



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Sujatmiko (2012:156), “komputer adalah mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program. Alat serbaguna ini memegang peranan penting dalam teknologi komunikasi”.

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

Jadi, komputer adalah mesin yang dipakai untuk mengolah data, memproses data, serta menghasilkan keluaran berupa informasi yang bisa digunakan sesuai keperluan.

2.1.2 Pengertian Basis Data (*Database*)

Sujatmiko (2012:40), “*database* bisa dikatakan sebagai kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut”.

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

Jadi, basis data adalah kumpulan data yang tersimpan di dalam komputer sehingga dapat diolah menjadi informasi.

2.1.3 Pengertian Perangkat Lunak

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



Sujatmiko (2012:256) berpendapat “Perangkat lunak adalah kumpulan beberapa perintah yang dieksekusi oleh mesin komputer dalam menjalankan pekerjaannya. Perangkat lunak ini merupakan catatan bagi bagi mesin komputer untuk menyimpan perintah ataupun dokumen”.

Jadi, perangkat lunak merupakan kumpulan beberapa perintah yang diproses oleh mesin komputer sehingga menghasilkan informasi yang dapat digunakan oleh *user*.

2.1.4 Pengertian Sistem

Sutabri (2012:10) mengemukakan bahwa, “Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu”. Menurut Kristanto (2011:2), “Sistem juga merupakan kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan (*input*) yang ditujukan kepada sistem tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (*output*) yang diinginkan. Hal senada juga diungkapkan oleh Yakub (2012:1) yang mengemukakan bahwa, “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu.

Dari beberapa definisi di atas penulis menyimpulkan bahwa sistem adalah kesatuan unsur yang saling bekerja sama dan terorganisir untuk mencapai tujuan tertentu.

2.1.4.1 Elemen-elemen Sistem

Menurut Mcleod (dikutip Yakub, 2012:3), tidak semua sistem memiliki elemen-elemen yang sama, tetapi susunan dasarnya sama. Ada beberapa elemen yang membentuk sebuah sistem yaitu:

- a. Tujuan, tujuan ini menjadi motivasi yang mengarahkan pada sistem, karena tanpa tujuan yang jelas sistem menjadi tak terarah dan tak terkendali.
- b. Masukan, masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan dapat berupa



hal-hal berwujud maupun tidak berwujud. Masukan berwujud adalah bahan mentah, sedangkan yang tidak berwujud adalah informasi.

- c. Proses, proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai.
- d. Keluaran, keluaran (*output*) merupakan hasil dari pemrosesan sistem dan keluaran dapat menjadi masukan untuk subsistem lain.
- e. Batas, batas (*boundary*) sistem adalah pemisah antara sistem dan daerah di luar sistem. Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.
- f. Mekanisme pengendalian dan umpan balik, mekanisme pengendalian (*control mechanism*) diwujudkan dengan menggunakan umpan balik (*feedback*), sedangkan umpan balik ini digunakan untuk mengendalikan masukan maupun proses. Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan.
- g. Lingkungan, lingkungan adalah segala sesuatu yang berada di luar sistem.

2.2.4.2 Metode Pengembangan Sistem

Sukanto et al (2013:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

- a. Analisis kebutuhan perangkat lunak

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

- b. Desain

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



c. Pengkodean Sistem

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain

d. Pengujian Sistem

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

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

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

2.1.5 Pengertian Aplikasi

Asropudin (2013:7), “aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Microsoft Wors, Microsoft Excel*”.

Sujatmiko (2012:23) berpendapat “Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas tertentu, misalnya Microsoft Word dan Microsoft Excel”.

Jadi menurut beberapa pengertian di atas, aplikasi merupakan program komputer yang dibuat dengan tujuan membantu manusia dalam mengerjakan tugas tertentu.



2.1.6 Pengertian Pengolahan Data

Ladjamudin (2013:13) mengatakan “pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Sutabri (2012:6), “Pengolahan data terdiri dari kegiatan penyimpanan dan penanganan data. Penyimpanan data meliputi pengumpulan, pencarian, dan pemeliharaan. Sedangkan penanganan data meliputi berbagai kegiatan seperti pemeriksaan, perbandingan, pemilihan, peringkasan, dan penggunaan”.

Dari beberapa definisi di atas, penulis menyimpulkan bahwa pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan berupa kegiatan penyimpanan data dan penanganan data.

2.1.7 Pengertian Alat Kontrasepsi

Kamus Besar Bahasa Indonesia menerangkan bahwa alat kontrasepsi adalah benda yang dipakai untuk memudahkan pekerjaan dalam hal ini digunakan untuk mencegah kehamilan seperti, spiral, pil dan kondom.

