

LAPORAN HASIL PRAKTIKUM
PEMROGRAMAN WEB DAN MOBILE I



NAMA : OKTA ARYO

NIM : 193020503035

KELAS : A

MODUL : II Form Handling

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS PALANGKA RAYA

2021

BAB I

TUJUAN DAN LANDASAN TEORI

1.1.TUJUAN

1..1.1. Mahasiswa mampu membuat handling yang mampu mengolah data dari form HTML.

1.1.2. Mahasiswa mampu membuat batasan-batasan untuk menangani inputan dari form HTML.

1.2.LANDASAN TEORI

HTML

HTML adalah bahasa mark up web yang utama dan dijalankan secara alami di setiap browser serta di-maintain oleh World Wide Web Consortium. Berdasarkan cara kerja HTML, Anda bisa menggunakannya untuk membuat struktur konten pada website dan aplikasi web (dengan memakai coding HTML sederhana). HTML merupakan level terbawah dari teknologi frontend dan berfungsi sebagai dasar *styling* yang bisa Anda tambahkan dengan CSS dan fungsionalitas yang dapat dijalankan menggunakan JavaScript.

Dokumen HTML adalah file yang diakhiri dengan ekstensi **.html** atau **.htm**. Ekstensi file ini bisa dilihat dengan menggunakan web browser apa pun (seperti Google Chrome, Safari, atau Mozilla Firefox). Browser tersebut membaca file HTML dan me-render kontennya sehingga user internet bisa melihat dan membacanya. Biasanya, rata-rata situs web menyertakan sejumlah halaman HTML yang berbeda-beda. Contohnya, beranda utama, halaman ‘tentang kami’, halaman kontak yang semuanya memiliki dokumen HTML terpisah.

Masing-masing halaman HTML terdiri atas seperangkat **tags** (bisa disebut juga **elements**), yang mengacu pada *building block* halaman website. Tag tersebut membuat hirarki yang menyusun konten hingga menjadi bagian,

paragraf, heading, dan *block* konten lainnya. Sebagian besar element HTML memiliki tag pembuka dan penutup yang menggunakan syntax **<tag></tag>**.

Tag HTML memiliki dua tipe utama: **block-level** dan **inline tags**.

1. Elemen *block-level* memakai semua space yang tersedia dan selalu membuat line baru di dalam dokumen. Contoh dari tag *block* adalah heading dan paragraf.
2. Elemen *inline* hanya memakai space sesuai dengan kebutuhannya dan tidak membuat line baru di halaman. Biasanya elemen ini akan memformat isi konten dari elemen *block-level*. Contoh dari tag inline adalah link dan *emphasized strings*.

Tag Block-Level

Tiga tag block-level yang harus dimiliki oleh setiap dokumen HTML adalah **<html>**, **<head>**, dan **<body>**.

1. Tag **<html></html>** adalah elemen level tertinggi yang menyertakan setiap halaman HTML.
2. Tag **<head></head>** menyimpan informasi meta, seperti judul dan *charset* halaman.
3. Tag **<body></body>** melampirkan semua konten yang muncul pada suatu halaman.
4. Heading memiliki 6 level di HTML. Level tersebut bervariasi, mulai dari **<h1></h1>** sampai ke **<h6></h6>**, di mana h1 merupakan level heading tertinggi dan h6 adalah level terendah. Paragraf dibuka dan ditutup dengan tag **<p></p>**, sedangkan blockquote menggunakan tag **<blockquote></blockquote>**.
5. Division merupakan bagian konten yang lebih besar dan biasanya terdiri atas beberapa paragraf, gambar, kadang-kadang blockquote, dan elemen lebih kecil lainnya. Kita bisa membuat *mark up* dengan menggunakan tag **<div></div>**. Di dalam elemen div juga terdapat tag div lainnya.

6. Anda juga bisa menggunakan tag `` untuk list yang berurutan dan `` untuk list yang tidak berurutan. Masing-masing list item harus dibuka dan ditutup dengan tag ``.

TAG INLINE

Sebagian besar tag inline digunakan untuk memformat teks. Sebagai contoh, tag `` akan render elemen ke format **bold**, sedangkan tag `` akan ditampilkan dalam format *italic*. Hyperlink adalah elemen inline yang mewajibkan adanya tag `<a>` dan attribute **href** untuk mengindikasikan tujuan link

KELEBIHAN DAN KEKURANGAN HTML

Sama seperti hal teknis lainnya dalam dunia web, HTML juga punya kelebihan dan kekurangannya.

