

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 Sistem

Berikut ini adalah beberapa definisi sistem menurut beberapa ahli, di antaranya:

Pengertian sistem menurut Davis dalam Sutabri (2012:6) menyatakan bahwa “sistem bias berupa abstrak atau fisik”. Sistem yang abstrak adalah susunan atau gagasan-gagasan atau konsepsi yang teratur yang saling berhubungan. Sedangkan sistem yang bersifat fisik adalah serangkaian unsur yang bekerja sama untuk mencapai suatu tujuan.

Menurut Norman L. Enger dalam Sutabri (2012:12) menyatakan, suatu sistem dapat terdiri dari atas kegiatan-kegiatan yang berhubungan guna mencapai tujuan-tujuan perusahaan seperti pengendalian inventaris atau penjadwalan produksi.

Menurut Gordon B. Davis dalam bukunya (2012:17) menyatakan, sistem bisa berupa abstrak atau fisik. Sistem yang abstrak adalah susunan yang teratur dari gagasan-gagasan atau konsepsi yang saling bergantung. Sedangkan sistem yang bersifat fisik adalah serangkaian unsur yang bekerjasama untuk mencapai suatu tujuan.

Menurut Sutarman (2009:5) “Sistem adalah kumpulan elemen yang saling berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama”.

Menurut Jogiyanto (2009:2) “Sistem dapat didefinisikan dengan pendekatan prosedur dan dengan pendekatan komponen”.

Menurut Moekijat dalam Prasajo (2011:152) “Sistem adalah setiap sesuatu terdiri dari objek-objek, unsur-unsur atau komponen-komponen yang bertata kaitan dan bertata hubungan satu sama lain sedemikian rupa, sehingga unsur-unsur tersebut merupakan satu kesatuan pemrosesan atau pengelolaan yang tertentu”.

2.1.2 Karakteristik Sistem

Menurut Mulyanto (2009:2) dalam bukunya Sistem Informasi Konsep dan Aplikasi mempunyai:

1. Komponen sistem (*Component*)

komponen sistem adalah suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem tersebut dapat berupa subsistem.

Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batasan sistem (*Bondary*)

Batasan sistem merupakan pembatas atau pemisah antar suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*)

Segala sesuatu diluar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.

4. Penghubung Sistem (*Interface*)

Penghubung sistem merupakan media penghubung antara suatu sistem dengan subsistem lainnya.

5. Masukan Sistem (*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. Keluaran Sistem (*Output*)

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

7. Pengolahan sistem (*Process*)

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.

8. Sasaran Sistem (*Objective*) 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 agar 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 dan dikembangkan 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 sutabri (2012:22) sistem merupakan bentuk intergrasi antara satu komponen dengan komponen lain, karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi dalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya:

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

Sistem abstrak merupakan sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologi, yaitu sistem berupa pemikiran-pemikiran hubungan antara manusia dengan Tuhan. Sedangkan sistem fisik merupakan sistem yang ada secara fisik. Misalnya sistem computer, sistem autansi, dan sistem persediaan barang.

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

sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia. Misalnya sistem perputaran bumi. Sistem buatan manusia adalah sistem yang melibatkan interaksi manusia dan mesin yang dibuat *human machine system*. Misalnya sistem informasi berbasis komputer.

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

Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi. Sebagai contoh adalah hasil pertandingan sepak bola. Sistem tak tertentu adalah sistem yang kondisi masa depannya tidak dapat di prediksi karena mengandung unsur probabilitas. Misalnya kematian seseorang.

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

Sistem tertutup adalah sistem yang tidak terhubung dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya campur tangan dari pihak diluarnya. Secara teoritis sistem tertutup ini ada, tetapi pada kenyataan tidak ada sistem yang benar-benar tertutup. Contohnya sistem adat masyarakat baduy. Sedangkan sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya. Misalnya sistem musyawarah.

2.1.4 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 informasi tersebut terdiri dari blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), dan basis data (*database block*).

1. Blok masukan (*input block*), input memiliki data yang masuk ke dalam sistem informasi, juga metode-metode untuk menangkap data yang dimasukkan.
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.

3. Blok keluaran (*output block*), produk dari sistem informasi adalah keluaran yang merupakan informasi berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Blok teknologi (*technology 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*).
5. 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.

2.1.5 Basis Data (*Database*)

Menurut Rosa A.S (2013:43) “Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah di olah atau informasi dan membuat informasi tersedia saat dibutuhkan”. Penyusunan basis data meliputi proses memasukan data kedalam media penyimpanan data, dan diatur dengan menggunakan perangkat Sistem Manajemen Basis Data (*Database Manajemen System DBDMS*).

Menurut Connolly dan Begg (2010:65) “*Database* adalah sekumpulan data tersebar yang terhubung secara logis dan penjelasan dari data ini dirancang untuk memenuhi kebutuhan informasi dari suatu organisasi”.

Menurut Gottschalk dan Saether (2010:41) “*Database* adalah sekumpulan data yang terorganisir untuk mendukung banyak aplikasi secara efisien dengan memusatkan data dan mengontrol data *redundant*”.

