



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

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

Hamacher dalam Mulyono (2010:1), Komputer adalah mesin penghitung elektronik yang cepat dan dapat menerima informasi *input* digital, kemudian memprosesnya sesuai dengan program yang tersimpan di memorinya, dan menghasilkan *output* berupa informasi.

Blissmer dalam Mulyono (2010:1), Komputer adalah alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima *input*, memproses input sesuai dengan instruksi yang diberikan, menyimpan perintah dan hasil pengolahannya, serta menyediakan *output* dalam bentuk informasi.

Fuori dalam Mulyono (2010:1), “Komputer adalah pemroses data yang dapat melakukan perhitungan besar secara cepat, termasuk perhitungan aritmatika dan operasi logika, tanpa campur tangan manusia.”

Dari pengertian tersebut, dapat disimpulkan bahwa komputer adalah mesin penghitung elektronik yang cepat yang dapat melakukan beberapa tugas, yaitu menerima *input*, memproses *input* sesuai dengan instruksi yang diberikan, serta menghasilkan *output* berupa informasi.

2.1.2. Pengertian Software

Shalahuddin et. al (2013:2), “*Software* adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).”

Sutarman (2012:14), “*Software* adalah kumpulan program-program komputer yang memungkinkan *hardware* memproses data.”



Siallagan (2009:3), “*Software* adalah program-program komputer yang berguna untuk menjalankan atau mengoperasikan suatu pekerjaan sesuai dengan yang dikehendaki.”

Dari pengertian tersebut, dapat disimpulkan bahwa *software* adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak yang memungkinkan *hardware* memproses data dan menjalankan atau mengoperasikan suatu pekerjaan sesuai dengan yang dikehendaki.

2.1.3. Pengertian Program

Sutarman (2012:3), “Program adalah barisan perintah/instruksi yang disusun sehingga dapat dipahami oleh komputer dan kemudian dijalankan sebagai barisan perhitungan numerik, dimana barisan perintah tersebut berhingga, berakhir, dan menghasilkan *output*.”

Shillagan (2009:3), “Program dapat dianalogikan sebagai instruksi atau perintah-perintah untuk mengoperasikan atau menjalankan *hardware*.”

Dari pengertian tersebut, dapat disimpulkan bahwa program adalah kumpulan instruksi atau perintah-perintah yang disusun sehingga dapat dipahami oleh komputer untuk melakukan fungsinya.

2.1.4. Pengertian *Internet*

Dadang et. al (2013:68), “*Internet* merupakan komunikasi jaringan komunikasi global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem operasi dan mesin.

Salim (2013:279), “*Internet* adalah jaringan komunikasi elektronik yang menghubungkan jaringan komputer dan fasilitas komputer yang terorganisasi di seluruh dunia melalui telepon atau satelit.”

Sutarman (2012:283), “*Internet* adalah kumpulan dari berbagai macam jenis komputer yang saling terhubung dengan menggunakan media telekomunikasi (telepon, *wireless*, satelit, dan sebagainya) dengan jangkauan seluruh dunia/global.”



Dari pengertian tersebut, dapat disimpulkan bahwa *internet* adalah jaringan komunikasi elektronik yang menghubungkan seluruh komputer di dunia dengan menggunakan media telekomunikasi.

2.1.5. Pengertian Web

Dadang et. al (2013:69), “*Web* adalah halaman informasi yang disediakan melalui jalur *internet* sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan *internet*.”

Asropudin (2013:109), “*Web* adalah sebuah kumpulan halaman (*webpage*) yang diawali dengan halaman muka (*homepage*) yang berisikan informasi, iklan, serta program interaksi.”

Salim (2013:513), “*Web* adalah sistem untuk mengakses, memanipulasi dan mengunduh dokumen *hepertaut* yang terdapat dalam komputer yang dihubungkan melalui *internet*.”

Dari pengertian tersebut, dapat disimpulkan bahwa *web* adalah kumpulan halaman informasi yang disediakan melalui jalur *internet* selama terkoneksi dengan jaringan *internet*.

2.2. Teori Khusus

2.2.1. Pengertian DFD (Data Flow Diagram)

Saputra (2013:118), “*Data Flow Diagram* atau disingkat *DFD* merupakan suatu diagram yang menggambarkan aliran data dalam suatu entitas ke sistem atau sistem ke entitas.”

Shalahuddin et. al (2013:70), *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*).

Kristanto (2008:61), “*Data Flow Diagram* merupakan 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

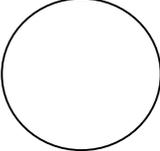


menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.”

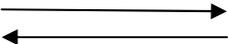
Dari pengertian tersebut, dapat disimpulkan bahwa *Data Flow Diagram* adalah proses yang dibuat untuk menggambarkan aliran data dari masukan (*input*) dan keluaran (*output*).

