

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Konsep Dasar Sistem**

Pemahaman tentang sistem harus diketahui sebelumnya, karena mempunyai peranan penting dalam melakukan penelitian terhadap sistem yang akan diteliti serta untuk mencapai tujuan dan sasaran yang hendak dicapai dalam melakukan penelitian.

Untuk itu dengan adanya penelitian ini, penulis berencana membuat sebuah perancangan pada bagian kasir yang berbasis *desktop*. Dalam pembuatan perancangan ini, penulis menggunakan *Microsoft Visual Basic 6.0* dengan bahasa pemrograman *visual basic*.

##### **2.1.1. Pengertian Sistem**

Sekarang ini sistem banyak dibicarakan dan dipakai orang-orang untuk bagian atau komponen yang saling berkaitan untuk suatu tujuan tertentu. Menurut Pratama (2014:7) Mendefinisikan "Sistem sebagai sekumpulan prosedur yang saling berkaitan dan saling berhubungan untuk melakukan suatu tugas bersama-sama".

Menurut Rosa dan Shalahuddin (2015:18) Mendefinisikan "Sistem adalah kegiatan untuk melihat sistem yang sudah berjalan, melihat bagai mana yang bagus dan tidak bagus, dan kemudian mendokumentasikan kebutuhan yang akan dipenuhi dalam sistem yang baru".

Kesimpulan sistem menurut penulis adalah suatu rangkaian tugas yang telah terorganisir dengan baik yang dapat dilakukan secara bersamaan.

## A. Karakteristik Sistem

Sesuai definisi yang dikemukakan diatas, suatu sistem merupakan kumpulan dari komponen-komponen yang saling terstruktur dan terpadu serta saling berkerjasama untuk melakukan fungsi dari sistem sehingga adanya ketercapaian tujuan dari sistem. Dengan demikian, suatu sistem mempunyai karakteristik atau sifat-sifat tertentu.

Sedangkan menurut Sutabri (2012:20) mendefinisikan bahwa “Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bias dikatakan sebagai suatu sistem”. Adapun karakteristik yang dimaksud adalah sebagai berikut(Sutabri, 2012:20):

1. Komponen Sistem (*Component System*)

Dimana suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan.

2. Batas Sistem (*Boundary*).

Merupakan pembatas atau pemisah suatu sistem dengan sistem lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem (*Environment*).

Merupakan sesuatu di luar batas sistem dari sistem yang dapat mempengaruhi operasi sistem, baik yang bersifat merugikan atau menguntungkan sistem tersebut.

4. Penghubung Sistem (*Interface*).

Penghubung juga dapat didefinisikan sebagai tempat dimana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.

5. Masukan Sistem (*Input*).

Merupakan energi yang dimasukkan ke dalam sistem, dimana masukan tersebut dapat berupa bahan yang dimasukkan agar sistem tersebut beroperasi (*maintenanceinput*) dan masukan yang diproses untuk mendapat keluaran (*signalinput*).

6. Keluaran (*Output*).

Merupakan hasil dari pemrosesan yang berupa informasi sebagai masukan pada sistem lain atau hanya sebagai sisa pembuangan.

7. Pengolahan sistem (*Process*).

Merupakan bagian yang melakukan perubahan dari masukan untuk menjadi keluaran yang diinginkan.

8. Sasaran (*Objectives*).

Sasaran sistem merupakan suatu tujuan yang ingin dicapai dalam sistem agar sistem menjadi terarah dan terkendali.

## **B. Klasifikasi Sistem**

Sistem merupakan suatu bentuk integrasi antara satu komponen dan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang.

Menurut Sutabri (2012:22) mengatakan bahwa “Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen yang lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi didalam sistem tersebut”. Adapun klasifikasi dari beberapa sudut pandang diantara lainnya (Sutabri, 2012:22):

1. Sistem abstrak (*abstract system*) dan sistem fisik (*physical system*).

Sistem abstrak adalah sistem berupa pemikiran atau gagasan yang tidak tampak secara fisik. Misalnya sistem agama. Sistem fisik merupakan sistem yang ada secara fisik dan dapat dilihat dengan mata. Misalnya, sistem komputer, sistem akuntansi, sistem transportasi.