2.1.6 Model Pengembangan Perangkat Lunak

Model *WaterFall*(air terjun) telah diperoleh dari proses *engineering*. Model *waterfall* juga menawarkan cara pembuatan perangkat lunak secara lebih nyata menurut Rosa A.S (2013:28) langkah-langkah yang penting dalam model ini adalah:

1. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif, untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibuuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk di dokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langka yang fokus ada desain pembuatan program perangkat lunak. Termaksud struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengkodean. Tahapan ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain. Agar dapat di implementsikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

Desain harus di translasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program computer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi *logic* dan fungsional dan memastikan bahwa semua bagian sudah diuji. Guna untuk meminimalisir kesalahan dan hasil sesuai apa yang diinginkan.

5. Pendukung (*Support*) atau pemeliharaan (*maintance*)

Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.2 Teori pendukung

2.2.1 *Data Flow Diagram*

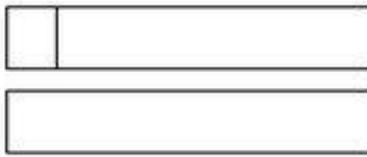
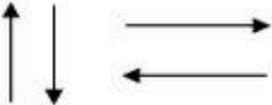
Menurut kendall & Kendall (2010:263) “Diagram Alir Data/Data Flow Diagram menggambarkan pandangan sejauh mungkin mengenai masukan, proses dan keluaran dari model sistem”.

Menurut Sutabri (2012:116) “*Data Flow Diagram* adalah suatu *network* yang mengembangkan suatu *system automat* atau terkomputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai aturan mainnya”.

1. Simbol Diagram Alir Data/*Data Flow Diagram*

Menurut Kendall dan Kendall (2010:265) simbol-simbol yang digunakan dalam menggambaran Diagram Alir Data/*Data Flow Diagram* Adalah:

Tabel II. Simbol *Data flow diagram*

Simbol	Keterangan
	<i>External Entity</i> , merupakan kesatuan di lingkungan luar sistem yang bisa berupa orang, organisasi atau sistem lain.
	<i>Process</i> , merupakan proses seperti perhitungan aritmatik penulisan suatu formula atau pembuatan laporan
	<i>Data Store</i> (Simpan Data), dapat berupa suatu file atau database pada sistem komputer atau catatan manual
	<i>Data Flow</i> (arus data), arus data ini mengalir diantara proses, simpan data dan kesatuan luar

Sumber: Rosa,(2013:71)

2. Beberapa Langkah Membuat DAD/DFD

a. Diagram Konteks

Diagram ini dibuat untuk menggambarkan sumber atau tujuan data yang akan di proses atau untuk menggambarkan sistem secara umum atau global dari keseluruhan sistem yang ada.

b. Diagram Nol

Diagram ini dibuat untuk menggambarkan tahapan proses yang ada di dalam diagram konteks, yang penjabarannya lebih terperinci.

c. Diagram Detail

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

3. Aturan Main Dalam Penggunaan DAD/DFD

Menurut Kenall & Kendall (2010:272) didalam pembuatan DAD/DFD terdapat aturan dan ketentuan yang berlaku, adapun aturan dalam pembuatan DAD/DFD adalah:

- a. Tidak boleh menghubungkan antara *external entity* dengan *external entity* lainnya secara langsung.
- b. Tidak boleh menghubungkan antara *data store* dengan *data store* lainnya secara langsung.
- c. Tidak boleh menghubungkan *data store* dengan *external entity* secara langsung.
- d. Setiap proses harus ada *data flow* yang masuk dan juga *data flow* yang keluar.
- e. Aliran data tidak boleh terbelah menjadi dua atau lebih aliran data yang berbeda.

2.2.2 Kamus Data

Kamus Data (Data Dictionary) menurut Kendall & Kendall (2013:333) adalah Suatu Aplikasi khusus dari jenis kamus-kamus yang digunakan sebagai referensi 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.

Menurut Rosa (2013:73) kamus data harus dapat mencerminkan keterangan yang jelas tentang data yang dicatat, adapun isi kamus data sebagai berikut:

a. Nama Arus Data

Karena kamus data dibuat berdasarkan arus data yang mengalir di *Data Flow Diagram*(DFD), maka nama dari arus data juga harus dicatat dikamus data.

b. Alias

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

c. Tipe Data

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

d. Arus Data

Arus data menunjukkan dari mana data mengalir dan kemana data akan menuju. Fungsinya untuk memudahkan mencari arus data di dalam *Data Flow Diagram*(DFD).

e. Penjelasa

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

f. Periode

Periode ini menunjukkan kapan terjadinya arus data umum. Periode ini dicatat dikamus data karena dapat digunakan untuk mendefinisikan kapan

input data harus dimasukkan kedalam sistem, kapan proses program harus dilakukan dan kapan laporan – laporan harus dihasilkan.

g. Volume

Volume yang perlu dicatat dalam kamus data adalah tentang volume rata-rata dan volume puncak dari kamus data. Volume rata-rata menunjukkan banyaknya arus data yang mengalir dalam satu periode tertentu, sedangkan volume puncak menunjukkan volume yang terbanyak.

