

**LAPORAN PRAKTIKUM  
PEMROGRAMAN WEB DAN MOBILE I**



**NAMA : ABIGAEL GERRY PRATAMA**  
**NIM : 193020503021**  
**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

Form merupakan bagian pada HTML yang dapat digunakan untuk membuat elemen Form pada halaman Web. Elemen Form dapat terdiri dari check box, radio button, menu, text box, text area, dan button.

Tag	Function
<form></form>	Specifies a form for user data input.
<input type="text" ...	Specifies a text box.
<input type="password" ...	Specifies a text box where typed text is hidden.
<input type="submit" ...	Specifies a submit button.
<input type="reset" ...	Specifies a cancel/reset button.
<input type="checkbox" ...	Specifies a check box.
<input type="radio" ...	Specifies a radio button.
<textarea></textarea>	Specifies a text area in a form.
<select></select>	Specifies a menu/drop-down list.
<option></option>	Specifies an option in select element.
<optgroup></optgroup>	Specifies a group of options.

Gambar 1.1 Element HTML

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>

```

Gambar 1.2 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.

```

<html>
  <body>
    Welcome <?php echo $_POST["name"]; ?><br>
    Your email address is: <?php echo $_POST["email"];
    ?> </body>
</html>

```

Gambar 1.3 HTML dan PHP

Jika field nama diinputkan dengan Tono dan email diinputkan dengan tono@mail.com maka output yang akan tampil adalah sebagai berikut: Welcome Budi Your email address is tono@mail.com Hasil yang sama juga akan tampil dengan menggunakan method get sebagai berikut:

```

<html>
  <body>
    <form action="welcome_get.php" method="get">
      Name: <input type="text" name="name"><br>
      E-mail: <input type="text" name="email"><br>
      <input type="submit">
    </form>
  </body>
</html>

```

Gambar 1.4 HTML dan PHP

dengan file “welcome\_get.php” sebagai berikut:

```
<html>
  <body>
    Welcome <?php echo $_GET["name"]; ?><br>
    Your email address is: <?php echo $_GET["email"];
    ?> </body>
</html>
```

Gambar 1.5 HTML dan PHP

## GET vs. POST

GET dan POST membuat sebuah array (contoh array(kunci => nilai, kunci2 => nilai2, kunci3 => nilai3, ...)). Array ini menyimpan pasangan kunci/nilai, dimana kuncikunci 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.

## 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>
```

Gambar 1.6 Text Field

## BAB II

### PEMBAHASAN

Buatlah program web untuk menginputkan username dan password menggunakan form dan penanganan input data dengan kriteria sebagai berikut:

1. username yang diinputkan tidak boleh lebih dari tujuh karakter.
2. password yang diinputkan harus terdiri dari huruf kapital, huruf kecil, angka dan karakter khusus.
3. Jumlah karakter password tidak boleh kurang dari sepuluh karakter.

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST"){
    $username = $_REQUEST["username"];
    $password = $_REQUEST["password"];
    $user = strlen($username);
    $pass = strlen($password);
    $x = false;

    if($user>7){
        echo "<script>alert('password lebih dari 7');</script>";
        $x = true;
    }
    if (!preg_match("/[A-Z]/", $password) ) {
        echo "<script>alert('password kapital');</script>";
        $x = true;
    }
    if (!preg_match("/[a-z]/", $password)) {
        echo "<script>alert('password kecil');</script>";
        $x = true;
    }

    if (!preg_match("/[^\a-zA-Z\d]/", $password)) {
        echo "<script>alert('password special character');</script>";
        $x = true;
    }

