



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Sujatmiko (2012:156) menyatakan bahwa, “Komputer adalah mesin yang dapat mengelola digital dengan mengikuti serangkaian perintah atau program.”

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

Dari pernyataan diatas dapat disimpulkan bahwa komputer adalah sistem elektronik untuk memanipulasi data yang cepat dan tepat serta dirancang dan diorganisasikan supaya secara otomatis menerima dan menyimpan data input, memprosesnya, dan menghasilkan output dibawah pengawasan suatu langkah-langkah intruksi program yang tersimpan di memori.

2.1.2 Pengertian Internet

Sujatmiko (2012:138) menyatakan bahwa, “*internet* adalah jaringan global yang menghubungkan berjuta-juta komputer di seluruh dunia melalui jalur telepon kabel maupun satelit.”

Ahmadi dan Hermawan (2013:2) menyatakan bahwa, “*internet* adalah sistem informasi global berbasis komputer.”

Dari pernyataan diatas dapat disimpulkan bahwa *internet* adalah sistem global dari seluruh jaringan komputer yang saling terhubung sehingga antara komputer dapat saling mengakses informasi dan bertukar data.

2.1.3 Pengertian Aplikasi

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



Sedangkan menurut 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.”

Dari pernyataan diatas dapat disimpulkan bahwa aplikasi adalah suatu perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna.

2.1.4 Pengertian Web

Ahmadi dan Hermawan (2013:69) menyatakan bahwa, “*World Wide Web*, lebih terkenal disingkat WWW, adalah suatu ruang informasi yang dipakai oleh pengenal global yang disebut Pengidentifikasi Sumber Seragam untuk mengenal pasti sumber daya berguna.”

Saptika (2015:1409) menyatakan bahwa, “*Web* adalah sistem untuk mengakses, memanipulasi, dan mengunduh dokumen hipertaut yang terdapat di komputer yang dihubungkan melalui *internet*.”

Dari pernyataan diatas dapat disimpulkan bahwa *Web* adalah halaman situs sistem informasi yang dapat diakses secara cepat.

2.1.5 Pengertian e-Commerce

Ahmadi dan Hermawan (2013:7) menyatakan bahwa, “*Electronic Commerce* adalah penjualan atau pembelian barang dan jasa antara perusahaan, rumah tangga, individu, pemerintah, dan masyarakat atau organisasi swasta lainnya yang dilakukan melalui komputer pada media jaringan.”

Menurut Yuhefizar (2013:2), “*e- Commerce* adalah singkatan dari *electronic commerce*, yaitu sebuah layanan berbasis elektronik (*internet*) untuk bertransaksi/berdagang secara *online*.”

Sedangkan menurut Saputra (2013:116), “*e- Commerce* adalah segala aktivitas transaksi produk ataupun jasa antara penjual dan pembeli dengan



memanfaatkan kecanggihan elektronik, sehingga proses transaksi dapat dilakukan meskipun antara penjual dan pembeli tidak secara langsung bertatap muka.”

Dari berbagai pengertian diatas, pada prinsipnya memiliki kesamaan yang mencakup komponen transaksi (adanya pembeli, penjual, barang, jasa dan informasi), subjek dan objek yang terlibat, serta media yang digunakan, yaitu *internet*.

2.1.6 Model Transaksi *e-Commerce*

e-Commerce merupakan aktivitas pembelian dan penjualan melalui jaringan internet dimana pembeli dan penjual tidak bertemu secara langsung, melainkan berkomunikasi melalui media *internet*. *e-Commerce* memiliki berbagai macam jenis transaksi dalam menerapkan sistemnya.

