

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Secara umum sistem dapat diartikan sebagai kumpulan dari elemen – elemen yang berinteraksi untuk mencapai suatu tujuan tertentu sebagai satu kesatuan. Menjelaskan tentang teori yang berhubungan dengan Tugas Akhir ini, penjelasannya sebagai berikut :

2.1.1. Sistem

Kemajuan teknologi yang berkembang sangat cepat ini memungkinkan masyarakat untuk menikmati kemudahan dalam melakukan sesuatu pekerjaan. Terutama perusahaan baik pemerintahan dan swasta wajib menggunakan sistem.

Menurut Agus Mulyanto (2009:3) “Sistem diartikan sebagai sekelompok komponen yang saling berhubungan, bekerja sama untuk mencapai tujuan bersama dengan menerima *input* serta menghasilkan *input* dalam proses transformasi yang teratur.

Menurut Pratama (2013:7) “Sistem didefenisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama – sama.

Berdasarkan uraian diatas dapat disimpulkan bahwa sistem merupakan bagian yang saling berhubungan secara erat antara satu dengan yang lainnya. Sistem tersebut mempunyai bentuk yang tertentu dan saling mempengaruhi untuk mencapai suatu tujuan yang sama. Sistem terdiri dari beberapa struktur dan proses, sebagai berikut :

1. Karakteristik Sistem

Menurut Mulyono (2009:2) suatu sistem mempunyai beberapa karakteristik, yaitu, komponen atau elemen (*component*), batas sistem (*boundary*), lingkungan luar sistem (*environment*), penghubung (*interface*), masukan (*input*), pengolah (*process*), keluaran (*output*), sasaran (*objective*) atau tujuan (*goal*).

a) Komponen Sistem (*component*)

Suatu sistem tidak beradadalam lingkungan yang kosong, tetapi sebuah sistem berada dan berfungsi di dalam lingkungan yang berisi sistem lainnya. Apabila suatu sistem merupakan salah satu dari komponen sistem lain yang lebih besar disebut dengan *subsistem*, sistem yang lebih besar adalah lingkungannya. Setiap *subsistem* mempunyai sifat – sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b) Batasan Sistem (*boundary*)

Batas sistem merupakan pembatas atau pemisah antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.

c) Lingkungan Luar Sistem (*environment*)

Lingkungan luar sistem adalah apa pun di luar batas dari sistem yang dapat mempengaruhi operasi sistem, baik pengaruh yang menguntungkan dan yang merugikan. Pengaruh yang menguntungkan harus dijaga sehingga mendukung kelangsungan operasi sebuah sistem. Lingkungan yang merugikan harus dikendalikan agar tidak mengganggu kelangsungan sebuah sistem.

d) Penghubung Sistem (*interface*)

Penghubung merupakan media penghubung antara satu *subsistem* dengan *subsistem* yang lainnya. Penghubung inilah yang akan menjadi media yang digunakan data dari masukan (*input*) hingga keluaran (*output*). Dengan adanya penghubung, suatu *subsistem* dapat berinteraksi dan berintegrasi dengan *subsistem* yang lain membentuk satu kesatuan.

e) Masukan Sistem (*input*)

Masukan (*input*) merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah bahan yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah masukan yang diproses untuk mendapatkan keluaran. Contoh lain di dalam perusahaan, karyawan sebagai *maintenance input* yang akan mengoperasikan sistem, data sebagai *signal input* yang akan diolah menjadi informasi.

f) Keluaran Sistem (*output*)

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk *subsistem* yang lain.

g) Pengolahan Sistem (*process*)

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

h) Sasaran Sistem (*objective*)

Suatu sistem mempunyai tujuan atau sasaran, jika sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh terhadap masukan dan keluaran yang dihasilkan.