Shalahuddin et. al (2013:71), menjelaskan notasi pada *DFD* adalah sebagai berikut :

Tabel 2.1. Simbol-simbol *Data Flow Diagram*

No	Simbol	Keterangan
1.		<p>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</p> <p>catatan :</p> <p>nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>
2.		<p><i>File</i> 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 (ERD)</i>, <i>Conceptual Data Model(CDM)</i>, <i>Physical Data Model (PDM)</i>).</p> <p>catatan :</p> <p>nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>

Lanjutan Tabel 2.1 Simbol-simbol *Data Flow Diagram*

No	Simbol	Keterangan
3.		<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 pada masukan (<i>input</i>) atau keluaran (<i>output</i>) berupa kata benda.</p>
4.		<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 pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.</p>

(Sumber : Shalahuddin et. al, 2013:71)

Shalahuddin et. al (2013:72) menjelaskan tentang tahapan-tahapan perancangan dengan menggunakan *Data Flow Diagram* yaitu:

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

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 *2*.

2.2.2. Pengertian Blockchart

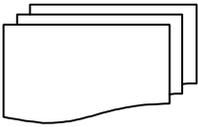
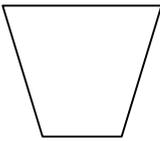
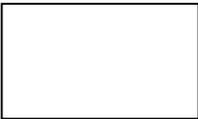
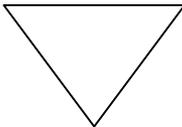
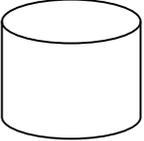
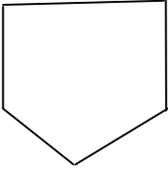
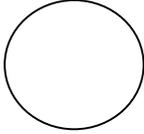
Kristanto (2008:75) menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.”

Kristanto (2008 : 75) menjelaskan, “Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

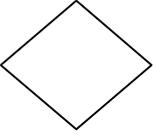
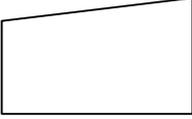
Tabel 2.2. Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.

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

No	Simbol	Keterangan
2.		Multi dokumen
3.		Proses Manual
4.		Proses dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (<i>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.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.

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

No	Simbol	Keterangan
10.		Terminal yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan (<i>Decision</i>).
12.		Layar peraga (<i>monitor</i>).
13.		Pemasukkan data secara manual.

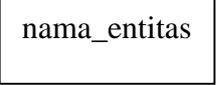
(Sumber : Kristanto, 2008:75)

2.2.3. Pengertian ERD (*Entity Relational Diagram*)

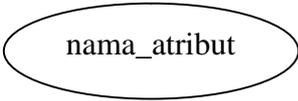
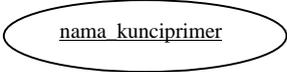
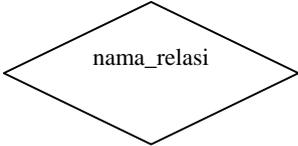
Shalahuddin et. al (2013:289), “*Entity Relationship Diagram (ERD)* adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relational.”

Shalahuddin et. al (2013:50), menjelaskan simbol-simbol yang digunakan dalam *ERD*, yaitu:

Tabel 2.3. Simbol-simbol *Entity Relationship Diagram (ERD)*

No	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya adar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.

Lanjutan Tabel 2.3. Simbol-simbol *Entity Relationship Diagram (ERD)*

No	Simbol	Keterangan
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3.	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)
4.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
5.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian 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

(Sumber : Shalahuddin et. al, 2013:50)

2.2.4. Pengertian *Flowchart*

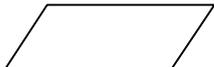
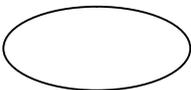
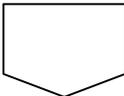
Saputra (2013:120), “*Flowchart* merupakan suatu diagram yang menggambarkan alur kerja suatu sistem.”



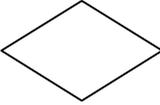
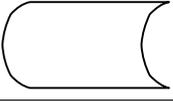
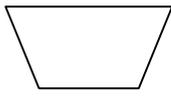
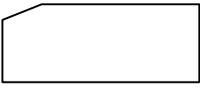
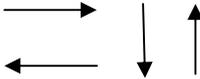
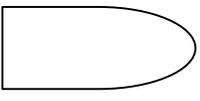
Siallagan (2009:6), “*Flow Chart* adalah suatu diagram alir yang mempergunakan simbol atau tanda untuk menyelesaikan masalah.”

Dari pengertian tersebut, dapat disimpulkan bahwa *Flowchart* adalah suatu diagram yang menggambarkan alur kerja suatu sistem dengan mempergunakan simbol atau tanda untuk menyelesaikan masalah.

Tabel 2.4. Simbol-simbol dalam *Flowchart*

No	Simbol	Keterangan
1.		Terminal menyatakan awal atau akhir dari suatu algoritma.
2.		Menyatakan proses.
3.		Proses yang terdefinisi atau sub program.
4.		Persiapan yang digunakan untuk memberi nilai awal suatu besaran.
5.		Menyatakan masukan dan keluaran (<i>input/output</i>).
6.		Menyatakan penyambung ke simbol lain dalam satu halaman.
7.		Menyatakan penyambung ke halaman lainnya.
8.		Menyatakan pencetakan (dokumen) pada kertas.