Ahmadi dan Hermawan (2013:35) menjelaskan jenis-jenis transaksi *e-Commerce* diantaranya sebagai berikut:

- a. *Collaborative Commerce (C-Commerce)*
Collaborative Commerce yaitu kerjasama secara elektronik antar rekan bisnis. Kerjasama ini biasanya terjadi antara rekan bisnis yang berada pada jalur penyediaan barang (*supply chaint*).
- b. *Business to Business (B2B)*
e-Commerce tipe ini meliputi transaksi antar organisasi yang dilakukan di *electronic-market*.
- c. *Business to Consumers (B2C)*
Business to Consumers yaitu penjual adalah suatu organisasi dan pembeli adalah individu.
- d. *Consumer to Business (C2B)*
Dalam C2B konsumen memberitahukan kebutuhan atas suatu produk atau jasa tertentu dan para pemasok bersaing untuk menyediakan produk atau jasa tersebut ke konsumen.
- e. *Consumer to Consumer (C2C)*
C2C yaitu konsumen menjual secara langsung ke konsumen lain atau mengiklankan jasa pribadi di internet. Dalam C2C seseorang menjual produk atau jasa ke orang lain. Dapat juga disebut sebagai pelanggan, yaitu orang yang menjual produk dan jasa ke satu sama lain.



2.1.7 Keuntungan *e-Commerce*

Saputra (2013:116) menjelaskan beberapa keuntungan dalam *e-Commerce* diantaranya adalah:

1. Modal Kecil

Untuk membangun sebuah *e-Commerce* tidak dibutuhkan sebuah modal yang besar. Tidak perlu memikirkan lahan toko maupun jika kita tidak memiliki produk, kita dapat bekerja sama dengan toko-toko yang sudah mempercayai kita untuk menjualkan produk mereka. Kita hanya cukup memiliki keahlian untuk mempromosikan *web* tersebut. Lalu untuk membangunnya, di *website* banyak menyediakan fasilitas untuk membangun toko *online (e-Commerce)* secara *free/gratis*. Serta tersedia juga *domain* dan *hosting* gratis.

2. Biaya Operasional Kecil

Toko *online/e-Commerce* dapat tidak membutuhkan kantor maupun petugas keamanan, disini anda benar-benar sebagai *owner* (pemilik).

3. Sistem Promosi yang Mudah

Anda tidak perlu direpotkan untuk membuat brosur maupun iklan yang memakan cukup biaya dan waktu. Anda cukup mempromosikan toko anda melalui *internet*.

4. Transaksi 24 jam

Toko *online / e-Commerce* dapat melayani pelanggan selama 24 jam nonstop. Pelanggan pun akan merasa tetap dilayani karena ditunjang oleh otomatisasi toko *online / e-Commerce* tersebut.

2.1.8 Kerugian *e-Commerce*

Ahmadi dan Hermawan (2013:13) menjelaskan beberapa kerugian dalam *e-Commerce* diantaranya adalah:

1. Kehilangan segi finansial secara langsung karena kecurangan. Seorang penipu mentransfer uang dari rekening satu ke rekening lainnya atau dia telah mengganti semua data finansial yang ada.
2. Pencurian informasi rahasia yang berharga. Gangguan yang timbul bisa menyingkap semua informasi rahasia tersebut kepada pihak-pihak yang tidak berhak dan dapat mengakibatkan kerugian yang besar bagi si korban.
3. Kehilangan kesempatan bisnis karena gangguan pelayanan. Kesalahan ini bersifat kesalahan nonteknis seperti aliran listrik tiba-tiba padam.
4. Penggunaan akses ke sumber oleh pihak yang tidak berhak. Misalkan *hacker* yang berhasil membobol sebuah sistem perbankan. Setelah itu dia memindahkan sejumlah rekening orang lain ke rekeningnya sendiri.



5. Kehilangan kepercayaan dari para konsumen. Hal ini disebabkan berbagai macam faktor, seperti usaha yang dilakukan sengaja oleh pihak yang berusaha menjatuhkan reputasi perusahaan tersebut.
6. Kerugian yang tak terduga. Disebabkan oleh gangguan yang dilakukan sengaja, ketidakjujuran, praktik bisnis yang tidak benar, kesalahan faktor manusia, atau kesalahan sistem elektronik.

