



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”.

Sutarman (2012:3), “komputer adalah alat yang dapat membaca input data dan mengolahnya sesuai dengan program yang ditetapkan untuk menghasilkan informasi yang merupakan output hasil pemrograman input data”.

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”.

Siallagan (2009:1), “komputer adalah sebagai sekumpulan alat elektronik yang saling bekerja sama, dapat menerima data (*input*), mengolah data (*proses*), memberikan informasi (*output*), dan terkoordinasi di bawah kontrol program yang tersimpan dalam memorinya”.

Dari pengertian diatas, dapat disimpulkan bahwa komputer adalah sekumpulan alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima data (*input*), memproses data sesuai dengan instruksi yang diberikan, dan menghasilkan output berupa informasi.

2.1.2. Pengertian *Software*

Sukanto dan Shalahuddin (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 diatas, dapat disimpulkan bahwa *software* adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak yang memungkinkan *hardware* memproses data dan mengoperasikan suatu pekerjaan sesuai dengan yang dikehendaki.

2.1.3. Pengertian Program

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

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

Dari pengertian diatas, dapat disimpulkan bahwa program adalah kumpulan instruksi atau perintah-perintah yang disusun sehingga dapat dipahami oleh komputer untuk mengoperasikan atau menjalankan *hardware*.

2.1.4. Pengertian *Internet*

Asropudin (2013:48), “*internet* adalah istilah umum yang dipakai untuk menunjukkan *network* tingkat dunia yang terdiri dari komputer dan layanan servis atau sekitar 30 sampai 50 juta pemakai komputer dan puluhan sistem informasi termasuk *e-mail*, *Gopher*, *FTP* dan *World Wide Web*”.



Ahmadi dan Dadang (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”.

Dari pengertian diatas, dapat disimpulkan bahwa *internet* adalah jaringan komunikasi global yang menghubungkan jaringan komputer dan fasilitas komputer yang terorganisasi melalui media telekomunikasi seperti telepon atau satelit.

2.1.5. Metode Pengembangan Sistem

Sukanto dan Shalahuddin (2013:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga disebut model sekuensial linear (*sequential linear*) atau alur hidup klasik (*clasic life cycle*). Metode air terjun (*waterfall*) menyediakan pendekatan alur hidup terurut sebagai berikut:

1. Analisis

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user* kemudian didokumentasikan.

2. Desain

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

3. Pembuatan Kode Program (Pengodean)

Desain harus diterjamhkan kedalam program perangkat lunak. Hasil dari tahap ini adalah program sesuai dengan desain yang dibuat.



4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung atau pemeliharaan sistem

Pada tahap ini tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirim ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak yang baru.

2.1.6. Metode Pengujian Sistem

Yakub (2012:150) menjelaskan tentang metode pengujian sistem yaitu, *black box*. *Black box* merupakan pendekatan komplementer dari teknik *white box testing*, karena pengujian *black box* berfokus pada pengujian persyaratan fungsional perangkat lunak, karena untuk mendapatkan serangkaian kondisi *input* yang sesuai dengan persyaratan fungsional suatu program.

2.2. Teori Judul

2.2.1. Konsep Dasar Sistem

2.2.1.1. Pengertian Sistem

Sutabri (2012:3), “sistem merupakan suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu”.

Sutarman (2012:5), “sistem adalah kumpulan elemen yang saling berhubungan dan berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama”.

Kristanto (2008:1), “sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu”.

Dari pengertian diatas, dapat disimpulkan bahwa sistem adalah suatu kumpulan elemen-elemen yang saling berhubungan dan saling tergantung satu sama lain untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

2.2.1.2. Karakteristik Sistem

Sutabri (2012:13) menjelaskan tentang karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya:

1. Komponen Sistem (*Component*) : Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan.
2. Batasan Sistem (*Boundary*) : Daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya.
3. Lingkungan Luar Sistem (*Environment*) : Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem.
4. Penghubung Sistem (*Interface*) : Media yang menghubungkan sistem dengan subsistem yang lain.
5. Masukan Sistem (*input*) : Energi yang dimasukkan ke dalam sistem yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).
6. Keluaran Sistem (*output*) : Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.
7. Pengolahan Sistem (*process*) : Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.
8. Sasaran Sistem (*Objective*) : Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik.