2. Klasifikasi Sistem

Menurut Ladjamuddin (2013:6) sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya sebagai berikut :

- a) Sistem diklasifikasikan sebagai sistem abstrak dan sistem fisik. Sistem abstrak adalah sistem yang berupa pemikiran atau ide – ide yang tidak tampak secara fisik, contohnya sistem teologi. Sistem fisik adalah sistem yang ada secara fisik, contohnya sistem komputer.
- b) Sistem diklasifikasikan sebagai sistem alamiah dan sistem buatan manusia. Sistem alamiah adalah sistem yang terjadi karena proses alam tidak dibuat oleh manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia.
- c) Sistem diklasifikasikan sebagai sistem tertentu dan sistem tak tentu. Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Sistem tertentu relatif stabil dalam jangka waktu yang lama. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

d) Sistem diklasifikasikan sebagai sistem tertutup dan sistem terbuka. Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

3. Konsep Dasar Sistem Informasi

Informasi merupakan salah satu sumber daya yang sangat diperlukan dalam suatu organisasi. Suatu sistem jika tidak mendapatkan informasi yang cukup tentu saja tidak akan bertahan lama.

Menurut Agus Mulyanto (2009:12), 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 nyata. Informasi merupakan pengetahuan dari hasil pengolahan data – data yang berhubungan menjadi sebuah kesimpulan.

4. Kualitas Informasi

Menurut Ladjamudin (2013:11). Kualitas informasi sangat dipengaruhi atau ditentukan oleh tiga hal sebagai berikut :

a) Relevan (*relevancy*)

Seberapa jauh tingkat relevansi informasi tersebut terhadap kenyataan kejadian masa lalu, kejadian hari ini, dan kejadian yang akan datang. Informasi yang berkualitas akan mampu menunjukkan benang merah relevansi kejadian masa lalu, hari ini, dan masa depan sebagai sebuah bentuk aktivitas yang masuk akal dan mampu dilaksanakan, dibuktikan oleh siapa saja.

b) Akurat (*accuracy*)

Suatu informasi dikatakan berkualitas jika seluruh kebutuhan informasi tersebut sudah tersampaikan, seluruh pesan benar atau sesuai, pesan yang disampaikan lengkap atau hanya sistem yang diinginkan user (*security*).

c) Tepat Waktu (*timeliness*)

Berbagai proses dapat diselesaikan dengan tepat waktu, laporan – laporan yang dibutuhkan dapat disampaikan tepat waktu.

d) Ekonomis (*economy*)

Informasi yang dihasilkan mempunyai daya jual yang tinggi, serta biaya operasional untuk menghasilkan informasi tersebut minimal, informasi tersebut juga mampu memberikan dampak yang luas terhadap pertumbuhan ekonomi dan teknologi informasi.

e) Efisien (*efficiency*)

Informasi yang berkualitas memiliki sintaks atau kalimat yang sederhana, mampu memberikan makna dan hasil yang mendalam, bahkan menggetarkan setiap orang yang menerimanya.

f) Dapat Dipercaya (*reliability*)

Informasi tersebut berasal dari sumber yang dapat dipercaya. Sumber tersebut juga telah teruji tingkat kejujurannya. Contohnya *output* program komputer, bisa dikategorikan sebagai *reliability*, karena program komputer akan memberikan *output* sesuai dengan *input* yang diberikan.

2.1.2. Basis Data

Menurut Fathansyah (2012:2) Basis Data didefinisikan sebagai “himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

Sedangkan menurut Rosa dan Shalahuddin (2013:43). Basis Data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan.

Berdasarkan uraian diatas, basis data adalah kumpulan data yang saling berhubungan untuk menyimpan semua data dan informasi yang dibutuhkan oleh suatu organisasi.

Dalam penulisan tugas akhir ini, penulis menggunakan bahasa SQL (*Structured Query Language*).

Menurut Rosa dan Shalahuddin (2013:46), menyatakan bahwa “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS (*Relationship Database Management System*)”.

