



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-5: Multi Level User
Mata Kuliah Pemrograman Web Lanjut
Pengampu: Tim Ajar Pemrograman Web Lanjut
Maret 2020

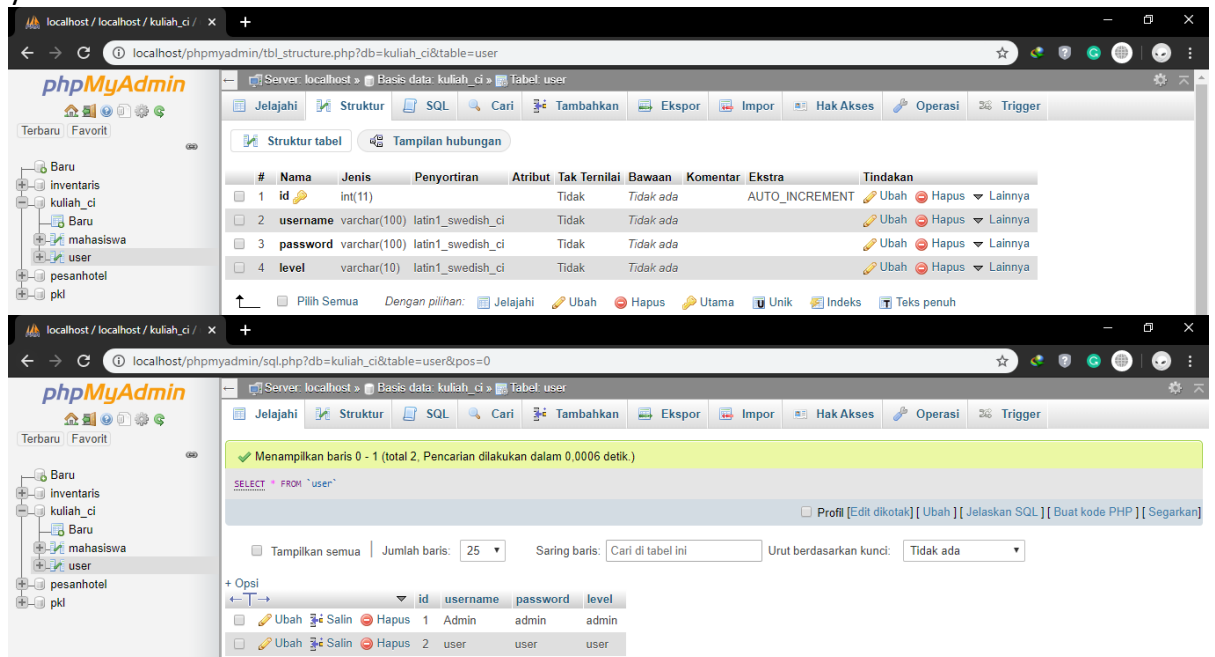
Nama : Septianda Reza Maulana
Kelas : TI-2A
NIM : 1841720135

Praktikum – Bagian 1: Membuat Login dengan Multi Level User

Langkah	Keterangan
1	<p>Buat tampilan login (views/login/index.php), dan tambahkan code berikut pada views/login/index.php:</p> <pre>index.php X application > views > login > index.php > ... 1 <?= 2 form_open('login/proses_login'); 3 ?> 4 <div class="container py-5"> 5 <div class="row"> 6 <div class="col-md-12"> 7 <div class="row"> 8 <div class="col-md-6 mx-auto"> 9 10 <!-- Form Card Login --> 11 <div class="card rounded-0"> 12 <div class="card-header"> 13 <h3 class="mb-0">Login</h3> 14 </div> 15 <div class="card-body"> 16 <form class="form" role="form" autocomplete="off" id="formLogin" novalidate="" method="post"> 17 <div class="form-group"> 18 <label for="uname1">Username</label> 19 <input type="text" class="form-control form-control-lg rounded-0" name="uname1"> 20 <div class="invalid-feedback">Oops, you missed this one.</div> 21 </div> 22 <div class="form-group"> 23 <label>Password</label> 24 <input type="password" class="form-control form-control-lg rounded-0" name="password"> 25 <div class="invalid-feedback">Enter your password too!</div> 26 </div> 27 <button type="submit" class="btn btn-success btn-lg float-right" id="btnLogin">Login</button> 28 </form> 29 </div> 30 </div> 31 </div> 32 </div> index.php X application > views > login > index.php > ... 33 </div> 34 </div> 35 </div> 36 37 <?= 38 form_close(); 39 ?></pre>

