

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

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

2.1.1. Sistem

1. Definisi Sistem

Defenisi Sistem berdasarkan pendapat para pakar sistem adalah sebagai berikut :

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

Sedangkan menurut Norman L. Enger dalam Sutabri (2012:7) menyatakan bahwa suatu” sistem dapat terdiri atas kegiatan-kegiatan yang berhubungan guna mencapai tujuan-tujuan perusahaan seperti pengendalian inventaris atau penjadwalan produksi”.

Sedangkan menurut Prof. Dr.Mr. S.Prajudi Atmosudirdjo dalam Sutabri (2012:7) Suatu sistem terdiri atas objek-objek atau unsur-unsur atau komponen-komponen yang berkaitan dan berhubungan satu sama lainnya sedemikian rupa sehingga unsur-unsur tersebut merupakan suatu kesatuan pemrosesan atau pengolahan yang tertentu”.

Dari beberapa definisi yang dikemukakan , dapat diambil kesimpulan bahwa suatu sistem merupakan keterpaduan dari elemen-elemen (*subsistem*) yang terdiri dari jaringan prosedur yang saling bekerjasama dan berhubungan untuk mencapai suatu tujuan atau sasaran.

2. Karakteristik Sistem

Menurut Sutabri (2012:13) Sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut :

a. Komponen Sistem (*Components*)

Suatu sistem terdiri dari jumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu subsistem. Setiap subsistem memiliki sifat-sifat sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang besar yang disebut dengan Supra sistem.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lain atau sistem dengan lingkungan luar. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisah.

c. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada diluar lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan. Kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

d. Penghubungan sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. Keluaran suatu subsistem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung. Dengan demikian terjadilah suatu integrasi sistem yang membentuk satu kesatuan.

e. Masukan sistem (*Input*)

Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance*) dan sinyal (*signal input*). Sebagai contoh

didalam suatu unit sistem komputer, “Program” dalah *maintenance input* yang digunakan untuk mengoperasikan komputer. Sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

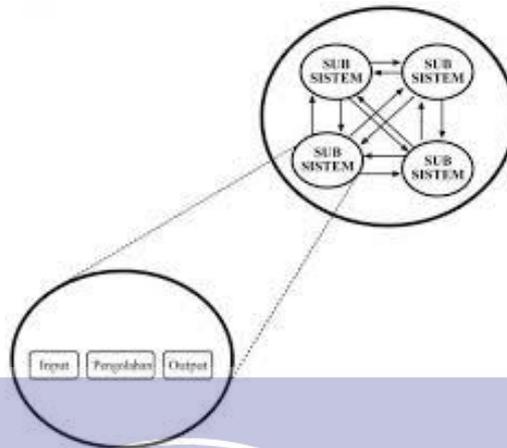
Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, dimana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan input bagi subsistem lainnya.

g. Pengolahan Sistem (*Procces*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

h. 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.



Sumber : Tata Sutabri , (2012:14)

Gambar II.1 Karakteristik Sistem

3. Klafifikasi Sitem

Menurut Sutabri (2012:15) Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

a. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan; sedangkan sistem fisik merupakan sistem yang secara fisik, seperti sistem komputer, sistem produksi, sistem penjualan, sistem administrasi, personalia dan sebagainya.

b. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadi siang dan malam, dan pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan

human machine system. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut pengguna komputer yang berinteraksi dengan manusia.

c. Sistem Deterministik dan sistem Probabilistik

Sistem yang beroperasi dengan tingkah laku 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 diprediksikan, karena mengandung unsur probabilitas.

d. Sistem terbuka dan sistem tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.1.2. Pengertian Data

Menurut Prof. Dr. Winardi, SE. (2010:510) Data merupakan fakta mentah dalam bentuk terpisah, yang apabila ditempatkan dalam konteks yang berarti, melalui operasi pemrosesan data, memungkinkan orang menarik sejumlah inferensi dari padanya.

Menurut Sutabri (2012:25) Data merupakan bentuk mentah yang belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model agar menghasilkan informasi.

Dari beberapa definisi yang dikemukakan diatas, dapat diambil kesimpulan bahwa data merupakan bahan yang akan diolah menjadi suatu bentuk yang lebih berguna dan bermanfaat.

2.1.3. Informasi

Definisi Informasi menurut Sutabri (2012:22) Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya .

Kualitas dari suatu informasi tergantung dari 3 (tiga) hal menurut sutabri (2012:33), yaitu informasi harus akurat (*accurate*), tepat waktu (*timelines*), dan relevan (*relevance*).

a. Akurat (*accurate*)

Informasi harus bebas dari kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi mungkin banyak mengalami gangguan (*noise*) yang dapat mengubah atau merusak informasi tersebut.

b. Tepat waktu (*timelines*)

Informasi yang sampai kepada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan di dalam pengambilan keputusan. Bila pengambilan keputusan terlambat maka dapat berakibat fatal bagi organisasi. Dewasa ini informasi bernilai mahal karena harus cepat dikirim dan didapat sehingga memerlukan teknologi mutakhir untuk mendapatkan, mengolah, dan mengirimkannya.

c. Relevan (*relevance*)

Informasi tersebut mempunyai manfaat untuk pemakaiannya. Relevansi informasi untuk setiap orang berbeda. Menyampaikan informasi tentang penyebab kerusakan mesin produksi kepada akuntan perusahaan tentunya kurang relevan. Akan lebih relevan bila ditujukan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi disampaikan untuk ahli teknik merupakan informasi yang kurang relevan, tetapi akan sangat relevan untuk seorang akuntan perusahaan.



