



## 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”.

Wahyudi (2012:3), “Komputer adalah peralatan (*device*) yang menerima data (*input*) dan menyimpan (*storage*) kemudian di proses (*process*) untuk menghasilkan data dalam bentuk lain (*output*)”.

Husda dan Wangdra (2016:2), “Komputer merupakan alat mengolah data sesuai perintah yang sudah di rumuskan”.

Jadi, komputer adalah alat bantu yang menerima data, menyimpan kemudian di proses untuk menghasilkan data dalam bentuk lain sesuai perintah yang sudah dirumuskan.

##### 2.1.2 Pengertian Data

Wahyudi (2012:3), “Data adalah suatu *file* atau *field* yang berupa karakter atau tulisan dan gambar”.

Ladjamudin (2013:20), “Data merupakan komponen dasar dari informasi yang akan di peroleh lebih lanjut untuk menghasilkan informasi”.

Jadi, data adalah komponen dasar dari informasi yang berupa karakter atau tulisan dan gambar untuk menghasilkan informasi.

##### 2.1.3 Pengertian Internet

Hidayatullah dan Kawistara (2017:1), “Internet adalah jaringan global yang menghubungkan komputer-komputer diseluruh dunia”.

Husda dan Wangdra (2016:127), “Internet merupakan sistem global jaringan komputer yang berhubungan menggunakan standar *Internet Protocol Suite* (*TCP/IP*) untuk melayani milyaran pengguna di seluruh dunia”.



Menurut Febrian (2006:2), “Internet merupakan tempat terhubungnya berbagai mesin komputer yang mengolah informasi di dunia ini, baik berupa server, komputer pribadi, handphone, komputer genggam, PDA, dan lain sebagainya”.

Jadi, internet adalah jaringan global tempat terhubungnya berbagai mesin komputer yang mengolah informasi untuk melayani milyaran pengguna di seluruh dunia.

#### **2.1.4 Pengertian Aplikasi**

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

Sujatmiko (2012:23), “Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya Ms-word, Ms-excel”.

Jadi, aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu.

#### **2.1.5 Pengertian Pendaftaran**

Menurut Kamus Besar Bahasa Indonesia, “Pendaftaran adalah proses, cara, perbuatan pencatatan nama, alamat, dan sebagainya dalam perihal mendaftarkan”.

Suharso dan Retnoningsih (2014:112) menjelaskan, “Pendaftaran adalah pencatatan nama, alamat, dan sebagainya dalam daftar perihal mendaftarkan (mendaftarkan).”

Hizair (2013:139) menjelaskan, “Pendaftaran adalah kegiatan mencatat nama, alamat dsb dalam daftar perihal mendaftarkan.”

#### **2.1.6 Pengertian BPHTB**

Mardiasmo (2009:340), “Bea Perolehan Hak atas Tanah dan Bangunan (BPHTB) adalah pajak yang dikenakan atas perolehan hak atas tanah dan atau bangunan”.

Undang-Undang Nomor 28 Tahun 2009, “Bea Perolehan Hak atas Tanah dan Bangunan adalah pajak atas perolehan hak atas tanah dan atau bangunan”.

### **2.1.7 Pengertian Aplikasi Pendaftaran Bea Perolehan Hak atas Tanah dan Bangunan Pada Badan Pengelolaan Pajak Daerah Kota Palembang**

Aplikasi Pendaftaran Bea Perolehan Hak atas Tanah dan Bangunan Pada Badan Pengelolaan Pajak Daerah Kota Palembang adalah suatu aplikasi yang dibuat untuk mempermudah wajib pajak dalam melakukan proses pendaftaran Bea Perolehan Hak atas Tanah dan Bangunan yang dapat diakses melalui internet.

## **2.2 Teori Khusus**

### **2.2.1 *Unified Modelling Language (UML)***

Pratama (2013:48), “*Unified Modelling Language (UML)* adalah standarisasi internasional untuk notasi dalam bentuk grafik, yang menjelaskan tentang analisis dan desain perangkat lunak yang dikembangkan dengan pemrograman berorientasi objek”.

Rosa dan Shalahuddin (2013:133), “*UML* merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”.

Jadi *Unified Modelling Language (UML)* adalah bahasa visual untuk pemodelan yang menjelaskan tentang analisis dan desain perangkat lunak dengan menggunakan diagram dan teks-teks pendukung.



