

# DOKUMENTASI PEMBUATAN FORM REGISTRASI EMAIL

*Albertus Kelvin / 16514297 / STEI*

Untuk uji coba : ADMIN – username : Admin dan password : 123

USER BIASA – username : guest dan password : guest

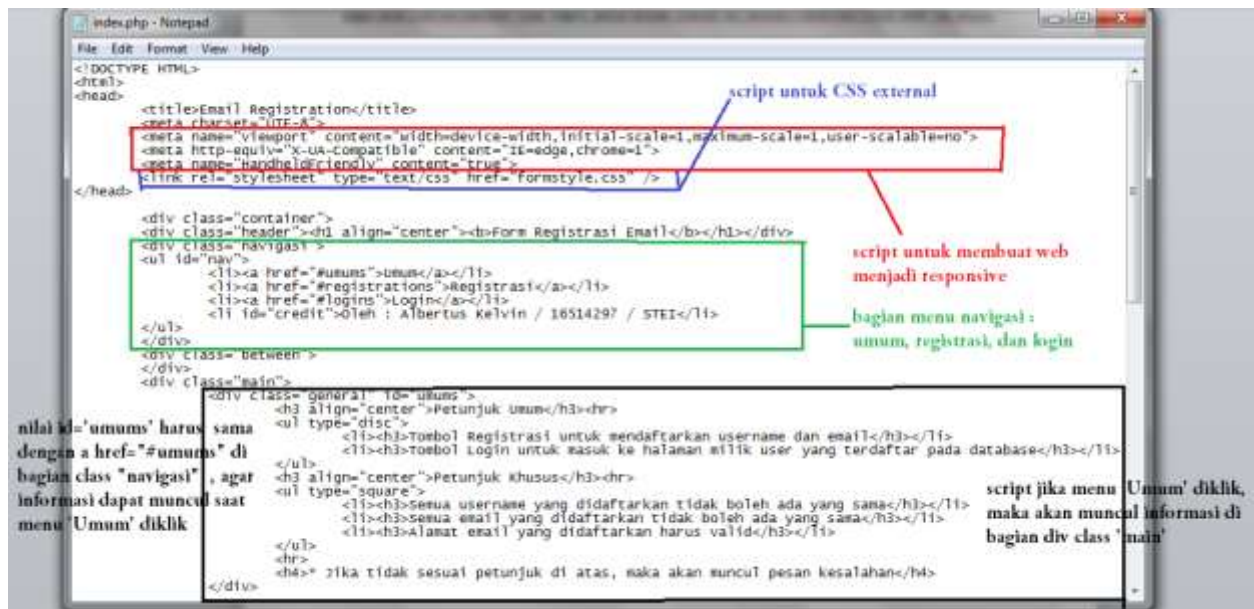
Form registrasi email ini dibuat dengan memanfaatkan PHP, HTML, dan CSS. Fitur yang saya buat untuk situs registrasi email ini adalah : **responsive (pc desktop, tablet landscape dan portrait), navigasi mudah, dan ada pembatasan akses URL bagi Admin dan User biasa (khususnya akses mailing-list.php).**

## Gambaran Umum

User akan memasukkan nama, username, dan alamat email. Username dan email yang dimasukkan haruslah valid, yaitu belum pernah didaftarkan sama sekali. Setelah mendaftar, maka nama, username, dan email akan disimpan di dalam sebuah database (bukan array). User juga bisa memanfaatkan fitur login, yaitu untuk masuk ke sebuah halaman milik user tersebut. Login dinyatakan berhasil jika username dan email yang dimasukkan sudah terdaftar di database dan ada kecocokan antara username dan email.

## Dokumentasi Pembuatan : index.php

index.php adalah file untuk pembuatan halaman utama form registrasi, dimana user dapat memasukkan nama, username, dan alamat email. Berikut penjelasan singkatnya.