2.1.8 Pengertian KB (Keluarga Berencana)

Menurut Kamus Besar Bahasa Indonesia, keluarga berencana merupakan suatu gerakan untuk menciptakan keluarga yang sehat dan sejahtera dengan mengatur kelahiran.

Muttaqin (2016:692), “keluarga berencana merupakan suatu upaya mengatur kelahiran anak, jarak dan usia melahirkan, mengatur kehamilan melalui promosi, perlindungan dan bantuan sesuai dengan hak reproduksi untuk mewujudkan keluarga yang berkualitas”.

Jadi, dari beberapa pengertian KB (Keluarga Berencana) diatas penulis menarik kesimpulan bahwa KB (Keluarga Berencana) adalah suatu gerakan untuk mengatur kelahiran anak demi terciptanya keluarga yang sehat dan sejahtera.



2.1.9 Pengertian Badan Kependudukan dan Keluarga Berencana Nasional (BKKBN)

Badan Kependudukan dan Keluarga Berencana Nasional (dahulu Badan Koordinasi Keluarga Berencana Nasional), disingkat BKKBN, adalah Lembaga Pemerintah Non Departemen Indonesia yang bertugas melaksanakan tugas pemerintahan di bidang keluarga berencana dan keluarga sejahtera.

2.1.10 Pengertian Aplikasi Pengolahan Data Permintaan Alat Kontrasepsi KB (Keluarga Berencana) Pada Badan Kependudukan dan Keluarga Berencana Nasional Provinsi Sumatera Selatan.

Pengertian Aplikasi Pengolahan Data Permintaan Alat Kontrasepsi KB (Keluarga Berencana) Pada Badan Kependudukan dan Keluarga Berencana Nasional Provinsi Sumatera Selatan adalah aplikasi yang dibuat untuk membantu SubBag Keuangan dan BMN dalam mengolah data permintaan alat kontrasepsi KB (Keluarga Berencana) dari Dinas Pengendalian Penduduk dan KB (Keluarga Berencana) di Provinsi Sumatera Selatan.

2.2 Teori Khusus

2.2.1 Data Flow Diagram (DFD)

2.2.1.1 Pengertian Data Flow Diagram (DFD)

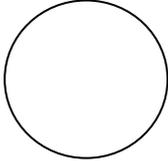
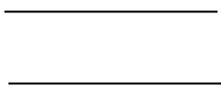
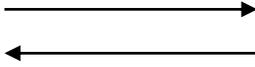
Sukanto et al. (2013:70) menyatakan “*Data Flow Diagram* atau dalam bahasa Indonesia menjadi diagram alir data adalah representatif grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

2.2.1.2 Simbol-simbol pada Data Flow Diagram (DFD)

Edward Yourdon dan Tom DeMarco membagi simbol atau notasi pada *Data Flow Diagram* menjadi 4 (empat) buah, yaitu sebagai berikut:



Tabel 2.1 Simbol-simbol pada *Data Flow Diagram*

No	Nama dan Simbol	Keterangan
1	Proses 	Diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program.
2	Basis data 	Diimplementasikan dengan pemograman terstruktur maka pemodelan inilah yang dibuat menjadi table-table basis data yang sesuai dengan table-table basis data.
3	Entitas Luar (<i>External Entity</i>) 	Digunakan untuk menggambarkan asal atau tujuan data.
4	Data Flow 	Digunakan untuk menggambarkan aliran data yang sedang berjalan.

(Sumber: Sukamto et al. 2013:71)

2.2.1.3 Tahapan-tahapan Perancangan *Data Flow Diagram* (DFD)

Dalam merancang suatu diagram alir data, perlu diketahui bahwa diagram alir data memiliki tahapan dalam merancanginya. Adapun tahapan dalam merancang diagram alir data, yaitu:

1. *Data Flow Diagram* (DFD) Level 0 atau *Context Diagram*

Data Flow Diagram Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain ataupun entitas luar.

2. *Data Flow Diagram* (DFD) Level 1



Data Flow Diagram Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. *Data Flow Diagram* Level 1 merupakan hasil *breakdown* dari *Data Flow Diagram* Level 0.

3. *Data Flow Diagram* Level 2

Data Flow Diagram Level 2 merupakan *breakdown* dari *Data Flow Diagram* Level 1. Jumlah *Data Flow Diagram* Level 2 sama dengan modul pada *Data Flow Diagram* Level 1 dalam suatu sistem.

4. *Data Flow Diagram* Level 3 dan seterusnya

Data Flow Diagram Level 3 dan seterusnya merupakan *breakdown* dari *data flow diagram* sebelumnya, namun jarang digunakan karena dapat menyebabkan sistem yang dikembangkan menjadi rumit.