**Gambar 2.1** Logo *UML*



*UML* diaplikasikan untuk maksud tertentu, biasanya antara lain untuk:

1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

### **2.2.1.1 Jenis-Jenis Diagram *UML***

Adapun jenis-jenis diagram *UML* sebagai berikut:

1. Diagram Kelas, bersifat statis. Memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi.
2. Diagram Paket (*Package Diagram*), bersifat statis. Memperlihatkan kumpulan kelas-kelas, merupakan dari diagram komponen.
3. Diagram *Use Case*, bersifat statis. Diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas).
4. Diagram Interaksi dan *Sequence* (Urutan), bersifat dinamis. Dinamis urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.
5. Diagram Komunikasi (*Communication Diagram*), bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi *UML* 1.4 yang menekankan organisasi struktural dari obyek-obyek yang menerima serta mengirim pesan.
6. Diagram *Statechart*, bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi, kejadian serta aktivitas.
7. Diagram Aktivitas (*Activity Diagram*), bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem.
8. Diagram Komponen (*Component Diagram*), bersifat statis. Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem / perangkat lunak pada komponen-komponen yang telah ada sebelumnya.

9. Diagram *Deployment* (*Deployment Diagram*), bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*).

### 2.2.1.2 Use Case Diagram

Menurut Pratama (2013:48), “*Use case* diagram menggambarkan aliran kegiatan dan proses bisnis yang dilakukan oleh pengguna (aktor)”.

Rosa dan Shalahuddin (2013:155) menyatakan, “*Use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”.

Jadi *use case* adalah pemodelan yang menggambarkan aliran kegiatan dari sistem informasi yang akan dibuat.

Komponen pembentuk diagram *use case* adalah:

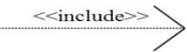
1. Aktor (*actor*), menggambarkan pihak-pihak yang berperan dalam sistem.
2. *Use case*, aktivitas/sarana yang disiapkan oleh bisnis / sistem.
3. Hubungan (*link*), aktor mana saja yang terlibat dalam *use case* ini.

Adapun simbol-simbol yang digunakan dalam *use case* adalah sebagai berikut:

**Tabel 2.1** Simbol *Use Case* Diagram

No.	Gambar	Nama	Keterangan
1		<i>Use Case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2		<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
3		<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

**Lanjutan Tabel 2.1** Simbol *Use Case* Diagram

No	Gambar	Nama	Keterangan
5		<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lain.
6		<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya sebagai syarat dijalankan <i>use case</i> .

Sumber: Rosa dan Shalahuddin (2013 : 162)

### 2.2.1.3 Class Diagram

Pratama (2013:48), “*Class* diagram menggambarkan struktur kelas yang terdapat pada perangkat lunak yang akan dibangun”.

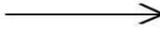
Rosa dan Shalahuddin (2013:141), “*Class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem”.

Adapun simbol-simbol yang digunakan dalam *class* diagram adalah sebagai berikut:

**Tabel 2.2** Simbol *Class* Diagram

No.	Gambar	Nama	Keterangan			
1	<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;">Nama_kelas</td> </tr> <tr> <td style="padding: 2px;">+atribut</td> </tr> <tr> <td style="padding: 2px;">+operasi()</td> </tr> </table>	Nama_kelas	+atribut	+operasi()	Kelas	Kelas pada struktur sistem.
Nama_kelas						
+atribut						
+operasi()						
2		<i>Interface</i>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.			
3		<i>Association</i>	Relasi antarkelas dengan makna umum.			
4		<i>Directed Association</i>	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain.			
5		Generalisasi	Relasi antarkelas dengan makna			

Lanjutan Tabel 2.2 Simbol *Class Diagram*

No.	Gambar	Nama	Keterangan
			generalisasi-spesialisasi (umum khusus).
6		<i>Dependency</i>	Relasi antarkelas dengan makna kebergantungan antarkelas.
7		<i>Aggregation</i>	Relasi antarkelas dengan makna semua-bagian ( <i>whole-part</i> ).

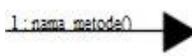
Sumber: Rosa dan Shalahuddin (2013 : 162)

#### 2.2.1.4 Sequence Diagram

Menurut Pratama (2013:48), “*Sequence diagram* menggambarkan *sequence* (aliran) pengiriman pesan (*message*) yang terjadi di aplikasi, sebagai bentuk interaksi dengan pengguna (*user*)”.

Adapun simbol-simbol yang digunakan dalam *sequence diagram* adalah sebagai berikut:

Tabel 2.3 Simbol *Sequence Diagram*

No.	Gambar	Nama	Keterangan
1		Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
2		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
3		Waktu Aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
4		Pesan tipe <i>call</i>	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.