Menurut Priyadi (2014:82) “*Structured Query Language (SQL)* merupakan bahasa basis data standar untuk basis data tipe relational”. Bahasa basis data SQL dibagi menjadi tiga (3) jenis, yaitu *Data Definition Language (DDL)*, *Data Manipulation Language (DML)*, *Data Control Language (DCL)*, penjelasannya sebagai berikut :

1. *Data Definition Language (DDL)*

Menurut Priyadi (2014:83) “*Data Definition Language (DDL)* merupakan perintah SQL yang digunakan untuk melakukan definisi awal suatu basis data dan tabel pada konsep RDBMS. Penulisan perintah *Data Definition Language (DDL)* terdiri dari :

a) *Create*

Perintah *create* pada SQL digunakan untuk membuat suatu basis data dan tabel pendukung di dalam pembangunan basis data tersebut.

b) *Alter*

Perintah *alter* pada SQL digunakan untuk mengubah struktur tabel yang terdapat di dalam basis data.

c) *Drop*

Perintah *drop* pada SQL digunakan untuk menghapus tabel yang terdapat di dalam basis data.

2. *Data Manipulation Language (DML)*

Menurut Priyadi (2014:90) “*Data Manipulation Language (DML)* merupakan perintah SQL yang digunakan untuk melakukan pengolahan *record* atau manipulasi *record* pada tabel dalam suatu basis data”. Penulisan perintah *Data Manipulation Language (DML)* terdiri dari :

a) *Insert*

Perintah *insert* pada SQL digunakan untuk melakukan entry atau penambahan *record* pada tabel dalam basis data.

b) *Select*

Perintah *select* pada SQL digunakan untuk memilih *record* yang akan ditampilkan berdasarkan data pada tabel dalam basis data.

c) *Update*

Perintah *update* pada SQL digunakan untuk mengubah data dalam suatu tabel pada *field* tertentu dengan *record* baru berdasarkan suatu *field* sebagai kriteria perubahan *record*-nya.

d) *Delete*

Perintah *delete* pada SQL digunakan untuk menghapus data dalam suatu tabel berdasarkan suatu *field* sebagai kriteria penghapusan *record*-nya.

3. *Data Control Language* (DCL)

Menurut Priyadi (2014:94) *Data Control Language* (DCL) merupakan perintah SQL yang digunakan untuk melakukan pengaturan hak akses suatu objek data para pengguna dalam basis data. Penulisan perintah *Data Control Language* (DCL) terdiri dari :

a) *Grant*

Perintah *grant* pada SQL digunakan oleh seorang administrator basis data untuk memberikan hak akses kepada pengguna tertentu, agar dapat mengakses suatu tabel dalam basis data.

b) *Revoke*

Perintah *revoke* pada SQL digunakan oleh seorang administrator basis data untuk membatalkan atau menghentikan hak akses yang telah diberikan kepada pengguna tertentu, agar tidak dapat mengakses tabel dalam basis data.

Menurut Zaki (2008:94), mengatakan bahwa “MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk tabel – tabel yang saling berhubungan”.

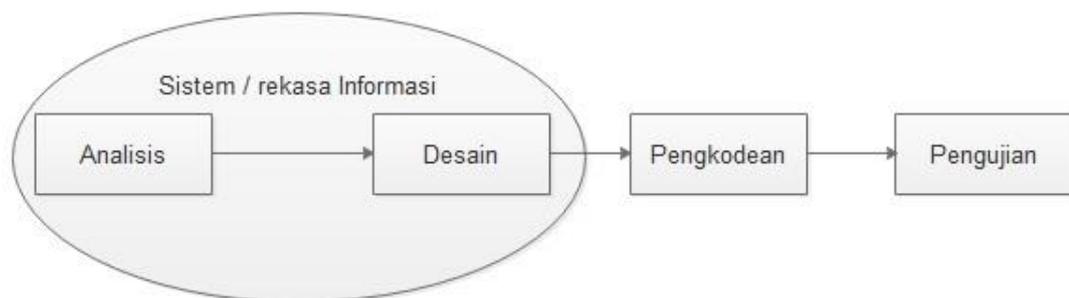
Menurut Sibero (2014:97) “MySQL atau di baca “My Sekuel” dengan adalah suatu RDBMS (*Relational Data Base Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data.