Gambar 1. Salah satu isi file index.php

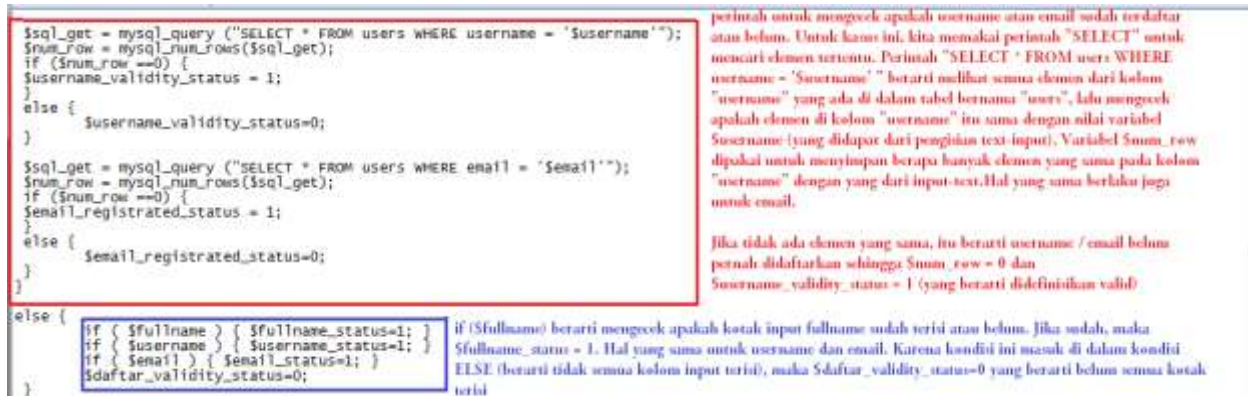
<p>class "general" untuk bagian CSS</p> <p>nilai id="registrations" harus sama dengan &lt;a href="#registrations" agar isi menu bisa ditampilkan saat button diklik</p>	<pre> &lt;div class="general" id="registrations"&gt;   &lt;form action="desktop-status.php" method="post"&gt;     &lt;h2&gt;FORM REGISTRASI&lt;/h2&gt;&lt;hr&gt;     &lt;table border="0" align="center"&gt;       &lt;tr&gt;         &lt;td align="left"&gt;&lt;h2&gt;Nama Lengkap&lt;/h2&gt;&lt;/td&gt; &lt;td width="50"&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td align="left"&gt;&lt;h2&gt;Username&lt;/h2&gt;&lt;/td&gt; &lt;td width="50"&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td align="left"&gt;&lt;h2&gt;Email Address&lt;/h2&gt;&lt;/td&gt; &lt;td width="50"&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td align="left"&gt;&lt;input type="submit" id="button-registrasi" value="SUBMIT" name="submission"&gt;&lt;/td&gt;       &lt;/tr&gt;     &lt;/table&gt;   &lt;/form&gt; &lt;/div&gt; &lt;div class="general" id="logins"&gt;   &lt;form action="login-desktop.php" method="post"&gt;     &lt;h2&gt;FORM LOGIN&lt;/h2&gt;&lt;hr&gt;     &lt;table border="0" align="center"&gt;       &lt;tr&gt;         &lt;td align="left"&gt;&lt;h2&gt;Username&lt;/h2&gt;&lt;/td&gt; &lt;td width="50"&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td align="left"&gt;&lt;h2&gt;Email Address&lt;/h2&gt;&lt;/td&gt; &lt;td width="50"&gt;&lt;/td&gt;       &lt;/tr&gt;       &lt;tr&gt;         &lt;td align="left"&gt;&lt;input type="submit" id="button-login" value="LOGIN" name="submission_login"&gt;&lt;/td&gt;       &lt;/tr&gt;     &lt;/table&gt;   &lt;/form&gt; &lt;/div&gt; </pre>	<p>script untuk menampilkan isi menu "Registrasi" yang berbentuk sebuah form. Jika tombol "SUBMIT" diklik, maka akan langsung menuju file "desktop-status.php"</p> <p>script untuk menampilkan isi menu "Login" yang berbentuk form. Jika tombol "LOGIN" diklik, maka akan langsung menuju file "login-desktop.php"</p>
---	---	---

Gambar 2. Bagian menampilkan isi menu "Registrasi" dan "Login"

**Dokumentasi pembuatan : desktop-status.php** --- file yang berisi informasi mengenai apa yang terjadi setelah user memasukkan nama, username, dan email. Status registrasi dapat berupa "SUKSES" (jika semua input text pada form terisi semua dan alamat email valid serta username dan email belum pernah didaftarkan sebelumnya), "USERNAME SUDAH TERSEDIA" (jika username yang dimasukkan sudah ada dalam database), "ADA KOTAK INPUT YANG MASIH KOSONG" (jika ada beberapa/ semua kotak input yang belum terisi), dll.

