



## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Teori Umum

##### 2.1.1. Data

Menurut Rusdian dan Irfan (2014:68), “Data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih memerlukan adanya pengolahan”.

Selain itu juga menurut Susanto (dalam Rusdiana dan Irfan, 2014:68), “Data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih memerlukan adanya pengolahan”.

Berdasarkan kedua pendapat di atas, penulis menyimpulkan bahwa data adalah sesuatu yang belum mempunyai arti, dapat berupa benda, kejadian dan lainnya yang selanjutnya dibutuhkan pengolahan untuk menghasilkan informasi.

##### 2.1.2. Informasi

Menurut Hutahaean (2015:9), “Informasi adalah data yang diolah menjadi bentuk yang lebih berarti bagi penerimanya”.

Sedangkan menurut Sidharta (dalam Rusdiana dan Irfan, 2014:75), “Informasi adalah data yang disajikan dalam bentuk yang berguna untuk membuat keputusan”.

Berdasarkan kedua pendapat di atas, penulis menyimpulkan bahwa Informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang.

##### 2.1.3. Perangkat Lunak

Menurut Sukamto dan Shalahuddin (2018:2), “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)”.



Selain itu menurut Kristanto (2018:1), “Perangkat lunak (*software*) adalah instruksi (program komputer) yang ketika dijalankan menyediakan fungsi dan tampilan yang diinginkan, struktur data yang memberi kesempatan program untuk memanipulasi informasi dan dokumen yang mendeskripsikan operasi dan penggunaan program”.

Berdasarkan kedua kutipan di atas penulis menyimpulkan bahwa, perangkat lunak merupakan program komputer berupa perintah yang menghasilkan fungsi serta tampilan dan terhubung dengan file-file perangkat lunak seperti kebutuhan, desain, serta cara penggunaannya.

#### **2.1.4. Basis Data**

Menurut Hidayatullah dan Kawistara (2017:142), “Basis Data adalah himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah”.

Menurut Rusdiana dan Irfan (2014:302), “*Database* adalah kumpulan informasi yang disimpan dalam komputer secara sistematis untuk memperoleh informasi dari basis data”.

Jadi penulis menyimpulkan bahwa Basis Data (*Database*) adalah himpunan kelompok data yang saling berhubungan yang disimpan di dalam komputer secara sistematis sehingga dapat dimanfaatkan kembali dengan cepat dan mudah untuk memperoleh informasi basis data.

#### **2.1.5. Internet**

Menurut Hidayatullah dan Kawistara (2017:1), “Internet adalah jaringan global yang menghubungkan komputer-komputer di seluruh dunia”,

Menurut Ahmadi dan Hermawan (2013:68), “Internet adalah komunikasi jaringan komunikasi global yang menghubungkan seluruh di dunia meskipun berbeda sistem komputer dan mesin”.



Jadi penulis menyimpulkan pengertian Internet adalah jaringan komunikasi global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem komputer dan mesin.

## **2.2. Teori Judul**

### **2.2.1. Aplikasi**

Pengertian aplikasi dikemukakan oleh para ahli, menurut Santoso (2015:9), “Aplikasi adalah suatu kelompok *file (form, class, report)* yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi *Payroll*, aplikasi *Fixed Asset*”.

Menurut Asropudin (2013:19) “Aplikasi adalah *softwarei* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Microsoft Excel* dan *Microsoft Word*”.

Menurut Budiharto (2013:5) “Aplikasi merupakan program yang dapat berjalan di komputer tersendiri, dari mulai program yang simpel sampai dengan program besar dan rumit. Aplikasi bisa berjalan lewat *command line* atau GUI. Aplikasi juga dapat menggunakan antarmuka untuk koneksi ke jaringan, *database*, atau aplikasi lain”.

Jadi yang dapat penulis simpulkan dari pernyataan-pernyataan tersebut yaitu aplikasi adalah sebuah program yang dibuat khusus yang tersimpan dalam memori masing-masing komputer untuk tujuan tertentu yang mempunyai kemampuan dan fungsi tersendiri.

