



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Asropudin (2013:19), “Komputer merupakan alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer”.

Sujatmiko (2012:156) mengatakan bahwa, “Komputer adalah mesin yang dapat mengelolah data digital dengan mengikuti serangkaian perintah atau program. Alat serbaguna ini memegang peran penting dalam teknologi komunikasi”.

2.1.2. Pengertian Data

Sujatmiko (2012:76) mengemukakan bahwa, “Data adalah kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi”.

Sutabri (2012:6), “Data merupakan bahan mentah untuk diolah yang hasilnya kemudian menjadi informasi. Dengan kata lain, data yang telah diperoleh harus diukur dan dinilai baik dan buruk, berguna atau tidak dalam hubungannya dengan tujuan yang dicapai”.

2.1.3. Pengertian Informasi

Rusdiana dan Irfan (2014:74), “Informasi merupakan sesuatu yang dihasilkan dari pengolahan data. Data yang sudah ada dikemas dan diolah sedemikian rupa sehingga menjadi sebuah informasi yang berguna”.

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



2.1.4 Metode Pengembangan Sistem

Sukanto dan Shalahuddin (2013:28) 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 sekuensial atau terurut mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

b. Desain

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pengodean

Tahap pengodean desain harus ditranlasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Tahap pengujian fokus pada sistem dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

Tahap pendukung atau pemeliharaan tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis



spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.

2.2. Teori Judul

2.2.1 Pengertian Aplikasi

Sujatmiko (2012:23), “Aplikasi merupakan program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya Ms-Word, Ms-Excel. Aplikasi berbeda dengan sistem operasi (yang menjalankan komputer), *atilly* (yang melaksanakan perawatan atau tugas-tugas umum) dan bahasa”.

Asropudin (2013:6), “Aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Ms-Word, Ms-Excel”.

2.2.2. Pengertian Agenda

Menurut Kamus Saku Bahasa Indonesia, “Agenda merupakan buku catatan yang bertanggung untuk satu tahun acara (yang akan dibicarakan dalam rapat)”.

2.2.3. Pengertian Surat Masuk

Nuraida (2014:88), “Surat masuk adalah surat yang masuk ke dalam suatu instansi/perusahaan atau bagian dalam suatu instansi/perusahaan yang berasal baik instansi/perusahaan lain atau dari bagian lain di instansi/perusahaan yang sama”.

Surat masuk adalah semua jenis surat yang diterima dari instansi lain dari perorangan, baik yang diterima melalui pos (kantor pos).

<http://evanurafivah14.blogspot.co.id/2012/11/prosedur-pengelolaan-surat-masuk-dan.html>

2.2.4. Pengertian Surat Keluar

Nuraida (2014:91), “Surat keluar adalah surat yang dikirim oleh suatu instansi/perusahaan atau dari bagian di dalam instansi/perusahaan tersebut yang



ditujukan ke instansi/perusahaan lain atau ke bagian lain dalam instansi/perusahaan yang sama”.

Surat keluar adalah segala komunikasi tertulis yang diterima oleh suatu badan usaha dari instansi lain perorangan.

<http://evanurafivah14.blogspot.co.id/2012/11/prosedur-pengelolaan-surat-masuk-dan.html>

2.2.5. Pengertian Sekretaris

Menurut Kamus Saku Bahasa Indonesia, “Sekretaris adalah orang yang disertai pekerjaan tulis-menulis, surat menyurat dalam organisasi”.

2.2.6. Pengertian Direktur

Menurut Kamus Saku Bahasa Indonesia, “Direktur adalah pemimpin tertinggi suatu perusahaan”.

2.2.7. Pengertian Aplikasi Agenda Surat Masuk dan Surat Keluar Sekretaris Direktur pada Politeknik Negeri Sriwijaya

Aplikasi agenda surat masuk dan surat keluar adalah suatu aplikasi yang mengelola data dan memantau pengiriman surat sampai pada proses bagian yang dituju berdasarkan isi atau maksud dari surat tersebut.

2.3. Teori Khusus

2.3.1. Pengertian DFD (*Data Flow Diagram*)

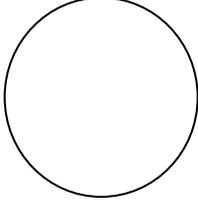
Indrajani (2015:27) mengatakan bahwa “*Data Flow Diagram* (DFD) adalah sebuah alat yang menggambarkan aliran data sampai sebuah sistem selesai, dan kerja atau proses dilakukan dalam sistem tersebut”.

Kristanto (2008:61) menyatakan, “*Data Flow Diagram* atau DFD adalah suatu model logika atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut”.