2.2.1.3. Klasifikasi Sistem

Sutabri (2012:15) menjelaskan bahwa sistem dapat diklasifikasikan menjadi beberapa beberapa bagian, yaitu:

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan, sedangkan sistem fisik merupakan sistem fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sedangkan sistem buatan manusia berupa sistem yang melibatkan hubungan manusia dengan mesin, yang dibuat dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

3. Sistem Deterministik dan Sistem Probabilistik

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

4. Sistem Terbuka dan Sistem Tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.2.1.4. Tahap-tahap Pengembangan Sistem

Sutabri (2012:59) menjelaskan tahapan utama dalam proses pengembangan sistem adalah sebagai berikut:

1. Investigasi Sistem

Tahap investigasi sistem menentukan masalah-masalah atau kebutuhan yang timbul. Pada tahap ini sistem yang ada tetap berjalan tanpa perlu perubahan maupun pembangunan sistem yang baru, hanya melakukan perbaikan-perbaikan pada sistem tanpa harus menggantinya.

2. Analisis Sistem

Tahap analisis sistem mempelajari lebih mendalam sistem yang sedang berjalan, konsepsi dan usulan dibuat untuk menjadi landasan bagi sistem yang baru yang akan dibangun.

3. Desain Sistem

Tahap desain sistem meninjau dan menyempurnakan kembali spesifikasi perangkat keras dan perangkat lunak yang telah disusun pada tahap sebelumnya. Sebagian besar kegiatan pada tahap ini berorientasi ke komputer dan juga dilakukan testing program.

4. Implementasi Sistem

Tahap implementasi sistem adalah tahap prosedur yang dilakukan untuk menyelesaikan desain sistem yang ada dalam dokumen desain sistem yang disetujui dan menguji, menginstall dan memulai sistem baru atau diperbaiki.

5. Pemeliharaan Sistem

Tahap pemeliharaan sistem adalah bentuk evaluasi untuk memantau supaya sistem informasi yang dioperasikan dapat berjalan secara optimal dan sesuai dengan harapan pemakai yang menggunakan sistem tersebut.

2.2.2. Pengertian Informasi

Sutarman (2012:4), “informasi (*information*) adalah hasil kegiatan pengolahan data yang memberikan bentuk yang lebih berarti dari suatu kejadian”.

Sutarman (2012:13), “informasi adalah fakta (*data*) yang diorganisasikan dengan cara tertentu sehingga mereka mempunyai arti bagi si penerima”.



Kristanto (2008:7), “informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima”.

Dari pengertian diatas, dapat disimpulkan bahwa informasi adalah hasil dari kegiatan pengolahan data dengan cara tertentu sehingga mempunyai arti bagi penerimanya.

2.2.3. Pengertian Sistem Informasi

Sutabri (2012:38), “sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu”.

O’Brian dalam Yakub (2012:17), “sistem informasi merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi”.

2.2.4. Pengertian Ikatan

Salim (2013:269), “ikatan adalah alat untuk mengikat (menyatukan, memberkas, menggabungkan)”.

Maulina dan Setya (2013: 247), “ikatan adalah perserikatan, perkumpulan, rangkaian dan pertalian”.

Dari pengertian diatas, dapat disimpulkan bahwa ikatan adalah alat atau cara yang digunakan untuk mengikat, menggabungkan, merangkai perserikatan atau perkumpulan.

2.2.5. Pengertian Alumni

Salim (2013:31), “alumni adalah tamatan atau lulusan suatu sekolah atau perguruan tinggi”.

Maulina dan Setya (2013:28), “alumni adalah orang-orang yang pernah belajar di sekolah atau perguruan tinggi”.



Dari pengertian diatas, dapat disimpulkan bahwa alumni adalah orang-orang yang telah mengikuti dan lulus dari suatu sekolah atau perguruan tinggi.

2.2.6. Pengertian Web

Ahmadi dan Dadang (2013:68), “*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 diatas dapat disimpulkan bahwa *web* adalah kumpulan halaman informasi yang disediakan melalui jalur *internet* dan dapat diakses di seluruh dunia selama terkoneksi dengan jaringan *internet*.

2.2.6. Pengertian Sistem Informasi Ikatan Alumni

Sistem Informasi Ikatan Alumni adalah suatu program yang dibuat untuk alumni dan Jurusan Manajemen Informatika agar dapat memperoleh informasi tentang data-data alumni Jurusan Manajemen Informatika dan media untuk berbagi informasi yang bertujuan untuk pengembangan Jurusan manajemen Informatika yang dapat diakses dengan menggunakan media *internet*.

2.3. Teori Khusus

2.3.1. Data Flow Diagram (DFD)

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”.


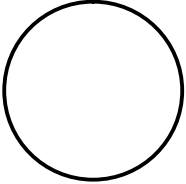
Sukamto dan Shalahuddin (2013:70), “*Data Flow Diagram (DFD)* atau dalam bahasa indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang

diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

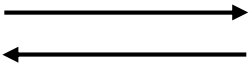

Sutabri (2012:117), “*Data Flow Diagram* adalah suatu *network* yang menggambarkan suatu sistem automat/komputerisasi, manulisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya”.