2.1.9 Pengertian Pabrik Roti Bobby

Pabrik Roti Bobby Palembang merupakan perusahaan yang bergerak di bidang manufaktur dengan mengolah bahan mentah berupa gandum menjadi roti-roti yang dijual di pasaran. Daerah pemasarannya meliputi Palembang dan sekitarnya.

2.1.10 Pengertian Aplikasi Web e-Commerce pada Pabrik Roti Bobby Palembang

Aplikasi Web e-Commerce pada Pabrik Roti Bobby Palembang adalah aplikasi yang berfungsi mengolah data transaksi penjualan secara *online* pada Pabrik Roti Bobby Palembang yang meliputi promosi produk, pemesanan produk, dan penjualan produk sehingga memudahkan konsumen dalam berbelanja produk dan memudahkan perusahaan dalam menjual produk yang dimilikinya serta memperluas jaringan pemasaran dengan menggunakan bahasa pemrograman *PHP* dan *database MySQL*.

2.1.11 Metode Pengujian Perangkat Lunak

2.1.11.1 Pengertian Metode Pengujian

Shihab (2011), “Metode pengujian adalah cara atau teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.”

Pengujian perangkat lunak perlu dilakukan untuk mengevaluasi baik secara manual maupun otomatis untuk menguji apakah perangkat lunak sudah memenuhi persyaratan atau belum, dan untuk menentukan perbedaan antara hasil yang diharapkan dengan hasil sebenarnya.



2.1.11.2 Metode Pengujian

Perangkat lunak dapat diuji dengan dua cara, yaitu:

1. Pengujian dengan menggunakan data uji untuk menguji semua elemen program (data internal, loop, keputusan dan jalur). Data uji dibangkitkan dengan mengetahui struktur internal (kode sumber) dari perangkat lunak.
2. Pengujian dilakukan dengan mengeksekusi data uji dan mengecek apakah fungsional perangkat lunak bekerja dengan baik. Data uji dibangkitkan dari spesifikasi perangkat lunak.

2.1.11.2.1 Metode *Black Box Testing*

2.1.11.2.1.1 Pengertian Metode *Black Box Testing*

Iskandaria (2012), “Pengujian *blackbox (blackbox testing)* adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada *input* dan *output* aplikasi (apakah sudah sesuai dengan apa yang diharapkan atau belum). Tahap pengujian atau *testing* merupakan salah satu tahap yang harus ada dalam sebuah siklus pengembangan perangkat lunak (selain tahap perancangan atau desain).”

Shihab (2011), “*Black Box Testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengetesan pada spesifikasi fungsional program.”

Shihab (2011), mengemukakan ciri-ciri *blackbox testing*, diantaranya sebagai berikut:

1. *Black Box Testing* berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.
2. *Black Box Testing* bukan teknik alternative daripada *white box testing*. Lebih dari pada itu, ia merupakan pendekatan pelengkap dalam mencakup error dengan kelas yang berbeda dari metode *white box testing*.
3. *Black Box Testing* melakukan pengujian tanpa pengetahuan detail struktur *internal* dari sistem atau komponen yang dites, juga disebut sebagai *behavioral testing*, *specification-based testing*, *input/output testing* atau *functional testing*.



Pada *black box testing* terdapat jenis teknik *design tes* yang dapat dipilih berdasarkan pada tipe *testing* yang akan digunakan, diantaranya sebagai berikut:

1. *Equivalence Class Partitioning*
2. *Boundary Value Analysis*
3. *State Transitions Testing*
4. *Cause-Effect Graphing*

Kategori kesalahan/error yang akan diketahui melalui *black box testing*:

1. Fungsi yang hilang atau tak benar/salah
2. Error dari antar-muka/*interface*
3. Error dari struktur data atau akses *eksternal database*
4. Error dari kinerja atau tingkah laku/*perform*
5. Error dari inisialisasi dan terminasi

2.2 Teori Khusus

2.2.1 Pengertian DFD (*Data Flow Diagram*)

Kristanto (2008:61) menyatakan bahwa, “*DFD* adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana 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.”