<pre> desktop-status.php - Notepad File Edit Format View Help &lt;!DOCTYPE HTML&gt; &lt;html&gt; &lt;head&gt;   &lt;link rel="stylesheet" type="text/css" href="formstyle.css"&gt; &lt;/head&gt; &lt;body&gt; &lt;div class="header_status"&gt;&lt;h1 align="center"&gt;STATUS REGISTRASI&lt;/h1&gt;&lt;/div&gt; &lt;?php   include "mydatabase.php";   include "tabelanddatabase.php";   \$fullname = addslashes(strip_tags (\$_POST['fullname']));   \$username = addslashes(strip_tags (\$_POST['username']));   \$email = addslashes(strip_tags (\$_POST['email']));    //script ini untuk mengecek apakah form sudah terisi semua   \$email_validity_status=0;   \$daftar_validity_status=0;   \$username_validity_status=0;   \$email_registered_status=0;   \$fullname_status=0; \$username_status=0; \$email_status=0;    if (\$fullname &amp;&amp; \$username &amp;&amp; \$email)   // untuk validasi alamat email si user   \$daftar_validity_status=1;   \$fullname_status=1; \$username_status=1; \$email_status=1;    \$email = filter_var(\$email, FILTER_SANITIZE_EMAIL);   if (filter_var(\$email, FILTER_VALIDATE_EMAIL) === false) {     \$email_validity_status=0;   } else {     \$email_validity_status=1;   } </pre>	<p>mengambil file bernama "mydatabase.php" (file pembuat database) dan "tabelanddatabase.php" (file pembuat tabel di database)</p> <p>mengambil / mendeteksi isi dari input text pada form registrasi. 'fullname', 'username', dan 'email' di dalam perintah \$ _POST merupakan nama dari input text yang bersangkutan</p> <p>kondisi awal. \$email _validity _status untuk cek apakah emailnya valid, \$daftar _validity _status untuk cek apakah form sudah terisi semua, \$username _validity _status mengecek apakah username sudah terdaftar / belum, \$email _registered _status mengecek apakah email sudah terdaftar / belum, \$fullname _status mengecek apakah kotak 'fullname' sudah terisi/ belum dan definisi yang sama seperti \$fullname _status untuk \$username _status dan \$email _status</p> <p>jika form sudah terisi semua (lihat perintah if di atasnya), maka langkah berikutnya adalah mengecek validitas alamat email. Jika form terisi semua, maka otomatis kotak input fullname, username, dan email juga sudah terisi. Maka, nilai \$fullname _status, \$username _status, dan \$email _status menjadi sama dengan 1.</p> <p>Kondisi filter_var(\$email, FILTER_VALIDATE_EMAIL)===false berarti hasil validasi bernilai True dan \$email _validity _status menjadi 1.</p> <p>Perintah filter_var(\$email, FILTER_SANITIZE_EMAIL) berfungsi untuk menghilangkan beberapa karakter spesifik pada \$email, untuk kemudian divalidasi lewat FILTER_VALIDATE_EMAIL</p>
--	--

Gambar 3. Salah satu isi file desktop-status.php



Gambar 4. Script untuk mengecek apakah username/ email sudah terdaftar atau belum



Gambar 5. Script untuk menampilkan pesan kesalahan

## Dokumentasi pembuatan : login-desktop.php

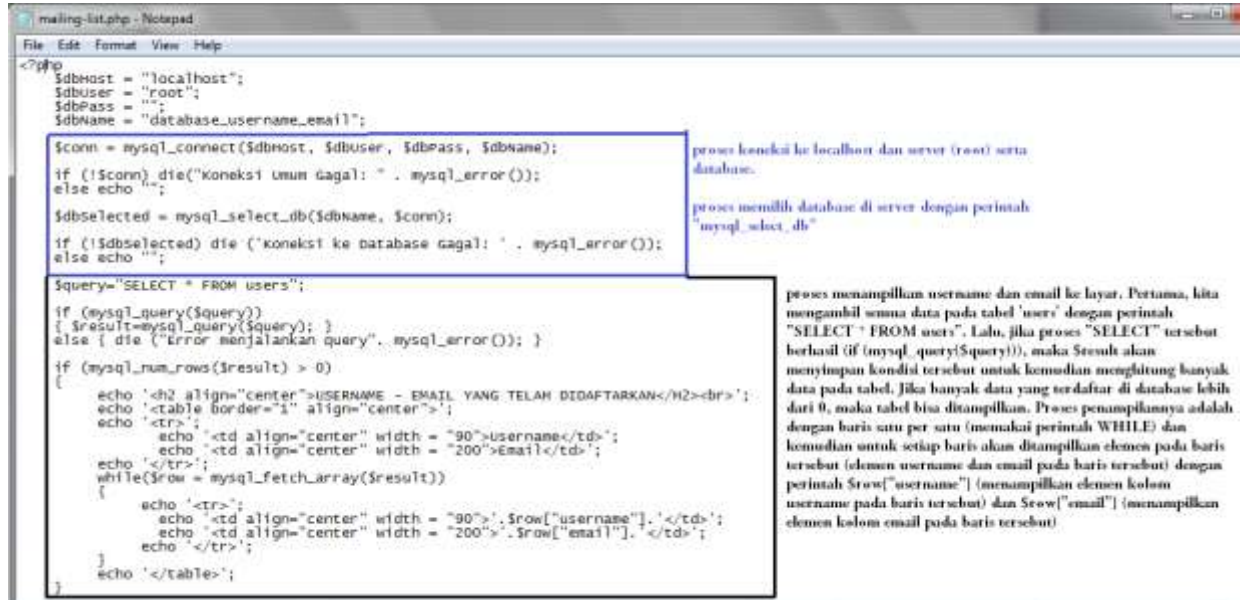
Intinya adalah file ini bertujuan untuk menampilkan status login si user, apakah login itu berhasil atau tidak.