Sukanto dan Shalahuddin (2013:71), menjelaskan notasi pada *Data Flow Diagram* adalah sebagai berikut:

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

Nama	Simbol	Keterangan
<i>External Entity</i>		<p><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>
<i>Proses</i>		<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 : nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>

Lanjutan Tabel 2.1. Simbol-simbol *Data Flow Diagram*

Nama	Simbol	Keterangan
<i>Data Flow</i>		<p><i>Data Flow</i>, 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 :</p> <p>nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data pegawai” atau tanpa kata data misalnya “pegawai”.</p>
<i>Data Storage</i>		<p><i>Data Storage</i>, <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</i> (ERD), <i>Conceptual Data Model</i>(CDM), <i>Physical Data Model</i> (PDM).</p>

(Sumber: Sukamto dan Shalahuddin, 2013:71)

Sukamto dan Shalahuddin (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

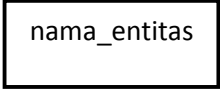
DFD Level 3,4,5, dan seterusnya merupakan *breakdown* dari modul pada *DFD Level* di-atasnya. *Breakdown* pada *level 3,4,5*, dan seterusnya aturannya sama persis dengan *DFD Level 1* atau 2.

2.3.2. *Entity Relationship Diagram (ERD)*

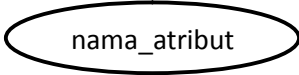
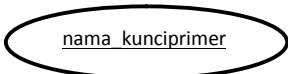
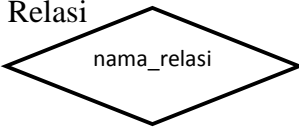
Sukamto dan Shalahuddin (2013:50) menjelaskan bahwa *Entity Relationship Diagram (ERD)* digunakan untuk pemodelan basis data relasional. *Entity Relationship Diagram* memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker), Ian (dikembangkan oleh Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lainnya.

Berikut adalah simbol-simbol yang digunakan pada *Entity Relationship Diagram (ERD)* dengan notasi Chen.

Tabel 2.2. 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 agar dapat diakses oleh aplikasi komputer;

Lanjutan Tabel 2.2. Simbol-simbol *Entity Relationship Diagram*


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.

(Sumber : Sukamto dan Shalahuddin, 2013:50)




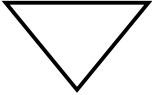


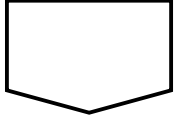
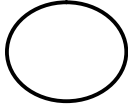

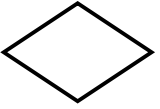


2.3.3. Blockchart

Kristanto (2008:75), *blockchart* adalah satu model logika data yang 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*, yaitu:

Tabel 2.3. Simbol-simbol dalam *Blockchart*

Simbol	Keterangan
	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.

Lanjutan Tabel 2.3. Simbol-simbol dalam *Blockchart*

Simbol	Keterangan
	Multi dokumen.
	Proses manual.
	Proses yang dilakukan oleh komputer.
	Menandakan dokumen yang diarsipkan (arsip manual).
	Data penyimpanan (<i>data storage</i>).
	Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
	Terminasi yang menandakan awal dan akhir dari suatu aliran.
	Pengambilan keputusan (<i>decision</i>).
	Layar peraga (<i>monitor</i>).
	Pemasukan data secara manual.

(Sumber : Kristanto, 2008,75)





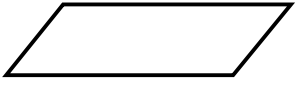
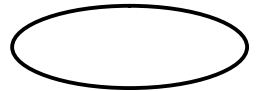


2.3.4. Flowchart

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

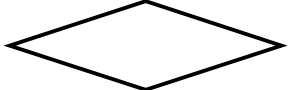



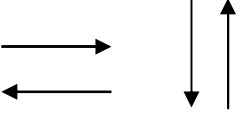


Siallagan (2009:6), “*Flowchart* adalah suatu diagram alir yang menggunakan simbol atau tanda untuk menyelesaikan masalah”.

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

Tabel 2.4. Simbol-simbol dalam *Flowchart*

Simbol <i>Flowchart</i>	Fungsi <i>Flowchart</i>
	Terminal menyatakan awal atau akhir dari suatu algoritma.
	Menyatakan Proses.
	Proses yang terdefinisi atau sub program.
	Persiapan yang digunakan untuk memberi nilai awal suatu besaran.
	Menyatakan masukan dan keluaran (<i>input/output</i>)
	Menyatakan penyambung ke simbol lain dalam satu halaman.
	Menyatakan penyambung ke halaman lainnya.
	Menyatakan pencetakan (dokumen) pada kertas.