Lanjutan Tabel 2.4. Simbol-simbol dalam *Flowchart*

No	Simbol	Keterangan
9.		Menyatakan <i>desicion</i> (keputusan) yang digunakan untuk penyeleksian kondisi di dalam program.
10.		Menyatakan media penyimpanan drum magnetik.
11.		Menyatakan <i>input/output</i> menggunakan disket.
12.		Menyatakan operasi yang dilakukan secara manual.
13.		Menyatakan <i>input/output</i> dari kartu plong.
14.		Menyatakan arah aliran pekerjaan (proses).
15.		Multidocument (banyak dokumen).
16.		Delay (penundaan atau kelambatan).

(Sumber : Siallagan, 2009:6)

2.2.5. Pengertian Kamus Data

Shalahuddin et. al (2013:73), "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)."

Kristanto (2008:72) menjelaskan, "Kamus data adalah kumpulan elemen-



elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* di dalam sistem.”

Dari pengertian tersebut, dapat disimpulkan bahwa kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran setiap *field* atau *file* di dalam sistem agar dapat dipahami secara umum.

Shalahuddin et. al (2013:73), menjelaskan simbol-simbol yang di gunakan dalam kamus data, yaitu :

Table 2.5. Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	disusun atau terdiri atas
2	+	Dan
3	[]	baik ...atau...
4	{ } ⁿ	n kali diulang/ bernilai banyak
5	()	data operasional
6	*...*	batas komentar

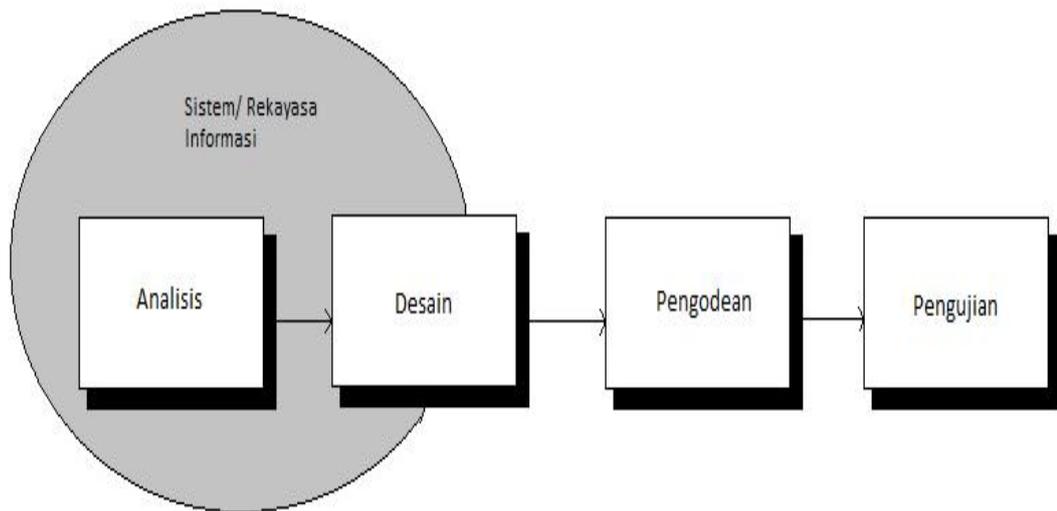
(Sumber : Shalahuddin et. al, 2013:73)

2.2.6. Tahap-Tahap Pengembangan Sistem

Pada tahapan ini penulis menggunakan *SDLC* atau *Software Development Life Cycle* yaitu proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem perangkat lunak. *SDLC* memiliki beberapa model penerapan tahapan prosesnya, disini kami memilih model *waterfall* (Shalahuddin et. al, 2013:26).

2.2.6.1. Model *Waterfall*

Shalahuddin et. al (2013:28), “model *waterfall* (air terjun) sering juga disebut model sekunsial linier atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup secara terurut mulai dari analisis, desain, pengkodean, pengujian.”



Sumber: Shalahuddin et. al, (2013:29)

Gambar 2.1. *Ilustrasi Model Waterfall*

2.3. Teori Judul

2.3.1. Pengertian Aplikasi

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

Sutabri (2012:147), “Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimiliki.”

Dari pengertian tersebut, dapat disimpulkan bahwa aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu untuk mengerjakan tugas-tugas tertentu.

2.3.2. Sekilas Tentang *e-commerce*

2.3.2.1. Pengertian *e-commerce*

Saputra (2013:116), “*Electronic Commerce* yang mempunyai definisi 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.



Shalahuddin et. al (2013:7), *Electronic Commerce* adalah penjualan atau pembelian barang dan jasa, antar perusahaan, rumah tangga, individu, pemerintah, dan masyarakat atau organisasi swasta lainnya, yang dilakukan melalui komputer pada media jaringan.