h. Struktur Data

Struktur yang menunjukkan 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 2.1 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
-	<i>Hypen</i> sebagai tanda penghubung
/	<i>Slash</i> sebagai tanda pembagi

Sumber: Rosa, (2013:73)

2. Notasi Struktur Data

Notasi digunakan untuk membuat spesifikasi elemen data sebagai berikut:

Tabel 2.2 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: Rosa, (2013:73)

2.2.3 Entity Relationship Diagram (ERD)

1. Definisi ERD

Menurut Sutanta (2011:91) “Entity Relationship Diagram (ERD) merupakan suatu model data yang dikembangkan berdasarkan objek.” Entity Relationship Diagram (ERD) digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. Entity Relationship Diagram (ERD) didasarkan pada suatu persepsi bahwa real world terdiri atas obyek-obyek dasar tersebut. Penggunaan Entity Relationship Diagram (ERD) relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis

sistem, Entity Relationship Diagram (ERD) berguna untuk memodelkan sistem yang nantinya, basis data akan di kembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antardata didalamnya.

Menurut Ladjamudin (2013:142) “ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”.

Menurut Simarmata (2010:67) “Entity Relation Diagram adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek kedalam entitas-entitas dan menentukan hubungan antar entitas”.

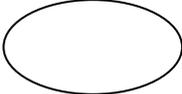
2. Komponen ERD

Komponen Entity Relationship Diagram menurut Sutanta (2011:91) adalah sebagai berikut :

- a.  Entitas persegi panjang mewakili kumpulan entitas

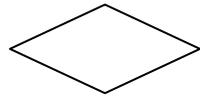
Entitas adalah sesuatu yang nyata atau abstrak dimana kita akan menyimpan data.

- b. Atribut(*attribute*)

 Ellipse mewakili atribut

Atribut adalah ciri umum semua atau sebagian besar instansi pada entitas tertentu. Sebutan lain atribut adalah property, elemen data, field.

c. Relasi (*relationship*)



Belah ketupat mewakili relasi

Relasi adalah hubungan ilmiah yang terjadi antara satu atau lebih entitas, misalnya proses pembayaran pegawai. Kardinalitas menentukan kejadian suatu entitas untuk satu kejadian pada entitas yang berhubungan.

3. Derajat Relasi

Derajat Relasi menurut Sutanta (2011:91) adalah sebagai berikut :

a. One to One (1:1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota B, begitu pula sebaliknya.

b. One to many (1:M/Many)

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

c. Many to Many (M:M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

2.2.4 *Logical Record Structure*

Menurut Friyadie (2007:13) “LRS merupakan hasil dari pemodelan *Entity Relational Ship* beserta atributnya sehingga bisa terlihat hubungan-hubungan antar

entitas". Dalam pembuatan LRS terdapat 3 hal yang dapat mempengaruhi (Friedyadie, 2007:13) yaitu:

1. Jika tingkat hubungan (*cardinality*) satu pada satu (*one-to-one*), maka digabungkan dengan entitas yang lebih kuat (*strong entity*), atau digabungkan dengan entitas yang memiliki atribut yang lebih sedikit.
2. Jika tingkat hubungan (*cardinality*) satu pada banyak (*one-to-many*), maka hubungan relasi atau digabungkan dengan entitas yang tingkat hubungannya banyak.
3. Jika tingkat hubungan (*cardinality*) banyak pada banyak (*many-to-many*), maka hubungan relasi tidak akan digabungkan dengan entitas manapun, melainkan menjadi sebuah LRS.

2.2.5 Key

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

1. Kunci Kandidat (*Candidate Key*)
 Suatu atribut atau set minimal atribut mengidentifikasi secara unik suatu kejadian yang spesifikasi dari suatu *entity*.
2. Kunci Primer (*Primary Key*)
 Suatu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian yang secara spesifikasi, akan tetapi juga dapat mewakili setiap kejadian dari suatu *entity*. Setiap kandidat punya peluang menjadi *primary key*, akan tetapi

sebaliknya dipilih satu saja yang dapat mewakili entitas yang ada secara menyeluruh.

3. Kunci Alternatif (*Alternate Key*)

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

4. Kunci Tamu (*Foreign Key*)

Suatu atribut atau set minimal atribut yang melengkapi satu hubungan yang menunjukkan keinduknya.

2.2.6 Pengkodean

Menurut Kadir (2009:298) bahwa “kode digunakan untuk tujuan mengklasifikasikan data, memasukan data kedalam komputer dan untuk mengambil bermacam-macam informasi yang berhubungan dengannya”.

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 khusus 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.

2.2.7 HIPO

Menurut Praptiningsih (2012:03) mengumumkan bahwa HIPO (*Hierarchy Input Process Output*) yaitu alat bantu yang digunakan untuk membuat spesifikasi program yang merupakan struktur yang berisi diagram dimana di dalam program ini berisi *input* yang diproses dan menghasilkan *output*.

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

Gambar 2.3 Diagram HIPO