Lanjutan Tabel 2.4. Simbol-simbol dalam *Flowchart*

Simbol <i>Flowchart</i>	Fungsi <i>Flowchart</i>
	Menyatakan <i>decision</i> (keputusan) yang digunakan untuk penyeleksian kondisi di dalam program.
	Menyatakan media penyimpanan drum magnetik.
	Menyatakan <i>input/output</i> menggunakan disket.
	Menyatakan <i>input/output</i> dari kartu plong.
	Menyatakan arah alir pekerjaan (proses).
	<i>Multidocument</i> (banyak dokumen)
	<i>Delay</i> (penundaan atau kelambatan).

(Sumber : Siallagan, 2009:6)

2.3.5. Kamus Data

Sukamto dan Shalahuddin (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 penulisan)”.

Kristanto (2008:72), “kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* di dalam sistem”.



Sukamto dan Shalahuddin (2013:74), menjelaskan bahwa kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan, yaitu sebagai berikut:

Tabel 2.5. Simbol-simbol dalam kamus data

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

(Sumber : Sukamto dan Shalaluddin, 2013:73)

2.4. Teori Program

2.4.1. Basis Data (*Database*)

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

Sukamto dan Shalaludin (2013:43), “basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”.

Priyadi (2014:3) menjelaskan beberapa elemen basis data, di antaranya 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 vertikal.

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

Database berfungsi untuk menyimpan informasi atau data. Untuk mengenal *database* diperlukan *software* yang sering disebut dengan *DBMS (Database Management System)*. Dengan *DBMS* pengguna dapat membuat, mengolah, mengontrol, dan mengakses *database* dengan mudah, praktis dan efisien.

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 (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*”.

2.4.3. Sekilas Tentang *PHP*

2.4.3.1. Pengertian Pemrograman *PHP* (*Hypertext Preprocessor*)

Saputra (2013:1), “*PHP* adalah singkatan dari (*PHP Hypertext Preprocessor*) yang merupakan suatu bahasa pemrograman yang hanya dapat berjalan pada sisi *server* (*Server Side Scripting*)”.

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

2.4.3.2. Sintaks *PHP*

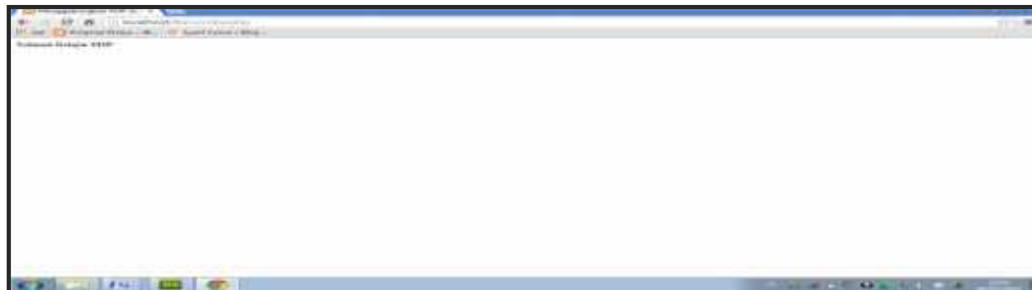
Badiyanto (2013:32), “*PHP* yang merupakan sebuah bahasa *scripting* yang terpasang pada *HTML* dan skrip *HTML* adalah bahasa standar untuk membuat halaman *web*”.

Sintaks *PHP* berkedudukan sebagai tag dalam bahasa *HTML*. Sebagaimana diketahui, *HTML* adalah bahasa standar untuk membuat halaman-halaman *web*.

Berikut adalah sintaks *PHP* yang ditulis didalam skrip *HTML*.

```
<html>
<head>
<title> Menggabungkan PHP dan HTML </title>
</head>
<body>
    <?php
        echo “Selamat Belajar PHP”;
    ?>
</body>
</html>
```

Bila dijalankan melalui browser, skrip tersebut membentuk tampilan seperti berikut:



Gambar 2.1. Tampilan PHP Sederhana

2.4.4. Sekilas Tentang MySQL

2.4.4.1. Pengertian MySQL

Kadir (2008:348), “*MySQL* adalah salah satu jenis *database server* yang sangat terkenal. *MySQL* menggunakan SQL sebagai bahasa dasar untuk mengakses *databasenya*, dan bersifat *open Source* pada berbagai platform (kecuali untuk jenis *enterprise*, yang bersifat komersial”.

MySQL memiliki beberapa kelebihan, antara lain:

1. *MySQL* dapat digunakan oleh beberapa *user* dalam waktu bersamaan tanpa mengalami masalah.
2. *MySQL* memiliki kecepatan yang bagus dalam menangani *query* sederhana.
3. *MySQL* memiliki operator dan fungsi secara penuh dan mendukung perintah *select* dan *where* dalam perintah *Query*.
4. *MySQL* memiliki keamanan yang bagus karena beberapa lapisan sekuritas seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perjanjian yang mendetail serta sandi terenkripsi.
5. *MySQL* mampu menangani basis data dalam skala besar dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta kurang lebih 5 milyar baris. Selain itu batas indeks yang dapat ditampung menvcapai 32 indeks pada setiap tabelnya.
6. *MySQL* dapat melakukan koneksi dengan *client* menggunakan *protocol TCP/IP*, *Unix Soket (UNIX)* atau *Named Pipes (NT)*.