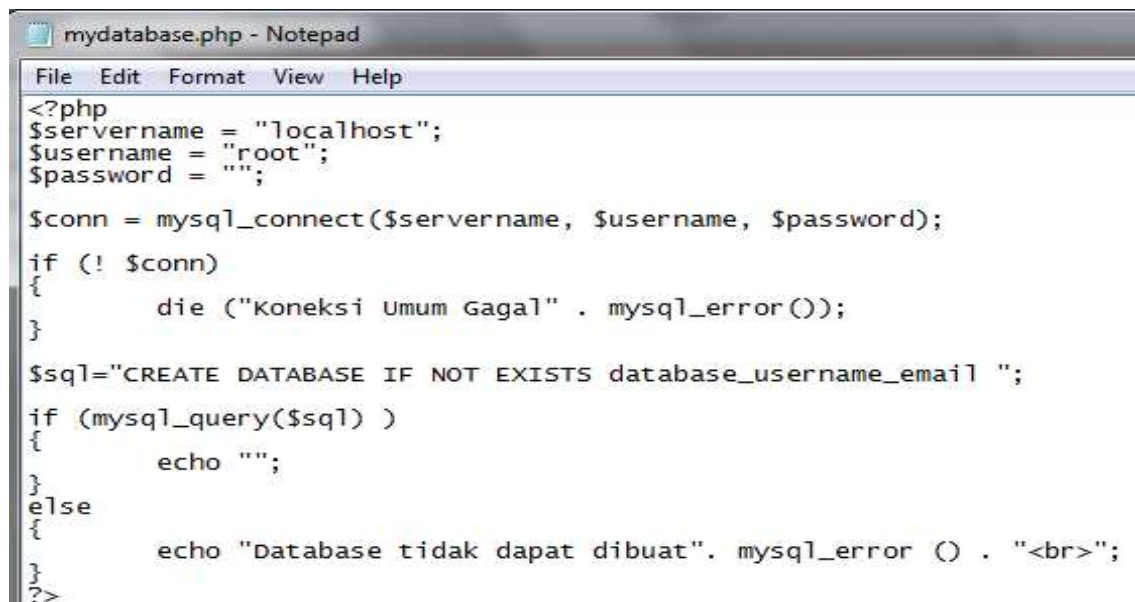
## Dokumentasi pembuatan : mailing-list.php

mailing-list.php digunakan untuk menampilkan semua username dan email yang telah didaftarkan.