Berdasarkan uraian diatas, MySQL adalah sebuah perangkat lunak sistem manajemen basis data terutama untuk pemilihan atau seleksi dan pemasukan data yang dikerjakan dengan mudah secara otomatis.

2.1.3. Model Pengembangan Perangkat Lunak

Metode yang di gunakan pada pengembangan perangkat lunak ini menggunakan *waterfall* Rosa dan Shalahuddin (2013:29) yang terbagi dalam beberapa bagian, yaitu :

Gambar II.I
Ilustrasi Model Waterfall



Sumber : Rosa dan Shalahuddin (2013:29)

Dari gambar II.I ilustrasi model waterfall penjelasannya sebagai berikut :

1. Analisis Kebutuhan Perangkat Lunak

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

2. Desain

Proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.

3. Pengkodean

Pembuatan kode program desain harus di translasikan kedalam program perangkat lunak.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

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.

2.2. Teori Pendukung

Teori pendukung dalam penulisan tugas akhir ini sangat diperlukan sebagai referensi untuk menunjang atau memperdalam pemahaman tentang informasi. Beberapa teori yang akan dibahas adalah model Diagram Alir Data (DAD), Kamus Data, *Entity Relationship Diagram* (ERD), Pengkodean dan HIPO.

2.2.1. Data Flow Diagram (DFD)

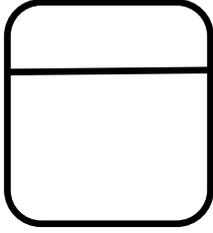
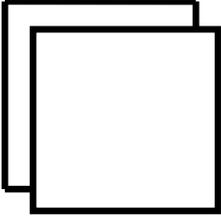
Dalam penulisan tugas akhir ini, penulis menggunakan Diagram Alir Data (*Data Flow Diagram*) untuk menggambarkan pandangan mengenai masukan, proses dan keluaran sistem yang berhubungan dengan mempresetasikan dan menganalisis prosedur – prosedur mendetail dalam sistem yang lebih besar.

A. Konsep

Menurut Rosa dan Shalahuddin (2013:70) “diagram alir data adalah representasi grafik yang menggambarkan aliran informasi dan transformasi yang di aplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*). Notasi – notasi pada DFD (Rosa dan Shalahuddin) adalah sebagai berikut :

B. Simbol – Simbol

Tabel II.I
Notasi – Notasi DFD

Notasi	Keterangan
	<p>Proses merupakan pemodelan perangkat lunak yang akan di implementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi fungsi atau prosedur di dalam kode program.</p>
	<p>Pemodelan perangkat lunak yang akan di implementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus dibuat menjadi tabel – tabel basis data yang di butuhkan, tabel harus sesuai dengan perancangan tabel pada basis data (ERD, CMD, PDM).</p>
	<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang berinteraksi dengan perangkat lunak yang di modelkan atau suatu sistem lain yang terkait dengan aliran data dari sistem yang di modelkan.</p>

	<p>Dipresentasikan dalam bentuk anak panah yang menuju kea tau dari proses dan digunakan untuk menggambarkan gerakan paket data atau informasi dari satu bagian kebagian lain dari sistem dimana penyimpanan mewakili lokasi penyimpanan data</p>
---	---

Sumber: Rosa dan Shalahuddin (2013:71)

Menurut Rosa dan Salahuddin ada beberapa tahapan – tahapan untuk merancang dengan menggunakan DFD yaitu :

1. Diagram Konteks

Diagram ini di buat untuk menggambarkan sistem yang akan di buat suatu entitas tunggal yang berinteraksi dengan orang atau sistem lain.