7. *MySQL* dapat mendeteksi pesan kesalahan pada *client* dengan menggunakan lebih dari dua puluh bahasa.
8. *MySQL* dapat berjalan stabil dari berbagai sistem operasi seperti *Windows*, *Linux*, *Freebsd*, *Mac Os X Server*, *Solaris*, *Amiga*, dan masih banyak lagi.
9. *MySQL* didistribusikan secara *open source*, dibawah lisensi *GPL* sehingga dapat digunakan secara gratis.

2.4.4.2. Fungsi-Fungsi *MySQL*

Kadir (2008:360), menjelaskan tentang sejumlah fungsi yang berawalan *mysql_* yang digunakan untuk mengakses *database server MySQL* sebagai berikut:

Tabel 2.6. Fungsi-Fungsi *MySQL*

No	Nama Fungsi	Kegunaan
1	<i>MySQL_connect()</i>	Membuka koneksi ke <i>server MySQL</i>
2	<i>MySQL_close()</i>	Menutup koneksi <i>MySQL</i>
3	<i>MySQL_select_db()</i>	Membuka/mengaktifkan suatu basis data
4	<i>MySQL_query()</i>	Mengeksekusi <i>string query</i>
5	<i>MySQL_db_query()</i>	Melakukan <i>query</i>
6	<i>MySQL_num_fields()</i>	Menghitung jumlah <i>field</i> dalam tabel tertentu
7	<i>MySQL_fetch_assoc()</i>	Mengambil hasil <i>query</i> sebelumnya
8	<i>MySQL_fetch_result()</i>	Membebaskan <i>resource</i> dari hasil <i>query</i>
9	<i>MySQL_fetch_row()</i>	Menampilkan hasil <i>query</i> ke dalam <i>array</i>
10	<i>MySQL_fetch_array()</i>	Menampilkan hasil <i>query</i> sebagai <i>array</i>
11	<i>MySQL_fetch_field()</i>	Menampilkan informasi <i>field</i> termasuk tipe datanya

Lanjutan Tabel 2.6. Fungsi-fungsi *MySQL*

No	Nama Fungsi	Kegunaan
12	<i>MySQL_affected_rows()</i>	Menampilkan jumlah baris yang dikenai operasi <i>SQL</i>
13	<i>MySQL_errno</i>	Menampilkan kode kesalahan (<i>error number</i>) dari perintah <i>SQL</i>
14	<i>MySQL_create_db()</i>	Membuat <i>database MySQL</i>
15	<i>MySQL_drop_db()</i>	Menghapus <i>database MySQL</i>
16	<i>MySQL_list_dbs()</i>	Menghasilkan seluruh daftar <i>database MySQL</i>
17	<i>MySQL_list_tables()</i>	Menampilkan seluruh tabel dalam <i>database</i> tertentu
18	<i>MySQL_list_fields()</i>	Menampilkan seluruh <i>field</i> dalam tabel tertentu

(Sumber : Kadir, 2008:360)

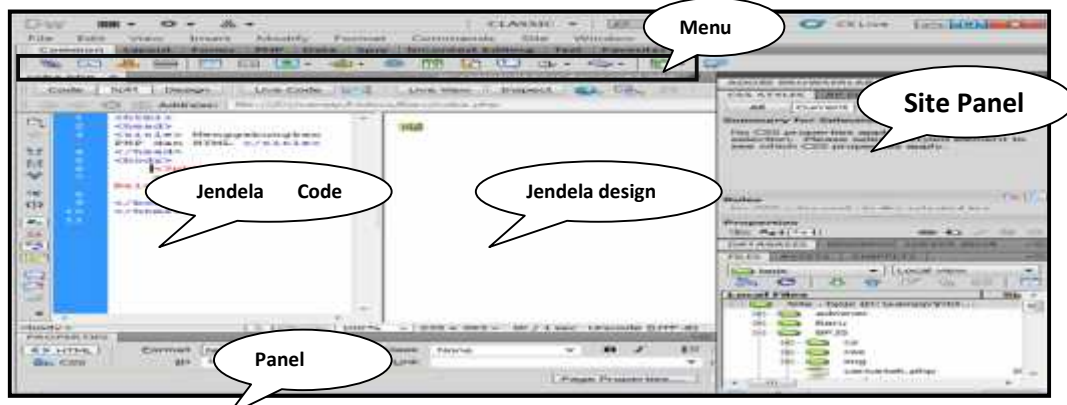
2.4.5. Mengenal tentang *Adobe Dreamweaver CS5*

2.4.5.1. Pengertian *Adobe Dreamweaver CS5*

Madcoms (2011:2), *Adobe Dreamweaver CS5* adalah sebuah *HTML editor profesional* untuk mendesain *web* secara visual dan mengelola situs atau halaman *web*. *Dreamweaver* merupakan software utama yang digunakan oleh *Web Designer* maupun *Web Programmer* dalam mengembangkan suatu situs *web*.