2. Sistem alamiah (*natural system*) dan sistem buatan manusia (*human made system*).

Sistem alamiah adalah sistem yang terjadi karena proses alam, bukan buatan manusia. Misalnya, sistem tata surya, sistem rotasi bumi. Sistem buatan manusia adalah sistem yang terjadi melalui rancangan atau campur tangan manusia. Misalnya, sistem komputer, sistem transportasi.

3. Sistem tertentu (*deterministic system*) dan sistem tak tentu (*probabilistic system*).

Sistem tertentu adalah sistem yang operasinya dapat diprediksi secara cepat dan interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti. Misalnya, sistem komputer karena operasinya dapat diprediksi berdasarkan program yang dijalankan. Sistem tak tentu adalah sistem yang hasilnya tidak dapat diprediksi karena mengandung unsur probabilitas. Misalnya, sistem persediaan.

4. Sistem tertutup (*closed system*) dan sistem terbuka (*open system*).

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sebenarnya sistem tertutup tidak ada, yang ada adalah relatif tertutup. Sebaliknya sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan

luarnya. Sistem ini menerima masukan dan menghasilkan *output* untuk subsistem yang lain.

### **2.1.2. Pengertian Sistem Informasi**

Menurut Mcleod & Schell (2008:10), “Sistem Informasi adalah suatu sistem virtual yang memungkinkan manajemen mengendalikan operasi system fisik perusahaan”.

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan (Hutahaean, 2015:13).

Komponen-komponen dalam sistem informasi menurut Hutahaean (2015:13) terdiri dari:

1. Blok masukan (*Input Block*)

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dasar.

2. Blok model (*Model Block*)

Blok ini terdiri dari kombinasi prosedur, logika dan metode matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang sudah diinginkan.

### 3. Blok keluaran (*Output Block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

### 4. Blok teknologi (*Technologi Block*)

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian diri secara keseluruhan. Teknologi terdiri dari unsur utama:

- a. Teknisi (*human ware* atau *brain ware*)
- b. Perangkat lunak (*software*)
- c. Perangkat keras (*hardware*)

### 5. Blok basis data (*Database Block*)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

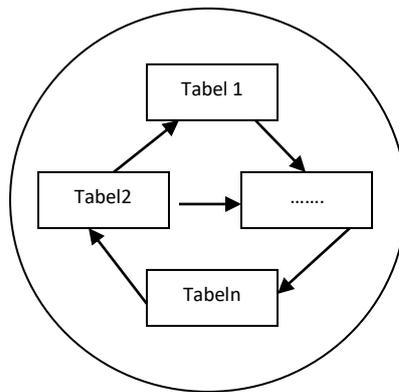
### 6. Blok kendali (*Control Block*)

Banyak faktor yang dapat merusak sistem informasi, misalnya bencana alam, api, temperature tinggi, air, debu, kecurangan-kecurangan, kegagalan sistem itu sendiri, kesalahan-kesalahan ketidakefisienan, sabotase dan sebagainya.

## **2.1.2. Basis Data**

Dalam pembuatan suatu aplikasi sangat dibutuhkan rancangan struktur data/*database* yang nantinya berfungsi untuk menentukan tabel-tabel yang akan kita buat.

Menurut Priyadi (2014:2) mengemukakan “Basis data adalah sekumpulan fakta berupa representasi tabel yang saling berhubungan dan disimpan dalam media penyimpanan secara digital”. Sedangkan menurut Shelly dan Vermant (2012:514) “Basis data adalah kumpulan data yang ditata dengan cara yang memungkinkan diakses, dicari, dan digunakan datanya”.



Sumber : Rosa dan Shalahudin (2015:44)

### Gambar II.I Ilustrasi Data

*MySQL* adalah *database management system* (DBMS) yang digunakan dalam penyimpanan data.

*MySQL* menurut Raharjo (2012:21) “*Software RDBMS* (atau *server database*) yang dapat mengolah *database* dengan cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user* (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi threaded*)”.

Berdasarkan kutipan diatas, dapat disimpulkan bahwa basis data adalah suatu penyimpanan data yang dapat digunakan untuk menampung data-data yang ada pada komputer.

### 2.1.3. Model Pengembangan Perangkat Lunak

Model *waterfall* sangat dibutuhkan bagi seorang *programmer* yang dimana nantinya model ini bisa membantu dalam penyelesaian suatu program yang sudah dibuat.