### **2.2.2. e-Reporting**

Proses pelaporan yang dilakukan secara manual atau tidak terprogram bukan hanya akan menghabiskan lebih banyak waktu dalam proses pelaporannya, akan tetapi juga dapat mengakibatkan kesalahan (*error*). Data-data dari laporan yang dibuat akan di-*input* ulang secara manual ke dalam komputer menggunakan *software* sehingga dapat mengakibatkan keterlambatan dalam penyampaian informasi secara cepat dan tepat waktu. Maka dari itu pengolahan dan penyajian data menjadi informasi yang akurat dan tepat waktu merupakan hal yang sangat

---



penting. Dalam upaya untuk meningkatkan kinerja baik dalam melakukan proses pelaporan, keberadaan pelaporan secara elektronik sangat dibutuhkan.

Menurut Baskoro (2013:23), “*e-Reporting* atau sering dikenal dengan sistem pelaporan dan penyajian data secara elektronik merupakan pola penyampaian data elektronik kepada publik melalui akses internet.” Sedangkan menurut Firmansyah (2013:24), “*E-Reporting* adalah suatu pelaporan yang disampaikan secara elektronik menggunakan media elektronik.... Media elektronik yang dipakai untuk mengakses pelaporan ini berupa komputer dengan menggunakan jaringan internet.”

Dari pernyataan-pernyataan di atas, penulis menyimpulkan bahwa *e-reporting* adalah suatu proses pelaporan informasi secara elektronik menggunakan media elektronik komputer yang tersambung dengan jaringan komputer untuk mempermudah proses pelaporan informasi dan menjadikannya lebih akurat dan tepat waktu.

### **2.2.3. *e-Monitoring***

*Monitoring* perkembangan dalam suatu proyek merupakan suatu faktor penting keberhasilan proyek yang dilakukan tersebut. Namun bila teknologi diterapkan pada sistem pelaporan, hal tersebut dikenal dengan istilah *e-monitoring*. Menurut Afriansyah, dkk (2015:7), “*E-monitoring* merupakan pemantauan dan pelaporan dengan penyampaian data secara elektronik (*online*) serta dapat dipantau serta dapat dipantau terus menerus untuk penilaian kualitas dan efektivitas sistem pengendalian untuk meyakinkan bahwa pengendalian telah berjalan sebagaimana yang diharapkan dan diperbaiki sesuai dengan kebutuhan.”

Berdasarkan pernyataan tersebut penulis menyimpulkan bahwa *e-monitoring* adalah suatu kegiatan pemantauan yang dilakukan secara elektronik (*online*) sehingga dapat dilakukan terus menerus yang bersifat efektif dan efisien. Sebagaimana diketahui proses pemantauan merupakan salah satu faktor penting keberhasilan suatu proyek.

---



#### **2.2.4. Jalan**

Jalan lingkungan merupakan jalan umum yang berfungsi melayani angkutan lingkungan dengan ciri perjalanan jarak dekat dan kecepatan rata-rata rendah. Seperti yang tertera pada Peraturan Pemerintah Republik Indonesia Nomor 34 Tahun 2006 Pasal 1 tentang Jalan, “Jalan adalah prasarana transportasi darat yang meliputi segala bagian jalan, termasuk bangunan pelengkap, dan perlengkapannya yang diperuntukkan bagi lalu lintas, yang berada permukaan tanah, di atas permukaan tanah dan atau air, serta di atas permukaan air, kecuali jalan kereta api dan jalan kabel.”

Sedangkan di dalam Undang-Undang Republik Indonesia Nomor 38 Tahun 2004 tentang Jalan, “Jalan adalah prasarana transportasi darat yang meliputi segala bagian jalan, termasuk bangunan pelengkap dan perlengkapannya yang digunakan untuk lalu lintas, yang berada pada permukaan tanah, di bawah permukaan tanah, dan atau di bawah permukaan air, kecuali jalan kereta api, jalan lori, dan jalan kabel.”

Kemudian menurut Pasal 8 Undang Undang Republik Indonesia No. 38 Tahun 2004, jalan menurut fungsinya dikelompokkan ke dalam jalan arteri, jalan kolektor, jalan lokal, dan jalan lingkungan, dengan perincian sebagai berikut:

1. Jalan arteri merupakan jalan umum yang berfungsi melayani angkutan utama dengan ciri perjalanan jarak jauh, kecepatan rata-rata tinggi, dan jumlah jalan masuk dibatasi secara berdaya guna.
  2. Jalan kolektor merupakan jalan umum yang berfungsi melayani angkutan pengumpul atau pembagi dengan ciri perjalanan jarak sedang, kecepatan rata-rata sedang, dan jumlah jalan masuk dibatasi.
  3. Jalan lokal merupakan jalan umum yang berfungsi melayani angkutan setempat dengan ciri perjalanan jarak dekat, kecepatan rata-rata rendah, dan jumlah jalan masuk tidak dibatasi.
-