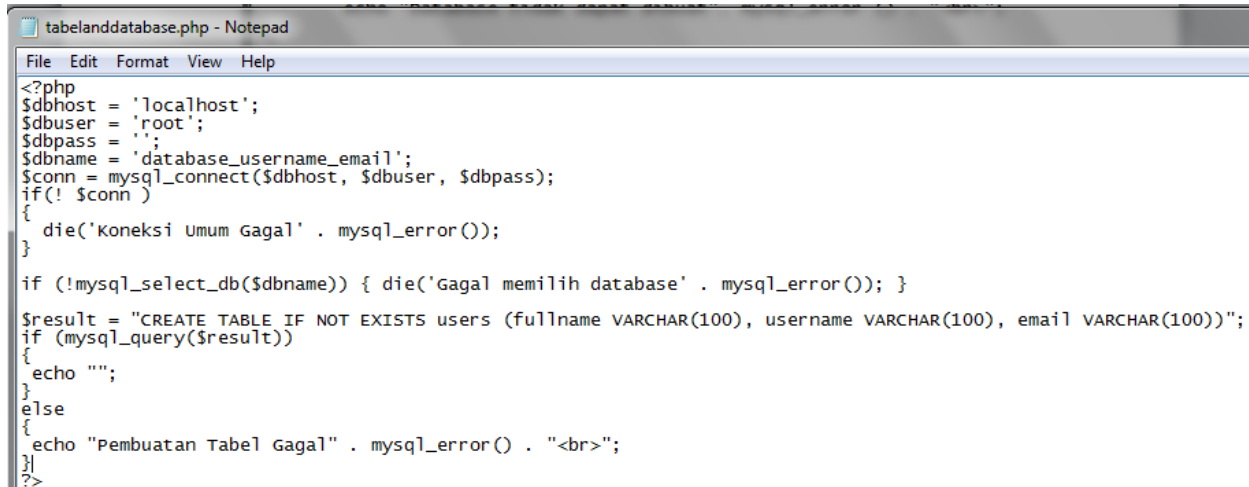
Gambar 7. Isi file mailing-list.php

## Dokumentasi pembuatan : mydatabase.php



Gambar 8. Membuat database terlebih dahulu. Nama databasenya : database\_username\_email

## Dokumentasi pembuatan : tabelanddatabase.php



```
tabelanddatabase.php - Notepad
File Edit Format View Help
<?php
$dbhost = 'localhost';
$dbuser = 'root';
$dbpass = '';
$dbname = 'database_username_email';
$conn = mysql_connect($dbhost, $dbuser, $dbpass);
if(! $conn )
{
    die('koneksi Umum Gagal' . mysql_error());
}

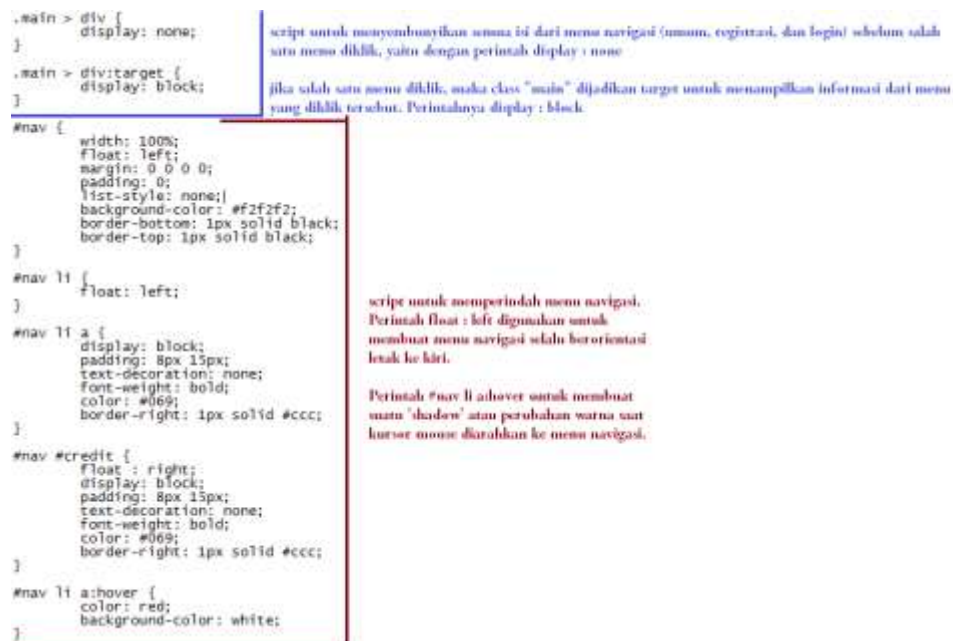
if (!mysql_select_db($dbname)) { die('Gagal memilih database' . mysql_error()); }

$result = "CREATE TABLE IF NOT EXISTS users (fullname VARCHAR(100), username VARCHAR(100), email VARCHAR(100))";
if (mysql_query($result))
{
    echo "";
}
else
{
    echo "Pembuatan Tabel Gagal" . mysql_error() . "<br>";
}
?>
```

Gambar 9. Membuat tabel bernama users dengan elemen fullname, username, dan email dengan masing-masing memiliki jumlah karakter maksimal 100 buah

Karena pembuatan database dan tabel dilakukan di sebuah server root di localhost, maka perlu dilakukan koneksi (`mysql_connect`). Proses koneksi ini dapat berhasil dan dapat pula gagal. Demikian pula dalam pembuatan database maupun tabel (`CREATE`). Maka, jika proses pembentukan database/ tabel gagal, akan muncul pesan kesalahan sesuai script di atas.