2. DFD Level 1

Diagram ini di buat untuk menggambarkan modul – modul yang ada dalam sistem yang akan di kembangkan.

3. DFD Level 2

Diagram ini di buat untuk menggambarkan arus secara lebih mendetail lagi dari tahapan proses yang ada di dalam diagram nol.

4. DFD Level 3 dan Seterusnya

Diagram ini merupakan *breakdown* dari modul pada DFD level di atasnya. *Breakdown* pada level 3,4,5 dan seterusnya, aturannya sama persis dengan DFD level 1 atau level 2.

C. Aturan Main

Didalam pembuatan DFD menurut Rusmawan (2011:24) DFD memiliki beberapa aturan yang harus diikuti, yaitu :

1. Tidak boleh menghubungkan entitas luar dengan entitas luar lainnya tanpa adanya proses.
2. Tidak boleh menghubungkan entitas luar dengan *data store* atau sebaliknya tanpa adanya proses.
3. Tidak boleh menghubungkan *data store* dengan *data store* lainnya tanpa adanya proses
4. Tidak boleh menghubungkan proses dengan proses tanpa melalui *data store*.

D. Tahapan Proses Pembuatan DAD (Diagram Alir Data)

Menurut Ladjamudin (2013:64) pembuatan DAD mempunyai beberapa tahapan sebagai berikut :

1. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari satu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh *input* ke sistem atau *output* dari sistem.

2. Diagram Nol

Diagram nol adalah diagram yang menggambarkan proses dari *data flow diagram*.

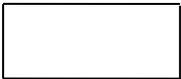
3. Diagram Rinci

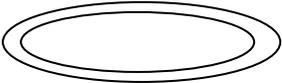
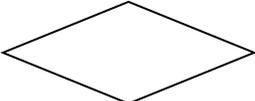
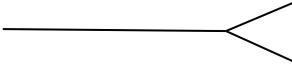
Diagram rinci adalah diagram yang mengurai proses apa yang ada didalam diagram nol atau diagram level di atasnya.

2.2.2. Entity Relationship Diagram (ERD)

Menurut Rosa dan Shalahuddin (2013:50) “ ERD digunakan untuk pemodelan basis data relasional. Basis data tidak perlu menggunakan OODBMS, sehingga perancangan basis tidak perlu menggunakan ERD”. ERD memiliki beberapa aliran notasi seperti notasi chen (dikembangkan oleh Peter Chen), dan Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain. Berikut adalah simbol – simbol yang digunakan pada ERD dengan notasi Chen :

Tabel II.2
Entity Relationship Diagram (ERD)

Simbol	Deskripsi
Entitas 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih kekata benda dan belum merupakan nama tabel.
Atribut 	Filed atau kolom data yang butuh disimpan dalam suatu entitas.
<u>Atribut kunci primer</u> 	Filed atau kolom yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat berbeda.

Atribut multinilai 	Filed atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
Asolisasi 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas lain disebut dengan kardinalitas. Misalnya ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

Sumber : Rosa dan shalahuddin (2013:50)

A. *Logical Record Structure (LRS)*

Menurut Ladjamuddin (2013:159) *Logical Record Structure (LRS)* merupakan transformasi dari *Entity Relationship Diagram (ERD)* adapun aturan – aturan dalam melakukan transformasi E-R Diagram ke *logical record structure* adalah sebagai berikut :

1. Setiap entity akan diubah kebentuk sebuah kotak dengan nama entity berada di luar kotak dan atribut berada didalam kotak.
2. Sebuah relasi kadang disatukan dalam sebuah kotak bersama entity, kadang dipisah dalam sebuah kotak tersendiri.