Sutabri (2012:108), “*Electronic Commerce* atau *e-commerce* adalah penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti *internet* atau televisi, *www*, atau jaringan komputer lainnya.”

Sutarman (2012:281), “*E-commerce* adalah pembelian, penjualan dan pertukaran barang atau layanan dan informasi secara elektronik, yaitu melalui jaringan komputer terutama *internet*.”

Dari pengertian tersebut, dapat disimpulkan bahwa *electronic commerce* atau *e-commerce* adalah penjualan atau pembelian barang dan jasa antar penjual dan pembeli melalui jaringan komputer terutama *internet*.

2.3.2.2. Model Transaksi *e-commerce*

Shalahuddin et. al (2013:35), *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. Jenis-jenis transaksi *e-commerce* di antaranya sebagai berikut :

1. *Collaborative Commerce (C-Commerce)*

Collaborative Commerce yaitu kerjasama secara elektronik antara rekan bisnis. Kerjasama ini biasanya terjadi antara rekan bisnis yang berada pada jalur penyediaan barang (*supply chain*).

2. *Business to Business (B2B)*

E-commerce tipe ini meliputi transaksi antarorganisasi yang dilakukan di *electronic market*. *Business to Business* memiliki karakteristik :

- a. *Trading partners* yang sudah diketahui dan umumnya memiliki hubungan yang cukup lama. Informasi hanya dipertukarkan dengan partner tersebut. Dikarenakan sudah mengenal rekan komunikasi, jenis informasi yang dikirimkan dapat disusun sesuai dengan kebutuhan dan kepercayaan (*trust*).



-
- b. Pertukaran data (*data exchange*) berlangsung berulang-ulang dan secara berkala, misalnya setiap hari, dengan format data yang sudah disepakati bersama. Dengan kata lain, servis yang digunakan sudah tertentu. Hal ini memudahkan pertukaran data untu dua entiti yang menggunakan standar yang sama.
 - c. Salah satu pelaku dapat melakukan inisiatif untu mengirimkan data, tidak harus menunggu partnernya.
 - d. Model yang umum digunakan adalah *peer-to-peer*, di mana *processing intelligence* dapat di distribusikan pada kedua pelaku bisnis.

3. ***Business to Consumers (B2C)***

Business to Consumers yaitu penjual adalah suatu organisasi dan pembeli adalah individu. B2C memiliki karakteristik sebagai berikut :

- a. Terbuka untuk umum, dimana informasi disebarkan ke umum;
- b. Servis yang diberikan bersifat umum (*generic*). Sebagai contoh, karea sistem *web* sudah umum digunakan maka sevis diberikan dengan menggunakan basis *web*.
- c. Servis diberikan berdasarkan permohonan (*on demand*). Komsumer melakukan inisiatif dan produser harus siap memberikan respon sesuai dengan permohonan.

4. ***Consumer to Business (C2B)***

Dalam *C2B* konsumen memberitahukan kebutuhan atau atas suatu produk atau jasa tertentu, dan para pemasok bersaing untu menyediakan produk atau jasa tersebut ke konsumen. Contohnya di *priceline.com*, dimana pelanggan menyebutkan produk dan harga yang diinginkan, dan *priceline* mencoba menemukan pemasok yang memenuhi kebutuhan tersebut.

5. ***Customer to Customer (C2C)***

C2C yaitu konsumen menjual secara langsung ke konsumen lain atau mengiklankan jasa pribadi di internet. Dalam *C2C* seseorang menjual produk dan jasa ke orang lain. Dapat juga disebut sebagai pelanggan ke pelanggan, yaitu orang yang menjual produk dan jasa ke satu sama lain.



2.3.2.3. Keuntungan *e-commerce*

Shalahuddin et. al (2013:13), keuntungan *e-commerce* :

1. *Revenue stream* (aliran pendapatan) baru yang mungkin lebih menjanjikan yang tidak bisa ditemui di sistem transaksi tradisional.
2. Dapat meningkatkan *market exposure* (pangsa pasar).
3. Menurunkan biaya operasional (*operating cost*).
4. Melebarkan jangkauan (*global reach*).
5. Meningkatkan *customer loyalty*.
6. Meningkatkan *supplier management*.
7. Memperpendek waktu produksi.
8. Meningkatkan *value chain* (mata rantai pendapatan).

2.3.2.4. Kerugian *e-commerce*

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 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 lain oleh pihak yang tidak berhak. Misalkan seorang *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 dengan sengaja oleh pihak lain yang berusaha menjatuhkan reputasi perusahaan tersebut.
6. Kerugian yang tak terduga. Disebabkan oleh gangguan yang dilakukan dengan sengaja, ketidakjujuran, praktik bisnis yang tidak benar, kesalahan faktor manusia, atau keasalahan sistem elektronik.



2.3.3. Pengertian Tupperware

Tupperware adalah nama merek dari peralatan rumah tangga yang terbuat dari plastik, termasuk didalamnya, wadah penyimpanan, wadah penyajian dan beberapa peralatan dapur yang diperkenalkan untuk khalayak umum pada tahun 1946.

