



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Prianto (2008:12), “Komputer adalah sebuah mesin yang berperan dalam memanipulasi data berdasarkan instruksi-instruksi yang diberikan.”

Puspitosari (2013:13), “Komputer adalah sebuah alat yang digunakan untuk mengolah data menurut perintah yang telah dirumuskan.”

Kesimpulannya, Komputer adalah Mesin yang dapat memanipulasi data digital dan dapat juga digunakan untuk mengolah data berdasarkan instruksi atau perintah.

2.1.2 Pengertian Program

Sahyar (2016:2), “Program adalah perintah-perintah atau instruksi yang disusun berdasarkan algoritma dengan menggunakan bahasa pemrograman untuk penyelesaian suatu masalah.”

Kristanto (2008:15), “Program dapat dianalogikan sebagai instruksi atau perintah-perintah untuk mengoperasikan atau menjalankan *hardware*.”

Kesimpulannya, Program adalah suatu instruksi atau perintah yang digunakan untuk mengoperasikan *hardware* yang disusun berdasarkan algoritma.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Riswaya dan Abdurrahman (2014:1), “Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai tujuan pembuatan aplikasi tersebut.”

Juansyah (2015:2), “Aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa



aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju.”

Kesimpulannya, Aplikasi adalah program yang siap digunakan untuk melaksanakan suatu perintah agar mencapai sasaran yang akan dituju.

2.2.2 Pengertian Pengolahan Data

Kristanto (2008:8), “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan.”

Ambarwati dan Marsela (2015:79) “Pengolahan data adalah suatu proses yang menerima masukan (input), memproses (processing) menggunakan program tertentu dengan mengeluarkan hasil data.”

Kesimpulannya, Pengolahan data adalah proses yang menerima masukan, memproses hingga menghasilkan keluaran informasi data.

2.2.3 Pengertian Mitra Binaan

Mitra Binaan adalah pelaku usaha kecil yang mendapatkan pinjaman dari program kemitraan.

Sarma dkk (2013:9) “Kemitraan adalah program yang bertujuan untuk meningkatkan kemampuan usaha kecil untuk menjadi tangguh dan mandiri melalui pemanfaatan dana.”

2.2.4 Pengertian Program Peningkatan Ekonomi Kerakyatan

Program Peningkatan Ekonomi Kerakyatan adalah suatu program Pemerintah Kota Palembang dalam rangkaian program Palembang EMAS 2018, yang bertujuan untuk meningkatkan ekonomi kerakyatan dengan pemberdayaan masyarakat.



2.2.5 Pengertian Website

Sujatmiko (2012:317), “Website adalah salah satu aplikasi internet yang terdiri dari perangkat lunak, kumpulan protokol dan seperangkat aturan yang memungkinkan untuk mengakses informasi di internet.”

Hidayat (2013:2), “Website adalah keseluruhan halaman-halaman web yang terdapat dari sebuah domain yang mengandung informasi.”

Kesimpulannya, Website keseluruhan halaman-halaman web yang terdapat dari sebuah domain yang memungkinkan untuk mengakses informasi di internet.

2.2.6 Pengertian Aplikasi Pengolahan Data Mitra Binaan Program Peningkatan Ekonomi Kerakyatan pada Dinas Koperasi dan Usaha Kecil Menengah Kota Palembang Berbasis Web

Aplikasi Pengolahan Data Mitra Binaan Program Peningkatan Ekonomi Kerakyatan pada Dinas Koperasi dan Usaha Kecil Menengah Kota Palembang Berbasis Web adalah suatu aplikasi yang dibuat dengan menggunakan bahasa pemrograman *PHP* dan *Database MySql* yang digunakan untuk mengoptimalkan efisiensi pengolahan data mitra binaan pada Dinas Koperasi dan Usaha Kecil Menengah Kota Palembang.

2.3 Teori Khusus

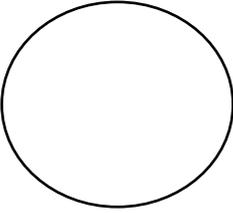
2.3.1 Pengertian DFD (*Data Flow Diagram*)

Sukanto dan Shalahuddin (2016:71), “*Data Flow Diagram* (DFD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*ouput*).”



Adapun notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut :

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

Notasi	Keterangan
	<p>Proses atau fungsi atau prosedur, pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program</p> <p>Catatan :</p> <p>Nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p>
	<p>File basisdata atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harusnya sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM))</p> <p>Catatan : Nama yang diberikan pada sebuah penyimpanan biasanya kata benda</p>

Lanjutan Tabel 2.1 Simbol-simbol *Data Flow Diagram*

Notasi	Keterangan
	<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) 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 :</p> <p>Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda</p>
	<p>Aliran data: merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses masukan (<i>input</i>) atau keluaran (<i>output</i>)</p> <p>Catatan:</p> <p>Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”</p>

(sumber : Sukamto dan Shalahuddin, 2016:71-72)



Sukanto dan Shalahuddin (2016:72-73) menjelaskan, 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 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 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 ada DFD Level 1 yang di-breakdown.