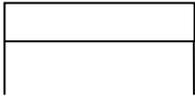
Sumber: Rosa dan Shalahuddin (2013 : 162)

### 2.2.1.5 Activity Diagram

Menurut Rosa dan Shalahuddin (2013:161), “Diagram aktivitas atau *activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

Adapun simbol-simbol yang digunakan dalam *activity diagram* adalah sebagai berikut:

**Tabel 2.4** Simbol *Activity Diagram*

No	Gambar	Nama	Keterangan
1		Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3		<i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4		<i>Join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabung menjadi satu.
5		Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6		<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Sumber: Rosa dan Shalahuddin (2013 : 162)

## 2.3 Teori Program

### 2.3.1 Pengertian Basis Data (*Database*)

Menurut Husda dan Wangdra (2016:119), “Basis data dapat disimpulkan sebagai himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.”

Menurut Sukamto dan Shalahudin (2014:43), “Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah di olah atau informasi dan membuat informasi tersedia saat di butuhkan”.



Menurut Fathansyah (2015:51), “Basis data merupakan kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.”

Jadi, basis data adalah himpunan kelompok data dengan sistem terkomputerisasi yang saling berhubungan tanpa pengulangan (redundansi) yang tidak perlu dan dapat dimanfaatkan kembali saat di butuhkan.

### **2.3.2 Sekilas tentang PHP**

#### **2.3.2.1 Pengertian PHP**

Menurut Hidayatullah dan Kawistara (2017:223), “*PHP Hypertext Preprocessor* atau disingkat PHP adalah suatu bahasa *scripting* khususnya digunakan untuk *web development*”.

Naista (2016:4), “*Hypertext Preprocessor* atau dingkat *PHP* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam *HTML*.”

Abdulloh (2016:3), “*PHP* singkatan dari *Hypertext Preprocessor* merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi *server*.”

Jadi, *Hypertext Preprocessor* atau *PHP* adalah suatu bahasa *scripting* yang dapat ditanamkan atau disisipkan ke dalam *HTML* dan diproses di sisi *server*.

#### **2.3.2.2 Sejarah PHP**

Hidayatullah dan Kawistara (2017:224), Pada awalnya *PHP* merupakan kependekan dari *Personal Home Page* (Situs Personal). *PHP* pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu *PHP* masih bernama *Form Interpreted (FI)*, yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari *web*.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya *PHP/FI*. Dengan perilisannya kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan *PHP*.



Pada November 1997, dirilis *PHP/FI* 2.0. Pada rilis 2.0 ini, interpreter *PHP* sudah diimplementasikan dalam program C. Di dalam rilis ini juga ikut sertakan modul-modul ekstensi yang meningkatkan kemampuan *PHP/FI* secara signifikan.

Kemudian pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter *PHP* menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk *PHP* dan meresmikan rilis tersebut sebagai PH 3.0 dan singkatan *PHP* diubah menjadi akronim berulang PHP: *Hypertext Preprocessing*.

### 2.3.2.3 Sintaks Dasar *PHP*

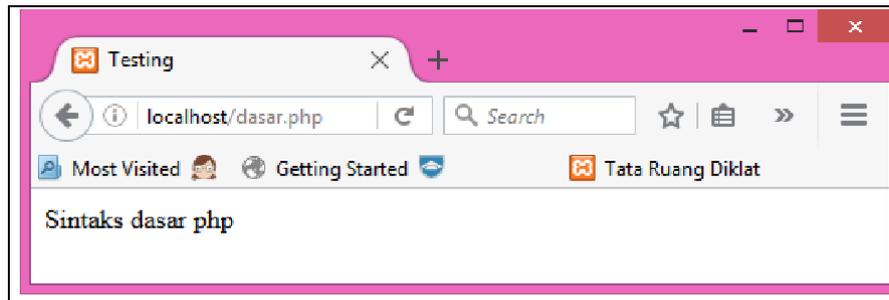
Sebelum menggunakan *PHP*, ada beberapa aturan sintaks yang harus dipenuhi ketika membuat file program *PHP*.

Dalam HTML diharuskan menulis *opening* dan *closing tag* seperti `<html>` `</html>`, dan dalam *PHP opening* dan *closing tag* harus dilakukan sebagai awal dan akhir program.