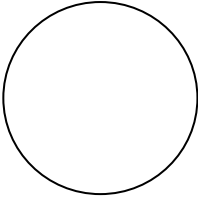
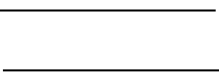
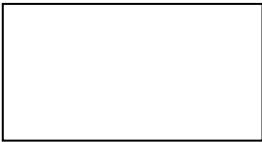
Rosa dan M. Shalahuddin (2016:70) menyatakan bahwa, “*DFD* dapat digunakan untuk mempresentasikan sebuah sistem atau perangkat lunak pada beberapa level yang lebih *detail* untuk merepresentasikan aliran informasi atau fungsi yang lebih *detail*.”

Dari pernyataan diatas dapat disimpulkan bahwa *Data Flow Diagram* (*DFD*) adalah suatu cara atau metode untuk membuat rancangan sebuah sistem yang mana berorientasi pada alur data yang nantinya bergerak pada sebuah sistem.



Rosa dan Shalahuddin (2016:71) juga menjelaskan notasi- notasi pada *DFD* (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut:

Tabel 2.1 Notasi-notasi pada *DFD*

Notasi	Keterangan
	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.
	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 tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (Entity Relationship Diagram (ERD), Conceptual Data Model (CDM), Physical Data Model.
	Entitas luar (external entity) atau masukan (input) atau keluaran (output) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau system lain yang terkait dengan aliran data dari system yang dimodelkan. Catatan: Nama yang digunakan pada masukan (input) atau keluaran (output) biasanya berupa kata benda.

Lanjutan **Tabel 2.1** Notasi-notasi pada *DFD*

	Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (input) atau 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: Shalahuddin et.al, 2016:71)

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan *DFD* sebagai berikut:

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

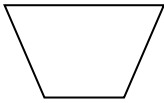
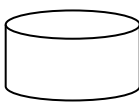
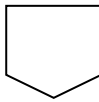


2.2.2 Pengertian *Block Chart*

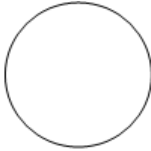
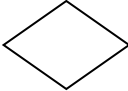
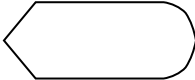
Kristanto (2008:75) menyatakan bahwa, “*block chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.”

Kristanto (2008:75) juga menjelaskan tentang simbol-simbol yang sering digunakan dalam *block chart* dapat dilihat pada tabel berikut ini:

Tabel 2.2 Simbol-simbol dalam *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

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

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
11.		Pengambilan keputusan (decision)
12.		Layar peraga (monitor)
13.		Pemasukkan data secara manual.

(Sumber: Kristanto, 2008:75)

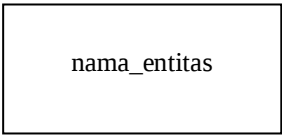
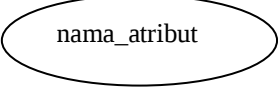
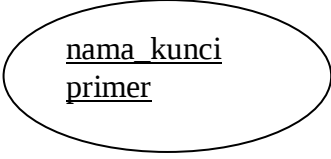
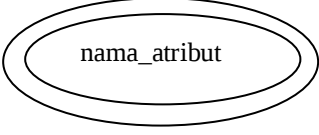
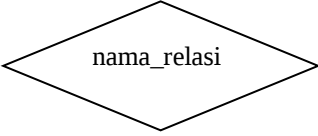
2.2.3 Pengertian *ERD (Entity Relationship Diagram)*

Rosa dan Shalahuddin (2016 : 50) menjelaskan tentang pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram (ERD)*. *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 (dikembangkan oleh Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lainnya.