    if (!preg_match("/[0-9]/", $password)) {
        echo "<script>alert('password digit');</script>";
        $x = true;
    }
    if($pass<10){
        echo "<script>alert('password kurang dari 10');</script>";
        $x = true;
    }
    if( $x == false ){
        echo "<script>alert('login berhasil');</script>";
    }
}
```



Gambar 2.1 PHP

Variabel `SERVER` memiliki `REQUEST_METHOD` berupa method `POST`. Hal ini untuk memvalidasi form jika sudah disubmit akan tervalidasi. Variabel `username` dan `password` diinisialisasikan untuk menampung nilai `username` dan `password`. Variabel `user` dan `pass` untuk menghitung panjang string dari `username` dan `password`. Variabel `x` diinisiasikan sebagai nilai `false`.

`if($user>7)` digunakan untuk mengecek apakah panjang karakter dari `$username` lebih besar dari 7 atau tidak, jika lebih besar maka program tersebut akan menampilkan tulisan dalam `echo` dan `$error` akan bernilai `true`.

`if($pass<10)` digunakan untuk mengecek apakah panjang karakter dari `$password` lebih kecil dari 10 atau tidak, jika lebih kecil maka program tersebut akan menampilkan tulisan dalam `echo` dan `$error` akan bernilai `true`.

`if (!preg_match("/[A-Z]/"` digunakan untuk mengecek apakah `$password` mengandung huruf kapital atau tidak, dapat dilihat terdapat fungsi `preg_match`, dimana fungsi ini digunakan untuk mengecek apakah suatu kecocokan terdapat dalam string atau tidak, dimana kecocokan yang dimaksud sesuai dengan yang berada pada bagian `"`. Dalam sintaks tersebut, kondisi dari kecocokannya adalah `"/[A-Z]/"`. Jika `password` tidak mengandung huruf kapital, maka program tersebut akan menampilkan tulisan dalam `echo` dan `$error` akan bernilai `true`.

`if (!preg_match("/[a-z]/"` digunakan untuk mengecek apakah `$password` mengandung huruf kecil atau tidak, jika tidak maka program tersebut akan menampilkan tulisan dalam `echo` dan `$error` akan bernilai `true`. Sintaks di atas juga menggunakan `preg_match` dengan kondisi `"/[a-z]/"`.

if (!preg\_match("/^[a-zA-Z\d]/") untuk mengecek apakah \$password mengandung karakter spesial atau tidak, jika tidak maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true. Sintaks di atas juga menggunakan preg\_match dengan kondisi "/^[a-zA-Z\d]/".

if (!preg\_match("/[0-9]/) digunakan untuk mengecek apakah \$password mengandung angka atau tidak, jika tidak maka program tersebut akan menampilkan tulisan dalam echo dan \$error akan bernilai true. Sintaks di atas juga menggunakan preg\_match dengan kondisi "/[0-9]/".

```
<!DOCTYPE html>
<html lang="en">
<head>
  <link rel="stylesheet" href="style.css">
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login | Page</title>
</head>
<body>
<div class="border">
<h3>Login</h3>
<form action="" method="post" name="Login_Form">
  <label class="un">Username</label><br>
  <input class="pw" name="username" type="text" placeholder="username">
<br>
  <label class="un">Password</label><br>
  <input class="pw" name="password" type="text" placeholder="password">
<br>
  <input class="lg" name="Submit" type="submit" value="Login">
<h1>Syarat dan Ketentuan</h1>
<p> Username yang diinputkan tidak boleh lebih dari 7 karakter.<br>
  Password yang diinputkan harus terdiri dari huruf kapital, huruf kecil, angka dan karakter khusus.<br>
  Jumlah karakter password tidak boleh kurang dari 10 karakter.
</p>
</form>
</div>
</body>
</html>
```

Gambar 2.2 HTML

Elemen <form> adalah wadah untuk berbagai jenis elemen masukan, seperti: bidang teks, kotak centang, tombol radio, tombol kirim, dll. Semua elemen

formulir yang berbeda dibahas dalam bab ini: Elemen Formulir HTML. Pada html ini juga ditambahkan file css untuk memperindah web login. Elemen `<input>` HTML adalah elemen formulir yang paling sering digunakan. Sebuah elemen `<input>` dapat ditampilkan dengan banyak cara, bergantung pada atribut type. Judul pada halaman web ini diberi nama Login | Page. Tag `<label>` mendefinisikan label untuk beberapa elemen. Elemen `<input>` menentukan bidang masukan tempat pengguna dapat memasukkan data yang merupakan elemen formulir yang paling penting dan elemen ini dapat ditampilkan dalam beberapa cara, bergantung pada atribut type. Dan terdapat kelas disetiap inputan agar dapat di desain pada css. Elemen `<br>` menyisipkan satu baris baru. Berikut adalah kode css yang dibuat untuk memperindah halaman html.