Berikut ini contoh sintaks dasar penggunaan *PHP* :

```
<!doctype html>
<HTML>
  <HEAD>
    <META charset="utf-8">
    <TITLE>Testing</TITLE>
  </HEAD>
  <?php
    Echo"Sintaks dasar php";
  ?>
</BODY>
</BODY>
</HTML>
```

Bila di jalankan di browser maka sintak dasar tersebut akan menjadi seperti gambar di bawah ini:



**Gambar 2.2** Tampilan di Browser

### 2.3.3 Pengertian HTML

Asropudin (2013:4), “*HTML (Hyper Text Markup Language)* adalah bahasa komputer yang digunakan untuk membuat sebuah halaman *webpage*”.

Hidayatullah dan kawistara (2017:15), “*Hyper Text Markup Language (HTML)* adalah bahasa *standard* yang di gunakan untuk menampilkan halaman *web*”.

Naista (2016:2), “*HyperText Markup Language* atau lebih singkatnya *HTML* adalah bahasa markup yang digunakan untuk membuat sebuah halaman *web*.”

Jadi, *HTML (Hyper Text Markup Language)* adalah bahasa komputer yang digunakan untuk membuat dan menampilkan halaman *web*.

### 2.3.4 Pengertian MySQL

Hidayatullah dan Kawistara (2017:175), “*MySQL* adalah salah satu aplikasi DBMS yang sudah sangat banyak digunakan oleh para pemrogram aplikasi *web*”.

Raharjo (2011:21), “*MySQL* merupakan *software RDBMS (server database)* yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (*multi-user*) dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*).”

Jadi, *MySQL* adalah aplikasi yang tergolong DBMS (*Database Management System*) yang bersifat *open source*, dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (*multi-user*) dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*).

### 2.3.5 Pengertian XAMPP

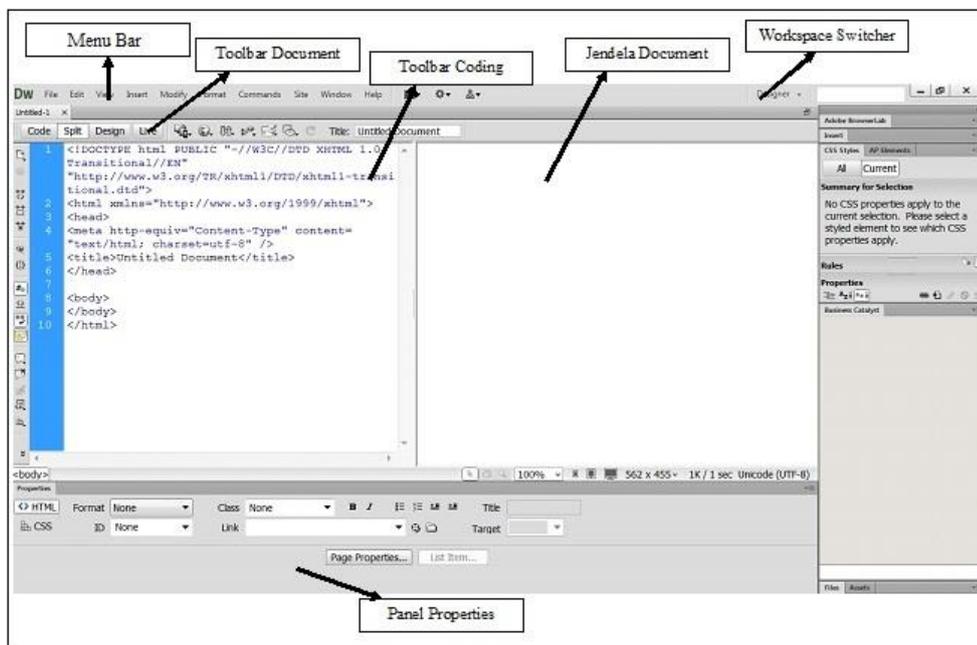
Komputer Wahana (2014:72), “XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. XAMPP adalah tool yang menyediakan paket perangkat lunak dalam satu buah paket. Dalam paket XAMPP sudah terdapat Apache (*web server*), MySQL (*database*), PHP (*server side scripting*), Perl, FTP Server, *PhpMyAdmin* dan berbagai pustaka bantu lainnya”.

Abdulloh (2016:7), “XAMPP adalah salah satu paket *installer* yang berisi Apache yang merupakan *web server* tempat menyimpan file-file yang dibutuhkan *website*, dan *PhpMyAdmin* sebagai aplikasi yang digunakan untuk perancangan *database MySQL*.”