Sumber, (Sutabri, 2012:26)

Gambar II.2 Siklus Informasi

2.1.4. Basis Data

Menurut Fathansyah (2015:2) menjelaskan bahwa “Basis Data terdiri atas 2 kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan Data adalah refresentasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa,

pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang berwujud dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya”. Sebagai satu kesatuan istilah, Basis Data (*Database*) sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file* atau tabel atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Basis Data dan lemari arsip sesungguhnya memiliki prinsip kerja dan tujuan yang sama. Prinsip utamanya adalah pengaturan data atau prinsip. Dan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data atau arsip. Perbedaannya hanya terletak pada media penyimpanan yang digunakan. Jika lemari arsip menggunakan lemari besi atau kayu sebagai media penyimpanan, maka basis data menggunakan media penyimpanan elektronik seperti cakram magnetis (*magnetic disk* atau disingkat sebagai *disk* saja). Hal ini merupakan konsekuensi yang logis, karena lemari arsip langsung dikelola oleh manusia, sementara basis data dikelola melalui perantaramesin pintar elektronik (yang kita kenal sebagai komputer).

Satu hal yang juga harus diperhatikan, bahwa basis data bukan hanya sekedar penyimpanan data secara elektronik (dengan bantuan komputer). Artinya,

tidak semua bentuk penyimpanan data secara elektronik bisa disebut basis data. Kita dapat menyimpan dokumen berisi data dalam *file* teks (dengan program pengolah kata), *file spread sheet*, dan lain-lain, tetapi tidak bisa disebut sebagai basis data. Hal ini, karena di dalamnya tidak ada pemilihan dan pengelompokan data sesuai jenis data. Kelak ketika *file-file* tersebut sudah cukup banyak, maka situasi ini tentu akan menyulitkan pencarian data tertentu. Yang sangat ditonjolkan dalam basis data adalah pengaturan, pemilihan, pengelompokan, pengorganisasian data yang akan kita simpan sesuai fungsi atau jenisnya. Pemilihan, pengelompokan, pengorganisasian ini dapat berbentuk sejumlah tabel terpisah atau dalam bentuk pendefinisian kolom-kolom (*field*) data dalam setiap tabel.

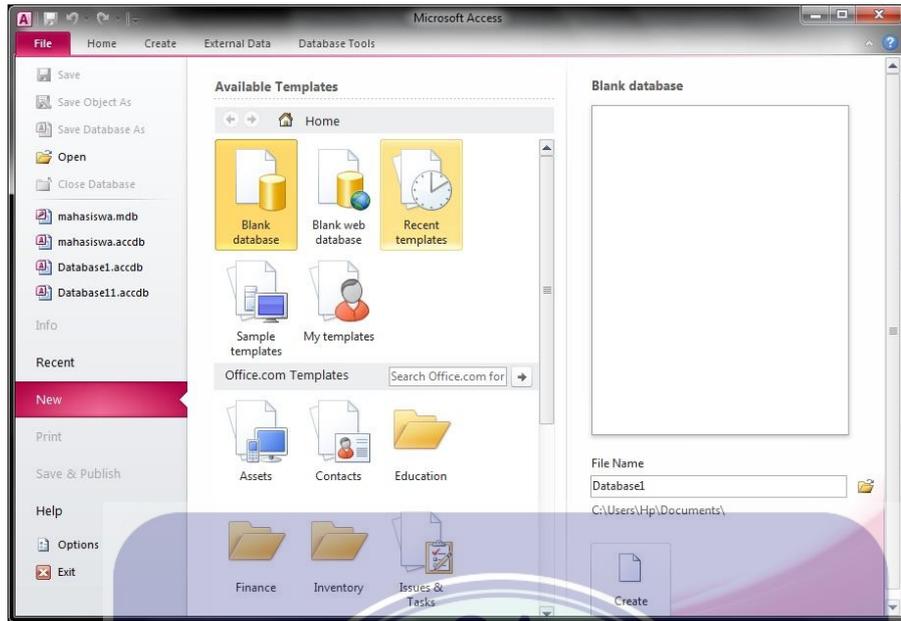
1. Microsoft Access

Menurut Suarna (2008 : 11) *Microsoft office Access* adalah sebuah Program aplikasi untuk mengolah database (Basis data) model relasional, karena terdiri dari lajur kolom dan lajur baris.

