



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Sujatmiko (2012:156) menyatakan bahwa, komputer adalah mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program. Alat serbaguna ini memegang peran penting dalam teknologi komunikasi”.

Menurut Kadir (2017:2) menyatakan bahwa, komputer merupakan peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia”.

2.1.2 Pengertian Informasi

Menurut Pratama (2014:9) menyatakan bahwa, informasi merupakan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah, sehingga memberikan nilai, arti dan manfaat”.

Sutabri (2012:22), “Informasi adalah data yang telah diklasifikasikan atau diolah atau diimplementasikan untuk digunakan dalam proses pengambilan keputusan.”

Jadi, dapat disimpulkan bahwa informasi adalah data yang diolah menjadi bentuk yang berguna dan bermanfaat untuk suatu keperluan.

2.1.3 Pengertian Aplikasi

Menurut Sutabri (2012:147) menyatakan bahwa, aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya”.

Menurut Sujatmiko (2012:23) menyatakan bahwa, *application* (aplikasi) adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya Ms-Word, Ms-Excel”.



2.2 Teori Khusus

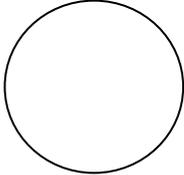
2.2.1 Data Flow Diagram (DFD)

Menurut Djamidin .T (2017:20) DFD (*Data Flow Diagram*) adalah representasi grafik dari sebuah sistem.

Menurut Rosa dan Shalahuddin (2013:70) menyatakan bahwa, *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

Notasi-notasi pada DFD adalah sebagai berikut :

Tabel 2.1 Simbol-simbol pada *DFD*

No.	Simbol	Keterangan
1.		Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. Catatan : Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.
2.		File atau basis data atau penyimpanan (storage); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi table-table pada basis data yang dibutuhkan, table-table ini juga harus sesuai dengan perancangan table-table pada basis data (Entity

Lanjutan **Tabel 2.1.** Simbol-simbol pada *DFD*

No	Simbol	Keterangan
		<p>Relationship Diagram (ERD), Conceptual Data Model (CDM), Physical Data Model (PDM)).</p> <p>Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>
3.		<p>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</p> <p>Catatan: Nama yang digunakan pada masukan (input) atau keluaran (output) biasanya kata benda</p>
4.		<p>Aluran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (input) atau keluaran (output).</p> <p>Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata benda, dapat diawali dengan kata data misalnya "data siswa" atau tanpa kata data misalnya "data siswa" atau tanpa kata data misalnya "siswa"</p>

(Sumber : Rosa dan Shalahuddin, 2013:71-72)



Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD :

1. 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.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan datang dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*.

4. Membuat DFD Level 3 dan seterusnya

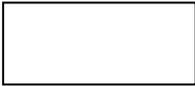
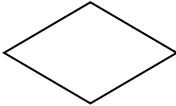
DFD Level 3, 4, 5, dan seterusnya merupakan *breakdown* dan modul pada DFD Level di-atasnya. *Breakdown* pada Level 3,4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.2.2 Entity Relationship Diagram (ERD)

Menurut Rosa dan Shalahuddin (2013:50) mengemukakan bahwa, *entity relationship diagram* digunakan untuk permodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan *Object Oriented Database Management System (OODBMS)* maka perancangan basis data tidak perlu menggunakan *ERD*.



Tabel 2.2 Simbol-Simbol Entity Relationship Diagram (ERD)

No.	Nama	Simbol	Keterangan
1.	Entitas / entity		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		File 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 / multivalue		Field atau kolom data yang dibutuhkan disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
5	Relasi		Relasi yang menghubungkan antara entitas; biasanya siawali dengan kata kerja

Lanjutan **Tabel 2.2** Simbol-Simbol Entity Relationship Diagram (ERD)

No.	Nama	Simbol	Keterangan
6	<i>Asosiasi</i> / <i>association</i>		Penghubung antara relasi dan entitas dimana kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian.

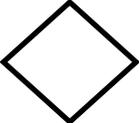
Sumber: Rosa dan Shalahuddin (2013:50)

2.2.3 Pengertian *Flowchart*

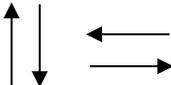
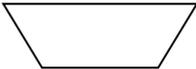
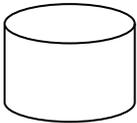
Ladjamuddin (2013:211) menyatakan bahwa, *flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma.