### 2.2.5. Kerusakan Jalan

Menurut Fauzan dan Triadi (2013:4), “Kerusakan Jalan merupakan kondisi jalan yang kurang baik dapat berupa retak-retak (*cracking*), berupa gelombang (*corrugation*), juga kerusakan berupa alur/cekungan arah memanjang jalan sekitar jejak roda kendaraan (*rutting*) ada juga berupa genangan aspal di permukaan jalan (*bleeding*), dan ada juga berupa lobang-lobang (*pothole*).”

Maka dari pernyataan tersebut, penulis mengambil kesimpulan bahwa kerusakan jalan adalah kondisi struktural dan fungsional jalan sudah tidak mampu memberikan pelayanan optimal terhadap lalu lintas yang melintasi jalan tersebut yang disebabkan oleh kerusakan berupa *cracking*, *corrugation*, *rutting*, *bleeding*, dan *pothole*.

### 2.2.6. Website

Menurut Bekti (2015:35), “Website merupakan kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing masing dihubungkan dengan jaringan-jaringan halaman”.

Menurut Abdullah (2015:1), “Web dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa text, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet”.

Sedangkan menurut Nilasari (2014:2), “Website atau sering disebut situs merupakan kumpulan halaman web yang dijalankan dari suatu alamat web domain”.

Penulis menyimpulkan bahwa *website* adalah sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk *digital* berupa teks, gambar, multimedia dan lainnya pada jaringan internet sehingga dapat diakses dari seluruh dunia yang memiliki koneksi internet.

---



## 2.3. Teori Program

### 2.3.1. PHP: *Hypertext Processor* (PHP)



**Gambar 2.1.** PHP: *Hypertext Processor* (PHP)

(Sumber: <https://www.php.net/images/logos/new-php-logo.png>)

Menurut Saputra (2014:4), “PHP adalah kependekan dari *PHP:Hypertext Preprocessor* (rekursif, mengikuti gaya penamaan di *\*nix*), merupakan bahasa utama *script server side* yang disisipkan pada HTML yang dijalankan di *server*, dan juga bisa digunakan untuk membuat aplikasi desktop”.

Menurut EMS, dkk (2015:170), menyatakan bahwa “PHP atau *PHP Hypertext Preprocessor* merupakan bahasa pemrograman *web* yang menggunakan prinsip *server side* paling terkenal di dunia. Dengan menggunakan PHP, anda dapat membuat berbagai aplikasi yang membutuhkan performa halaman *web* dinamis”.

Berdasarkan dari pendapat di atas dapat disimpulkan bahwa *PHP:Hypertext Processor* merupakan salah satu bahasa pemrograman yang menggunakan *editor HTML* untuk membuat halaman *web* menjadi lebih interaktif dan dinamis.

### 2.3.2. *Hypertext Markup Language* (HTML)

Menurut Hidayatullah dan Kawistara (2014:13), *Hypertext Markup Language* (HTML) adalah bahasa standar yang digunakan untuk menampilkan halaman *web*. Hal yang dapat dilakukan HTML, yaitu:

1. Mengatur tampilan dari halaman *web* dan isinya.
2. Membuat tabel dalam halaman *web*.
3. Mempublikasikan halaman *web* secara *online*.
4. Membuat *form* yang bisa digunakan untuk menangani registrasi dan transaksi *via web*.



5. Menambahkan objek-objek seperti citra, audio, video, animasi, *java applet* dalam halaman *web*.
6. Menampilkan area gambar (*canvas*) di *browser*.

### 2.3.3. *phpMyAdmin*



**Gambar 2.2.** *phpMyAdmin*

(*Sumber: <https://www.google.co.id/logo+phpmyadmin+png&>*)

Sebagaimana yang dikemukakan oleh Hikmah, dkk (2012:2), “*phpMyAdmin* merupakan aplikasi yang dapat digunakan untuk membuat database, pengguna (*user*), memodifikasi tabel, maupun mengirim database secara cepat dan mudah tanpa harus menggunakan perintah (*command*) SQL”.