2.4.5.2. Area Kerja *Adobe Dreamweaver CS5*

Area kerja *Adobe Dreamweaver CS5* adalah lingkungan yang digunakan untuk merancang halaman *web*.



Gambar 2.2. Tampilan lembar kerja *Dreamweaver CS5*

Area kerja yang ada pada *Adobe Dreamweaver CS5* antara lain:

1. **Application Bar**, berada di bagian paling atas jendela aplikasi *Dreamweaver CS5*. Baris ini berisi tombol *workspace (workspace switcher)*, menu, dan aplikasi lainnya.



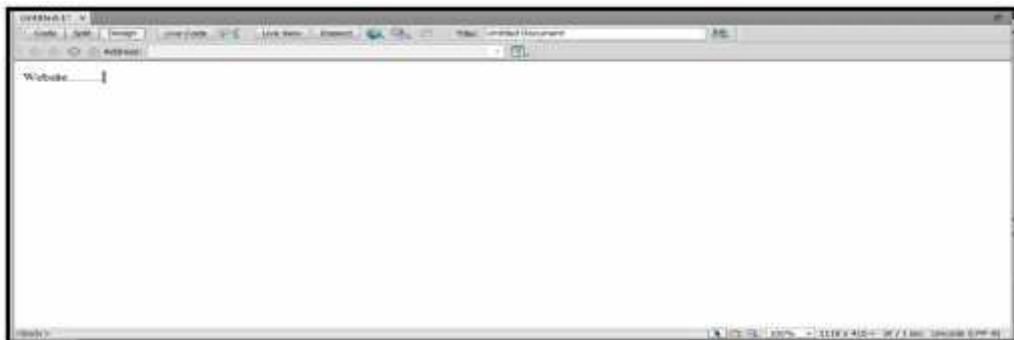
Gambar 2.3. Tampilan *Application Bar*

2. **Toolbar Document**, berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan *Design* atau tampilan *Code*.



Gambar 2.4. Tampilan *Toolbar Document*

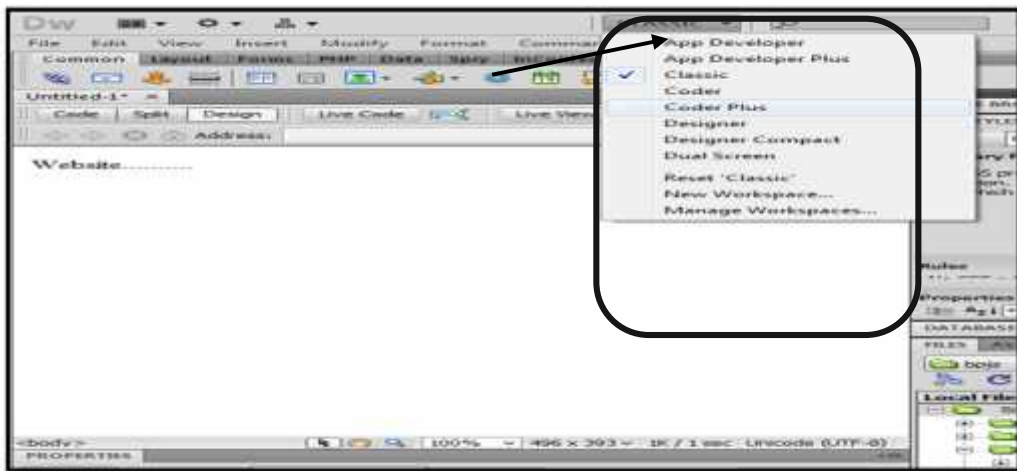
3. **Jendela Dokumen**, adalah lembar kerja tempat membuat dan mengedit *design* halaman *web*.