2.3.4. Pengertian Aplikasi *e-commerce* Tupperware pada PT. Cahaya Putri Selatan Palembang

Aplikasi *e-commerce* Tupperware pada PT. Cahaya Putri Selatan Palembang adalah aplikasi yang berfungsi untuk mengolah transaksi penjualan produk Tupperware secara *online* pada PT. Cahaya Putri Selatan Palembang meliputi promosi produk, pemesanan produk serta penjualan produk sehingga memudahkan konsumen dalam berbelanja produk Tupperware tanpa harus datang langsung ke lokasi penjualan produk dan memudahkan pemilik usaha dalam menjual produk yang dimilikinya.

2.4. Teori Program

2.4.1. Sekilas Tentang Basis Data (*Database*)

2.4.1.1. Pengertian Basis Data

Priyadi (2014:2), “Basis data adalah sekumpulan fakta berupa representasi tabel yang saling berhubungan dan disimpan dalam media penyimpanan secara digital.”

Shalahuddin et. al (2013:43), “Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan.”

Heryanto (2012:1), “*Database* (basis data) adalah kumpulan dari tabel. Satu tabel mempresentasikan suatu entitas tertentu. Suatu entitas terdiri dari beberapa atribut.”

Dari pengertian tersebut, dapat disimpulkan bahwa *database* (basis data) adalah sekumpulan fakta berupa representasi tabel yang saling berhubungan dan



disimpan dalam media penyimpanan agar mempermudah pengguna dalam mendapatkan informasi saat dibutuhkan.

2.4.1.2. Pengertian Tabel

Priyadi (2014:3), “Tabel direpresentasikan menjadi suatu bentuk segiempat berupa matriks, yang terdiri dari kolom dan baris.”

Heryanto (2012:3), “Tabel adalah kumpulan dari kolom atau *field* dan baris atau *record*.”

Dari pengertian diatas, dapat disimpulkan bahwa tabel adalah kumpulan dari kolom dan baris.

Berikut adalah tampilan dari tabel :

id	id_pegawai	no_ktp	nama	tempat	tgl_lahir	jk	alamat	status	jumlah_anak
2	P002	1234	Raya Puspita Sari	Palembang	1993-03-18	Perempuan	jl seruni	Belum Menikah	0
3	P003	123213	Daffa' Jiyad Pratama	Palembang	2003-10-17	Laki-laki	plaju	Belum Menikah	0

Gambar 2.2. Tampilan Tabel

2.4.1.3. Pengertian *Field* (Kolom)

Priyadi (2014:3), “Kolom merupakan representasi untuk sebuah nama *field* yang pembacaan datanya dilakukan secara vertikal.”

Heryanto (2012:3), “*Field* atau kolom merupakan atribut dari tabel.”

Dari pengertian tersebut, dapat disimpulkan bahwa kolom atau *field* adalah atribut dari tabel yang pembacaan datanya dilakukan secara vertikal.



Berikut adalah tampilan dari *field* :

id	id_pegawai	no_ktp	nama	tempat	tgl_lahir	jk	alamat	status	jumlah_anak
2	P002	1234	Raya Puspita Sari	Palembang	1993-03-18	Perempuan	jl seruni	Belum Menikah	0
3	P003	123213	Daffa' Jiyad Pratama	Palembang	2003-10-17	Laki-laki	plaju	Belum Menikah	0

Gambar 2.3. Tampilan *Field*

2.4.1.4. Pengertian *Record* (Baris)

Priyadi (2014:4), Baris merupakan 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 tabek tersebut.

Heryanto (2012:3), “*Record* atau baris merupakan isi dari tabel.”

Dari pengertian tersebut dapat disimpulkan bahwa baris atau *record* adalah isi dari tabel yang pembacaan datanya dilakukan secara horizontal.

Berikut adalah tampilan dari *record* :

id	id_pegawai	no_ktp	nama	tempat	tgl_lahir	jk	alamat	status	jumlah_anak
2	P002	1234	Raya Puspita Sari	Palembang	1993-03-18	Perempuan	jl seruni	Belum Menikah	0
3	P003	123213	Daffa' Jiyad Pratama	Palembang	2003-10-17	Laki-laki	plaju	Belum Menikah	0

Gambar 2.4. Tampilan *Record*

2.4.2. Pengertian *HTML*

Asropudin (2013:44), “*HTML* atau *Hyper Text Markup Language* adalah bahasa komputer yang digunakan untuk membuat sebuah halaman *webpage*.”



Badiyanto (2013:23), *HTML (Hyper Text Markup Language)* adalah bahasa standar pemrograman untuk membuat halaman *web* yang terdiri dari kode-kode *tag* tertentu, kemudian kode-kode tersebut diterjemahkan oleh *web browser* untuk menampilkan halaman *web* yang terdiri dari beberapa macam format tampilan seperti teks, grafik, animasi *link*, maupun audio-video.

Saputra (2013:1), “*HTML* merupakan bahasa pemrograman yang digunakan untuk membangun kerangka atau pondasi *web*.”