### 2.3.4. *XAMPP*



**Gambar 2.3.** *XAMPP*

(*Sumber: <https://www.apachefriends.org/about.html>*)

Menurut Haqi dan Setiawan (2019:8), “*XAMPP* adalah perangkat lunak bebas (*free software*) yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsi *XAMPP* sendiri sebagai server yang berdiri sendiri (*localhost*), yang terdiri dari beberapa program, antara lain: *Apache HTTP Server*,



*MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*”.

## 2.4. Teori Khusus

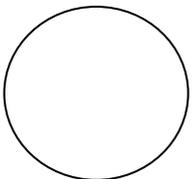
### 2.4.1. Data Flow Diagram (DFD)

Menurut Sukamto dan Shahaluddin (2018:244), “DFD sering juga disebut DAD (Diagram Aliran Data) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

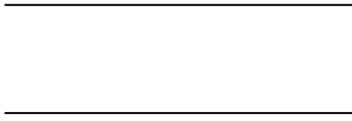
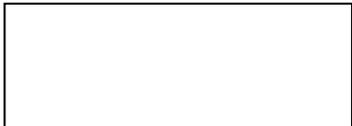
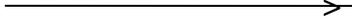
Sedangkan menurut Kristanto (2018:55), “DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem”.

Supardi (2013:8), notasi atau simbol pada DFD adalah sebagai berikut:

**Tabel 2.1.** Simbol-simbol *Data Flow Diagram* (DFD)

No.	Simbol	Keterangan
1.		<b>Proses</b> ( <i>Process</i> ) atau <b>fungsi</b> ( <i>Function</i> ) atau <b>prosedur</b> ( <i>Procedure</i> ). Pada pemrograman terstruktur, notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama proses Biasanya berupa kata kerja.

**Tabel 2.1.** Simbol-simbol *Data Flow Diagram* (DFD)

No.	Simbol	Keterangan
2.		File atau basis data atau penyimpanan (Storage) pada pemrograman terstruktur. Notasi inilah yang harusnya menjadi tabel-tabel basis data yang dibutuhkan. Tabel-tabel ini harus sesuai dengan ERD (Entity Relationship Diagram), CDM (Conceptial Data Model), PDM (Physical Data Model). Catatan: Nama penyimpanan biasanya berupa kata benda.
3.		Entitas luar (external entity) atau masukan (input) atau keluaran (output) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.
4.		Aliran data merupakan data yang dikirim antar-proses penyimpanan ke proses, atau dari proses ke masukan (input) dan keluaran (output). Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data. Misalnya “data Siswa” atau tanpa kata data, misalnya “Siswa”.

(Sumber: Supardi, 2013:8)



Supardi (2013:9), tahapan-tahapan perancangan dengan menggunakan *Data Flow Diagram* yaitu:

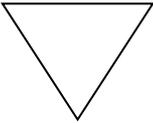
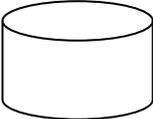
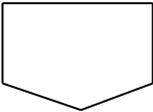
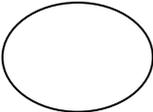
- a. Membuat DFD Level 0 atau sering disebut juga Context Diagram DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.
- b. Membuat DFD Level 1 DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil breakdown DFD Level 0 yang sebelumnya sudah dibuat.
- c. Membuat DFD Level 2 Modul-modul pada DFD Level 1 dapat di-breakdown menjadi DFD Level. Modul mana saja yang harus di-breakdown lebih detail tergantung pada kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu di-breakdown lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*.
- d. Membuat DFD Level 3 dan seterusnya DFD Level 3,4,5, dan seterusnya 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 2.

#### **2.4.2. Block Chart**

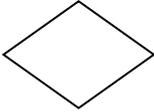
Menurut Kristanto (2018:68), “*Block Chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Block Chart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

Kristanto (2018:68), simbol-simbol yang sering digunakan dalam *Block Chart* dapat dilihat pada tabel berikut:

Tabel 2.2. Simbol-simbol *Block Chart*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/benda/berkas/cetakan.
2.		Multi dokumen.
3.		Proses manual.
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual).
6.		Data penyimpanan (data storage).
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran.

**Lanjutan Tabel 2.2.** Simbol-simbol *Block Chart*

No.	Simbol	Keterangan
11.		Pengambilan keputusan ( <i>decision</i> ).
12.		Layar peraga ( <i>monitor</i> ).
13.		Pemasukan data secara manual.