2.2.2. *Entity Relationship Diagram (ERD)*

2.2.2.1. Pengertian *Entity Relationship Diagram (ERD)*

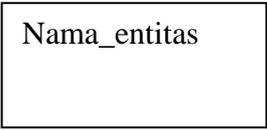
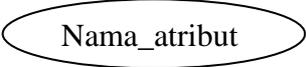
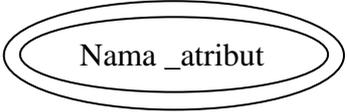
Sukamto et al. (2013:50) menyatakan “*Entity Relationship Diagram (ERD)* merupakan bentuk paling awal dalam melakukan perancangan basis data relasional”.

Entity relationship diagram dikembangkan berdasarkan teori himpunan dalam bidang matematika. *ERD* juga memiliki beberapa aliran notasi atau simbol seperti notasi *Chen* (dikembangkan oleh Peter Chen) notasi *Barker* (dikembangkan oleh Richard Barker, Ian Palmer, dan Harry Ellis), notasi *Crow's Foot*, dan juga notasi lainnya.

2.2.2.2 Simbol-simbol pada *Entity Relationship Diagram (ERD)*

Peter Chen membagi simbol-simbol yang digunakan dalam *Entity Relationship Diagram* menjadi 6 notasi atau simbol. Adapun notasi atau simbol yang digunakan, yaitu:


Tabel 2.2 Simbol-simbol pada *Entity Relationship Diagram*

Nama	Simbol	Keterangan
Entitas/ <i>entity</i>		Merupakan data inti yang akan disimpan sehingga dapat diakses oleh aplikasi komputer.
Atribut		Merupakan <i>field</i> atau kolom data yang akan disimpan dalam suatu entitas.
Atribut kunci primer		Merupakan <i>field</i> atau kolom data yang disimpan dalam suatu entitas serta menggunakan kunci akses <i>record</i> yang diinginkan ataupun unik.
Atribut multinilai/ <i>multivalue</i>		Merupakan <i>field</i> atau kolom data yang disimpan dalam suatu entitas dimana memiliki nilai lebih dari satu.
Relasi		Merupakan relasi yang menghubungkan antar entitas dan biasanya diawali dengan kata kerja.
Asosiasi/ <i>association</i>		Merupakan penghubung antara relasi dimana kedua ujungnya memiliki <i>multiplicity</i> dalam jumlah pemakaian.

Sumber: Sukamto et al. 2013:50-51)



2.2.3 Block Chart

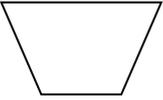
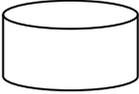
2.2.3.1 Pengertian Block Chart

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

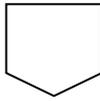
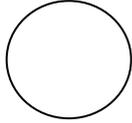
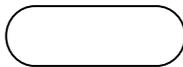
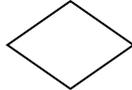
2.2.3.2 Simbol-simbol pada Block Chart

Adapun simbol-simbol dalam *Block Chart* adalah sebagai berikut:

Tabel 2.3. Simbol-Simbol pada *Block Chart*

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

Lanjutan Tabel 2.3. Simbol-Simbol pada *Block Chart*

No.	Simbol	Keterangan
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
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 (<i>decision</i>)
12.		Layar peraga (<i>monitor</i>)
13.		Pemasukan data secara manual

(Sumber: Kristanto, 2008:75-77)

2.2.4 Flowchart

2.2.4.1 Pengertian *Flowchart*

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

Sitorus (2015:14), “*flowchart* merupakan sebuah alat bantu yang berbentuk diagram alir di gunakan untuk menggambarkan sebuah algoritma yang terstruktur yang mudah dipahami oleh orang lain”.

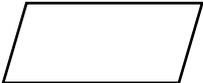
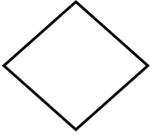
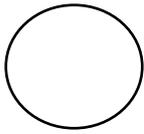
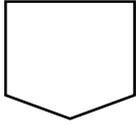
Jadi, *flowchart* merupakan sebuah alat bantu berbentuk diagram alir yang digunakan untuk menggambarkan urutan prosedur dalam sebuah program.



2.2.4.2 Simbol-simbol pada *Flowchart*

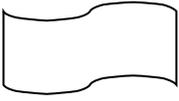
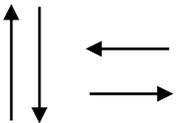
Sitorus (2015:14-15), menjelaskan tentang simbol-simbol yang digunakan untuk menggambarkan algoritma dalam bentuk diagram alir dan kegunaan dari simbol-simbol yang bersangkutan dapat dilihat pada tabel berikut ini:

Tabel 2.4 Simbol-simbol pada *Flowchart*

No	Simbol	Nama	Keterangan
1.		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program
2.		<i>Input / Output</i>	Menyatakan proses input atau output tanpa tergantung jenis peralatannya
2.		<i>Proses</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
4.		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya / tidak.
5.		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
6.		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
7.		<i>Predefined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.