Fungsi Microsoft Office Access

- a. Untuk membuat program aplikasi persediaan barang
- b. Untuk membuat program aplikasi gaji karyawan/pegawai
- c. Untuk membuat program aplikasi gaji
- d. Untuk membuat program aplikasi kehadiran
- e. Untuk membuat program aplikasi antrian kunjungan berobat.

Ini Contoh halaman jendela Utama *Microsoft Access* :

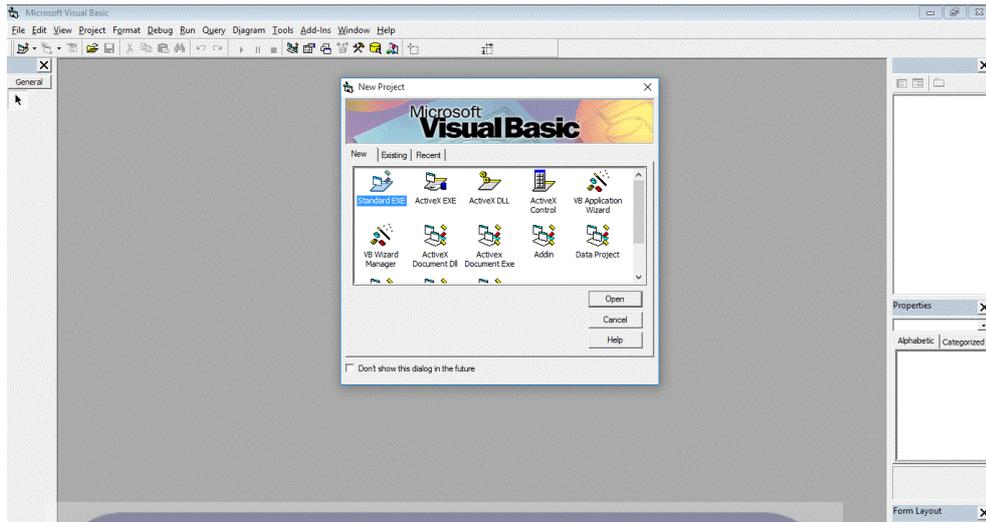


Sumber, suarna (2008 : 28)

Gambar II.3 Jendela Menu Utama Microsoft Acces

2. Visual basic

Menurut Aminudin (2016:2) “*visual basic 6.0* adalah bahasa pemrograman *event-driven* yang berasal dari *BASIC*”. *Event driven* artinya program menunggu sampai adanya respons dari pemakai berupa kejadian tertentu, misalnya tombol diklik atau menu dipilih. Ketika *event* terdeteksi, *event* yang berhubungan akan melakukan aksi sesuai dengan kode yang diberikan. *Visual basic* merupakan salah satu *development tool* yaitu alat bantu untuk membuat berbagai macam program komputer, khususnya yang menggunakan sistem operasi *windows*. *Visual basic* merupakan salah satu bahasa pemrograman komputer yang mendukung *object* (*Object Oriented Programming = OOP*).



Sumber, Aminudin (2016 : 32)

Gambar II.4. Tampilan Menu Utama Program Vb.6.0

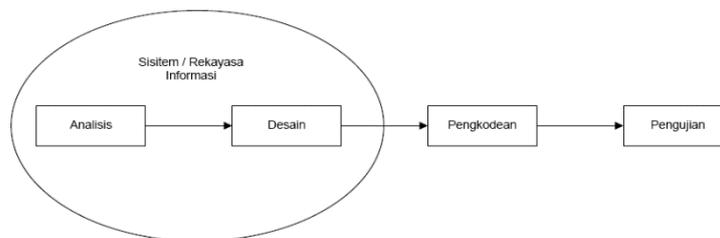
2.1.5. Model Pengembangan perangkat lunak

1. Metode *Waterfall*

Menurut Shalahuddin dan Rosa (2013:28) Menjelaskan bahwa “Model air terjun (*waterfall*) sering juga disebut model sekurnsial linier (sequential linier) atau alur hidup klasik (*classic life cycle*)”.

Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahapan pendukung. (*support*).

Berikut ini adalah gambar model air terjun :



Sumber : Shalhuddin (2013:28).

Gambar II.5 Ilustrasi Model *Waterfall*

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahapan ini perlu untuk didokumentasikan.

b. 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 mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pembuatan kode program

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

d. Pengujian

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

e. 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 yang baru. 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.

Model air terjun (*watfall*) sangat cocok digunakan kebutuhan pelanggan sangat di pahami dan kemungkinan terjadinya perubahan kebutuhan selama pengembangan perangkat lunak kecil, hal positif dari model air terjun adalah struktur tahap pengembangan sistem jelas, dokumentasi dihasilkan disetiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap).

2. Model Prototipe

Model prototipe dapat digunakan untuk menyambungkan ketidakpahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak (Rosa A.S dan M.Shalahudin, 2016:31).

Model prototipe (*prototyping model*) dimulai dari mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat. Lalu dibuatlah program prototipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi. Program ini biasanya menyediakan tampilan dengan simulasi alur perangkat lunak yang sehingga tampak seperti perangkat lunak yang sudah jadi. Program prototipe ini dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*.