```
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@600&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@300;400;600&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@300;400;600;800&display=swap');

body{
  font-family: 'Montserrat', sans-serif;
  color: #ffffff;
  background-color: rgb(255, 255, 255);
}

h3{
  top: 169px;
  text-align: center;
  font-family: Montserrat;
  font-weight: 800;
  font-size: 72px;
  line-height: 88px;
}

.border{
  position: absolute;
  width: 592px;
  height: 650px;
  left: 445px;
  top: 55px;
  background: #FF7C7C;
  border-radius: 20px;
}
```



```
.un{
  position: absolute;
  font-size: 15px;
  font-weight: 600;
  left: 260px;
}

.pw{
  border: 0;
  display: block;
  margin: 5px auto;
  text-align: center;
  border: 2px solid #d35555;
  padding: 10px 10px;
  width: 200px;
  outline: none;
  border-radius: 20px;
}

.lg{
  font-family: 'Montserrat', sans-serif;
  position: absolute;
  left: 255px;
  border: 2px solid #d35555;
  padding: 10px 20px;
  border-radius: 20px;
}

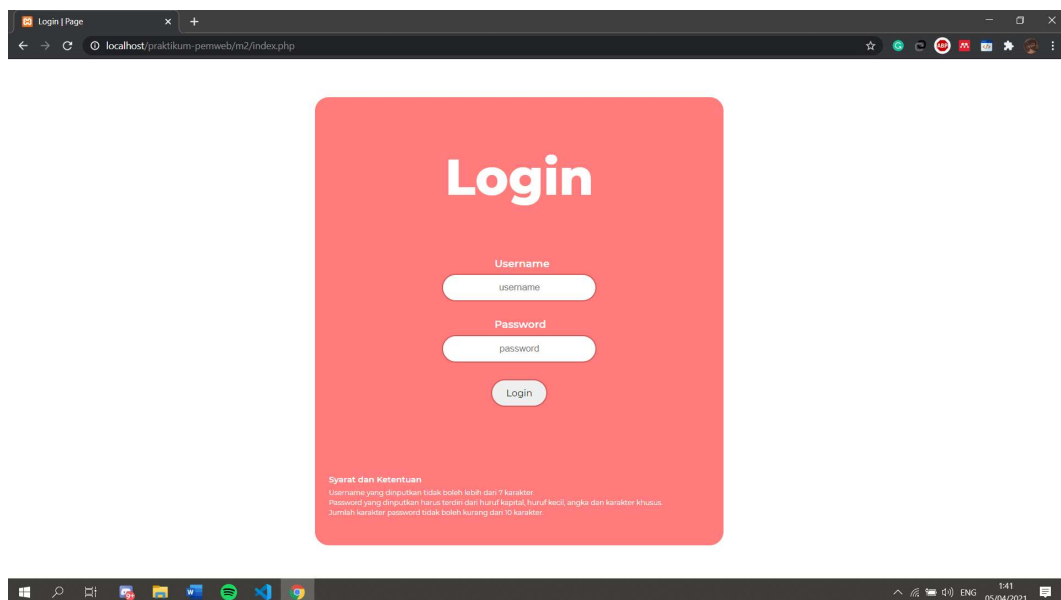
h1{
  margin-top: 100px;
  font-size: 12px;
  font-weight: 600;
  position: absolute;
  left: 20px;
  bottom: 80px;
}