Saputra (2012:1), “*HTML* merupakan singkatan dari *Hyper Text Markup Language*. *HTML* bisa disebut bahasa paling dasar dan penting yang digunakan untuk menampilkan dan mengelola tampilan pada halaman website.”

Sutarman (2012:163), “ *HTML* adalah bahasa standar yang digunakan untuk pembuatan halaman *web* atau *word wide wibe*, dengan *hypertext* dan informasi lain yang akan ditampilkan pada halaman *web*.”

Dari pengertian diatas, dapat disimpulkan bahwa *HTML* adalah bahasa pemrograman yang digunakan untuk membuat sebuah halaman *web*.

Berikut adalah *tag-tag* dalam *HTML* :

Tabel 2.6. Tag HTML

No	Tag HTML	Keterangan
1.	<html>	Tag Skrip HTML
2.	<head>	Tag bagian kepala HTML
3.	<title>	Tag untuk menempatkan judul
4.	<body>	Skrip untuk isi HTML
5.	<h1> sampai <h6>	Tag untuk menentukan ukuran <i>font</i> pada judul atau subjudul pada halaman <i>web</i>
6.	<p>	Tag untuk pemberian paragraph
7.	 	Tag untuk pemberian pindah baris, seperti <i>enter</i>

(Sumber: Badiyanto, 2013:23)



Lanjutan Tabel 2.6. Tag HTML

No	Tag HTML	Keterangan
8.		Tag menampilkan teks dengan <i>font</i> mencetak <i>font</i> tebal (<i>bold</i>)
9.	<i>	Tag menampilkan teks dengan <i>font</i> cetak miring (<i>italic</i>)
10.	<u>	Tag menampilkan teks dengan <i>font</i> garis bawah (<i>underline</i>)
11.		Tag menampilkan teks dengan <i>font</i> cetak menjadi tebal (mirip <i>bold</i>)
12.	<code>	Tag menampilkan teks dengan <i>font</i> cetak huruf menjadi <i>font courier</i>
13.	<tt>	Tag menampilkan teks dengan <i>font</i> cetak huruf menjadi <i>font teletype writer</i>
14.	<samp>	Tag menampilkan teks dengan <i>font</i> cetak huruf menjadi <i>font code</i>
15.	<pre>	Memformat tampilan teks semua spasi yang bukan <i>script</i> akan sama letaknya dengan yang ditulis
16.		Membuat nomor <i>bullet</i> , pada baris berikutnya harus ditulis sebelum menulis teks
17.		Membuat nomor urut angka
18.	<table border>	Gunanya untuk membuat table
19.	<tr>	Tag membuat baris dalam table
20.	<td>	Tag membuat kolom dalam table
21.	<th>	Tag <i>table header</i> , membuat kolom judul

(Sumber: Badiyanto, 2013:23)

2.4.3. Pengertian CSS

Badiyanto (2013:25), “CSS adalah kepanjangan dari *Cascading Style Sheets*. CSS skrip yang berisi rangkaian instruksi yang menentukan suatu teks akan tertampil di halaman *web browser*.”

Saputra (2012:27), CSS yang merupakan singkatan dari *Cascading Style Sheet* merupakan bahasa pemrograman *web* yang didesain khusus untuk mengendalikan dan membangun berbagai komponen dalam *web* sehingga tampilan *web* lebih rapih, terstruktur, dan seragam.



Dari pengertian tersebut, dapat disimpulkan bahwa CSS adalah skrip yang berisi rangkaian instruksi yang didesain khusus untuk mengendalikan dan membangun berbagai komponen dalam *web* sehingga tampilan *web* lebih rapi, terstruktur dan seragam.

2.4.4. Pengertian MySQL

Badiyanto (2013:57), “MySQL merupakan sebuah *database server SQL multiuser dan multi threaded*.”

Saputra (2013:14), “MySQL merupakan *database storage engine* yang paling banyak digunakan oleh *web developer* karena sifatnya yang *free*, alias gratis.”

Kadir (2008:02), “MySQL atau dibaca mai-se-Kyu-el merupakan yang tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source*. *Open Source* menyatakan bahwa *software* ini dilengkapi dengan *source code* (kode yang dibuat untuk membuat MySQL), selain bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi.”

Dari pengertian tersebut, dapat disimpulkan bahwa MySQL adalah sebuah *database server SQL multiuser dan multi threaded* yang sifatnya gratis.

2.4.5. Sekilas Tentang PHP

2.4.5.1. Pengertian PHP (*Hypertext Preprocessor*)

Badiyanto (2013:32), “PHP adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML/PHP banyak dipakai untuk membuat situs *web* dinamis.”

Saputra (2013:1), PHP (dibaca : *PHP Hypertext Preprocessor*) merupakan suatu bahasa pemrograman yang hanya berjalan pada sisi *server* (*Server Side Scripting*). Artinya proses yang dibuat dengan PHP tidak akan berjalan tanpa menggunakan *web server*.