Berikut adalah gambar dari model prototipe :



(Sumber : Rosa A.S dan M.Shalahuddin, 2016:32)

Gambar II.6 Ilustrasi model *prototipe*

2.2. Peralatan Pendukung

Sebuah sistem yang telah ada atau sistem baru akan dikembangkan memerlukan alat untuk untuk mempermudah mendefinisikan unsur dari suatu

sistem berikut ini dijelaskan alat pendukung berupa diagram arus data, Kamus data Normalisasi, Sebagai berikut :

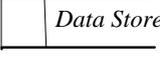
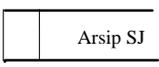
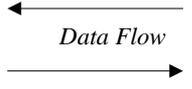
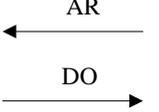
2.2.1. Data Flow Diagram

Menurut Sutabri (2012:177) *Data Flow Diagram* adalah suatu network yang menggambarkan suatu sistem automat/komputerisasi, manualisasi, atau gabungan dari keduanya yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya.

1. Simbol Data Flow Diagram (DFD)

Simbol atau lambang yang digunakan dalam membuat data flow diagram ada 4 buah, yaitu sebagai berikut :

Tabel II.1.
- Diagram Alir Data

NO	SIMBOL	KETERANGAN	CONTOH
1		<i>External Entity</i> , merupakan kesatuan dilingkungan luar sistem yang bisa berupa orang atau sistem lain.	
2		<i>Process</i> , merupakan proses seperti perhitungan aritmatik penulisan suatu formula atau pembuatan laporan.	
3		<i>Data Store</i> , dapat berupa suatu file atau database pada sistem komputer atau catatan manual.	
4		<i>Data Flow</i> , arus data ini mengalir diantara proses simpan data dan kesatuan luar.	

2. Aturan Main Data flow Diagram

Bentuk rambu-rambu atau aturan main yang baku dan berlaku dalam penggunaan data flow diagram untuk membentuk model sistem adalah sebagai berikut :

- a. Di dalam data *flow* diagram tidak boleh menghubungkan antara suatu external *entity* dengan *entity* lainnya secara langsung.
- b. Di dalam data *flow* diagram tidak boleh menghubungkan data *store* yang satu dengan yang lainnya secara langsung.
- c. Di dalam data *Flow* diagram tidak boleh menghubungkan data *store* dengan external *entity* secara langsung.
- d. Setiap proses harus ada memiliki data *flow* yang masuk dan ada data *flow* yang keluar.

3. Tehnik Membuat Data Flow Diagram

Teknik atau cara yang lazim digunakan di dalam membuat data flow diagram adalah :

- a. Mulai dari yang umum atau tingkatan yang lebih tinggi, kemudian diuraikan atau dijelaskan sampai yang lebih detail atau tingkatan yang lebih rendah, yang lebih dikenal dengan istilah *TOP-DOWN ANALYSIS*.
- b. Jabarkan proses yang terjadi di dalam data *flow* diagram sedetail mungkin sampai tidak dapat diuraikan.
- c. Peliharalah konsistensi proses yang terjadi di dalam DFD, mulai dari diagram yang tingkatannya lebih tinggi sampai dengan diagram yang tingkatannya lebih rendah.
- d. Berikan label yang bermakna untuk setiap simbol yang digunakan seperti
 - 1) Nama yang jelas untuk EXTERNAL ENTITY

- 2) Nama jelas untuk PROSES
- 3) Nama jelas untuk DATA *FLOW*
- 4) Nama yang jelas untuk DATA *STORE*

4. Tahapan Data Flow Diagram

Langkah-langkah di dalam membuat data *flow* diagram dibagi menjadi 3 (tiga) tahapan atau tingkat konstruksi DFD, yaitu sebagai berikut :

a. DIAGRAM KONTEKS

Diagram ini dibuat untuk menggambarkan sumber data serta tujuan data yang akan diproses atau dengan kata lain diagram tersebut digunakan untuk menggambarkan sistem secara umum/global dari keseluruhan.

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 data secara lebih mendetail lagi dari tahapan proses yang ada didalam diagram nol.

2.2.2 *Enterprise Relationship Diagram (ERD)*

Menurut Fathansyah (2015:81) mengemukakan bahwa: *Model Entity-Relationship* yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta dari 'dunia nyata' yang kita tinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan *Diagram Entity-Relationship* (Diagram E-R).

Sebelum kita membahas lebih jauh tentang bagaimana Diagram E-R tersebut dapat kita gambarkan, maka yang harus lebih dulu diketahui adalah komponen-komponen pembentukan *Model Entity-Relationship*. Sesuai namanya, ada 2 (dua) komponen utama pembentuk *Model Entity-Relationship*,

yaitu Entitas (*Entity*) dan Relasi (*Relation*). Kedua komponen ini didedikasikan lebih jauh melalui sejumlah Atribut atau Properti.