Adapun simbol-simbol *Flowchart* adalah sebagai berikut:

Tabel 2.3 Simbol-Simbol *FlowChart*

No.	Simbol	Arti
1.		Simbol <i>Start</i> atau <i>End</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja
3.		Simbol <i>Input/Output</i> yang mendefinisikan masukan dan keluaran proses
4.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu
5.		Simbol konektor untuk menyambung proses pada lembar kerja yang sama

Lanjutan Tabel 2.3. Simbol-Simbol *FlowChart*

No.	Simbol	Arti
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
7.		Simbol untuk menghubungkan antar proses atau antar symbol
8.		Simbol yang menyatakan piranti keluaran, seperti layar monitor, <i>printer</i> , dll
9.		Simbol yang mendefinisikan proses yang dilakukan secara manual
10.		Simbol masukan atau keluaran dari atau ke sebuah dokumen
11.		Simbol yang menyatakan bagian dari program (subprogram)
12.		Simbol masukan atau keluaran dari atau ke sebuah pita magnetic
13.		Simbol database atau basis data

(Sumber: Ladjamuddin, 2013:211)

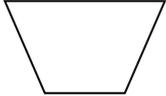
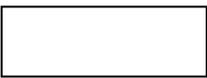
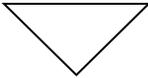
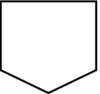
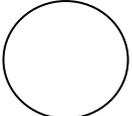
2.2.4 Pengertian *Block Chart*

Kristanto (2011:68) menyatakan bahwa, *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.

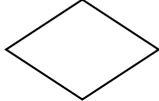


Adapun simbol-simbol *Block Chart* adalah sebagai berikut:

Tabel 2.4 Simbol-Simbol *Block Chart*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
2.		Multi dokumen
3.		Proses manual
4.		Proses yang dilakukan oleh komputer
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (<i>data storage</i>)
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

Lanjutan **Tabel 2.4** Simbol-Simbol *Block Chart*

No.	Simbol	Keterangan
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (decision)
12.		Layar peraga (monitor)
13.		Pemasukan data secara manual

(Sumber: Kristanto, 2011:68)

2.2.5 Pengertian Kamus Data

Menurut Rosa dan Shalahuddin (2013:73) menyatakan bahwa, kamus data (data dictionary) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. 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). Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya 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

Kamus data memiliki beberapa symbol untuk menjelaskan informasi tambahan sebagai berikut:



Tabel 2.5 Simbol-simbol kamus data

Simbol	Arti
=	Disusun atau terdiri dari
+	dan
[]	Baik ...atau...
{ }n	n kali diulang/ bernilai banyak
()	Data opsional
* ... *	Batas komentar

(Sumber: Rosa dan Shalahuddin, 2013:73)

2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Menurut Sutabri (2012:147) menyatakan bahwa, aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya”.

Menurut Sujatmiko (2012:23) menyatakan bahwa, *application* (aplikasi) adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya Ms-Word, Ms-Excel”.

2.3.2 Pengertian Pengolahan Data

Yakub (2012:75) mengemukakan bahwa, pengolahan data adalah manipulasi atau transformasi simbol-simbol seperti; angka dan abjad untuk tujuan meningkatkan kegunaannya.

2.3.3 Pengertian Pasien

Dalam Kamus Besar Bahasa Indonesia Edisi Ketiga (834), Pasien yaitu orang sakit (yang dirawat dokter); penderita (sakit).



2.3.4 Pengertian Rawat Jalan

Dalam Kamus Besar Bahasa Indonesia Edisi Ketiga (935), Rawat Jalan yaitu perawatan pasien dengan cara berobat jalan.

2.3.5 Pengertian Rawat Inap

Dalam Kamus Besar Bahasa Indonesia Edisi Ketiga (935), Rawat Inap yaitu perawatan pasien dengan menginap (di rumah sakit).

2.3.6 Pengertian Rumah Sakit Khusus

Dalam Kamus Besar Bahasa Indonesia Edisi Ketiga (968), Rumah Sakit Khusus yaitu rumah sakit yang memberikan layanan pemeriksaan, pengobatan, dan perawatan satu macam penyakit.

2.3.7 Pengertian Paru-paru

Dalam Kamus Besar Bahasa Indonesia Edisi Ketiga (832), Paru-paru yaitu organ tubuh yang berupa sepasang kantong berbentuk bulat toraks, terdapat di rongga dada, berfungsi sebagai alat pernapasan (untuk membersihkan darah dengan oksigen yang diisap dari udara pada manusia dan sebagian binatang).

2.3.8 Pengertian Aplikasi Pengolahan Data Pasien Rawat Jalan dan Rawat Inap pada Rumah Sakit Khusus Paru-paru

Aplikasi Pengolahan Data Pasien Rawat Jalan dan Rawat Inap pada Rumah Sakit Khusus Paru-paru merupakan sebuah aplikasi yang dibangun untuk mempermudah sistem kerja pengolahan data pasien rawat jalan dan rawat inap serta penyimpanan data yang lebih efektif, sehingga pencarian data mudah ditemukan.