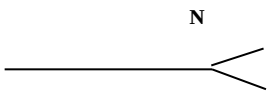


Berikut adalah simbol-simbol yang digunakan pada *ERD* dengan notasi Chen:

Tabel 2.3 Simbol-simbol *ERD*

Simbol	Deskripsi
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 computer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
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 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)
Atribut multinilai / multivalue 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja

Lanjutan **Tabel 2.3** Simbol-simbol *ERD*

Asosiasi / association 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki multiplicity kemungkinan jumlah pemakaian Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas
---	---

(Sumber: Shalahuddin et.al , 2016:50)

2.2.4 Pengertian *Flowchart*

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

Ladjamudin (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.”

Dari pernyataan diatas dapat disimpulkan bahwa *flowchart* adalah bagan air atau bagan-bagan yang mempunyai arus penyelesaian suatu masalah.

2.2.4.1 Simbol-simbol *Flowchart*

Flowchart disusun dengan simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses didalam program. Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yakni sebagai berikut:

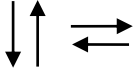
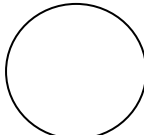
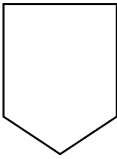
1. *Flow Direction Symbols* (Simbol penghubung/alur)
2. *Processing Symbols* (Simbol proses)
3. *Input-output Symbols* (Simbol input-output)



2.2.4.2 Flow Direction Symbols

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga *connecting line*. Simbol-simbol tersebut adalah sebagai berikut:

Tabel 2.4 Flow Direction Symbols

No.	Simbol	Arti
1.		Simbol arus / flow Untuk menyatakan jalannya arus suatu proses.
2.		Simbol Communication link Untuk menyatakan bahwa adanya transisi suatu data/informasi dari suatu lokasi ke lokasi lainnya.
3.		Simbol Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.
4.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.

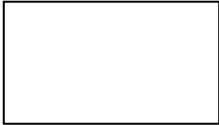
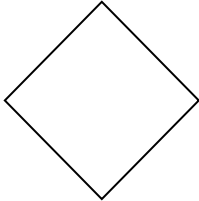
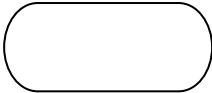
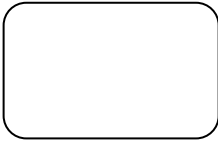
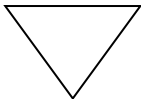
(Sumber: Ladjamudin, 2013:266)

2.2.4.3 Processing Symbols

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses atau prosedur. Simbol-simbol tersebut adalah sebagai berikut:



Tabel 2.5 Processing Symbols

No.	Simbol	Arti
1.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.
2.		Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).
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 memberi 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 <i>keyboard</i> .
7.		Simbol Offline Storage Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
8.		Simbol Manual Input Untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i> .

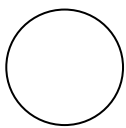
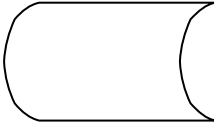
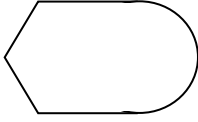
(Sumber: Ladjamudin, 2013:267)



2.2.4.4 Input-output Symbols

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

Tabel 2.6 *Input-output Symbols*

No.	Simbol	Arti
1.		Simbol Input-output Untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
2.		Simbol Punched Card Untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> dituliskan ke kartu.
3.		Simbol Magnetic-tape Unit Untuk menyatakan input berasal dari pita <i>magnetic</i> atau <i>output</i> disimpan ke pita <i>magnetic</i> .
4.		Simbol Disk Storage Untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
5.		Simbol Document Untuk mencetak laporan ke printer.
6.		Simbol Display Untuk menyatakan peralatan <i>output</i> yang digunakan berupa layar (video, komputer).