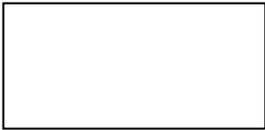
Notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut:

Tabel 2.1. Simbol–simbol *Data Flow Diagram* (DFD) menurut Edward Yourdon dan Tom DeMarco

Notasi	Keterangan
	<p>Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi fungsi atau prosedur di dalam kode program.</p> <p>Catatan: nama yang diberikan biasanya berupa kata kerja.</p>
	<p>File atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel–tabel basis data yang dibutuhkan, tabel–tabel ini juga harus 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* (DFD) menurut Edward Yourdon dan Tom DeMarco

	<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: nama yang digunakan biasanya berupa kata benda.</p>
	<p>Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).</p> <p>Catatan: nama yang digunakan biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa”.</p>

(Sumber: Sukanto dan Shalahuddin, 2014:71-72)

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*

DFD Level 0 menggambarkan system yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun system lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara system yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam system 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 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 system, 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* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.3.2. Pengertian *Blockchart*

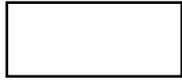
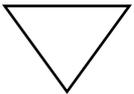
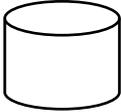
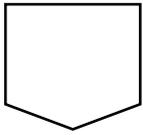
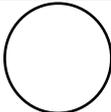
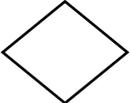
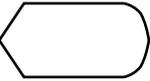
Kristanto (2008:75) menjelaskan,“*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol – simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

Adapun simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.2. Simbol-Simbol *Blockchart*

No	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
2.		Multi dokumen
3.		Proses manual

Lanjutan Tabel 2.2. Simbol-Simbol *Blockchart*

4.		proses yang dilakukan oleh komputer
5.		Menandakan dokumen yang diarsifkan (arsif manual)
6.		Data penyimpanan (<i>data storage</i>)
7.		Proses apa saja yang tidak terdefinisi termasuk aktifitas fisik
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
9.		Terminasi yang mewakili simbol tertentu digunakan pada aliran lain pada halaman yang sama
10.		Terminasi yang menandakan awal dan akhir suatu aliran
11.		Pengambilan keputusan (<i>decesion</i>)
12.		Layar peraga (<i>monitor</i>)
13.		Pemasukan data secara manual

(Sumber : Kristanto, 2008 : 75-77)

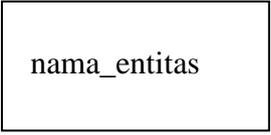
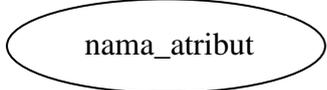
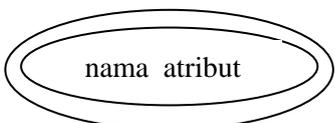


2.3.3. Pengertian ERD (*Entity Relational Diagram*)

Sukanto dan Shalahuddin (2014:50), “*Entity Relationship Diagram (ERD)* digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD”.

Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

Tabel 2.3. Simbol – simbol *Entity Relational Diagram (ERD)*

Simbol	Deskripsi
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 table
Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas
Atribut kunci primer 	Field 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)
Atribut multi nilai / <i>multi value</i> 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu



Lanjutan Tabel 2.3. Simbol – simbol *Entity Relational Diagram (ERD)*

<p>Relasi</p> 	<p>Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja</p>
<p>Asosiasi / <i>association</i></p> 	<p>Penghubung antara 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 : Sukanto dan Shalahudin, 2014:50-51)

2.3.4. Pengertian *Flowchart*

Indrajani (2015:36), “*Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempermudah penyelesaian masalah, khususnya yang perlu dipelajari dan dievaluasi lebih lanjut.”

Ladjamudin (2013:211), “*Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.”

Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yakni sebagai berikut:

1. *Flow Direction Symbols*

Digunakan untuk menghubungkan antara simbol satu dengan simbol yang lain. Simbol-simbol tersebut adalah sebagai berikut :

**Tabel 2.4.** Simbol – simbol Flow Directions Symbols

1.		<p>Simbol arus / flow</p> <p>Untuk menyatakan jalannya arus suatu proses</p>
2.		<p>Simbol Communication link</p> <p>Untuk menyatakan bahwa adanya transisi suatu data/informasi dari satu lokasi ke lokasi lainnya</p>
3.		<p>Simbol Connector</p> <p>Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.</p>
4.		<p>Simbol Offline Connector</p> <p>Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.</p>

(Sumber : Ladjamudin, 2013:266)

2. Processing Symbols

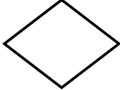
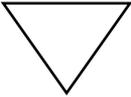
Menunjukkan jenis operasi pengolahan dalam suatu proses / prosedur. Simbol-simbol tersebut adalah sebagai berikut:

Tabel 2.5. Simbol – simbol Processing Symbols

1.		<p>Simbol Offline Connector</p> <p>Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.</p>
2.		<p>Simbol Manual</p> <p>Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).</p>



Lanjutan Tabel 2.5. Simbol – simbol Processing Symbols

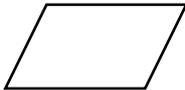
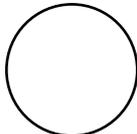
3.		Simbol Decision/ logika Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/tidak.
4.		Simbol Predefined Proses Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk member harga awal.
5.		Simbol Terminal Untuk menyatakan permulaan atau akhir suatu program
6.		Simbol Keying Operation Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard
7.		Simbol off-line storage Untuk menunjukkan bahwa data dalam symbol ini akan disimpan ke suatu media tertentu
8.		Simbol Manual input Untuk memasukan data secara manual dengan menggunakan online keyboard.