Kadir (2008:2), “ PHP adalah bahasa berbentuk skrip yang ditempatkan dalam *server* dan diproses di *server*, hasilnya dikirim ke klien.”

Dari pengertian tersebut, dapat disimpulkan bahwa PHP adalah bahasa pemrograman yang berjalan dalam sebuah *webserver* yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis.



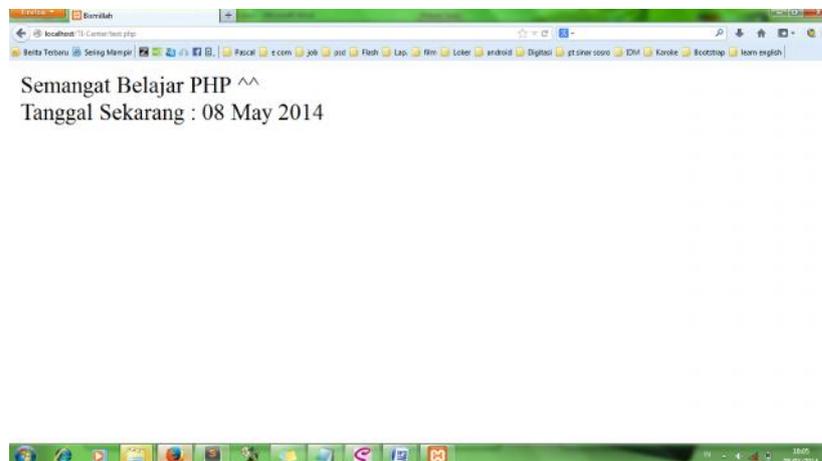
2.4.5.2. Skrip PHP

Badiyanto (2013:32), “*skrip PHP, PHP* yang merupakan sebuah bahasa *scripting* yang terpasang pada *HTML* dan skrip *HTML (Hyper Text Markup Language)* adalah bahasa standar untuk membuat halaman *web*.”

Berikut ini contoh sintaks *PHP* yang berada di dalam kode *HTML* :

```
< HTML>
<HEAD>
    <TITLE>Bismillah</TITLE>
</HEAD>
<BODY>
    Semangat Belajar PHP ^^<BR>
    <?php
    printf("Tanggal Sekarang : %s", Date("d F Y")); ?>
</BODY>
</HTML>
```

Bila dijalankan melalui *browser*, kode tersebut membentuk tampilan seperti berikut:



Gambar 2.5. Tampilan PHP Sederhana

2.4.6. Sekilas tentang *Sublime Text*

2.4.6.1. Pengertian *Sublime Text*

Sibarani (2013: 24), “*Sublime Text 2.0.2* adalah editor teks yang digunakan untuk menerjemahkan hasil analisa dan rancangan kedalam bentuk *code*.”



Berikut beberapa fitur yang diunggulkan dari aplikasi *Sublime Text*:

a. *Goto Anything*

Fitur yang sangat membantu dalam membuka *file* ataupun menjelajahi isi dari *file* hanya dengan beberapa *keystrokes*.

b. *Multiple Selections*

Fitur ini memungkinkan *user* untuk mengubah secara interaktif banyak baris sekaligus, mengubah nama variabel dengan mudah, dan memanipulasi *file* lebih cepat dari sebelumnya.

c. *Command Pallete*

Dengan hanya beberapa *keystrokes*, *user* dapat dengan cepat mencari fungsi yang diinginkan, tanpa harus menavigasi melalui menu.

d. *Distraction Free Mode*

Bila *user* memerlukan fokus penuh pada aplikasi ini, fitur ini dapat membantu *user* dengan memberikan tampilan layar penuh. *Split Editing* dapatkan hasil yang maksimal dari *monitor* layar lebar dengan dukungan *editing* perpecahan. Mengedit sisi *file* dengan sisi, atau mengedit dua lokasi di satu *file*. Anda dapat mengedit dengan banyak baris dan kolom yang *user* inginkan.

e. *Instant Project Switch*

Menangkap semua *file* yang dimasukkan ke dalam *project* pada aplikasi ini. Terintegrasi dengan fitur *Goto Anything* untuk menjelajahi semua *file* yang ada ataupun untuk beralih ke *file* dalam *project* lainnya dengan cepat.

f. *Plugin API*

Dilengkapi dengan *plugin* API berbasis *Python* sehingga membuat aplikasi ini sangat tangguh.

g. *Customize Anything*

Aplikasi ini memberikan *user* fleksibilitas dalam hal pengaturan fungsional dalam aplikasi ini.

h. *Cross Platform*

Aplikasi ini dapat berjalan hampir di semua *operating system modern* seperti *Windows*, *OS X*, dan *Linux based operating system*.