Aturan pokok di atas akan sangat dipengaruhi oleh elemen yang menjadi titik perhatian utama pada langkah transformasi yaitu kardinalitas. 1:1 (*one to one*). Pada kardinalitas *one to one*, sebaiknya panah di arahkan ke entity dengan jumlah atribut yang lebih sedikit. 1:M (*one to many*) Pada kardinalitas *one to many*, maka relasi harus digabungkan dengan entity pada pihak yang lebih banyak, dan tidak

perlu melihat banyak sedikitnya atribut pada atribut tersebut. M:M (*many to many*) Pada kardinlitas *many to many*, maka *relationship* berubah status menjadi *file konektor* sehingga baik entity maupun relasi akan menjadi struktur *record* tersendiri. Dengan demikian, maka panah dari entity A dan entity B akan mengarah ke *relationship* tersebut.

Menurut para ahli uraian di atas *Logical Record Structure* (LRS) terdiri dari *link – link* diantara tipe *record*, *link* ini menunjukkan arah dari satu tipe *record* lainnya.

2.2.3. Pengkodean

Menurut Fathansyah (2007:105) pengkodean adalah “cara yang ditempuh untuk menyatakan suatu data dalam bentuk lain”. Ada tiga jenis bentuk pengkodean yang dipilih menurut Fathansyah (2007:105).

1. Sekuensial

Pengkodean dilakukan dengan mengasosiasikan data dengan kode terurut (seperti bilangan asli atau abjad). Contoh, nilai mutu kuliah (“sempurna”, “baik”, “cukup”, “kurang baik”, “buruk”) dikodekan dengan ‘A’, ‘B’, ‘C’, ‘D’ dan ‘E’.

2. Mnemonic

Pengkodean dilakukan dengan membentuk suatu singkatan dari data yang ingin dibuat. Data jenis kelamin (‘laki - laki’ dan ‘perempuan’) ditulis dengan ‘L’ dan ‘P’.

3. Block

Pengkodean dinyatakan dalam format tertentu, data no induk mahasiswa dalam bentuk format XXYYYY yang berbentuk atas XX = dua digit terakhir angka tahun masuk dan YYY = no.urut mahasiswa

2.2.4 HIPO (*Hearchy Plus Input Process-Output*)

HIPO (*Hearchy Plus Input Process-Output*) adalah tehnik dokumentasi program yang digunakan untuk komunikasi spesifikasi sistem kepada progremer melalui proses perancangan. Hipo memiliki tujuan yang paling penting untuk menghasilkan *output* yang benar untuk memenuhi kebutuhan *user*.

Menurut Ladjamuddin (2013:211)” HIPO merupakan tehnik untuk mendokumentasikan sistem pemograman”.

HIPO dikembangkan oleh personal IDM yang percaya dokumentasi sistem yang akan mempercepat pencarian prosedur yang akan dimodifikasi, karena HIPO menyediakan fasilitas lokal dalam bentuk kode dari prosedur dalam suatu sistem. Hipo dapat digunakan untuk memenuhi kebutuhan dari berbagai macam pemakaian yang menggunakan dokumentasi untuk tujuan – tujuan yang berdeda.

2.2.5 Kamus Data

Kamus data menjelaskan tentang teori aliran data dan penyimpanan data dalam DFD. Menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran. Mendeskripsikan hubungan detail antara penyimpanan (yang akan menjadi titik perhatian dalam ERD).

A. Konsep

Menurut Rosa dan Shalahuddin (2013:73) “kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem sehingga masukan dan keluaran dapat dipahami secara umum.

1. Nama dan Arus

Nama arus data harus dicatat pada kamus data, sehingga mereka yang membaca DAD memerlukan penjelasan lebih lanjut tentang suatu arus data tertentu dan dapat langsung mencarinya dengan mudah di kamus data.

2. Alias

Alias atau nama lain dari data dapat ditulis bila ada. Untuk menyatakan nama lain dari suatu *dataelement* atau *datastore* yang sebenarnya sama dengan *dataelement* atau *datastore* yang telah ada.