## Dokumentasi pembuatan : formstyle.css



```
.main > div {
display: none;
}

.main > div:target {
display: block;
}

#nav {
width: 100%;
float: left;
margin: 0 0 0 0;
padding: 0;
list-style: none;
background-color: #f2f2f2;
border-bottom: 1px solid black;
border-top: 1px solid black;
}

#nav li {
float: left;
}

#nav li a {
display: block;
padding: 8px 15px;
text-decoration: none;
font-weight: bold;
color: #069;
border-right: 1px solid #ccc;
}

#nav #credit {
float: right;
display: block;
padding: 8px 15px;
text-decoration: none;
font-weight: bold;
color: #069;
border-right: 1px solid #ccc;
}

#nav li a:hover {
color: red;
background-color: white;
}
```

script untuk menyembunyikan semua isi dari menu navigasi (umum, registrasi, dan login) sebelum salah satu menu diklik, yaitu dengan perintah `display : none`

jika salah satu menu diklik, maka class "main" dijadikan target untuk menampilkan informasi dari menu yang diklik tersebut. Perintahnya `display : block`

script untuk memperindah menu navigasi. Perintah `float : left` digunakan untuk membuat menu navigasi selalu berorientasi letak ke kiri.

Perintah `#nav li a:hover` untuk membuat suatu 'shadow' atau perubahan warna saat kursor mouse diarahkan ke menu navigasi.

Gambar 10. Script CSS untuk menu navigasi (merah) dan main div (biru)

```

/* untuk jenis pc desktop */
@media screen and (max-width: 1440px) {
  .header {
    background-color : green;
    padding : 5px;
  }
  .between {
    background-color : white;
    width : 100%;
    margin : 0% 0% 5% 0%;
  }
  .main {
    background-color : #CCFFCC;
    margin : 0 auto;
    width : 50%;
    border-radius : 9px;
    box-shadow : 10px 10px 5px #99FFCC;
  }
}

```

pengaturan responsive layout untuk jenis pc desktop dengan lebar maksimum 1440px.

Untuk kasus ini, yang diatur perubahannya hanyalah div header, between (antara menu navigasi dan main), dan main

```

/* untuk jenis tablet landscape */
@media screen and (max-width: 1060px) {
  .header {
    background-color : green;
    padding : 5px;
  }
  .between {
    background-color : white;
    width : 100%;
    margin : 0% 0% 5% 0%;
  }
  .main {
    background-color : #CCFFCC;
    margin : 0 auto;
    width : 100%;
    border-radius : 9px;
    box-shadow : 10px 10px 5px #99FFCC;
  }
}

```

pengaturan responsive layout untuk media jenis tablet dengan orientasi landscape (lebar maksimum 1060px)

Gambar 11. Pengaturan layout untuk media pc desktop (biru) dan tablet landscape (merah)

```

/* untuk jenis tablet portrait */
@media screen and (max-width: 768px) {
  .header {
    background-color : green;
    padding : 5px;
  }
  .between {
    background-color : white;
    width : 100%;
    margin : 0% 0% 5% 0%;
  }
  .main {
    background-color : #CCFFCC;
    margin : 0 auto;
    width : 100%;
    border-radius : 9px;
    box-shadow : 10px 10px 5px #99FFCC;
  }
}

```

pengaturan responsive layout untuk media jenis tablet dengan orientasi portrait (lebar maksimum 768px).

Untuk kasus ini, lebar div main berubah menjadi 100% agar tidak terlalu kecil saat diaplikasikan di layar media yang juga kecil ukurannya

Gambar 12. Pengaturan layout untuk media tablet portrait (hitam)