p{
  font-size: 10px;
  line-height: 15px;
  font-weight: 500;
  position: absolute;
  left: 20px;
  bottom: 30px;
}
```

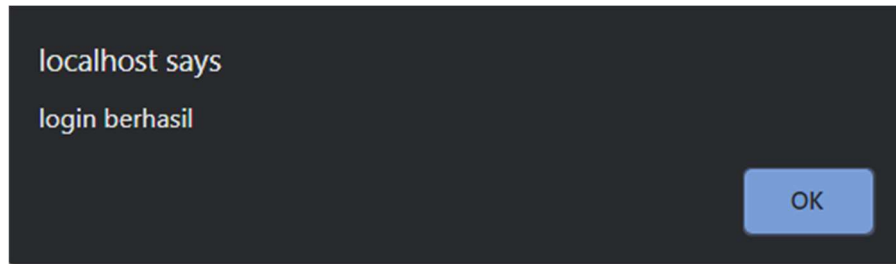
Gambar 2.3 CSS

Disini font Montserrat dari google font. Dengan cara menggunakan fungsi `@import url` lalu memasukkan url font yang akan di import. Pada tag body, fungsi yang dimasukkan akan berpengaruh ke web yang dibuat. Pada tag ini terdapat properti `font-family` menentukan font untuk sebuah elemen. Properti `background-color` menyetel warna latar belakang sebuah elemen. Latar belakang elemen adalah ukuran total elemen, termasuk padding dan batas (tetapi bukan margin). Properti `color` menentukan warna teks. Properti `font-size` menyetel ukuran font. Properti `font-weight` mengatur seberapa tebal atau tipis karakter dalam teks harus ditampilkan.

Properti `line-height` untuk mengatur jarak antar ke bawah kalimat. Elemen margin adalah ruang antara kontennya dan tepinya. Properti `position` menentukan jenis metode pemosisian yang digunakan untuk elemen (statis, relatif, absolut, tetap, atau lengket). Properti kiri dan kanan mempengaruhi posisi horizontal dari elemen yang diposisikan. Properti atas dan bawah memengaruhi posisi vertikal elemen yang diposisikan. Properti ini tidak berpengaruh pada elemen yang tidak diposisikan. Properti lebar mengatur lebar elemen. Lebar elemen tidak termasuk padding, border, atau margin. Berikut adalah hasil dari program yang dijalankan.



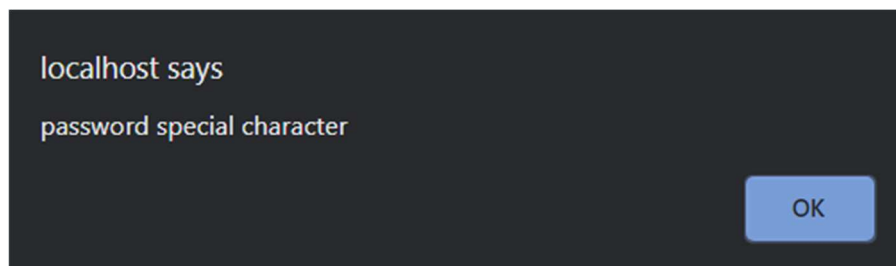
Gambar 2.4 Halaman Login



Gambar 2.5 Login Berhasil



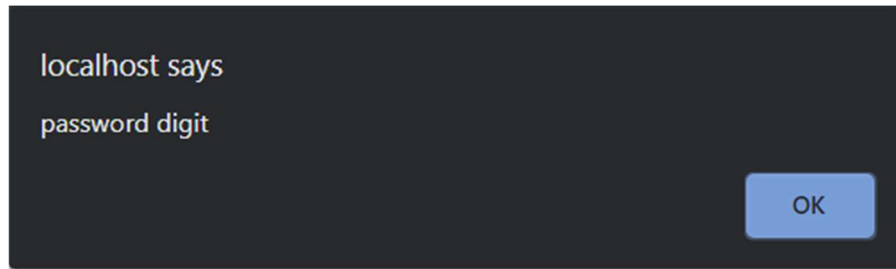
Gambar 2.6 Password Salah



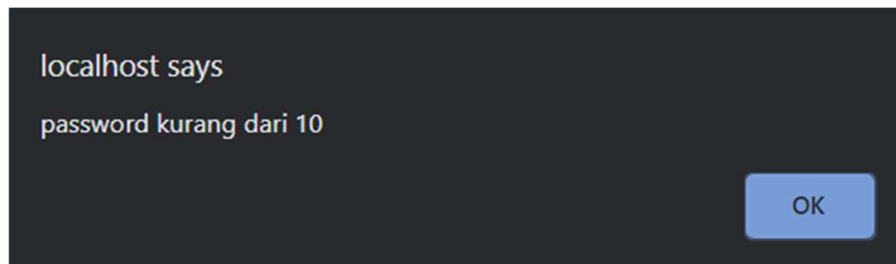
Gambar 2.7 Password Salah



Gambar 2.8 Password Salah



Gambar 2.9 Password Salah



Gambar 2.10 Password Salah

### **BAB III**

### **KESIMPULAN**

Kesimpulan dari praktikum ini agar dapat membuat handling yang mampu mengolah data dari form HTML dan mampu membuat batasan-batasan untuk menangani inputan dari form HTML. Form handling ialah suatu mekanisme untuk menangani suatu masukan dari form yang dikirim oleh pengguna. Form merupakan bagian pada HTML yang dapat digunakan untuk membuat elemen Form pada halaman Web. Umumnya digunakan untuk menerima masukan/input data dari pengguna, misalkan ketika registrasi kesuatu website atau ketika login pada halaman dan sebagainya.

## **BAB IV**

### **DAFTAR PUSTAKA**

Dosen Teknik Informatika. Pemrograman Web dan Mobile 1. 2021. MODUL PRAKTIKUM PEMROGRAMAN WEB I. Universitas Palangka Raya. Fakultas Teknik. Jurusan Teknik Informatika.

*Belajar HTML Part 11 : Membuat Form pada HTML - Malas Ngoding.* (n.d.).  
<https://www.malasngoding.com/belajar-html-membuat-form-pada-html/>

## LAMPIRAN

Tag	Function
<form></form>	Specifies a form for user data input.
<input type="text" ...	Specifies a text box.
<input type="password" ...	Specifies a text box where typed text is hidden.
<input type="submit" ...	Specifies a submit button.
<input type="reset" ...	Specifies a cancel/reset button.
<input type="checkbox" ...	Specifies a check box.
<input type="radio" ...	Specifies a radio button.
<textarea></textarea>	Specifies a text area in a form.
<select></select>	Specifies a menu/drop-down list.
<option></option>	Specifies an option in select element.
<optgroup></optgroup>	Specifies a group of options.

Gambar 1.1 Element HTML