Model (*waterfall*) sering juga disebut model sekuensial (*sequential linier*) 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, dan tahap pendukung (*support*). Menurut (Rosa dan Shalahudin, 2015:28), model *waterfall* memiliki 5 tahapan, yaitu:

1. Analisis kebutuhan perangkat lunak

Pada tahap ini penulis melakukan pengamatan dan wawancara secara langsung kepada karyawan untuk mendapatkan keterangan mengenai aplikasi perangkat lunak yang akan dibangun pada del market. Analisa ini dilakukan untuk menciptakan sebuah aplikasi yang sesuai dengan yang diharapkan oleh *user*. Pada proses ini pengumpulan kebutuhan dilakukan secara intensif agar dapat dipahami.

2. Desain

Pada tahap ini desain dilakukan untuk menggambarkan bentuk dari aplikasi yang akan dibangun. Desain ini dibuat setelah penulis mendapatkan hasil dari menganalisis kebutuhan perangkat lunak yang diinginkan oleh pihak del market Pontianak.

3. Pembuatan kode program

Desain yang telah digambarkan penulis harus dibuatkan kode program agar program yang dibangun dapat dipergunakan oleh pihak del market

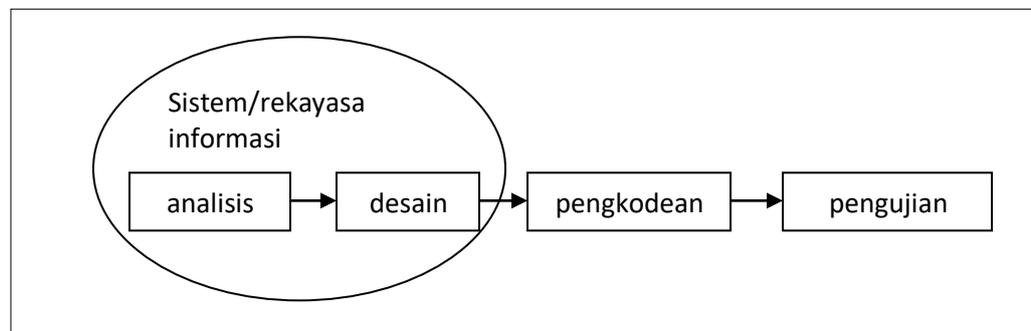
Pontianak. Pembuatan kode program merupakan langkah ketiga yang dilakukan untuk menghasilkan sebuah program aplikasi pengolahan data kasir pada del market Pontianak menggunakan bahasa pemrograman Visual basic 6.0.

#### 4. Pengujian

Pengujian merupakan tahapan dimana program yang akan dibuat harus melewati pengujian kelayakan. Pengujian ini dimaksudkan agar tercipta sebuah program aplikasi yang terhindar dari kesalahan (*error*). Untuk melakukan pengujian pada tahap ini, penulis menggunakan *flowchart* sebagai pengujian alur kerja program aplikasi yang dibangun.

#### 5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tahapan pendukung ataupun pemeliharaan ini merupakan tahapan terakhir dalam membangun sebuah aplikasi. Tahapan ini dimaksudkan sebagai pemeliharaan dari aplikasi yang telah dibangun. Tujuannya adalah supaya aplikasi yang dibangun selalu mengalami peningkatan dalam segi kualitas.



Sumber: Rosa dan Shalahudin (2015:28)

**Gambar II.2 Model *waterfall***

#### **2.1.4. Pengertian Judul**

Penjualan Menurut Rangkuti (2009:206) adalah, "Pemindahan hak milik atas barang atau pemberian jasa yang dilakukan penjualan kepada pembeli dengan harga yang disepakati bersama dengan jumlah yang dibebankan kepada pelanggan dalam penjualan barang/jasa dalam suatu periode".

Lia mulyani (2014:18) mengemukakan, "Transaksi adalah segala aktivitas perusahaan yang menimbulkan perubahan terhadap posisi harta keuangan perusahaan,

Seperti menjual, membeli, membayar gaji, serta membayar biaya-biaya lainnya.

Menurut Hutahaean (2015:2), "Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran tertentu.

Menurut Arif (2016:2), "Perancangan menggambarkan rencana umum suatu kegiatan rancangan proyek aktivitas-aktivitas khusus yaitu teknik atau metode-metode dalam merancang sesuatu".