(Sumber: Ladjamudin, 2013:267)

3. Input -Output Symbols

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*. Simbol-simbol tersebut adalah sebagai berikut:

**Tabel 2.6.** Simbol – simbol Input-Output Symbols

1.		Simbol Input-output Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
2.		Simbol Punched Card Untuk menyatakan input berasal dari kartu atau output ditulis ke kartu.
3.		Simbol Magnetic-tape unit Untuk menyatakan input berasal dari pita magnetic atau output disimpan ke pita magnetic.
4.		Simbol Disk storage Untuk menyatakan input berasal dari disk atau output disimpan ke disk.
5.		Simbol Document Untuk mencetak laporan ke printer.
6.		Simbol Display Untuk menyatakan peralatan output yang digunakan berupa layar (video, komputer).

(Sumber: Ladjamudin, 2013:268)

2.3.5. Kamus Data (*Data Dictionary*)

Sukanto dan Shalahuddin (2014:73) menjelaskan, “Kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada system perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)”.

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



Tabel 2.7. Simbol-simbol dalam 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, 2014:74)

2.4. Teori Program

2.4.1. Pengertian MySQL

Raharjo (2011:21), “MySQL merupakan *software* RDBMS (atau *server database*) yang dapat mengelola *database* dengan sangat 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*)”.

Winarno, dkk (2014:102) menjelaskan bahwa, “*MySQL* merupakan tipe data relasional yang artinya *MySQL* menyimpan datanya dalam bentuk *table-table* yang saling berhubungan”.

2.4.2. Pengertian PHP

Prasetyo (2015:130), “PHP (PHP:Hypertext Preprocessor) adalah bahasa script yang ditanam di sisi server. Proses PHP dijalankan di server saat sebuah halaman dibuka dan mengandung kode PHP, prosesor itu akan menerjemahkan dan mengeksekusikan semua perintah dalam halaman tersebut, dan kemudian menampilkan hasilnya ke browser sebagai halaman biasa”.

Saputra (2012:2), “PHP memiliki kepanjangan *PHP Hypertext Preprocessor*, merupakan suatu bahasa pemrograman yang difungsikan untuk membangun suatu website dinamis”.



2.4.3. Pengertian XAMPP

Madcoms (2009:1), “XAMPP adalah salah satu paket software web server yang terdiri dari Apache, MySQL, PHP dan phpMyAdmin”.

Nugroho (2013:1), “XAMPP adalah paket program web lengkap yang dapat anda pakai untuk belajar pemrograman *web* khususnya PHP dan MySQL, Paket ini dapat di *download* secara gratis dan illegal”.

Menurut Nugroho (2013:7), Dibawah folder utama xampp, terdapat beberapa folder penting yang perlu diketahui. Untuk penjelasan dan fungsinya sebagai berikut:

1. Apache : folder utama dari Apache Webserver.
2. Htdocs : Folder utama untuk menyimpan data-data latihan web, baik PHP maupun HTML biasa. Pada folder ini, anda dapat membuat subfolder sendiri untuk mengelompokkan file latihannya. Semua folder dan file program di htdocs bias diakses dengan mengetikkan alamat `http://localhost/` di browser.
3. Manual : Berisi subfolder yang di dalamnya terdapat manual program dan database, termasuk manual PHP dan MySQL.
4. MySQL : Folder utama untuk database MySQL server. Di dalamnya terdapat subfolder data (lengkapnya; `C:\xampp\mysql\data`) untuk merekam semua nama database, serta subfolder bin yang berisi tools klien dan server MySQL.
5. PHP : Folder utama untuk program PHP.

2.4.4. Pengertian Dreamweaver

Elcom (2013:1), ”Dreamweaver adalah sebuah editor professional yang menggunakan HTML untuk mendesain web secara visual dan mengelola situs atau halaman web. Dreamweaver merupakan software utama yang digunakan oleh desainer web dan programmer web untuk mengembangkan suatu situs web”.

Madcoms (2009:10), “Dreamweaver merupakan perangkat lunak yang ditujukan untuk membuat suatu web. Versi terbaru dari Dreamweaver adalah Dreamweaver CS4. Versi terbaru ini memiliki performa lebih baik dan memiliki tampilan yang memudahkan Anda untuk membuat dan mengelola halaman web,



termasuk diantaranya dalam hubungannya dengan bahasa pemrograman PHP dan database MySQL”.

Sadeli (2014:12), “Dreamweaver merupakan suatu perangkat lunak web editor keluaran Adobe System yang digunakan untuk membangun dan mendesain suatu website dengan fitur-fitur yang menarik dan kemudahan dalam penggunaannya”.

2.4.5. Pengertian Basis Data (*Database*)

Sujatmiko (2012:76) mengemukakan bahwa, “Database merupakan basis data atau representasi kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan”.

Indrajani (2015:70), “Basis data adalah sebuah kumpulan data yang saling berhubungan secara logis, dan merupakan sebuah penjelasan dari data tersebut, yang didesain untuk menemukan data yang dibutuhkan oleh sebuah organisasi”.