(Sumber: Ladjamudin, 2013:268)



2.2.5 Pengertian Kamus Data

Rosa dan Shalahuddin (2016:73) 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 memiliki beberapa simbol 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: Shalahuddin et.al, 2016:74)

2.2.6 Pengembangan Sistem

2.2.6.1 Tahap-Tahap Pengembangan Sistem

Kristanto (2008:41) menjelaskan tahap-tahap dalam pengembangan sistem adalah sebagai berikut:

1. Tahap Penyelidikan Awal dan Studi Kelayakan

Pada tahap penyelidikan awal, analis belajar dari pemakai mengenai apa yang diharapkan dari sebuah sistem informasi yang baru. Jika suatu fungsi tertentu telah dapat ditetapkan untuk dilaksanakan, maka langkah berikutnya adalah mengadakan studi kelayakan yang bertujuan untuk memutuskan usaha-usaha penyusunan sistem yang akan dilakukan untuk menyelesaikan masalah-masalah sistem informasi.

2. Tahap Penentuan Kebutuhan-Kebutuhan Sistem

Sasaran pertama pada tahap ini adalah mendefinisikan apa yang seharusnya dapat dilakukan oleh sistem yang baru. Kemudian menentukan kriteria yang dapat digunakan untuk mengevaluasi keandalan sistem yang baru.

3. Tahap Desain Sistem

Untuk melakukan perbaikan terhadap sistem informasi, terlebih dahulu harus dipahami dengan jelas kondisi sistem yang ada



sekarang dan yang dihadapi, setelah itu sasaran dan kebutuhan sistem di masa yang akan datang. Kemudian baru dapat dimasukkan ide-ide secara bersama-sama ke dalam suatu desain yang akan memenuhi tujuan-tujuan yang telah ditetapkan. Untuk itu dapat digunakan analisa terstruktur dengan diagram-diagram aliran data. Pada proses desain sistem, terdapat proses pemindahan dari apa yang harus dilakukan sistem dan bagaimana sistem nanti akan melakukannya.

4. Tahap Implementasi dan Evaluasi Sistem

Bila program dari sistem sesuai dengan spesifikasi, kemudian dapat dilaksanakan uji persetujuan pemakai untuk melihat apakah desain yang dinyatakan dengan spesifikasi sistem tersebut sesuai dengan kebutuhan pemakai. Pengujian program dengan data yang realistis memerlukan bantuan para pemakai sebagai penghasil data tersebut, demikian juga dengan pengujian sistem. Dengan demikian, pada tahap implementasi dan uji coba sistem baru merupakan tahapan yang paling menyita banyak waktu dan membutuhkan perhatian yang khusus.

2.2.7 Design Input dan Output

2.2.7.1 Input

Isa (2012:68) menyatakan bahwa, “*input* adalah sesuatu yang dibutuhkan oleh suatu sub-sistem untuk melakukan fungsinya.”

Input dapat berasal dari:

- Luar sistem yang baru (dari *internal* Sistem Informasi yang lain (misalnya *output* dari sistem pemesanan) atau dari *eksternal* sistem (misalnya dari *output* sistem paket aplikasi atau dari organisasi lain)).
- Sub-sistem yang lain, yang masih merupakan bagian dari sistem tersebut.
- Dari (pusat) kumpulan data/database.

2.2.7.2 Output

Isa (2012:68) menyatakan bahwa, “*output* merupakan suatu bentuk keluaran yang dikehendaki dari input yang telah melalui pemrosesan (baik berupa laporan yang berbentuk *hardcopy*, maupun data/file yang akan menjadi *input* untuk sub-sistem lain).”

Output dapat ditujukan kepada:

- Sub-sistem yang lain (yang akan berfungsi sebagai *input* untuk sub-sistem tersebut baik yang berupa program maupun *spreadsheet*).
- Internal* sistem yang lain (misalnya untuk sistem penggajian) atau *eksternal*



sistem (misalnya data untuk sistem pembayaran gaji karyawan yang dilakukan oleh bank).