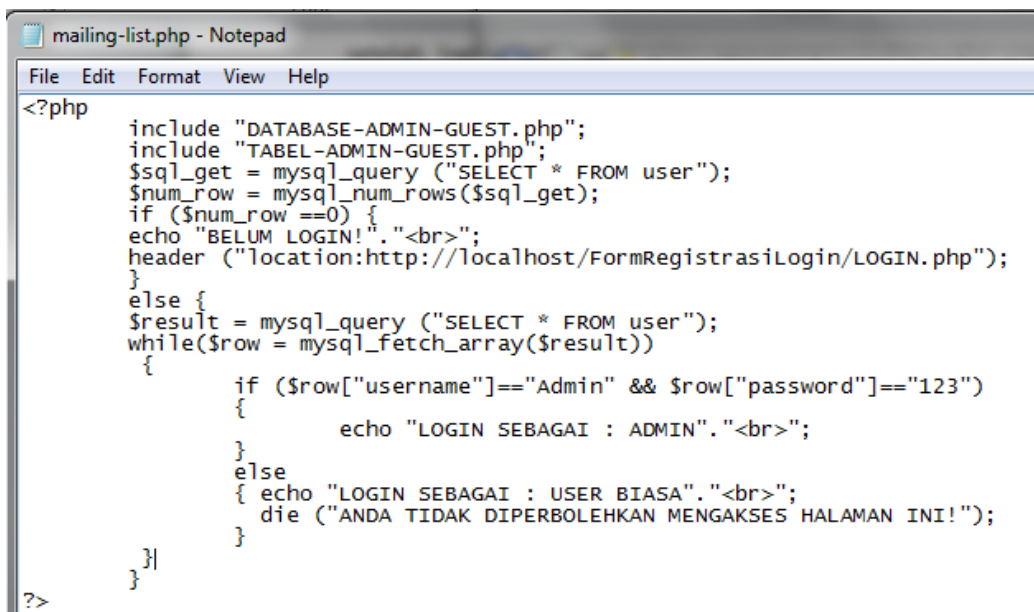
## [TUGAS BONUS]

### Dokumentasi pembuatan : Pembatasan Akses URL bagi Admin dan User biasa

Metode yang saya pakai untuk kasus pembatasan akses ini adalah dengan cara redirecting, penyimpanan di database admin-guest, serta pendeklarasian tipe user yaitu Admin atau user biasa.

Pertama, saya memiliki sebuah database bernama "database\_admin\_user" dan tabel bernama "user". Saya memprogram agar isi / elemen yang ada di tabel "user" maksimal 1 buah, sehingga setelah login maka tabel akan menyimpan 1 data, dan setelah logout maka tabel akan menyimpan 0 data. Jika user belum melewati halaman LOGIN.php, maka jika ia mencoba mengakses situs lewat URL, misalnya "localhost/FormRegistrasiLogin/index.php" (atau alamat apapun dalam folder FormRegistrasiLogin) maka program akan meredirect si user untuk kembali ke halaman LOGIN.php. Ia baru akan bisa masuk ke index.php setelah data username dan password yang dimasukkan di halaman LOGIN.php tersimpan di tabel "user".

Kedua, jika user berhasil login, ia akan mendapat status akses yaitu antara admin atau user biasa. Admin dapat melihat isi mailing-list.php, sedangkan user biasa tidak. Bagaimana cara agar user biasa tidak dapat melihat isi mailing-list.php?



```
<?php
include "DATABASE-ADMIN-GUEST.php";
include "TABEL-ADMIN-GUEST.php";
$sql_get = mysql_query ("SELECT * FROM user");
$num_row = mysql_num_rows($sql_get);
if ($num_row ==0) {
echo "BELUM LOGIN!". "<br>";
header ("location:http://localhost/FormRegistrasiLogin/LOGIN.php");
}
else {
$result = mysql_query ("SELECT * FROM user");
while($row = mysql_fetch_array($result))
{
    if ($row["username"]=="Admin" && $row["password"]=="123")
    {
        echo "LOGIN SEBAGAI : ADMIN". "<br>";
    }
    else
    {
        echo "LOGIN SEBAGAI : USER BIASA". "<br>";
        die ("ANDA TIDAK DIPERBOLEHKAN MENGAKSES HALAMAN INI!");
    }
}
}
```

Gambar 1. Script pembatas akses antara admin dan user biasa untuk melihat mailing-list. Script ini berada di file mailing-list.php bagian paling atas

## HASIL AKSES FORM REGISTRASI EMAIL (SUDAH TERMASUK SOAL BONUS)

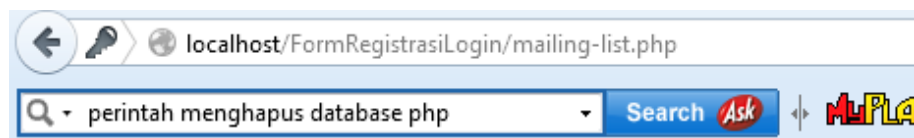
Jika login sebagai Admin dan menuliskan URL : localhost/FormRegistrasiLogin/mailling-list.php

LOGIN SEBAGAI : ADMIN

### USERNAME - EMAIL YANG TELAH DIDAFTRAKAN

Username	Email
albertus_kelvin_16514297	albertus95@gmail.com
albertus_16514297_kelvin	kelvin95@yahoo.com
kelvin_16514297_albertus	albertus95@yahoo.co.id

Jika bukan sebagai Admin...