```
<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>
```

Gambar 1.2 HTML

```
<html>
  <body>
    Welcome <?php echo $_POST["name"]; ?><br>
    Your email address is: <?php echo $_POST["email"];
  ?> </body>
</html>
```

Gambar 1.3 HTML dan PHP

```
<html>
  <body>

    <form action="welcome_get.php" method="get">
      Name: <input type="text" name="name"><br>
      E-mail: <input type="text" name="email"><br>
      <input type="submit">
    </form>
  </body>
</html>
```

Gambar 1.4 HTML dan PHP

```
<html>
  <body>
    Welcome <?php echo $_GET["name"]; ?><br>
    Your email address is: <?php echo $_GET["email"];
    ?> </body>
</html>
```

Gambar 1.5 HTML dan PHP

```
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>
```

Gambar 1.6 Text Field

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST"){
    $username = $_REQUEST["username"];
    $password = $_REQUEST["password"];
    $user = strlen($username);
    $pass = strlen($password);
    $x = false;

    if($user>7){
        echo "<script>alert('password lebih dari 7');</script>";
        $x = true;
    }
    if (!preg_match("/[A-Z]/", $password) ) {
        echo "<script>alert('password kapital');</script>";
        $x = true;
    }
}
```



```

        if (!preg_match("/[a-z]/", $password)) {
            echo "<script>alert('password kecil');</script>";
            $x = true;
        }

        if (!preg_match("/[^\a-zA-Z\d]/", $password)) {
            echo "<script>alert('password special character');</script>";
            $x = true;
        }

        if (!preg_match("/[0-9]/", $password)) {
            echo "<script>alert('password digit');</script>";
            $x = true;
        }
        if($pass<10){
            echo "<script>alert('password kurang dari 10');</script>";
            $x = true;
        }
        if( $x == false ){
            echo "<script>alert('login berhasil');</script>";
        }
    }
}
?>