1. Entitas (*Entity*)

Menurut Fathansyah (2015:75) “Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain”. Sebuah kursi yang kita duduki, seseorang yang menjadi pegawai di sebuah perusahaan dan sebuah mobil yang melintas di depan kita adalah Entitas.

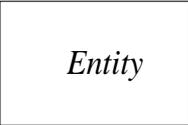
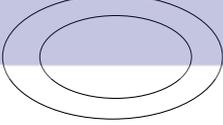
2. Atribut (*Attributes/Properties*)

Menurut Fathansyah (2015:76) “Setiap Entitas pasti memiliki Atribut yang mendeskripsikan karakteristik (properti) dari Entitas tersebut”. Sebagaimana telah disebutkan sebelumnya, penentuan atribut-atribut yang relevan bagi sebuah Entitas merupakan hal penting lainnya dalam pembentukan model data. Penetapan atribut bagi sebuah entitas umumnya memang didasarkan pada fakta yang ada. Tetapi tidak selalu seperti itu. Kelak akan kita lihat, karena proses normalisasi atau pertimbangan-pertimbangan tertentu, ada sejumlah atribut yang tidak ada di ‘dunia nyata’ tapi perlu kita tambahkan. Yang relevan untuk lebih diperhatikan dalam pembuatan Model E-R adalah kedudukan atribut dalam entitas. Harus dapat kita bedakan atau ketahui, mana atribut yang berfungsi sebagai *Key Primer (Primary Key)* dan mana yang bukan (atribut deskriptif).

3. Relasi (*Relationship*)

Menurut Fathansyah (2015:77) “Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda”.

Tabel II.2
Entity Relationship Diagram

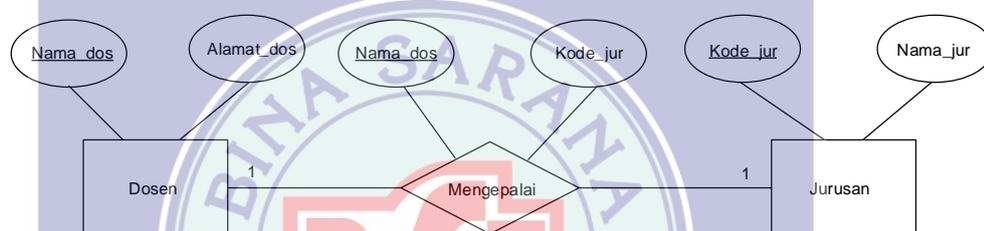
NO	SIMBOL	KETERANGAN	CONTOH
1		Suatu objek yang dapat dibedakan dengan objek lainnya dan berfungsi untuk memberikan identitas pada entitas yang memiliki label dan nama	Mahasiswa Kartu Anggota Perpustakaan (KAP)
2		Hubungan yang terjadi antara 1 entitas atau lebih yang tidak mempunyai fisik hanya sebagai konseptual dan berfungsi untuk mengetahui jenis hubungan yang ada antara file .	Relasinya adalah mendaftar
3		Karakteristik dari entitas atau relasi yang menyediakan penjelasan detail tentang entitas atau relasi tersebut dan berfungsi untuk memperjelas atribut yang dimiliki oleh sebuah entitas .	Atributnya nama mahasiswa
4		Memiliki fungsi untuk menghubungkan atribut dengan entitas dan entitas dengan relasi .	_____
5		Atribut Primary key adalah satu atau gabungan dari beberapa atribut yang dapat membedakan semua baris data(Row/record) dalam tabel secara unik.	Nomor Pokok Mahasiswa (NPM), dan NIM
6		Atribut multivalued adalah Nilai dari suatu atribut yang mempunyai lebih dari satu	Dari sebuah Buku, yaitu terdapat beberapa pengarang

Sumber : Rossa A.S. Dan M.Shalahuddin (2015:50)

Berikut adalah contoh penggambaran relasi antar himpunan entitas lengkap dengan kardinalitas relasi dan atribut-atributnya:

a. Relasi satu-ke-satu (*one-to-one*)

Adanya relasi antara himpunan entitas Dosen dengan himpunan entitas Jurusan. Himpunan relasinya kita beri nama 'Mengepalai'. Para relasi ini, setiap dosen paling banyak mengepalai satu jurusan (walaupun memang tidak semua dosen yang menjadi ketua jurusan). Dan setiap jurusan pasti dikepalai oleh paling banyak satu orang dosen. Maka penggambarannya adalah:

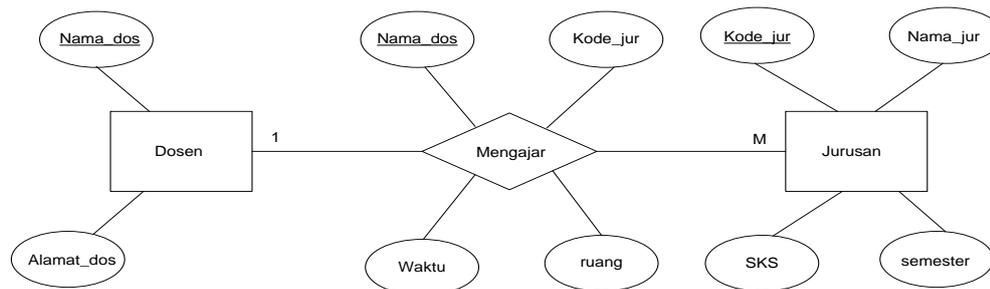


Sumber: Fathansyah, Basis Data.(2015:82)