LOGIN SEBAGAI : USER BIASA

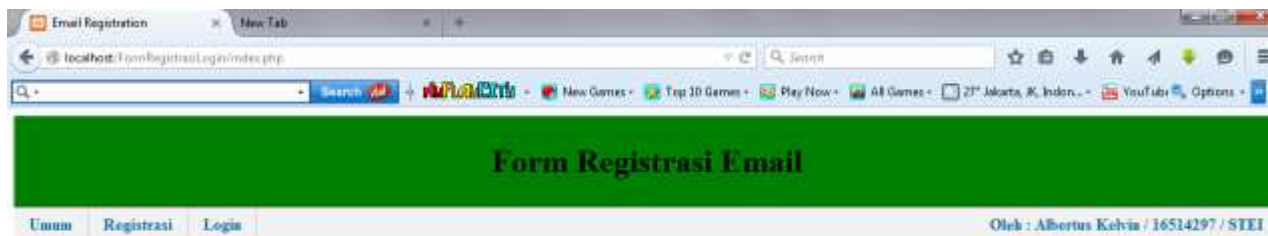
ANDA TIDAK DIPERBOLEHKAN MENGAkses HALAMAN INI!

Untuk gambaran lainnya dapat dicoba di programnya secara langsung.

## HASIL AKSES FORM REGISTRASI EMAIL MELALUI LOCALHOST SECARA UMUM (ini belum termasuk tugas bonus)

Pertama, tentu kita harus mengaktifkan MySQL dan Apache pada XAMPP Control Panel. Berikut screenshot ujicoba akses website dari localhost nya.

Gambar A. Tampilan Awal saat dibuka.





Gambar B. Tampilan saat menu 'Umum' dipilih.

**Form Registrasi Email**

Umum Registrasi Login Oleh : Albertus Kelvin / 16514297 / STEI

**Petunjuk Umum**

- Tombol Registrasi untuk mendaftarkan username dan email
- Tombol Login untuk masuk ke halaman milik user yang terdaftar pada database

**Petunjuk Khusus**

- Semua username yang didaftarkan tidak boleh ada yang sama
- Semua email yang didaftarkan tidak boleh ada yang sama
- Alamat email yang didaftarkan harus valid

\* Jika tidak sesuai petunjuk di atas, maka akan muncul pesan kesalahan

Gambar C. Tampilan saat menu 'Registrasi' dipilih dan diisi oleh user.

**Form Registrasi Email**

Umum Registrasi Login Oleh : Albertus Kelvin / 16514297 / STEI

**FORM REGISTRASI**

Nama Lengkap

Username

Email Address

**SUBMIT**

Gambar D. Tampilan saat tombol 'SUBMIT' diklik.



Gambar E. Tampilan saat ingin melihat username dan email di mailing-list.



Gambar F. Contoh proses input di form Registrasi yang tidak valid (username sudah terdaftar dan alamat email tidak valid)



Gambar G. Tampilan saat tombol 'SUBMIT' pada Gambar F diklik.



Gambar H. Tampilan menu Login dengan username albertus\_kelvin\_16514297 dan alamat email [albertusk95@gmail.com](mailto:albertusk95@gmail.com)



Gambar J. Tampilan saat ingin menambahkan username dan email baru (CONTOH : albertus\_16514297\_kelvin – [kelvin95@yahoo.com](mailto:kelvin95@yahoo.com) ;;;; kelvin\_16514297\_albertus – [albertus95@yahoo.co.id](mailto:albertus95@yahoo.co.id)) → lebar tabelnya saya ganti



## **FILE PHP, HTML, dan CSS yang dibuat :**

File Utama :

- index.php – halaman utama
- mailing-list.php – menampilkan daftar username dan email

File CSS :

- formstyle.css

File Cadangan untuk soal Wajib :

- mydatabase.php dan tabelanddatabase.php – membuat database dan tabel
- desktop-status.php dan login-desktop.php

File Soal Bonus

- LOGIN.php – halaman awal login admin/ user biasa
- LOGOUT-ADMIN-GUEST.php – halaman logout
- DATABASE-ADMIN-GUEST.php dan TABEL-ADMIN-GUEST.php – membuat database dan tabel penyimpan username dan password user
- ADD-ADMIN-GUEST.php – memasukkan username dan password user ke database dan tabel