2.4 Teori Program

2.4.1 Basis Data (*Database*)

Djamidin .T (2017:20) Basis data merupakan kumpulan data yang terstruktur dan sistematis dalam media penyimpanan komputer dan saling



berhubungan dan dapat dimanfaatkan dengan cepat jika dibutuhkan dan tidak terjadi redundansi (pengulangan) yang tidak perlu.

Menurut Sujatmiko (2012:40) basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data.

2.4.2 PHP (*Hypertext Preprocessor*)



Gambar 2.1 Logo PHP

Menurut Saputra, A. (2012:2) PHP atau yang memiliki kepanjangan PHP Hypertext Preprocessor, merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu website dinamis. PHP menyatu dengan kode HTML, maksudnya adalah beda kondisi, HTML digunakan sebagai pembangun atau pondasi dari kerangka layout web, sedangkan PHP difungsikan sebagai prosesnya, sehingga dengan adanya PHP tersebut, sebuah web akan sangat mudah di-maintenance.

Menurut Nugroho (2013:153) menyatakan bahwa, PHP (*PHP Hypertext Preprocessor*) itu adalah pemrograman berbasis Web. Jadi PHP itu adalah Bahasa pemrograman yang digunakan untuk membuat aplikasi berbasis web (website, blog atau aplikasi web).



2.4.2.1 Tipe Data *PHP*

2.4.2.1.1 Tipe Data String

Dalam penulisannya, tipe data string menggunakan tanda kutip tunggal (‘ ‘) atau menggunakan tanda kutip ganda (“ “).

```
$data1='Ini data string PHP';
$data1="Ini juga data string PHP";
```

menggunakan variabel. Jika menggunakan tanda petik tunggal maka apabila sebuah variabel berisi data **string** dan berisi variabel yang lain, yang terjadi adalah nilai dari variabel tersebut akan dibaca atau akan tetap dicetak nama variabel itu sendiri.

2.4.2.1.2 Tipe Data Integer

Tipe data integer merupakan semua data yang berisi bilangan bulat dan besarnya **range** sama dengan data pada bahasa C, yaitu antara -2,147,483,648 sampai +2,147,483,647 pada platform 32 bit, apabila data diluar kisaran tersebut, maka PHP akan secara otomatis mengkonversikan data tersebut dari tipe **Integer** menjadi tipe **Floating Point**.

2.4.2.1.3 Tipe Data Floating Point

Tipe data yang berisi bilangan pecahan atau bilangan desimal. Kisaran data floating adalah antara 1.7E-308 sampai 1.7E+308. Data ini berbentuk desimal atau bentuk pangkat. Perhatikan contoh berikut:

```
$float=0. 5;
$float2=5. 0e-4;
```

2.4.2.1.4 Tipe Data Object

Tipe data object dapat berupa bilangan, variabel ataupun fungsi. Tipe data ini dapat membantu programmer untuk membuat sebuah program. Data ini dapat disertakan ke dalam program.



2.4.3 XAMPP



Gambar 2.2 Logo XAMPP

Menurut Nugroho (2013:1) “Xampp adalah paket program web lengkap yang dapat anda pakai untuk belajar pemrograman web, khususnya PHP dan Mysql, paket ini dapat didownload secara gratis dan legal.”

Menurut Riyanto (2015:1) menyatakan bahwa, XAMPP merupakan paket PHP dan MySQL berbasis open source, yang dapat digunakan sebagai tool pembantu pengembangan aplikasi berbasis PHP. XAMPP mengkombinasikan beberapa paket perangkat lunak berbeda ke dalam satu paket.

2.4.4 MySQL



Gambar 2.3 Logo MySQL

Menurut Nugroho, B. (2013:26) *Database MySQL* adalah program berbasis DOS perintah dasarnya adalah SQL (*Structured Query Language*), Anda dapat mengaksesnya dari jendela DOS Prompt atau Command Prompt.

Menurut Saputra, A. (2012:7) MySQL merupakan standar penggunaan database di dunia untuk pengolahan data.



Kelebihan yang dimiliki oleh MySQL :

1. Bersifat Open source, yang memiliki kemampuan untuk dapat dikembangkan lagi.
2. Menggunakan bahasa SQL (Structure Query Language), yang merupakan standar bahasa dunia dalam pengolahan data.
3. Super performance dan reliable, tidak bisa diragukan, pemrosesan databasenya sangat cepat dan stabil.
4. Sangat mudah dipelajari (ease of use).
5. Memiliki dukungan support (group) pengguna MySQL.
6. Mampu lintas Platform, dapat berjalan di berbagai sistem operasi.
7. Multiuser, dimana MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami konflik.

2.4.5 Javascript

Menurut Sidik (2011:1) menyatakan bahwa, Javascript adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam browser menjadi lebih interaktif, tidak sekadar indah saja. Javascript memberikan beberapa fungsionalitas ke dalam halaman *web*, sehingga dapat menjadi sebuah program yang disajikan dengan menggunakan antarmuka *web*.