Gambar 2.5. Jendela Dokumen

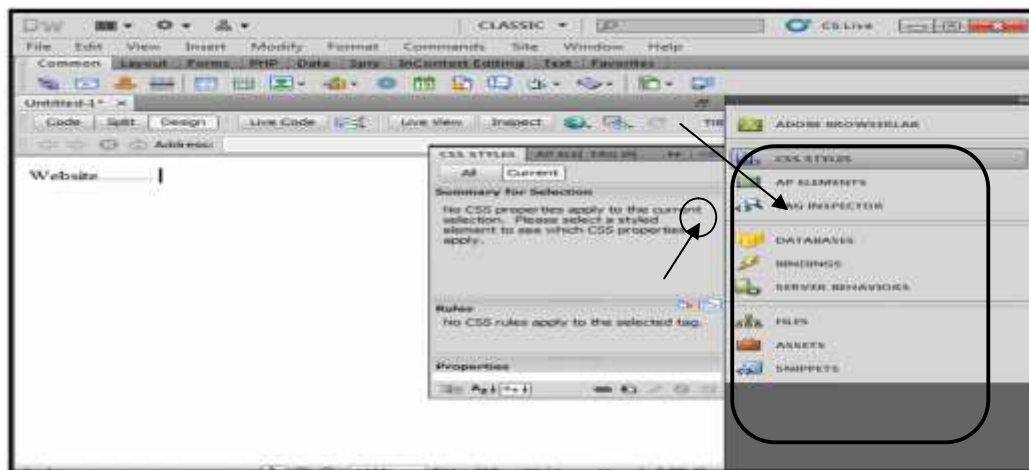


4. **Workspace Switcher**, digunakan untuk mengubah tampilan ruang kerja (*workspace*) *Dreamweaver CS5*. Sebagai contoh, mengubah tampilan menjadi tampilan *classic*, yaitu tampilan ruang kerja *Dreamweaver* versi sebelumnya.



Gambar 2.6. Tombol perintah untuk memilih jenis tampilan ruang kerja

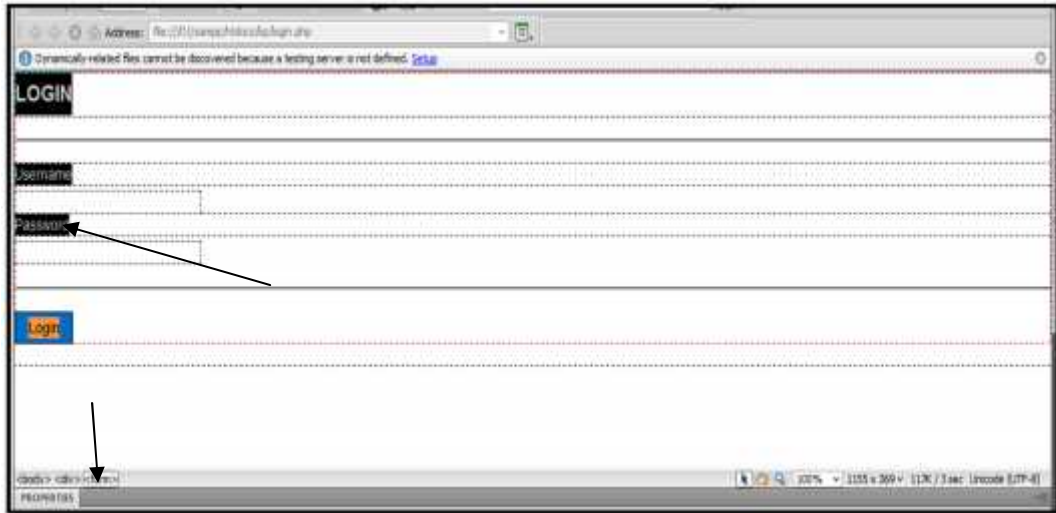
5. **Panel Groups**, adalah kumpulan panel yang saling berkaitan, panel-panel di kelompokkan pada judul-judul tertentu berdasarkan fungsinya. Panel ini digunakan untuk memonitor dan memodifikasi pekerjaan. Secara *default*, panel group berisi panel *Insert*, *CSS Style*, *Asset*, *AP Element* dan *Files*.



Gambar 2.7. Tampilan panel groups

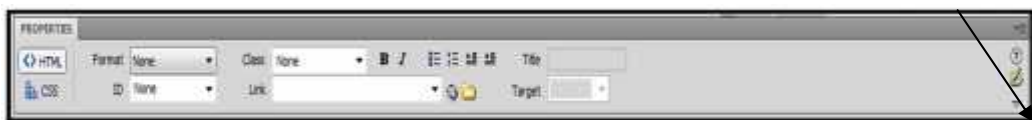


6. **Tag Selector**, diletakkan di bagian bawah jendela dokumen, satu baris dengan status bar. *Tag selector* juga menampilkan informasi format dari bagian yang sedang aktif pada lembar kerja *design*.



Gambar 2.8. Contoh tampilan *tag selector*

7. **Panel Property**, digunakan untuk melihat dan mengubah berbagai properti objek atau teks pada jendela *design*. Properti-properti dalam jendela panel property sebagian ada yang tidak tampil. Untuk menampilkan atau menyembunyikan properti dari objek terpilih, dapat dilakukan dengan cara, **klik panah di pojok kanan bawah dari jendela panel property**.