Lanjutan Tabel 2.4 Simbol-simbol pada *Flowchart*

No	Simbol	Nama	Keterangan
8.		<i>Punched Card</i>	Menyatakan input berasal dari kartu atau output ditulis ke kartu
9.		<i>Punch Tape</i>	Menunjukkan <i>input/output</i> menggunakan pita kertas berlubang.
10.		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui printer)
11.		<i>Flow</i>	Menyatakan jalannya arus suatu proses

(Sumber: Siturous, (2015:14-15))

2.2.5. Kamus Data (*Data Dictionary*)

Sukamto et al. (2013:73) menyatakan “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*ouput*) dapat dipahami secara umum (memiliki standar cara penulisan)”.

Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi:

1. Nama, merupakan nama dari data.
2. Digunakan pada, merupakan proses-proses yang terkait data.
3. Deskripsi, merupakan deskripsi data.
4. Informasi tambahan, merupakan tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

Kamus data memiliki beberapa simbol yang bertujuan untuk menjelaskan informasi tambahan. Adapun simbol dari kamus data, yaitu:

**Tabel 2.5.** Simbol-simbol pada Kamus Data

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

(Sumber: Sukanto et al. 2013:74)

2.2.6. Pengujian *Black Box Testing*

Menurut Rizky (dikutip oleh Hanifah, 2016:34), “Black Box Testing sendiri merupakan pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian black box ini menitik beratkan pada fungsi system”.

Berikut adalah 10 tipe pengujian dari metode Black Box menurut Julian (dikutip oleh Hanifah, 2016:34):

1. *Equivalence Partitioning*, Membagi inputan menjadi kelas data yang dapat digunakan untuk menggenerasi kasus uji.
2. *Boundary Value Analysis / Limit Testing*, Mengijinkan untuk menyeleksi kasus uji yang menguji batasan nilai input, Merupakan komplemen dari Equivalence Partitioning.
3. *Comparison Testing*, Uji setiap versi dengan data yang sama untuk memastikan semua versi menghasilkan keluaran yang sama.
4. *Sample Testing*, Melibatkan beberapa nilai yang terpilih dari sebuah kelas ekivalen.



-
5. *Robustness Testing*, Data input dipilih diluar spesifikasi yang telah didefinisikan, Tujuan dari pengujian ini adalah membuktikan bahwa tidak ada kesalahan jika masukan tidak valid.
 6. *Behavior Testing*, Hasil uji tidak dapat dievaluasi jika hanya melakukan pengujian sekali, tapi dapat dievaluasi jika pengujian dilakukan beberapa kali, misalnya pada pengujian struktur data stack.
 7. *Performance Testing*, Mengevaluasi kemampuan program untuk beroperasi dengan benar dipandang dari sisi acuan kebutuhan misalnya: aliran data, ukuran pemakaian memori, kecepatan eksekusi.
 8. *Requirement Testing*, Spesifikasi kebutuhan yang terasosiasi dengan perangkat lunak diidentifikasi pada tahap spesifikasi kebutuhan dan desain.
 9. *Endurance Testing*, Melibatkan kasus uji yang diulang-ulang dengan jumlah tertentu.

Cause – Effect Relationship Testing, Bagi-bagi spesifikasi kebutuhan menjadi bagian yang memiliki kemungkinan kerja

2.3. Teori Program

2.3.1. Pengertian PHP

Zaki et al. (2014:49), “*PHP Hypertext Preprocessor* adalah sebuah bahasa pemrograman *web* berbasis *server* (*server-side*) yang mampu memarsing kode *PHP* dari kode *web* dengan ekstensi *.PHP*, sehingga menghasilkan tampilan *website* yang dinamis disisi client (*browser*)”.

Sujatmiko (2012:213), “*PHP* merupakan bahasa pemrograman yang digunakan untuk membuat aplikasi *web*. Istilah *PHP* sebenarnya merupakan singkatan dari *Hypertext Preprocessor*”.

Jadi menurut beberapa pendapat di atas, *PHP (Hypertext Preprocessor)* adalah sebuah bahasa pemrograman berbasis *web* yang digunakan untuk membuat halaman HTML.