```

Gambar 2.1 PHP

```

<!DOCTYPE html>
<html lang="en">
<head>
    <link rel="stylesheet" href="style.css">
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login | Page</title>
</head>
<body>
<div class="border">
<h3>Login</h3>
<form action="" method="post" name="Login_Form">
    <lable class="un">Username</lable><br>
    <input class="pw" name="username" type="text" placeholder="username"
><br>
    <lable class="un">Password</lable><br>
    <input class="pw" name="password" type="text" placeholder="password"
><br>
    <input class="lg" name="Submit" type="submit" value="Login">
<h1>Syarat dan Ketentuan</h1>
<p> Username yang diinputkan tidak boleh lebih dari 7 karakter.<br>
    Password yang diinputkan harus terdiri dari huruf kapital, huruf kecil, a
ngka dan karakter khusus.<br>
    Jumlah karakter password tidak boleh kurang dari 10 karakter.
</p>
</form>

```

```
</div>
</body>
</html>
```

Gambar 2.2 HTML

```
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@600&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@300;400;600&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@300;400;600;800&display=swap');

body{
  font-family: 'Montserrat', sans-serif;
  color: #ffffff;
  background-color: rgb(255, 255, 255);
}

h3{
  top: 169px;
  text-align: center;
  font-family: Montserrat;
  font-weight: 800;
  font-size: 72px;
  line-height: 88px;
}

.border{
  position: absolute;
  width: 592px;
  height: 650px;
  left: 445px;
  top: 55px;
  background: #FF7C7C;
  border-radius: 20px;
}

.un{
  position: absolute;
  font-size: 15px;
  font-weight: 600;
  left: 260px;
}

.pw{
  border: 0;
  display: block;
  margin: 5px auto;
  text-align: center;
  border: 2px solid #d35555;
```

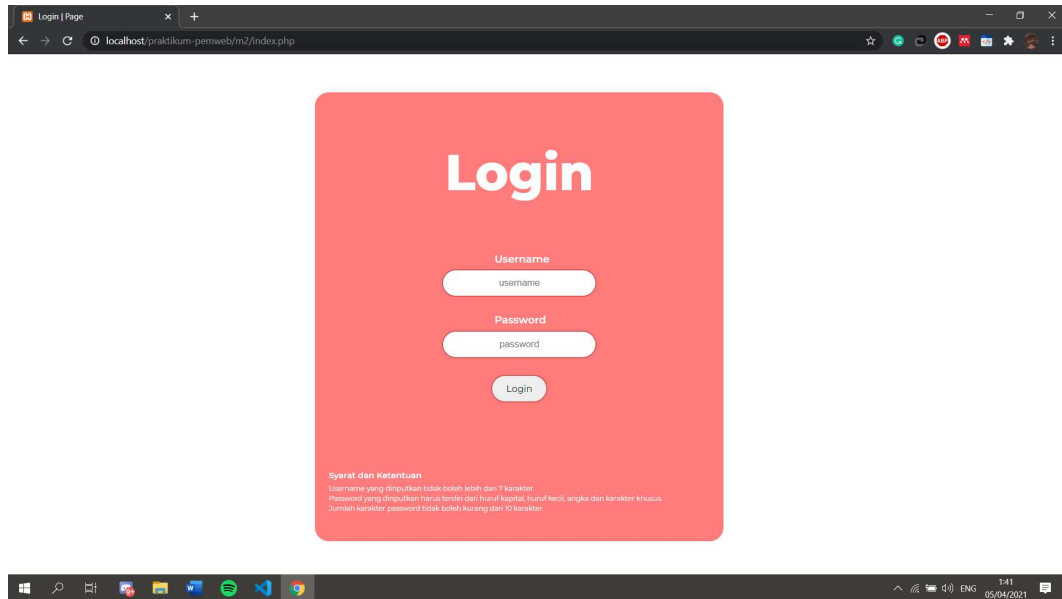
```
padding: 10px 10px;
width: 200px;
outline: none;
border-radius: 20px;
}

.lg{
  font-family: 'Montserrat', sans-serif;
  position: absolute;
  left: 255px;
  border: 2px solid #d35555;
  padding: 10px 20px;
  border-radius: 20px;
}

h1{
  margin-top: 100px;
  font-size: 12px;
  font-weight: 600;
  position: absolute;
  left: 20px;
  bottom: 80px;
}

p{
  font-size: 10px;
  line-height: 15px;
  font-weight: 500;
  position: absolute;
  left: 20px;
  bottom: 30px;
}
```