Gambar II.7 Diagram E-R Relasi Satu Ke Satu

b. Relasi satu-ke-banyak (*one-to-many*)

Adanya relasi antara himpunan entitas Dosen dengan himpunan entitas Kuliah. Himpunan relasinya kita berinama 'Mengajar'. Pada relasi ini, setiap dosen dapat mengajar lebih dari satu mata kuliah, sedang setiap mata kuliah diajar hanya oleh paling banyak satu orang dosen. Maka penggambarannya adalah:



Sumber: Fathansyah, Basis Data.(2015:83)

Gambar II.8 Diagram E-R Relasi Satu Ke Banyak

c. Relasi banyak-ke-banyak (*many-to-many*)

Adanya relasi antara himpunan entitas Mahasiswa dengan himpunan entitas Kuliah. Himpunan relasinya kita berinama 'Mempelajari'. Pada relasi ini, setiap mahasiswa dapat mempelajari lebih dari satu mata kuliah. Demikian juga sebaliknya, setiap mata kuliah dapat dipelajari oleh lebih dari satu orang mahasiswa. Maka penggambarannya adalah:



Sumber: Fathansyah, Basis Data.(2015:84)

Gambar II.9 Diagram E-R Relasi Banyak Ke Banyak

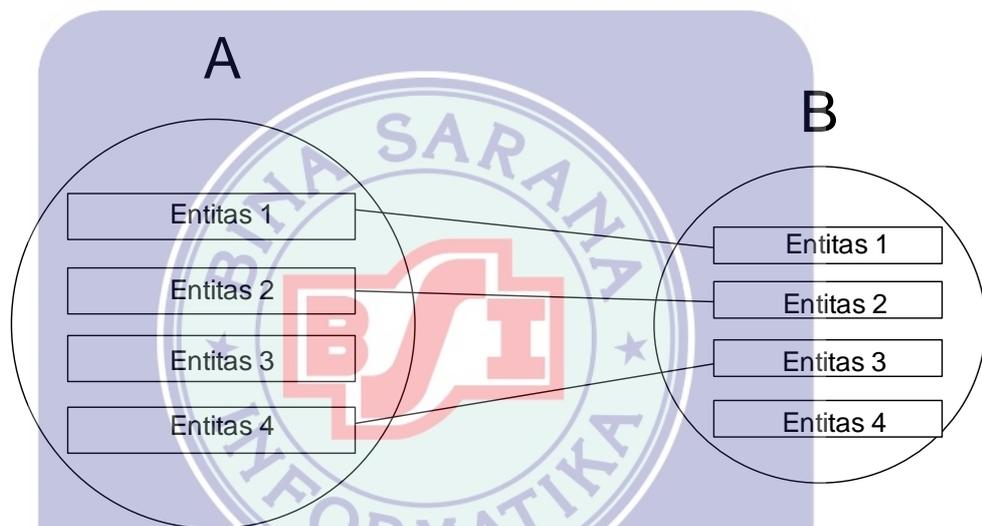
4. Kardinalitas Atau Derajat Relasi

Menurut Fathansyah (2015:78) "Kardinalitas relasi menunjukkan maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain". Kardinalitas relasi yang terjadi diantara dua himpunan entitas dapat berupa satu

ke satu (*one to one*), satu ke banyak (*one to many*), banyak ke satu (*many to one*) dan banyak ke banyak (*many to many*).

a. Satu ke Satu (*One to One*)

Berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas himpunan entitas A.

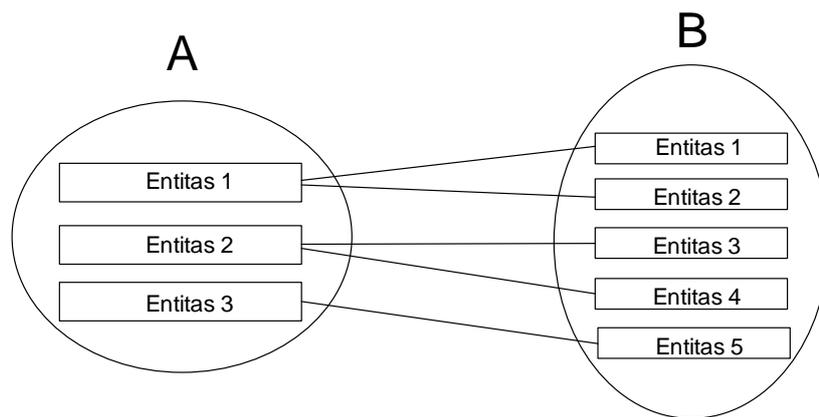


Sumber: Fathansyah, Basis Data.(2015:79)