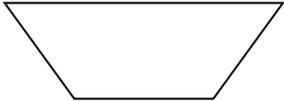
(Sumber: Kristanto, 2018:68)

### 2.4.3. Flow Chart

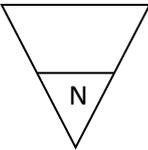
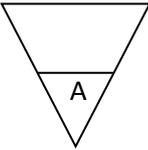
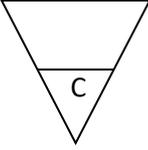
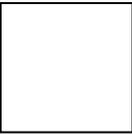
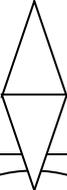
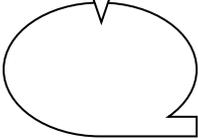
Menurut Supardi (2013:51), “*Flow Chart* merupakan Diagram Alur yang sering digunakan sistem analisis dalam membuat atau menggambarkan logika program”.

Supardi (2013:53), beberapa simbol yang digunakan dalam *Flow Chart* yaitu sebagai berikut:

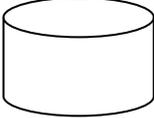
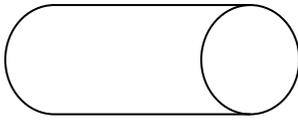
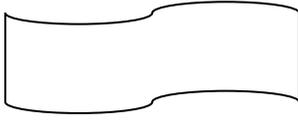
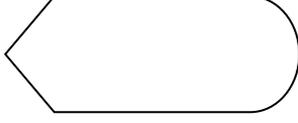
**Tabel 2.3.** Simbol-simbol *Flow Chart*

No.	Simbol	Keterangan
1.		Simbol <b>dokumen</b> yang menunjukkan dokumen <i>input</i> dan <i>output</i> baik untuk proses manual, mekanik atau komputer.
2.		Simbol <b>kegiatan manual</b> yang menunjukkan pekerjaan manual.

Lanjutan Tabel 2.3. Simbol-simbol *Flow Chart*

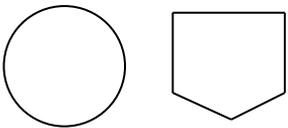
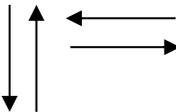
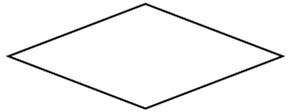
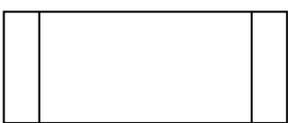
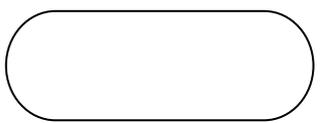
No.	Simbol	Keterangan
3.		Simbol simpanan offline yang berupa file non-komputer yang diarsip urut angka (numerical).
4.		Simbol simpanan offline yang berupa file non-komputer yang diarsip urut huruf (alphabetical).
5.		Simbol simpanan offline yang berupa file non-komputer yang diarsip urut tanggal (chronological).
6.		Simbol kartu plong yang menunjukkan dokumen input dan output yang menggunakan kartu plong (punched card).
7.		Simbol proses yang menunjukkan kegiatan proses dari operasi program komputer.
8.		Simbol operasi luar yang menunjukkan operasi yang dilakukan di luar proses operasi komputer.
9.		Simbol pengurutan offline yang menunjukkan proses pengurutan data dari luar proses komputer.
10.		Simbol pita magnetik yang menunjukkan bahwa input dan output menggunakan pita magnetik.

Lanjutan Tabel 2.3. Simbol-simbol *Flow Chart*

No.	Simbol	Keterangan
11.		Simbol hard disk yang menunjukkan input dan output menggunakan hard disk.
12.		Simbol diskette yang menunjukkan input dan output menggunakan diskette.
13.		Simbol drum magnetik yang menunjukkan input dan output menggunakan drum magnetik.
14.		Simbol pita kertas terhubung yang menunjukkan input dan output menggunakan pita kertas terhubung.
15.		Simbol keyboard yang menunjukkan input dan output menggunakan online keyboard.
16.		Simbol display yang menunjukkan output yang ditampilkan di monitor.
17.		Simbol <b>pita kontrol</b> yang menunjukkan penggunaan pita kontrol ( <i>control tape</i> ) dalam <i>batch control total</i> untuk pencocokan di proses <i>batch processing</i> .
18.		Simbol <b>hubungan komunikasi</b> yang menunjukkan proses transmisi data melalui <i>channel</i> komunikasi.
19.		Simbol <b>penjelasan</b> yang menunjukkan penjelasan dari suatu proses.