Gambar 2.6. Ruang kerja dari Sublime Text 2

2.4.6.2. Komponen Sublime Text

a. Minimap

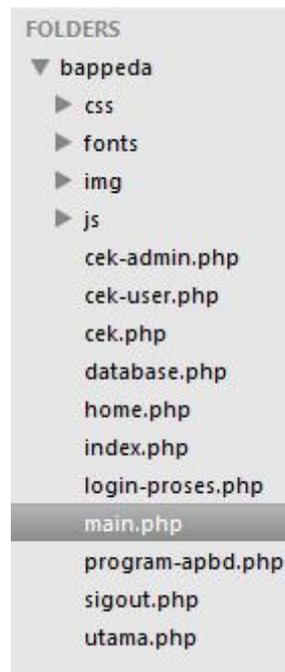
Minimap merupakan fitur menunjukkan kode pengguna di seluruh sudut kanan atas layar. Hal ini memungkinkan mereka untuk melihat garis besar keseluruhan kode mereka dan memiliki fungsi untuk melompat antara bagian yang berbeda dari *file*.



Gambar 2.7. Tampilan *Minimap* pada Sublime Text 2.0.2

b. Side Bar

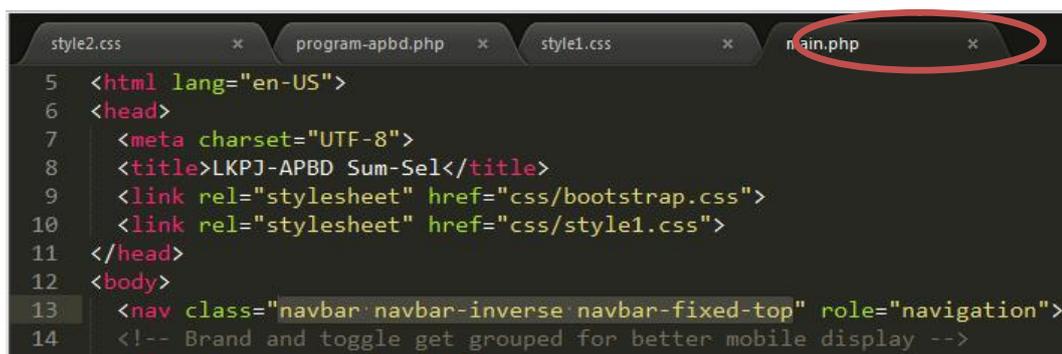
Komponen dalam sublime text yang berfungsi menunjukkan file yang sedang terbuka pada sublime text. Memungkinkan pengguna dalam memilih file – file yang terbuka secara cepat.



Gambar 2.8. Tampilan *Side Bar* pada *Sublime Text 2.0.2*

c. Tabs

Salah satu komponen pada *sublime text* yang berfungsi menampilkan nama file yang diubah dalam *sublime text*, dan memudahkan penggantian panel dengan mengklik tab yang diinginkan.



Gambar 2.9. Tampilan *Tabs* pada *Sublime Text 2.0.2*

d. Status Bar

Menunjukkan posisi baris dan kolom *syntax* yang sedang diubah dalam *sublime text*. Memudahkan pengguna mengetahui posisi *syntax* yang mengalami *error* setelah dijalankan.



Gambar 2.10. Tampilan *Status Bar* pada *Sublime Text 2.0.2*

e. Menu Bar

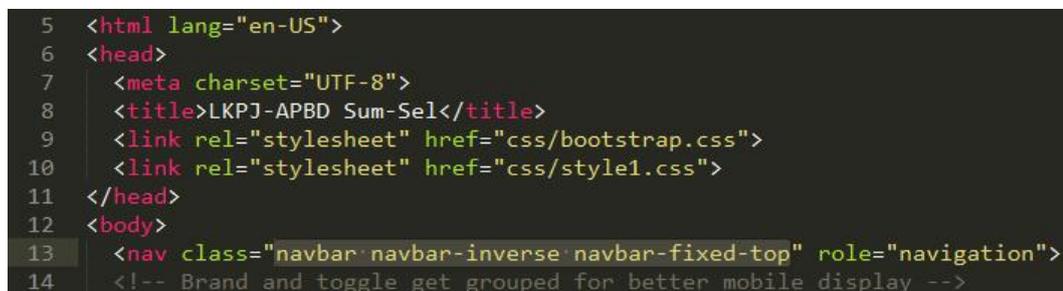
Berisi menu-menu yang digunakan untuk membantu dalam membuat dan mengubah syntax. Menu dalam sublime text, yaitu *file*, *edit*, *selection*, *find*, *view*, *goto*, *tools*, *project*, *preferences* dan *help*.



Gambar 2.11. Tampilan *Menu Bar* pada *Sublime Text 2.0.2*

f. Pane

Merupakan lembar kerja pada sublime text. Berisi kumpulan syntax dari file yang digunakan pada sublime text.



Gambar 2.12. Tampilan *Pane* pada *Sublime Text 2.0.2*

2.4.7. Javascript

Badiyanto (2013:27), *JavaScript* adalah bahasa pemrograman *java*, perintah-perintahnya ditulis dengan kode yang disebut skrip. *Java* adalah Bahasa pemrograman berorientasi objek, sedangkan *script* adalah serangkaian intruksi program.

Saputra (2013:3), “*Javascript* adalah bahasa *scripting* yang handal yang berjalan di sisi *client*.”