Kelebihan:

1. Bahasa yang digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar.
2. Dijalankan secara alami di setiap web browser.
3. Memiliki *learning curve* yang mudah.
4. Open-source dan sepenuhnya gratis.
5. Bahasa markup yang rapi dan konsisten.
6. Standard web yang resmi di-maintain oleh World Wide Web Consortium (W3C).
7. Mudah diintegrasikan dengan bahasa *backend*, seperti PHP dan Node.js.

Kekurangan:

1. Paling sering digunakan untuk halaman web statis. Untuk fitur dinamis, Anda bisa menggunakan JavaScript atau bahasa backend, seperti PHP.

2. HTML tidak memungkinkan user untuk menjalankan *logic*. Alhasil, semua halaman web harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti header dan footer.
3. Fitur-fitur baru tidak bisa digunakan secara cepat di sebagian browser.
4. Terkadang perilaku browser susah untuk diprediksi (misalnya, browser lama tidak selalu bisa render tag yang lebih baru).

Variabel superglobal PHP `$_GET` dan `$_POST` digunakan untuk mengumpulkan data-form. Contoh berikut menunjukkan form HTML sederhana dengan dua field input dan tombol submit:

```
<html>

<body>

<form action="welcome.php" method="post">

Name: <input type="text" name="name"><br> E-
mail: <input type="text" name="email"><br>

<input type="submit">

</form>

</body>

</html>
```

Ketika user mengisi form, dan menekan tombol click, data form dikirim untuk memproses file PHP dengan nama “welcome.php”. Data form dikirimkan dengan method HTTP POST. Untuk menampilkan data yang sudah disubmit bisa dilakukan dengan mencetak data tersebut menggunakan

perintah echo. File “welcome.php” adalah sebagai berikut:

```
<html>

<body>
```

Welcome <?php echo \$_POST["name"]; ?>

Your email address is: <?php echo \$_POST["email"];

?> </body>

GET dan POST membuat sebuah array (contoh array(kunci => nilai, kunci2 => nilai2, kunci3 => nilai3, ...)). Array ini menyimpan pasangan kunci/nilai, dimana kunci- kunci adalah nama-nama dari form control dan nilai-nilai adalah data input dari user. Method GET diakses menggunakan \$_GET dan method POST diakses menggunakan \$_POST. Kedua variabel ini adalah variabel superglobal, yang selalu bisa diakses, tanpa memperhatikan lingkup dan bisa diakses dari fungsi, class atau file yang berbeda tanpa harus melakukan teknik khusus. \$_GET adalah sebuah array dari variabel yang dikirimkan ke skrip melalui parameter URL. . \$_POST adalah sebuah array dari variabel yang dikirimkan ke skrip melalui method HTTP POST.

Informasi dikirim dari sebuah form dengan method GET bisa dilihat oleh semua orang (semua nama dan nilai variabel ditampilkan di URL). GET juga memiliki batas pada jumlah informasi yang dikirim. Batasannya adalah sekitar 2000 karakter. Namun, karena variabel ditunjukkan di URL, ia memungkinkan untuk dilakukan bookmark halaman. Dalam beberapa kasus, hal ini sangat bermanfaat. GET bisa digunakan untuk mengirimkan data yang tidak sensitive.

Informasi yang dikirim dari sebuah form dengan method POST tidak bisa dilihat oleh siapapun (semua nama-nama atau nilai-nilai tertanam didalam body request HTTP) dan tidak memiliki batasan jumlah informasi yang akan dikirim. POST juga mendukung fungsionalitas lanjutan seperti dukungan untuk input biner multi-part ketika sedang melakukan upload file ke server. Namun, karena variabel tidak ditampilkan di URL, tidak mungkin untuk dilakukan

bookmark halaman (data tidak ter-bookmark). Developer lebih baik menggunakan POST untuk mengirimkan data form.

Text Field

Field nama, email dan website adalah elemen-elemen text input, dan field komentar adalah textarea yaitu sebagai berikut:

Name: <input

type="text"

name="name"> E-

mail: <input

type="text"

name="email">

Website: <input

type="text"

name="website">

Comment: <textarea name="comment" rows="5" cols="40"></textarea>

Radio Button

Field jenis kelamin adalah radio button yaitu sebagai berikut:

Gender:

<input type="radio" name="gender" value="female">Female

<input type="radio" name="gender" value="male">Male

Form Element

Kode HTML untuk membentuk form pada gambar diatas adalah sebagai berikut:

```
<form          method="post"          action="<?php          echo  
htmlspecialchars($_SERVER["PHP_SELF"]);? >">.
```

BAB II

LANGKAH PENYELESAIAN DAN PEMBAHASAN

2.1. LANGKAH PENYELESAIAN

1. Buatlah program web untuk menginputkan username dan password menggunakan form dan penanganan input data dengan kriteria sebagai berikut:
 - a. username yang diinputkan tidak boleh lebih dari tujuh karakter.
 - b. password yang diinputkan harus terdiri dari huruf kapital, huruf kecil, angka dan karakter khusus.
 - c. Jumlah karakter password tidak boleh kurang dari sepuluh karakter.