4. Membuat DFD Level 3 dan seterusnya

DFD Level 3,4,5 dan seterusnya merupakan breakdown dari modul pada DFD diatasnya. Breakdown pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

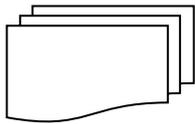
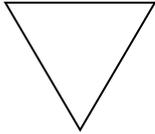
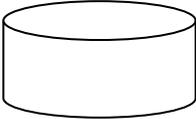
2.3.2 Pengertian *Block chart*

Kristanto (2008:75), “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.” Pembuatan blockhart harus memudahkan bagi pemakai dalm memahami alur dari sistem atau transaksi.”



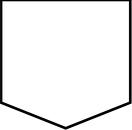
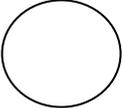
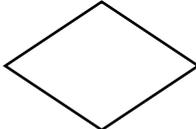
Adapun simbol-simbol yang sering digunakan dalam block chart dapat dilihat pada tabel berikut ini:

Tabel 2.2 Simbol-simbol *Block chart*

Simbol	Arti
	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/berkas atau cetakan.
	Multi Dokumen
	Proses Manual
	Proses yang dilakukan oleh komputer
	Menandakan dokumen yang diarsipkan (arsip manual)
	Data penyimpanan (<i>data storage</i>)



Lanjutan Tabel 2.2 Simbol-simbol *Block chart*

	Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
	Terminasi yang menandakan awal dan akhir dari suatu aliran
	Pengambilan keputusan (<i>decision</i>)
	Layar peraga (<i>monitor</i>).
	Pemasukkan data secara manual.

(Sumber : Kristanto, 2008:75-77)

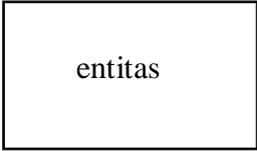
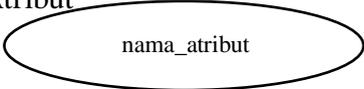
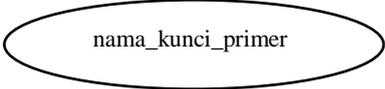


2.3.3 Pengertian ERD (*Entity Relational Diagram*)

Sukanto dan Shalahuddin (2016:50), “*Entity Relational Diagram* (ERD) digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain.”

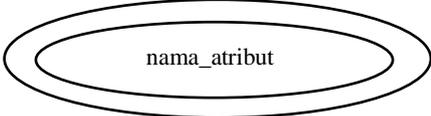
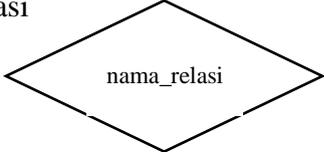
Adapun simbol-simbol yang digunakan pada ERD dengan notasi Chen adalah sebagai berikut:

Tabel 2.3 Simbol-simbol *Entity Relational Diagram*

Simbol	Deskripsi
Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimoan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> 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)



Lanjutan Tabel 2.3 Simbol-simbol *Entity Relational Diagram*

<p>Atribut multinilai / <i>multivalued</i></p> 	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu</p>
<p>Relasi</p> 	<p>Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja</p>
<p>Asosiasi / <i>association</i></p> 	<p>Penghubung antar relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> 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 <i>one to many</i> menghubungkan entitas A dan entitas B maka</p>

(Sumber: Sukamto dan Shalahuddin, 2016:50-51)

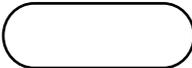
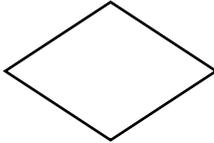
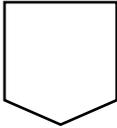


2.3.4 Pengertian *Flowchart*

Sitorus (2015:14) menjelaskan bahwa *Flowchart* adalah langkah-langkah menyelesaikan masalah yang dituliskan dalam simbol-simbol tertentu.”

Menurut Sitorus (2015:14-16), Gambaran simbol-simbol *Flowchart* sebagai berikut:

Tabel 2.4 Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program.
2		<i>Input/output</i>	Menyatakan proses input atau output tanpa tergantung jenis peralatannya.
3		<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
4		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya atau tidak.
5		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
6		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.

Lanjutan Tabel 2.4 Simbol-simbol *Flowchart*

7		<i>Predefined process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
8		<i>Punched Card</i>	Menyatakan input berasal dari kartu atau output data ke kartu.
9		<i>Punch Tape</i>	
10		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui Printer)
11		<i>Flow</i>	Menyatakan jalannya arus suatu proses

(Sumber: Sitorus, 2015:15-16)

2.3.5 Pengertian Kamus Data

Sukamto dan Shalahuddin (2016:73) menjelaskan, “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (input) dan keluaran (ouput) dapat dipahami secara umum (memiliki standar cara penulisan).”