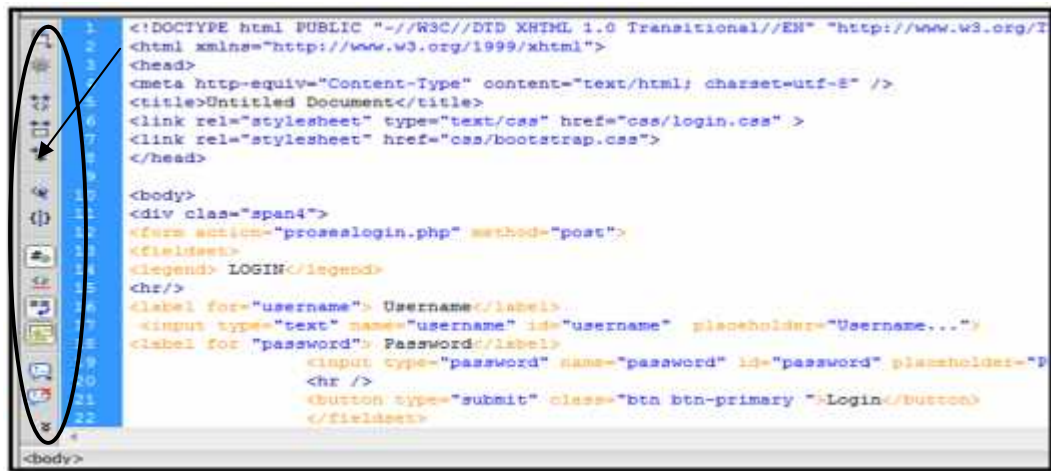


Gambar 2.9. Tombol panah pada *property*



Gambar 2.10. Contoh tampilan jendela *properties* untuk objek *form*

8. **Toolbar Coding**, berisi tombol-tombol yang digunakan untuk melakukan operasi kode-kode standar. *Toolbar* ini hanya tampil pada jendela *code*.

Gambar 2.11. Tampilan *toolbar Coding*

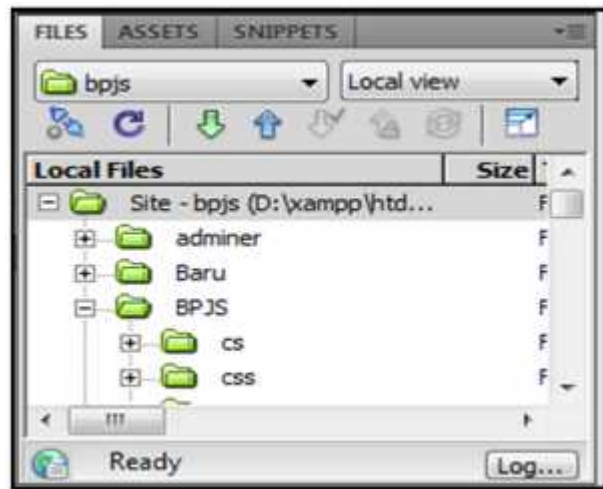
9. **Panel *Insert***, berisi tombol-tombol untuk menyisipkan berbagai jenis objek, seperti *image*, tabel, atau objek media ke dalam jendela dokumen.

Gambar 2.12. Tampilan *panel Insert*

10. **Panel *Groups***, adalah kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan di bawah satu judul, seperti *Insert*, *Files*, *CSS Styles* dan sebagainya.

Gambar 2.13. *Group panel*

11. **Panel *Files***, digunakan untuk mengatur *file-file* dan *folder-folder* yang membentuk situs *Web*. Panel *File* ditampilkan dalam panel *grup*.



Gambar 2.14. Contoh tampilan panel *Files*

2.4.6. 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*”.

Dari istilah di atas, dapat disimpulkan bahwa *JavaScript* adalah Bahasa pemrograman yang memiliki sekumpulan perintah khusus yang digunakan untuk membuat sebuah halaman *web* yang lebih responsif dan interaktif.

2.4.7. CSS

Badiyanto (2013:25), “CSS adalah kepanjangan dari *Cascading Style Sheets*, CSS skrip yang berisi rangkaian intruksi yang memantau suatu teks akan tertampil dihalaman *web browser*”. Contoh cara penggunaan CSS sebagai berikut :

```
<html>
<head>
<style>
body {
background-color: #d0e4fe;
}
```

```
h1{
color: orange;
text-align: center;
}
p{
font-family: "Times New Roman";
font-size: 20px;
color: blue;
}
</style>
</head>
<body>
<h1> Ini contoh CSS </h1>
<p>ini bagian dari paragraph.</p>
</body>
</html>
```

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



Gambar 2.15. Tampilan kode CSS