2.2. PEMBAHASAN

Berikut adalah program form login :

```
<!DOCTYPE html>
<html>
<head>
  <title>Form Login</title>
</head>
<body>
  <form action="login.php" method="POST">
    <fieldset>
      <legend>SELAMAT DATANG</legend>
      <p>
        <label>Username:</label>
        <input type="text" name="username" placeholder="username..." />
      </p>
      <p>
        <label>Password:</label>
```



```

        <input type="password" name="password"
placeholder="password..." />
    </p>
    <p>
        <label><input type="checkbox" name="remember"
value="remember" /> Remember me</label>
    </p>
    <p>
        <input type="submit" name="submit" value="Login" />
    </p>
</fieldset>
</form>
</body>
</html>

```

Penjelasan :

```

<!DOCTYPE html>
<html>

```

Pada code program diatas berfungsi untuk mendeklarasikan html sebagai begin dalam program. Dimana program tidak dapat dijalankan bila html tidak dideklarasikan.

```

<head>
    <title>Form Login</title>
</head>

```

Code program diatas merupakan deklarasi dari program yang dibuat yaitu program form login dimana state <head> berfungsi sebagai pengeksekusi title dari nama program.

```
<body>
  <form action="login.php" method="POST">
    <fieldset>
```

Untuk code program diatas berfungsi untuk mengeksekusi program pada saat login. Saat program berhasil dibuat maka ketika masuk ke web untuk melakukan login maka program akan menerima masukkan data saat login dilakukan.

```
<legend>SELAMAT DATANG</legend>
  <p>
```

Pada code program diatas berfungsi sebagai tabel dalam form login dimana dengan mendeklarasikan kata-kata dalam tabel login dengan kalimat ‘SELAMAT DATANG’.

```
    <label>Username:</label>
    <input type="text" name="username" placeholder="username..." />
  </p>
  <p>
    <label>Password:</label>
    <input type="password" name="password"
placeholder="password..." />
  </p>
  <p>
```

Code program diatas merupakan program utama dimana dalam code diatas terdapat deklarasi dari inputan username dan password. Pada inputan username tipe data atau tipe huruf inputan dalam username adalah text, dimana variable

name adalah username. Untuk inputan password tipe data atau tipe huruf inputan dalam password adalah password dimana dalam inputan user atau masukkan oleh user akan terlihat seperti huruf bulat hitam, dimana variable name adalah password.

```
<label><input type="checkbox" name="remember"
value="remember" /> Remember me</label>
</p>
<p>
```

Kemudian untuk code program diatas adalah code untuk menyimpan data dari inputan username dan password, dimana saat login pada program maka program akan menyimpan masukkan dari username dan password saat login.

```
<input type="submit" name="submit" value="Login" />
</p>
</fieldset>
</form>
</body>
</html>
```

Code program diatas merupakan akhir dari program dimana untuk variable inputan menggunakan submit dengan masukkan login. Kemudian deklarasi dari html sebagai end atau akhir dari program.

Hasil dari program diatas adalah sebagai berikut :

← → ↻ ⓘ localhost/modul%201%20pemweb/modul%202.html

Login

Username:

Password:

☐ Remember me

Gambar 2.1.1 hasil tampilan login pada web

BAB III

KESIMPULAN

Dari praktikum diatas kesimpulan yang didapatkan adalah pemrograman web pada html merupakan pemrogram bahasa mark up web yang utama dan dijalankan secara alami di setiap browser serta di-maintain oleh World Wide Web Consortium. Pada sebuah website sendiri terdapat beberapa kelebihan maupun kelebihannya baik bagi pemakai maupun pembuat. Web sendiri berfungsi untuk menyimpan, menampilkan atau meberikan data yang dalam berbagai bentuk seperti file, video, foto dan lainnya maupun informasi yang sedang terjadi.

DAFTAR PUSTAKA

Belajar HTML #03: Membuat Paragraf pada HTML. (n.d.).

<https://www.petanikode.com/html-paragraf/>

Christy, A. (2020). Apa Itu HTML? Fungsi dan Cara Kerja HTML. In *Hostinger International* (p. 1). <https://www.hostinger.co.id/tutorial/apa-itu-html/>

Praktikum, K. (n.d.). *MODUL PRAKTIKUM PEMROGRAMAN WEB I Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya.*