Gambar 2.3 CSS



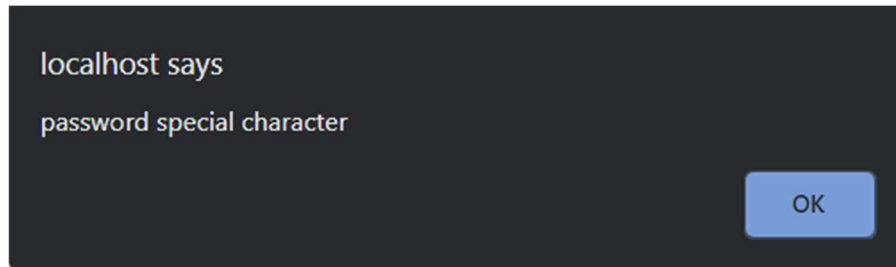
Gambar 2.4 Halaman Login



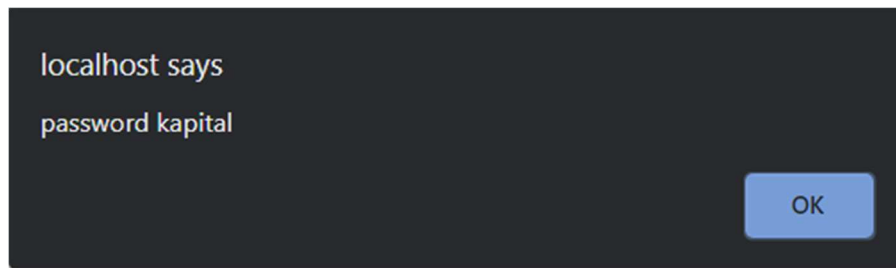
Gambar 2.5 Login Berhasil



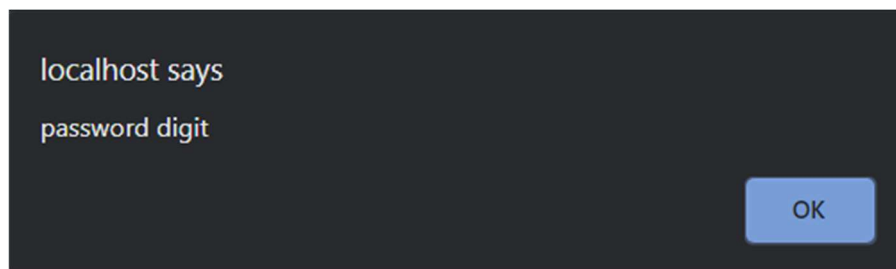
Gambar 2.6 Password Salah



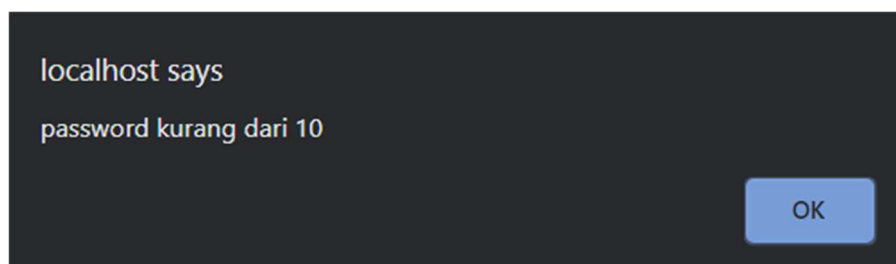
Gambar 2.7 Password Salah



Gambar 2.8 Password Salah



Gambar 2.9 Password Salah



Gambar 2.10 Password Salah