### **2.2.Peralatan Pendukung (*Tool System*)**

Adapun peralatan pendukung (*tool system*) yang digunakan dalam perancangansistem ini dijelaskan dalam sub-sub bab berikut:

#### **2.2.1. Diagram Alir Data (DAD)**

DAD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan tersimpan.

## A. Konsep Dasar DAD

Menurut Rosa dan Shalahuddin (2015:70) “Diagram Alir Data (DAD) representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

Sedangkan menurut Pratama (2014:49) “Diagram Alir data (DAD) merupakan diagram permodelan suatu perangkat lunak, yang mana didalamnya terdapat sejumlah notasi dengan aliran-aliran data”.

Berikut ini adapun berdasarkan simbol menurut Rosa dan Shalahuddin (2015:69) dari diagram aliran data yaitu:

### 1. *Ekternal Entity*

Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima input dan menghasilkan output kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada dilingkungan luarnya yang akan memberikan input atau menerima output dari sistem. Satu kesatuan luar dapat disimbolkan dengan suatu notasi kotak sebagai berikut:



Sumber :Rosa dan Shalahuddin (2015:69)

**Gambar II.3 Simbol *External Entity* pada DAD**

## 2. *Data Flow*

Simbol ini digunakan untuk menggambarkan data yang berjalan, menunjukkan arus data yang berupa masukan untuk sistem atau hasil dari proses sistem yang mengalir diantara proses (*process*), simpanan data (*data store*) dan entitas (*external entity*) digambarkan dengan arah panah.

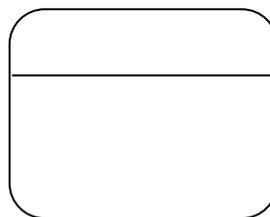


**Sumber** :Rosa dan Shalahuddin (2015:69)

**Gambar II.4 Simbol *Data Flow* pada DAD**

## 3. *Proses*

Simbol ini digunakan untuk proses pengolahan atau transformasi data, menunjukkan kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dan hasil suatu data yang masuk kedalam proses untuk menghasilkan arus data yang akan keluar dari proses, Suatu proses dapat ditunjukkan dengan simbol lingkaran atau dengan simbol empat persegi panjang tegak dengan sudut-sudutnya tumpul.



**Sumber** :Rosa dan Shalahuddin (2015:69)

**Gambar II.5 Simbol *Proses* pada DAD**

## 4. *Data Store*

Simbol ini digunakan untuk menggambarkan *Data Flow* yang sudah disimpan, menunjukan suatu tempat penyimpanan data yang dapat berupa

suatu file di sistem komputer, arsip atau catatan manual, tabel acuan dan lain-lain Simpanan data diDAD dapat disimbolkan dengan sepasang garis horizontal parallel yang tertutup disalah satu ujungnya.



Sumber :Rosa dan Shalahuddin (2015:69)

**Gambar II.6 Simbol *Data Store* pada DAD**

### **B. Aturan Main Diagram Alir Data**

Menurut Ladjamudin (2013:75) Adapun beberapa hal aturan main DAD atau larangan yang harus diperhatikan dalam menggambarkan atau mendesain DAD, agar dapat menggambarkan keseluruhan sistem yang dirancang.

1. Arus data tidak boleh dari entitas luar langsung menuju ke entitas luar lainnya. Tanpa memalui suatu proses.
2. Arus data tidak boleh dari simpanan data langsung menuju ke entitas luar, tanpa melalui suatu proses.
3. Arus data tidak boleh dari ke simpan data langsung menuju kesimpanan data lainnya, tanpa memlaui proses.
4. Arus data dari satu proses langsung menuju proses lainnya, tanpa melalui suatu simpanan data, sebaiknya/sebisa mungkin hindari.

#### **2.2.2. Jenis DAD**

Tahapan adalah urutan pelaksanaan atau kejadian yang terjadi secara alami atau desain, mungkin menggunakan waktu, ruang, keahlian atau sumber daya lainnya, yang menghasilkan suatu hasil. Menurut Rosa dan Shalahudin (2015:70) terdiri dari beberapa jenis dalam membuat DAD dibagi menjadi tiga, yaitu:

a. Diagram Konteks

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

b. Diagram Nol

Diagram ini dibuat untuk menggambarkan tahapan proses yang ada dalam diagram konteks yang penggambarannya secara lebih terperinci.

c. Diagram Detail

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

### 2.2.3. Kamus Data

Kamus data merupakan sebuah katalog fakta tentang sebuah data untuk sebuah keperluan informasi dari suatu sistem.