### 2.3.6 Pengertian Adobe Dreamweaver

Mandar (2017:1), “*Adobe Dreamweaver* merupakan aplikasi pengembang yang berfungsi untuk mendesain web yang dibuat, dikembangkan, dan diproduksi oleh Adobe System.”

Menu yang ditampilkan sebagai berikut:

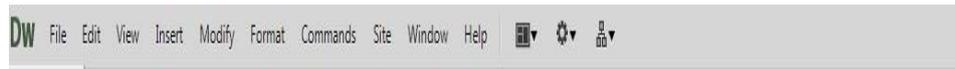


Gambar 2.3 Tampilan Halaman Dreamweaver CS6

Penjelasan:

a. *Menu Bar*

Kumpulan beberapa menu pada *Adobe Dreamweaver* yang dikelompokkan berdasarkan kategori penggunaan dan fungsinya.



**Gambar 2.4** Tampilan *Menu Bar*

b. *Toolbar Document*

Berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan *design* atau tampilan *code*. Juga dapat digunakan untuk operasi-operasi umum, misalnya untuk melihat hasil sementara halaman *web* pada jendela browser.



**Gambar 2.5** Tampilan *Toolbar Document*

c. *Toolbar Coding*

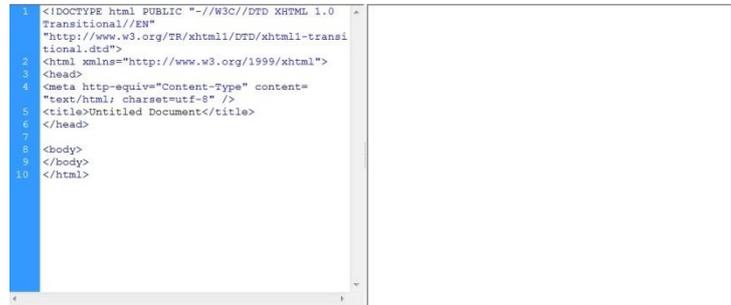
Berisi tombol-tombol yang digunakan untuk melakukan operasi kode-kode standar. *Toolbar* ini hanya tampilan pada jendela *code*.



**Gambar 2.6** Tampilan *Toolbar Coding*

#### d. Jendela *Document*

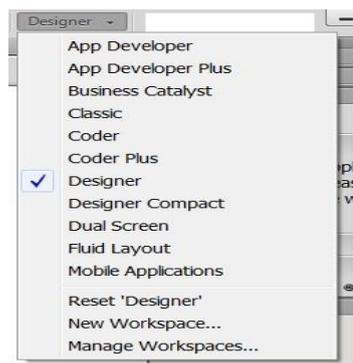
Berupa lembar kerja tempat anda membuat dan mengedit design halaman *web*.



**Gambar 2.7** Tampilan Jendela *Document*

#### e. *Workspace Switcher*

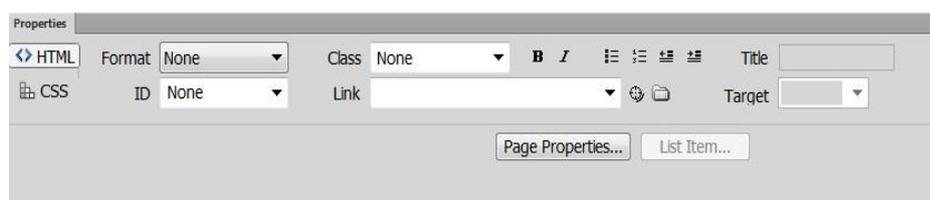
Digunakan untuk mengubah tampilan ruang kerja (*workspace*) dreamweaver CS6.



**Gambar 2.8** Tampilan *Workspace Switcher*

#### f. Panel *Properties*

Digunakan untuk melihat dan mengubah berbagai *property* objek teks pada jendela *design*. *Property* untuk satu objek dengan objek lainnya selalu berbeda-beda. Jendela ini tidak dapat diuraikan pada tampilan jendela *code*. Anda dapat mengatur *property* masing-masing objek dalam panel *properties*.



**Gambar 2.9** Tampilan Panel *Properties*



---

### 2.3.7. Pengertian CSS

Risnandar et al. (2013:39), “CSS (*Cascading Style Sheets*) adalah kumpulan aturan-aturan pemformatan yang mengontrol tampilan konten dalam sebuah halaman web.”

Naista (2016:3), “*Cascading Style Sheet* atau lebih singkatnya CSS merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam.”

Jadi, *Cascading Style Sheet* atau CSS adalah kumpulan aturan-aturan pemformatan untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam.