Entitas

2.)Atribut

Atribut merupakan pendekskripsian karakteristik dari entitas. Atribut digambarkan dalam bentuk lingkaran atau elips. Atribut yang menjadi kunci entitas atau *key* diberi garis bawah.

Simbol :



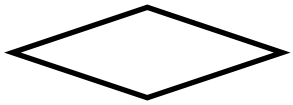
Sumber: Sukamto dan Shalahuddin (2013:53)

Gambar II.7.

Gambar Atribut

3.)Relasi

Relasi menunjukan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Dalam hal ini relasi digambarkan dengan garis lurus.

Simbol : 

Sumber: Sukamto dan Shalahuddin (2013:53)

Gambar II.8.
Gambar Relasi

2. *Logical Record Structure* (LRS)

Menurut Hasugian dan Shidiq (2012 : 608) "LRS *(logical record strukture*) adalah sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan mengikuti pola atau aturan permodelan tertentu dalam akitan dengan konvensi ke LRS". Beberapa tipe record digambarkan dengan kotak empat persegi panjang dengan nama yang unik. LRS juga terdiri dari hubungan antara tiga record. Salah satu metode pembuatan LRS yaitu dimulai dengan membuat ERD kemudian konversi ke dalam LRS.

2.2.4. Pengkodean

Menurut kendall (2008a:267) menyimpulkan bahwa “Pengkodean adalah proses dari meletakkan data yang berarti dua macam atau data yang sulit dipakai dengan segera, lebih mudah dimasukan digital atau huruf”.

Menurut Jogiyanto (2008:384) Struktur kode adalah “Suatu susunan digit (angka), huruf dan karakter-karakter khusus yang dapat dirancang dalam bentuk

kode". Kode dapat dibentuk dari kumpulan angka, huruf dan karakter khusu misalnya %, /, -, $, #, &, : dan sebagainya, yaitu sebagai berikut:

1.Harus mudah diingat

Agar kode mudah diingat,maka dapat dilakukan dengan cara menghubungkan kode tersebut dengan obyek yang diwakili dengan kodenya.

2.Harus unik

Kode harus unik untuk masing-masing item yang diwakilinya. Unik berarti tidak ada kode yang kembar.

3.Harus *Fleksibel*

Kode harus *fleksibel* sehingga memungkinkan perubahan-perubahan atau penambahan item baru dapat diwakili oleh kode.

4.Harus *Efisien*

Kode harus sependek mungkin, sehingga mudah diingat dan juga akan *efisien* bila direkam atau disimpan didalam komputer.

2.2.5. HIPO

Menurut Nugroho (2008:118) "HIPO ada suatu teknik menyusun diagram yang menunjuk input,output dan proses yang mengubah input menjadi "output".

Menurut Jogiyanto HM (2010:221) mengumumkan bahwa HIPO merupakan akronim dari *Hirarchy Input Proses Output*, HIPO merupakan paket yang berisikan suatu set diagram yang secara grafis menjelaskan fungsi suatu ssistem dari tingkat umum ketingkat khusus.