Menurut Rosa dan Shalahuddin (2015:73) “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).

Sedangkan menurut Ladjamudin (2013:70) “Kamus Data adalah catalog yang menjelaskan lebih detail tentang Diagram Alir Data yang mencakup proses, arus data tersimpan”.

Isi data harus mencerminkan keterangan yang jelas tentang data yang dicatat. Maka data harus memuat hal-hal sebagai berikut (Ladjamudin, 2013:70):

## 1. Hal-hal yang dimuat dalam kamus data

### a. Nama Arus Data

Karena kamus data dibuat berdasarkan arus data yang mengalir di diagram alir data, maka nama arus data juga harus dicatat di kamus data, sehingga mereka yang membaca diagram alir data dan memerlukan penjelasan lebih lanjut tentang suatu arus data tertentu di diagram alir data dapat langsung mencarinya dengan mudah di kamus data.

### b. Alias

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

### c. Bentuk Data

Data yang mengalir dari hasil suatu proses ke proses lainnya dalam bentuk dokumen dasar atau formulir, dokumen hasil cetakan komputer, laporan terarah, tampilan layar dimonitor, variabel, parameter dan *field-field* dalam bentuk data adalah bentuk data dari arus data yang mengalir yang perlu dicatat di kamus data.

### d. Arus Data

Arus data menunjukkan dari mana data mengalir dan kemana data akan menuju. Keterangan arus data ini perlu dicatat di kamus data supaya memudahkan mencari arus data di dalam *data flow diagram*.

### e. Periode

Periode ini menunjukkan kapan terjadinya arus data ini. Periode ini perlu dicatat di kamus data karena dapat digunakan untuk mendefinisikan kapan

input data harus dimasukkan ke dalam sistem, kapan proses program harus dilakukan dan kapan laporan-laporan harus dihasilkan.

f. Volume

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

g. Struktur Data

Struktur data menunjukkan arus data yang dicatat pada kamus data yang terdiri dari item-item atau elemen-elemen data.

h. Penjelasan

Untuk lebih memperjelas lagi tentang makna dari arus data yang dicatat di kamus data, maka bagian penjelasan dapat diisi dengan keterangan-keterangan tentang arus data tersebut.

2. Notasi Tipe Data

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

3. Notasi Struktur Data

Notasi ini untuk membuat spesifikasi elemen data.

**Tabel II.1**

**Simbol Kamus Data**

<b>SIMBOL</b>	<b>KETERANGAN</b>
=	Dimana atau terdiri dari
+	Dan
[[]]	Baik...atau

{ }n	n kali diulang/bernilai banyak
()	Data opsional
*...*	Batas komentar

Sumber: Rosa dan Shalahuddin (2015:74)

#### 2.2.4. *Enterprise Relationship Diagram (ERD)*

Sering kita ketahui bahwa ERD atau biasa disebut dengan *entity relationship diagram* merupakan aplikasi pendukung yang nantinya membantu dalam proses pembuatan kode program.

##### A. Definisi ERD

Menurut Marimin, dkk (2006:111) “*Enterprise Relationship Diagram (ERD)* merupakan alat yang digunakan dalam menggambarkan kebutuhan data dan asumsi-asumsi dalam sistem yang akan dibangun atau dikembangkan secara terstruktur dari bawah”. Sedangkan menurut Fatta (2007:121) “ERD adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan dan digunakan dalam sistem basis”.

Sedangkan menurut Raharjo (2011:11) *Entity Relationship Diagram (ERD)* merupakan salah satu alat bantu (berupa gambar) dalam model *database* relasional yang berguna untuk menjelaskan hubungan atau relasi antar *table* yang terdapat didalam *database*”.

Berdasarkan kutipan diatas, dapat disimpulkan bahwa ERD merupakan alat bantu dalam proses pembuatan *database* dan juga berfungsi untuk merelasikan sebuah *table*.