3. Bentuk Data

Bentuk data perlu dicatat di kamus data, karena dapat dipergunakan untuk mengelompokkan kamus data kegunaannya sewaktu perancangan sistem.

4. Arus Data

Arus data menunjukkan dari mana data mengalir dan kemana data menuju. Keterangan arus data ini perlu dicatat di kamus data untuk memudahkan mencari arus data DAD.

5. Penjelasan

Untuk menjelaskan tentang makna dari arus data yang dicatat di kamus data, sebagian penjelasan dapat diisi dengan keterangan – keterangan tentang arus data tersebut.

Dimana notasi kamus data lebih mudah menjelaskan data dibandingkan dengan narasi.

Notasi atau simbol yang digunakan dibagi menjadi dua macam yaitu sebagai berikut :

a. Notasi Tipe Data

Notasi ini digunakan untuk membuat spesifikasi format input maupun suatu data. Notasi yang umum digunakan antara lain :

Tabel II.3
Notasi Tipe Data

Notasi	Keterangan
X	Setiap Karakter
9	Angka Numerik
A	Karakter Alphabet
Z	Angka Nol Ditampilkan Sebagai Spasi Kosong
.	Titik, Sebagai pemisah Ribuan
,	Koma, Sebagai Pemisah Pecahan
-	<i>Hypen</i> , Sebagai Tanda Penghubung
/	<i>Slash</i> , Sebagai Tanda Pembagi

Sumber : Jogiyanto(2015:370)

b. Notasi Struktur Data

Notasi ini digunakan untuk membuat spesifikasi elemen data. Dimana notasi yang umum di gunakan adalah sebagai berikut :

Tabel II.4
Notasi Struktur Data

Notasi	Keterangan
=	Terdiri Dari
+	Duan (<i>And</i>)
()	Pilih (Ya atau Tidak)
{ }	Pengulangan Proses (Iterasi)
[]	Pilih Salah Satu Pilihan
	Pemisah Pilihan Didalam Data []
*	Keterangan atau Catatan
@	Petunjuk (<i>Key Field</i>)

Sumber : Jogiyanto (2015:371)

2.3 *Tools* Pendukung

1. *Database*

Menurut Hirin & Virgi (2011:29) “*database* atau basis data adalah sekumpulan informasi atau data secara sistematis sehingga dapat diperiksa program komputer untuk memperoleh informasi dari basis data tersebut.

Menurut Zaki (2008:94) “*database* adalah kemudahannya dalam penyimpanan dan menampilkan data karena dalam bentuk tabel”.

Jadi berdasarkan para ahli uraian diatas dapat disimpulkan bahwa *database* adalah suatu tempat penyimpanan yang besar dan dapat menyimpan

banyak data, dirancang sedemikian rupa dengan tujuan *database* tersebut dapat dijalankan dengan efisien dan efektif.

2. Microsoft Visual Basic 6.0

Menurut Kusriani dan Koniyo (2007:171), "Visual Basic adalah salah satu bahasa pemrograman komputer. Bahasa pemrograman adalah perintah – perintah yang dimengerti oleh komputer untuk melakukan tugas – tugas tertentu. Bahasa pemrograman visual basic, yang dikembangkan oleh microsoft sejak tahun 1991, merupakan pengembangan dari pendahulunya, yaitu bahasa pemrograman BASIC (*Beginner's All-purpose Symbolic Instruction Code*) yang dikembangkan pada era 1950-an. Visual basic merupakan salah satu *development tool*, yaitu alat bantu untuk berbagai macam program komputer, khususnya yang menggunakan sistem operasi windows. Visual basic merupakan salah satu bahasa pemrograman komputer yang mendukung pemrograman berorientasi objek (*Object Oriented Integrated Development Enviroment*) atau lingkungan kerja dari visual basic 6.0 itu sendiri.