2.3.1.1. Script PHP

PHP yang merupakan sebuah bahasa scripting yang terpasang pada

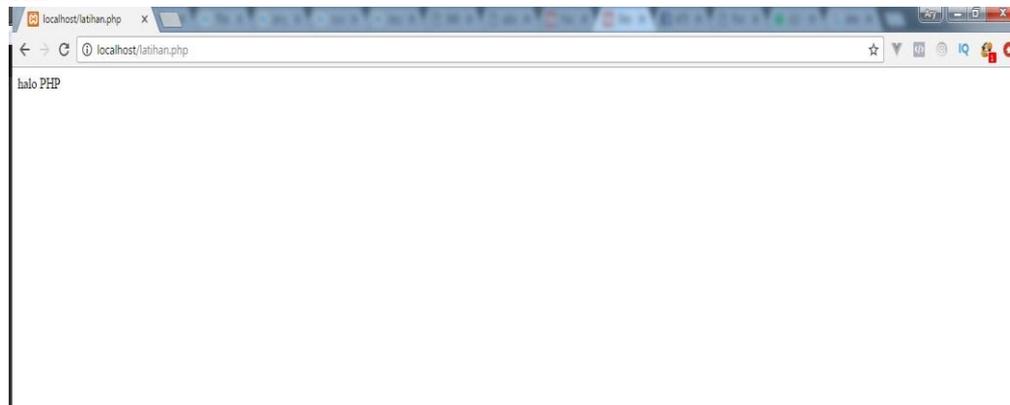


HTML, dan skrip HTML (*Hyper Text Markup Language*) adalah bahasa standar untuk membuat halaman *web*. Biasa *file* dituliskan dengan ekstensi *.htm* atau *.html*.

```
<html>
  <head>
    <title>Latihan PHP</title>

  </head>
  <body>
    halo PHP
  </body>
</html>
```

Bila dijalankan melalui *browser*, kode diatas akan menampilkan hasil seperti berikut:



Gambar 2.1 Tampilan program HTML pada *browser*

2.3.1.3. Teknik Penulisan Script *PHP*

Zaki et al. (2014:24), block kode *PHP* dimulai dengan sintak tag `<?PHP` dan di akhiri dengan `?>`, berikut adalah contoh blok kode *PHP*:

```
<?PHP
//kode tertentu diisikan di sini
?>
```

Selain itu ada beberapa tag lain yang bisa di pakai, seperti:

Tabel 2.6 Script Dasar *PHP*

Jenis Tag	Tag Pembuka	Tag Penutup
Tag standar	<code><?PHP</code>	<code>?></code>
Tag pendek	<code><?</code>	<code>?></code>



Lanjutan Tabel 2.6 Script Dasar PHP

Jenis Tag	Tag Pembuka	Tag Penutup
Tag ASP	<%	%>
Tag script	<script language="PHP">	</script>

(Sumber: Zaki et al. 2014:24-25)

Echo sering digunakan di dalam *PHP*. Fungsinya untuk meng-ouput data. Berikut ini contoh penggunaan echo:

```

<?PHP
Echo '<p> selamat datang </p>';
?>
```

Di dalam *script PHP* juga dapat membuat komentar dan komentar tersebut tidak akan dieksekusi sebagai sebuah *script*. Beberapa cara untuk memberi keterangan di dalam *script PHP* adalah sebagai berikut:

- 1) Gunakan *tag /** dan akhiri *tag */* apabila jumlah keterangan lebih dari 1 baris.
- 2) Gunakan *tag //*, *tag* ini digunakan untuk keterangan yang hanya terdiri dari 1 baris saja.
- 3) Gunakan *tag #*, *tag* ini juga digunakan untuk 1 baris komentar saja.

Dalam penulisannya, baris perintah *script PHP* selalu diakhiri dengan menuliskan tanda titik koma (;). Satu baris *script PHP* tidak harus berada dalam satu baris, melainkan dapat menuliskan perintah *script PHP* lebih dari satu baris.

2.3.2. Pengertian MySQL

Sukamto et al. (2013:46), “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus”.

Winarno (2014:102), “*MySQL* adalah sebuah software database. *MySQL* merupakan tipe data relasional yang artinya *MySQL* menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan”.

Jadi, *MySQL* merupakan sebuah bahasa yang digunakan untuk mengelola data dalam bentuk tabel-tabel yang saling berhubungan.



2.3.2.1 Fungsi-Fungsi MySQL

Sejumlah fungsi yang berawalan **mysql_** yang digunakan untuk mengakses *database server MySQL* adalah sebagai berikut:

Tabel 2.7. Fungsi-fungsi *MySQL*

No.	Fungsi MySQL	Penggunaan
1	<i>Mysql_connect()</i>	Membuat hubungan ke <i>database MySQL</i> yang terdapat pada suatu host
2	<i>Mysql_close()</i>	Menutup hubungan ke <i>database MySQL</i>
3	<i>Mysql_select_db()</i>	Memilih <i>database</i>
4	<i>Mysql_query()</i>	Mengeksekusi permintaan terhadap sebuah tabel atau sejumlah tabel
5	<i>Mysql_db_query()</i>	Menjalankan suatu permintaan terhadap suatu <i>database</i>
6	<i>Mysql_num_rows()</i>	Memperoleh jumlah baris dari suatu hasil permintaan yang menggunakan <i>SELECT</i>
7	<i>Mysql_affected_rows()</i>	Memperoleh jumlah baris yang dikenai operasi <i>INSERT, DELETE, UPDATE</i>
8	<i>Mysql_num_fields()</i>	Memperoleh jumlah kolom pada suatu hasil permintaan
9	<i>Mysql_fetch_row()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasil
10	<i>Mysql_fetch_array()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasilnya akan disimpan dua kali pada <i>array</i> hasil
11	<i>Mysql_fetch_field()</i>	Menghasilkan informasi suatu kolom
12	<i>Mysql_data_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke baris tertentu
13	<i>Mysql_field_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke kolom tertentu



Lanjutan Tabel 2.7 Fungsi-fungsi MySQL

No.	Fungsi MySQL	Penggunaan
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 daftar <i>database MySQL</i>
17	<i>Mysql_list_tables()</i>	Memperoleh daftar nama tabel dalam suatu <i>database</i>
18	<i>Mysql_list_fields()</i>	Memperoleh daftar nama kolom dalam suatu <i>database</i>

(Sumber: Abdul Kadir (2008:360-379))

2.3.2.2. Alasan Menggunakan MySQL

Berikut ini beberapa alasan mengapa mereka memilih *MySQL* sebagai *server database*

1. Fleksibel. *MySQL* dapat digunakan untuk mengembangkan aplikasi *desktop* maupun aplikasi *web* dengan menggunakan teknologi yang bervariasi. Ini berarti bahwa *MySQL* memiliki fleksibilitas terhadap teknologi yang akan digunakan sebagai pengembang aplikasi, apakah itu *PHP*, *JSP*, *Java*, *Delphi*, *C++*, maupun yang lainnya dengan cara menyediakan *plug-in* dan *driver* yang spesifik untuk masing-masing teknologi tersebut.
2. Performa Tinggi. *MySQL* memiliki *mein query* performa tinggi dengan demikian proses transaksional dapat dilakukan dengan sangat cepat.
3. Lintas Platform. *MySQL* dapat digunakan pada *platform* atau lingkungan sistem operasi yang beragam.
4. Gratis. *MySQL* dapat digunakan secara gratis.
5. Proteksi Data yang Handal. Perlindungan terhadap keamanan data merupakan hal nomor satu yang dilakukan oleh para profesional di bidang *database*. *MySQL* menyediakan mekanisme yang powerful untuk menangani hal tersebut, yaitu dengan menyediakan fasilitas manajemen *user*, enkripsi data, dan lain sebagainya.



6. Komunitas Luas. Karena penggunaanya banyak maka *MySQL* memiliki komunitas yang luas. Hal ini berguna jika kita menemui suatu permasalahan dalam proses pengolahan data menggunakan *MySQL*.

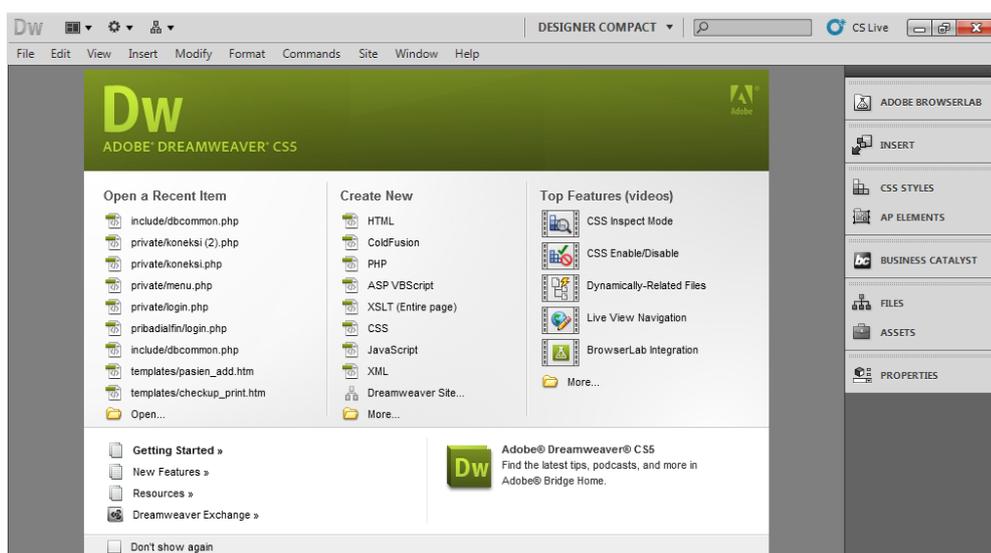
2.3.3. Pengertian *Adobe Dreamweaver*

Wahana Komputer (2012:2), “*dreamweaver* adalah salah satu aplikasi editor *web* yang digunakan untuk merancang secara visual yang dapat memudahkan *user* dalam membuat suatu *website*”.