- c. *E-Commerce*.
- d. Kumpulan data / *database*

2.3 Teori Program

2.3.1 Basis Data (*Database*)

Sujatmiko (2012:40) menyatakan bahwa, “Basis Data (*Database*) merupakan kumpulan informasi yang disimpan didalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut.”

Priyadi (2014:3) juga menjelaskan beberapa elemen basis data, diantaranya adalah sebagai berikut:

1. Tabel

Pada suatu basis data, tabel di representasikan menjadi suatu bentuk segiempat berupa matriks, yang terdiri dari kolom dan baris.

2. *Field*

Pada tabel basis data, kolom merupakan representasi untuk sebuah nama *field* yang pembacaan datanya dilakukan secara vertical.

3. *Record*

Pada tabel basis data, baris merupakan suatu representasi untuk sebuah *record* yang pembacaan datanya dilakukan secara horizontal. Satu baris pada sebuah tabel merupakan data yang dimiliki oleh satu *record*. Nilai-nilai yang dimiliki oleh sebuah *record* merupakan gabungan dari semua *field* yang terdapat dalam tabel tersebut.

4. Kardinalitas

Kardinalitas merupakan batasan dari banyaknya hubungan, yang dapat dilakukan oleh himpunan entitas dalam melakukan relasi dengan himpunan entitas lainnya. Variasi kemungkinan untuk melakukan relasi yang dimiliki oleh kardinalitas terdiri dari empat macam, yaitu:

- a. Satu ke satu (1:1)



- b. Satu ke banyak (1:N)
- c. Banyak ke satu (N:1)
- d. Banyak ke banyak (N:N)

2.3.2 Sekilas Tentang *PHP*

2.3.2.1 Pengertian *PHP (Hypertext Preprocessor)*

Kadir (2008:358) menyatakan bahwa, “*PHP* merupakan bahasa pemrograman skrip yang diletakkan dalam server yang biasa digunakan untuk membuat aplikasi *web* yang bersifat dinamis.”

MADCOMS (2012:206) menyatakan bahwa, “*PHP (Hypertext Preprocessor)* adalah salah satu bahasa pemrograman yang berjalan dalam sebuah *webserver* dan berfungsi sebagai pengolah data pada sebuah *server*.”

Menurut pernyataan diatas *PHP* adalah sebuah pemrograman *web* berbasis *server* yang mempunyai parsing kode *PHP* dari kode *web* dengan ekstensi *PHP* sederhana menghasilkan tampilan *website*.

2.3.2.2 Konsep Kerja *PHP*

Saputra (2013:4) menjelaskan tentang konsep kerja *PHP* adalah sebagai berikut:

1. Server membaca permintaan dari *client/browser*.
2. Kemudian dilanjutkan untuk mencari halaman/*page* pada *server*.
3. *Server* melakukan instruksi yang diberikan oleh *PHP* untuk melakukan modifikasi pada halaman/*page*.
4. Selanjutnya hasil modifikasi tersebut akan dikembangkan kepada *client/browser*.

2.3.2.3 Skrip *PHP*

Kadir (2008:3) menjelaskan tentang skrip *PHP*. “Skrip *PHP* berkedudukan sebagai *tag* dalam bahasa *HTML*. Sebagaimana diketahui, *HTML* adalah bahasa standar untuk membuat halaman-halaman *web*”.