Lanjutan Tabel 2.3. Simbol-simbol *Flow Chart*



No.	Simbol	Keterangan
20.		Simbol <b>penghubung</b> yang menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
21.		Simbol <i>input</i> dan <i>output</i> yang digunakan untuk mewakili data <i>input</i> dan <i>output</i> .
22.		Simbol <b>garis alir</b> yang digunakan untuk menunjukkan arus dari proses.
23.		Simbol <b>keputusan</b> ( <i>decision</i> ) yang digunakan untuk penyeleksian kondisi di dalam program.
24.		Simbol <b>proses terdefinisi</b> yang menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
25.		Simbol <b>persiapan</b> yang digunakan untuk memberi nilai awal suatu besaran.
26.		Simbol <b>titik terminal</b> yang digunakan untuk menunjukkan awal dan akhir dari suatu proses.

(Sumber: Supardi, 2013:53)

#### 2.4.4. Entity Relational Diagram (ERD)

Menurut Sukanto dan Shalahuddin. (2018:289), “*Entity Relational Diagram* (EDR) adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional”.

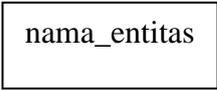
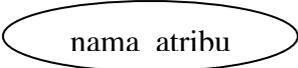
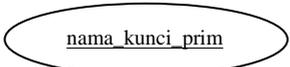
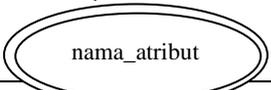
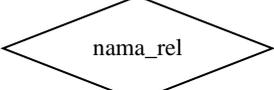
Menurut Supardi (2013:13), “ERD merupakan model konseptual yang mendeskripsikan hubungan antar–penyimpanan (dalam DFD)”.



Sedangkan menurut Ladjamudin (2013:142), “ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam system yang abstrak”.

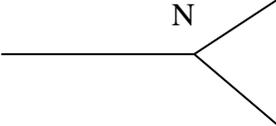
Menurut Sukamto dan Shalahuddin (2018:50), beberapa simbol-simbol pada *Entity Relational Diagram* (EDR) yaitu sebagai berikut:

**Tabel 2.4.** Simbol-simbol *Entity Relational Diagram* (ERD)

No	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	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 ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
4.	Atribut multivalai/ <i>multivalue</i> 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.



**Lanjutan Tabel 2.4.** Simbol-simbol *Entity Relational Diagram* (ERD)

No	Simbol	Keterangan
6.	Asosiasi / association  	<p>Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian</p> <p>Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan one to many menghubungkan entitas A dan entitas B.</p>

(*Simbol: Sukamto dan Shalahuddin, 2018:50*)

#### 2.4.5. Kamus Data

Sukamto dan Shalahuddin (2018:73), “Kamus data adalah daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)”. Pada umumnya kamus data berisi:

- a. Nama-nama dari data
- b. Digunakan pada – merupakan proses-proses yang terkait data
- c. Deskripsi – merupakan deskripsi data
- d. Informasi tambahan – seperti tipe data, nilai data, batas nilai data dan komponen yang membentuk data.

Sukamto dan Shalahuddin (2018:74), simbol-simbol yang ada dalam kamus data adalah sebagai berikut:



Tabel 2.5. Simbol-simbol Kamus Data

No.	Simbol	Keterangan
1.	=	Disusun atau terdiri atas
2.	+	Dan
3.	[]	Baik ... atau ...
4.	{ }n	n kali diulang/bernilai banyak
5.	()	Data opsional
6.	*...*	Batas komentar

(Sumber: Sukamto dan Shalahuddin, 2018:74)

#### 2.4.6. Black-Box Testing

Menurut Jaya (2018), “*Black-Box Testing* merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak”. *Black-Box Testing* bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya difokuskan pada informasi domain. *Black-Box Testing* memungkinkan pengembang *software* untuk membuat himpunan kondisi *input* yang akan melatih seluruh syarat-syarat fungsional suatu program.

Menurut Roger S. Pressman (2012:597), “Pengujian kotak hitam (*Black-Box Testing*) berupaya untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang salah atau hilang,
2. Kesalahan dalam struktur data atau akses basis eksternal,
3. Kesalahan perilaku atau kinerja, dan
4. Kesalahan inisialisasi dan penghentian.