Elcom (2013:1), “*dreamweaver* adalah sebuah editor profesional yang menggunakan HTML untuk medesain *web* secara visual dan mengelola situs halaman *web*”.

Jadi, berdasarkan pendapat beberapa ahli di atas, dapat disimpulkan bahwa *dreamweaver* merupakan *software* editor profesional untuk mengembangkan suatu situs *web*.

2.3.3.1. *Starting Page*



Gambar 2.2. Tampilan *Starting Page* pada *Adobe Dreamweaver*

Jendela diatas menampilkan 4 menu pilihan, yaitu:

1. *Open a Recent Item*

Menu ini menampilkan daftar nama dokumen yang pernah dibuka oleh *Adobe Dreamweaver*. Dokumen tersebut dapat dibuka dengan cara memilih nama dokumen yang ada.



2. *Create New*

Menu ini digunakan untuk membuat dokumen *web* yang baru. Dokumen *web* dapat dibuat dengan beberapa tipe yang disediakan oleh *Adobe Dreamweaver*..

3. *Top Features* (video)

Merupakan fitur yang disediakan *Dreamweaver* melalui *internet*.

4. *Help online*

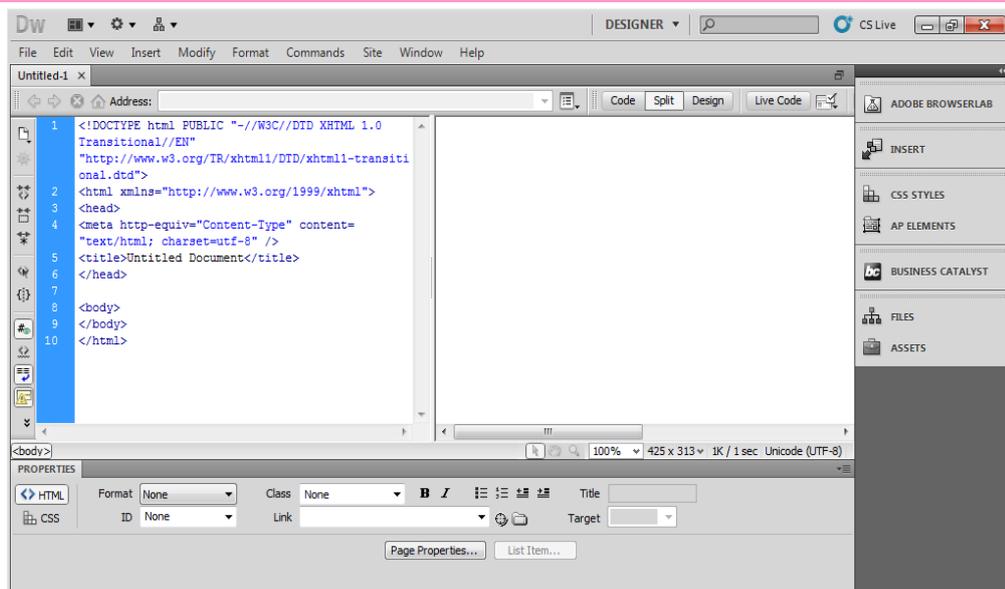
Merupakan menu pilihan untuk menampilkan fasilitas *help* secara *online* melalui *internet*.

2.3.3.2. Ruang Kerja *Adobe Dreamweaver*

Saat pertama kali menjalankan *Dreamweaver* dengan, akan ditampilkan sebuah kotak dialog *Workspace Setup* yang digunakan untuk memilih tampilan ruang kerja.

1. ***Designer Workspace***, adalah sebuah penggabungan ruang kerja menggunakan MDI (*Multiple Document Interface*), dimana semua jendela *document* dan panel-panel tergabung di dalam satu jendela aplikasi yang benar, dengan sisi panel group di sebelah kanan.
2. ***Coder Workspace***, adalah ruang kerja yang tergabung sama, tapi letak sisi panel group berada di sebelah kiri.

Ruang kerja *Adobe Dreamweaver* memiliki komponen-komponen yang memberikan fasilitas dan ruang untuk menuangkan kreasi saat bekerja, seperti *Application Bar*, *Toolbar Document*, *Document Window*, *Workspace Switcher*, *Panel Groups*, *Tag Selector*, *Property Inspector*, *Toolbar Browser Navigation*.