Contoh:

File latihan1.html

```
<HTML>

                                <HEAD>
    <TITLE> Latihan HTML </TITLE>
    </HEAD>
    <BODY>
    Mari Belajar Membuat Web
    </BODY>
</HTML>
```

Contoh di atas bisa ditulis dengan menggunakan *PHP* sebagai berikut yang kodenya di simpan dengan latihan1.php.

```
<HTML>
<HEAD>
<TITLE> Latihan HTML </TITLE>
</HEAD>
<BODY>
<?php
    Printf ("Mari Belajar Membuat Web");
    // atau
    Echo "<br>";
    Echo "Mari Belajar Membuat Web";
?>
</BODY>
</HTML>
```

2.3.3 Sekilas Tentang MySQL

2.3.3.1 Pengertian MySQL

Kadir (2008:2) menyatakan bahwa, “MySQL merupakan *software* yang tergolong DBMS (*Database Management System*) yang bersifat *Open Source*.”



Setiawan (2015:30) menyatakan bahwa, “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi diseluruh dunia.”

Dari kedua pernyataan diatas, dapat disimpulkan bahwa *MySQL* adalah *software database* untuk mengelolah *database* dengan cepat.



Gambar 2.1 Logo MySQL

2.3.4 Webserver

Untuk bisa mengakses aplikasi *web* yang akan dibuat oleh penulis, maka diperlukan yang namanya *webserver*. Penulis menggunakan *XAMPP* sebagai *webserver* dalam membangun Aplikasi *web e-Commerce* pada Pabrik Roti Bobby Palembang.

2.3.4.1 Pengertian XAMPP

Wardana (2016:4) menjelaskan bahwa, “*XAMPP* adalah paket *software* yang didalamnya terkandung *Web Server Apache*, *database MySQL*, dan *PHP Interpreter*.”

Nugroho (2013:1) menjelaskan bahwa, “*XAMPP* adalah paket program *web* lengkap yang dapat dipakai untuk belajar pemrograman *web*, khususnya *PHP* dan *MySQL*, paket ini dapat didownload secara gratis dan legal.”

Dari pernyataan diatas *XAMPP* adalah paket program *web* yang terdiri dari *Apache*, *MySQL*, *PHP Interpreter* yang dipakai untuk belajar pemrograman *web* secara gratis dan legal.



Gambar 2.2 Halaman download XAMPP

Adapun langkah-langkah untuk menginstall program XAMPP pada komputer sebagai berikut :

1. Klik 2x (*Double Klik*) file instalasinya.
2. Selanjutnya anda akan diminta untuk menentukan bahasa yang digunakan saat instalasi.



Gambar 2.3 Menjalankan file instalasi

2.3.5 Pengertian *phpMyAdmin*

Nugroho (2013:71) menyatakan bahwa, “*phpMyAdmin* adalah *tools MySQL client* berlisensi *Freeware*, serta dapat mengunduhnya secara gratis pada situs resminya di www.phpmyadmin.net harus dijalankan disisi *server web* (misalnya: *Apache web server*) dan pada komputer harus tersedia *PHP*, karena berbasis *web*.”

**Gambar 2.4** Logo phpMyAdmin

Fungsi dalam halaman ini adalah sebagai pengendali *database MySQL* sehingga pengguna *MySQL* tidak perlu *report* untuk menggunakan perintah-perintah *SQL*. Karena dengan adanya halaman ini semua yang ada pada halaman *phpMyAdmin*.

2.3.6 Pengertian *Sublime Text*

Faridl (2015:3) menjelaskan bahwa, “*Sublime Text* adalah teks *editor* berbasis *Phyton*, sebuah teks *editor* yang elegan, kaya *fitur*, *cross flatform*, mudan dan simple yang cukup terkenal dikalangan *developer* (pengembang), penulis dan desainer.”

Sublime Text mempunyai beberapa keunggulan-keunggulan yang dapat membantu pengguna dalam membuat sebuah *web development*. Berikut keunggulan-keunggulan *fitur* yang dimiliki *Sublime Text*, yaitu:

1. *Multiple Selection*
2. *Command Pallete*
3. *Distraction Free Mode*
4. *Find in Project*
5. *Plugin API Switch*
6. *Drag and Drop*
7. *Split Editing*
8. *Multi Platform*