##### B. Komponen ERD

Adapun komponen *Entity Relationship Diagram (ERD)* menurut Simarmata dan Prayudi (2010:67), yaitu:

### 1. Entitas(*Entity*)

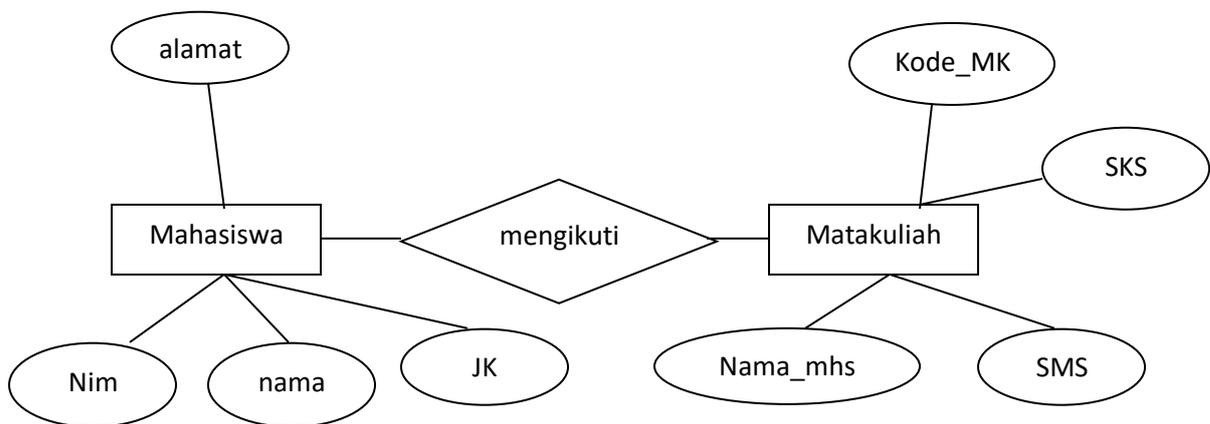
Entitas adalah sesuatu yang nyata atau abstrak dimana kita akan menyimpan data

### 2. Relasi (*Relationship*)

Relasi adalah hubungan alamiah yang terjadi antara suatu atau lebih entitas, misalnya proses pembayaran pegawai.

### 3. Atribut (*Attribute*)

Atribut adalah ciri umum semua atau sebagian instansi pada entitas tertentu.



Sumber: Supardi (2013:15)

**Gambar II.7 Enterprise Relationship Diagram**

#### 2.2.5. Logical Record Structure(LRS)

LRS pada umumnya digunakan *programmer* untuk membuat sebuah aturan-aturan yang diterapkan ke dalam *database*. Proses LRS itu sendiri membantu kinerja *programmer* dalam menyelesaikan suatu aplikasi.

Menurut Hasugian dan Shidiq, (2012:608) mengungkapkan: Sebuah model yang digambarkan dengan sebuah diagram-ER akan mengikuti pola/aturan permodelan tertentu dalam kaitannya dalam konversi ke LRS, maka perubahan yang terjadi adalah mengikuti aturan-aturan berikut: setiap entitas yang dirubah ke bentuk kotak, sebuah atribut relasi disatukan dalam sebuah kotak bersama entitas jika hubungan yang terjadi pada diagram-ER.

Aturan-aturan dalam melakukan transformasi E-R diagram ke *Logical RecordStructure* menurut Ladjamudin (2013:159) adalah sebagai berikut:

1. Setiap *entity* akan diubah ke bentuk sebuah kotak dengan nama *entity* berada diluar kotak dan atribut berada didalam kotak.
2. Sebuah relasi kadang disatukan dalam sebuah kotak bersama *entity*, kadang dipisah dalam sebuah tersendiri.

Aturan pokok diatas akan sangat dipengaruhi oleh elemen yang menjadi titikperhatian utama pada langkah transformasi yaitu *cardinality*/kardinalitas menurutLadjamudin (2013:159) adalah sebagai berikut:

1. 1:1 (*one to one* )

Pada kardinalitas *one to one*, sebaiknya pernah diarahkan ke *entity* dengan jumlahatribut yang lebih sedikit.

2. 1:M (*one to many*)

Pada kardinalitas relasi *one to many*, maka relasi harus dibangun dengan *entity* pada pihak ini yang *many*, dan tidak melihat banyak sedikitnya atribut pada *entity* tersebut.