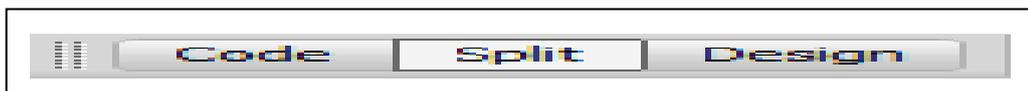
Gambar 2.3. Tampilan Ruang Kerja *Adobe Dreamweaver*

1. **Menu Bar**, menu bar pada *Adobe Dreamweaver* terdiri dari menu *File, Edit, View, Insert, Modify, Format, Commands, Site, Window, Help*. Masing-masing menu mempunyai submenu sesuai kategorinya dengan fungsi yang berbeda



Gambar 2.4. Menu Bar pada *Adobe Dreamweaver*

2. **Document Toolbar**, *document toolbar* berisi tombol-tombol yang digunakan untuk mengatur layar kerja dokumen pada *Dreamweaver* dengan cepat, di antaranya adalah *Show Code view, Show Code and Design views, dan Show Design view*.



Gambar 2.5. Document Toolbar pada *Adobe Dreamweaver*

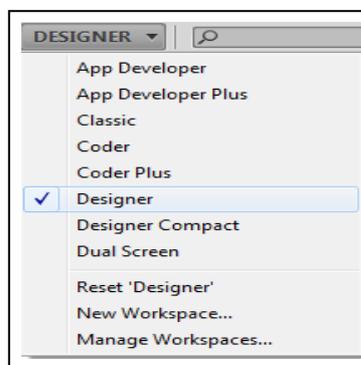


3. **Show Code View**, merupakan tampilan layar Code yang menampilkan kode-kode script halaman *web*. Lembar kerja tempat membuat dan mengedit desain halaman *web*.

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>Untitled Document</title>
6 </head>
7
8 <body>
9 </body>
10 </html>
```

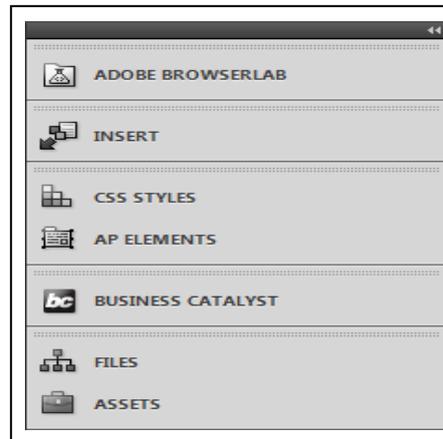
Gambar 2.6. *Show Code View* pada *Adobe Dreamweaver*

4. **Workspace Switcher**, digunakan untuk mengubah tampilan ruang kerja (*workspace*) *Adobe Dreamweaver*



Gambar 2.7. *Workspace Switcher* pada *Adobe Dreamweaver*

5. **Panel Groups**, adalah kumpulan panel yang saling berkaitan, panel-panel ini dikelompokkan pada judul-judul tertentu berdasarkan fungsinya. Panel ini digunakan untuk memonitor dan memodifikasi pekerjaan.



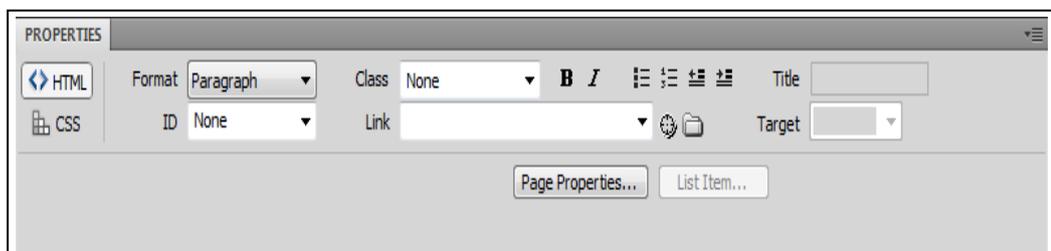
Gambar 2.8. *Panel Groups* pada *Adobe Dreamweaver*

6. **Tag Selector**, terletak di bagian bawah jendela dokumen, satu baris dengan *status bar*. Bagian ini menampilkan hirarki pekerjaan yang sedang terpilih pada jendela dokumen, dapat juga digunakan untuk memilih objek pada jendela desain berdasarkan jenis atau kategori objek tersebut. *Tag Selector* juga menampilkan informasi format dari bagian yang sedang aktif pada lembar kerja desain.



Gambar 2.9. *Tag Selector* pada *Adobe Dreamweaver*

7. **Property Inspector**, digunakan untuk melihat dan mengubah berbagai *property* objek atau teks pada jendela desain. *Property* untuk satu objek dengan objek lainnya selalu berbeda-beda. Jendela ini tidak dapat diuraikan pada tampilan jendela kode.



Gambar 2.10. *Property Inspector* pada *Adobe Dreamweaver*