Gambar II.10 Kardinalitas Relasi Satu Ke Satu

b. Satu ke Banyak (*One to Many*)

Yang berarti setiap entitas pada himpunan A dan dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

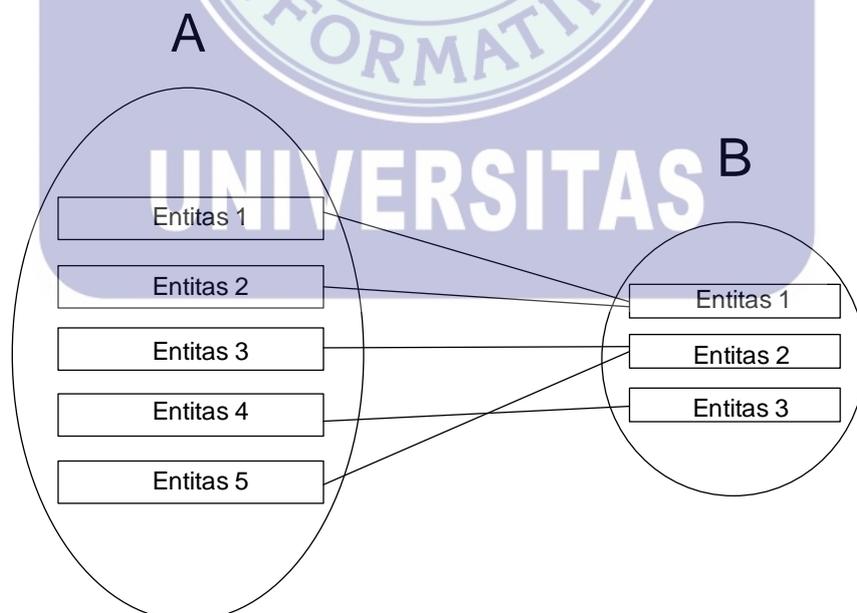


Sumber: Fathansyah, Basis Data.(2015:80)

Gambar II.11 Kardinalitas Relasi Satu Ke Banyak

c. Banyak ke Satu (*Many to One*)

Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.

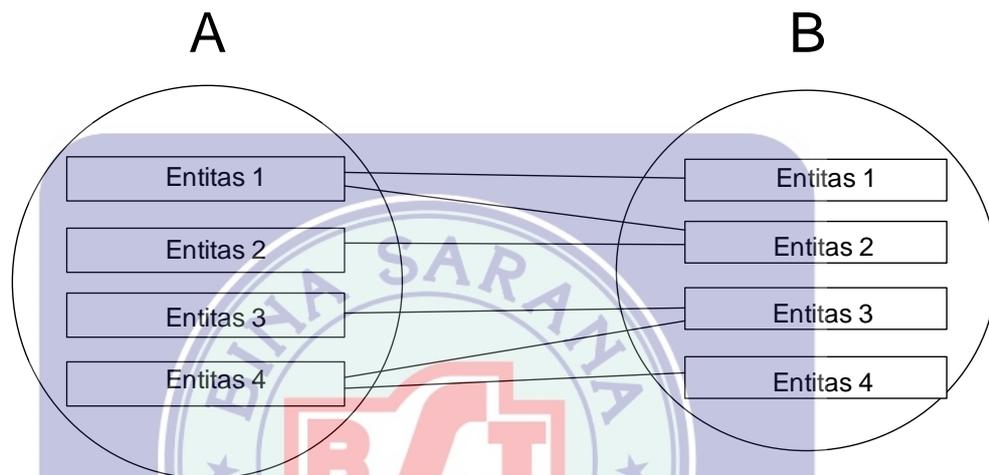


Sumber: Fathansyah, Basis Data.(2015:80)

Gambar II.12 Kardinalitas Relasi Banyak Ke Satu

d. Banyak ke Banyak (*Many to Many*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, demikian juga sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.



Sumber: Fathansyah, Basis Data.(2015:80)

Gambar II.13 Kardinalitas Relasi Banyak Ke Banyak

2.2.3. Kamus Data (*Data Dictionary*)

Menurut Shalahudin dan Rosa (2013 : 73) Kamus data adalah kumpulan daftar elemen elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluar (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).

Menurut Kristanto (2008:72) Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* di dalam sistem.

Simbol-simbol yang ada dalam kamus data adalah sebagai berikut :

Tabel II.3
Notasi Aljabar

=	Artinya adalah terdiri atas
+	Artinya adalah dan
()	Artinya adalah opsional
[]	Artinya adalah memilih salah satu alternatif
**	Artinya adalah komentar
@	Artinya adalah identifikasi atribut kunci
	Artinya adalah pemisah alternatif simbol []

Sumber : Kristanto (2008 :72)

2.2.4. Key

Field key yang digunakan :

1. *Candidate key* (kunci kandidat)

Menurut sutabri (2012:139) “Kunci kandidat adalah salah satu atribut atau satu set minimal atribut yang mengidentifikasi secara unik suatu kejadian yang spesifik dari suatu entitas.”