3. M:N (*Many to many*)

Pada kardinalitas *Many to many*, maka *relationship* berubah status menjadi *file* konektor (yang akan berubah kardinalitas *many to many* seolah-olah menjadi *one to many*), sehingga baik *entity* maupun relasi akan menjadi struktur *record* tersendiri.

### 2.2.6. Pengkodean

Dalam proses pembuatan suatu program kita sering mendengar tentang kata pengkodean yang dimana pengkodean ini merupakan *script* yang membantu

dalam pembuatan program, karena tanpa *script* kita tidak akan bisa membuat program.

Pengkodean (*coding*) merupakan Supranto (2007:82), “suatu kegiatan pemberian kode atau simbol pada keterangan-keterangan tertentu, kalau pengolahan dilakukan dengan komputer elektronik”.

Menurut Kusrini dan Koniyo (2007:23) ada beberapa macam tipe kode dalam kode akun meliputi:

1. Kode Numerical

Adalah cara pengkodean akun berdasarkan nomor urut yang dapat di mulai dari angka 1, 2, 3 dan selanjutnya.

2. Kode Desimal

Adalah cara pemberian kode akun dengan menggunakan lebih dari satu angka. Setiap angka mempunyai makna atau karakter sendiri. Kode desimal dapat di bedakan atas kode kelompok, dan kode stelsel akun decimal.

3. Kode Mnemonik

Adalah cara pengkodean akun dengan menggunakan huruf tertentu

4. Kode Akun dengan Sistem Kombinasi Huruf dan angka

Adalah dengan cara pengkodean dengan kombinasi antara huruf dan angka.

**2.2.7. Hieracky Plus Input-Proses-Output (HIPO)**

Diagram HIPO ini sangat membantu dalam prosedur pembuatan program. Diagram ini merupakan susunan alur aplikasi dalam proses awalnya program sampai proses program berakhir.

Menurut Hanif (2007:150) “HIPO adalah Teknik penggambaran modul-modul yang nantinya akan dikembangkan oleh *programmer* menjadi prosedur-prosedur dalam program sistem informasi”.

*Hieracky Plus Input-Proses-Output* (HIPO) menurut Ladjamudin (2013:213), “merupakan teknik untuk mendokumentasikan sistem pemrograman”.

Berdasarkan kutipan diatas, dapat disimpulkan bahwa HIPO adalah prosedur-prosedur yang nantinya dikembangkan seorang *programmer* supaya bisa menjadi program baku.

Tingkat digram HIPO terdiri dari tiga jenis diagram menurut Ladjamudin (2013:204), yaitu:

#### 1. Daftar Isi Visual (DIV)

DIV merupakan diagram pertama HIPO yang terdiri dari satu atau lebih diagram hirarki.

#### 2. Diagram Ringkas

Diagram ringkas ini merupakan diagram kedua dari paket HIPO yang menjelaskan fungsi dan refrensi utama yang diperlukan dalam program detail untuk memperluaskan fungsi sehingga cukup rinci.

#### 3. Diagram Rinci

Diagram rinci HIPO berisikan elemen-elemen dasar sistem, menerangkan fungsi-fungsi khusus, menampilkan item-item input dan output. Secara rinci (yaitu nama *field* input yang digunakan dan output yng dihasilkan), dan memberikan referensi terhadap diagram HIPO yang lain seperti *flowchart* dan *table* keputusan dari logika yang rumit.

### 2.2.8. Crystal Report

Setiap aplikasi sangatlah penting untuk membuat proses akhir atau biasa disebut dengan cetak. Proses cetak disini menggunakan *crystal report* akhir atau biasa dikoneksikan terlebih dahulu kedalam *visual basic*.

Menurut MADCOM (2010:10) mengemukakan “*Crystal Report* merupakan Program yang terpisah dengan program *Microsoft Visual Basic6.0*, tetapi keduanya dapat dihubungkan (*linkage*). Membuat laporan dengan *crystal report* hasilnya lebih baik dan lebih mudah, karena pada *Crystal Report* banyak tersedia objek-objek maupun komponen yang mudah digunakan”.

Berdasarkan kutipan diatas, dapat disimpulkan bahwa *crystal report* adalah proses membuat laporan supaya lebih baik dan mudah.