Kamus data memiliki beberapa simbol sebagai berikut :

Tabel 2.5 Simbol-simbol Kamus Data

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

(*Sumber: Sukamto dan Shalahuddin, 2016:74*)

2.4 Teori Program

2.4.1 Pengertian Basis Data

Sukamto dan Shalahuddin (2016:43), “Basis data adalah sistem komputerisasi yang tujuan utamanya adalah memelihara data yang sudah ada yang diolah atau informasi dan membuat informasi tersedia saat dibutuhkan.”

Kristanto (2008:14), “Basis data merupakan data-data yang saling berhubungan satu dengan yang lain yang disimpan dalam perangkat keras komputer dan akan diolah menggunakan perangkat lunak.”

Kesimpulannya, Basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis dan saling berhubungan satu dengan yang lain yang tujuan utamanya adalah memelihara data yang sudah ada yang diolah.

2.4.2 Pengertian XAMPP

Riyanto (2011:1), “XAMPP merupakan paket PHP dan MySQL berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP”



EMS (2016:59-60), “XAMPP merupakan salah satu paket web server yang praktis karna mengisntal langsung *Apache* (server), MySQL (database), dan PHP yang yang tersedia untuk multisistem operasi (windows dan Linux).”

Kesimpulannya, XAMPP adalah suatu program server yang terdiri dari *Apache* (server), MySQL (database), dan PHP digunakan untuk mengakses fungsi yang ada dalam halaman website tersebut agar bisa diakses oleh user.

2.4.3 Pengertian MySQL

Sukanto dan Shalahuddin (2016:46), “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus.”

Iriani (2013:56), “MySQL adalah sebuah konsep pengopeasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.”

Kesimpulannya, MySQL adalah bahasa yang digunakan untuk mengelola data pada RDBMS bersifat *open source* sehingga memungkinkan pengoperasian data dikerjakan dengan mudah.

2.4.4 Pengertian PHP

Enterprise (2016:97), “PHP merupakan pemrograman yang digunakan untuk membuat website interaktif.”

EMS (2016:55), “PHP merupakan bahasa pemrograman pelengkap HTML yang memungkinkan dibuatnya aplikasi web dinamis untuk pengolahan data, pemrosesan data dan sebagainya.”

Kesimpulannya, PHP adalah bahasa pemograman yang bisa kita gunakan untuk membuat aplikasi web yang disisipkan pada HTML.



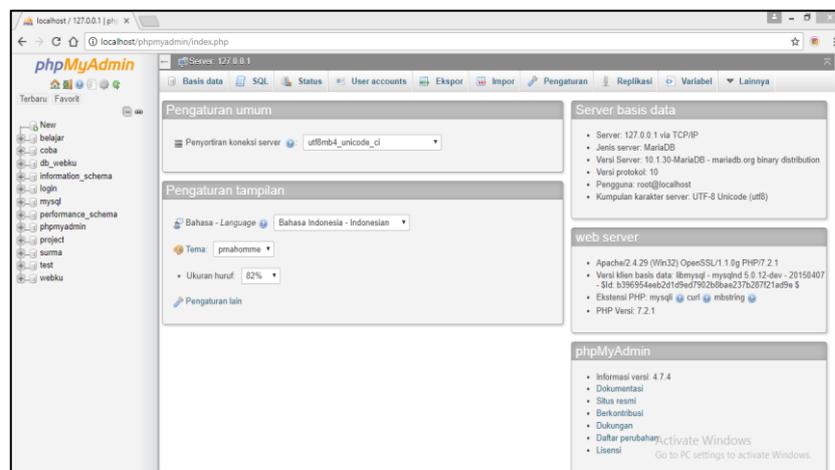
Gambar 2.1 Tampilan Logo PHP

(Sumber : www.yudana.id)

2.4.5 Pengertian *phpMyAdmin*

Nugroho (2013:15), “*phpMyAdmin* adalah aplikasi manajemen database server MySQL berbasis web”

Kita dapat mengunduh *phpMyAdmin* dengan gratis pada situs resminya di www.phpmyadmin.net. *Phpmyadmin* harus dijalankan di sisi server *web* (misalnya; *Apache web server*) dan pada komputer harus tersedia *PHP*, karna berbasis *web*.



Gambar 2.2 Tampilan *phpMyAdmin*

2.4.6 Metode Pengembangan Sistem

Sukamto dan Shalahudin (2016:28-30) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak



secara terurut mulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung (*support*).

a. Analisis Kebutuhan Perangkat Lunak

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

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengkodean.

c. Pembuatan Kode Barang

Pada tahap pengkodean, 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. Dalam penelitian ini pengujian sistem akan menggunakan pengujian Black-Box. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah mengirimkan ke *user*. Perubahan bisa terjadi adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.



2.4.6.1 Pengujian Kotak Hitam (*Black-Box Testing*)

Rosa dan Shalahuddin (2016:275) menyatakan bahwa *black-box testing* yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.