2. *Primary key* (kunci primer)

Menurut sutabri (2012:140) “Kunci primer adalah satu atribut atau set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian yang spesifik,akan tetapi juga dapat mewakili dari suatu entitas.”

3. *Alternate key* (kunci Alternatif)

Menurut sutabri (2012:140) “Kunci alternatif adalah kunci kandidat yang tidak terpakai sebagai *primary key*.”

4. *Forgen key* (Kunci tamu)

Menurut sutabri (2014:140) “Kunci tamu adalah satu artibut atau satu set artibut yang melengkapi satu *relationship* (hubungan) yang menunjukan induknya.”

2.2.5. Pengkodean

Menurut Fathansyah (2012:105), “Pengkodean adalah data atau informasi yang dapat dilihat oleh pemakai awam (*end user*) bisa berbeda dengan bagaimana data atau informasi itu disimpan”.

Ada 3 (tiga) bentuk pengkodean yang dapat kita pilih, yaitu:

1. Sekunsial

Dimana pengkodean dilakukan dengan mengasosiasikan data dengan kode terurut (biasanya berupa bilangan asli atau abjad), misalnya data mutu kuliah (‘Sempurna’, ‘Baik’, ‘Cukup’, ‘Kurang’, ‘Buruk’) dikodekan dengan ‘A’, ‘B’, ‘C’, ‘D’, dan ‘E’.

2. Mnemonic

Dimana pengkodean dilakukan dengan membentuk suatu singkatan dari data yang ingin dikodekan, misalnya data jenis kelamin (‘Laki-laki’ dan ‘Perempuan’) dikodekan dengan ‘L’ dan ‘P’.

3. Blok

Dimana pengkodean dinyatakan dalam format tertentu, misalnya no induk mahasiswa dengan format XXYYYY Yang terbentuk atas XX = dua dijit terakhir angka tahun masuk YYYY = no urut mahasiswa.

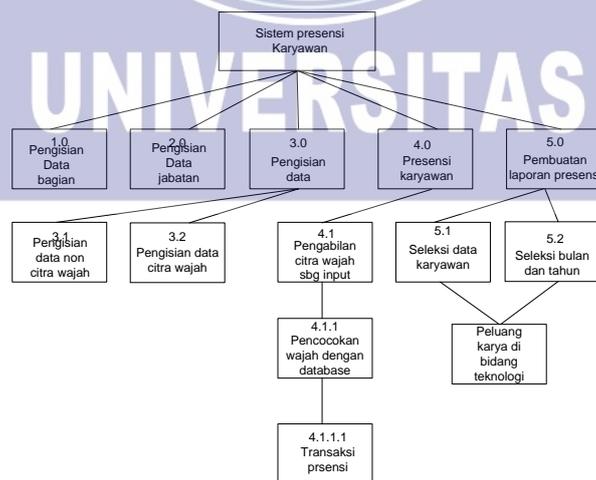
2.2.6. HIPO (*Hierarchy plus Input-Process-Output*)

Menurut Al Fatta (2007 : 147) *Hipo* adalah merupakan tehnik untuk mendokumentasikan pengembangan suatu sistem yang dikembangkan oleh IBM.

HIPO dapat digunakan untuk memenuhi kebutuhan beberapa pengguna untuk kepentingan berbeda-beda, antara lain :

- a. Seorang manajer dapat menggunakan dokumentasu HIPO untuk memperoleh gambaran umum sistem.
- b. Seorang programmer menggunakan HIPO untuk menentukan fungsi-fungsi dalam program yang dibuatnya.
- c. Programmer juga dapat menggunakan HIPO untuk mencari fungsi -fungsi yang dimodifikasi cepat.

Tehnik ini mempunyai beberapa tujuan utama. Pertama dapat dibuat sebuah struktur yang menggambarkan hubungan antra fungsi dalam program secara hierarkis. Perhatikan gambar berikut :



Sumber :Al Fatta (2007 :148)

Gambar II.14 Diagram DIV dari HIPO

1. Jenis Diagram HIPO

Paket Hipo terdiri dari 3 jenis diagram ,yaitu diagram daftar isi visual (*Visual table of content*) Diagram ringkas (*overvirw diagram*), dan diagram rinci (*detail diagram*).

a. Daftar isi Visual (DIV)

Diagram ini memuat semua modul yang ada dalam sistem berikut nama dan nomornya, yang nantinya akan diperinci dalam diagram ringkas dan diagram rinci. Dalam DIV juga bisa dilihat fungsi-fungsi utama yang menyusun sebuah sistem dan hubungan antra fungsi tersebut.

b. Diagram Ringkas

Diagram ringkas menerangkan input, proses dan output dari sistem diagram ringkas menggabarkan input dan output dari fungsi-fungsi yang telah didefinisikan dalam daftar isi visual.

c. Diagram Rinci

Diagram rinci HIPO digunakan untuk memperinci input,proses dan output yang telah digambarkan dalam diagram ringkas.