2	<p>Buatlah sebuah controller dengan nama login (controller/login)</p> <pre> Login.php X application > controllers > Login.php > ... 1 <?php 2 defined('BASEPATH') OR exit('No direct script access allowed'); 3 4 class Login extends CI_Controller { 5 6 public function index(\$name='') 7 { 8 \$data['title']='Login'; 9 \$this->load->view('template/header_login',\$data); 10 \$this->load->view('Login/index', \$data); 11 \$this->load->view('template/footer'); 12 } 13 } 14 /* End of file Controllername.php */ 15 > </pre>
3	<p>Tambahkan sebuah code pada views/template dengan nama header_login (views/template/header_login)</p> <pre> header_login.php X application > views > template > header_login.php > html > body > nav.navbar.navbar-expand-lg.navbar-light.bg-light 1 <!doctype html> 2 <html lang="en"> 3 <head> 4 <!-- Required meta tags --> 5 <meta charset="utf-8"> 6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no"> 7 8 <!-- Bootstrap CSS --> 9 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity 10 11 <title><?= \$title ?></title> 12 13 <!-- style css --> 14 <style> 15 .badge{ 16 margin-left: 3px; 17 } 18 </style> 19 20 <!-- <title>Hello, World!</title --> 21 <title><?= \$title ?></title> 22 23 </head> 24 <body> 25 </pre>
4	<p>Ketika dijalankan di browser, hasilnya akan seperti gambar berikut:</p> 

- 5 Setelah berhasil membuat sebuah form login maka selanjutnya kita akan membuat logic nya. Login ini digunakan untuk proses validasi data antara inputan user dengan database yang sudah ada.
- 6 Sebelum membuat *logic*-nya terlebih dahulu buat tabelnya pada **database Kuliah_CI** (yang pernah dibuat sebelumnya) dengan nama **tabel user** dan tambahkan 2 data, sebagai contoh yaitu **admin** dan **user**



The first screenshot shows the phpMyAdmin interface for the 'user' table. The table structure is as follows:

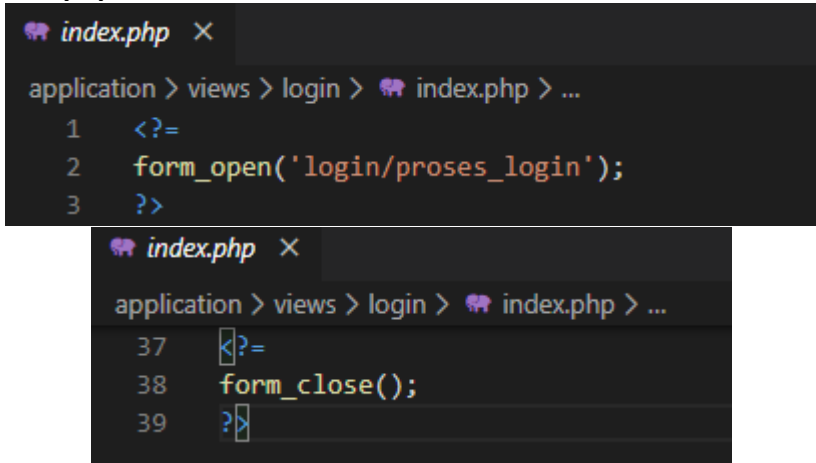
#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	username	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	password	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	level	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

The second screenshot shows the 'user' table with two records:

id	username	password	level
1	Admin	admin	admin
2	user	user	user

- 7 Setelah kita berhasil membuat sebuah tabel untuk menyimpan *username*, *password* dan *level* maka selanjutnya kita akan membuat model dengan nama **login_model** (**models/login_model**)

```
login_model.php X
application > models > login_model.php > Login_model
1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Login_model extends CI_Model {
5     public function Login($username, $password)
6     {
7         // var_dump($username);
8         // var_dump($password);
9         // die();
10        $this->db->select('username,password,level');
11        $this->db->from('user');
12        $this->db->where('username',$username); // maksudnya adalah username
13        $this->db->where('password',$password); // maksudnya adalah password
14        $this->db->Limit(1); // maksudnya data yang diambil cuma 1
15
16        $query = $this->db->get();
17        if ($query->num_rows()==1) // jika data ditemukan
18        {
19            return $query->result();
20        }else {
21            return false;
22        }
23    }
24 }
```

	<p>Keterangan :</p> <ol style="list-style-type: none"> 1. <i>Line 5</i> sebuah function <i>login</i> dimana terdapat 2 parameter yaitu <i>\$username</i> dan <i>\$password</i> 2. <i>Line 10</i> untuk melakukan <i>select</i> mana saja yang akan digunakan. Jika kita lihat terdapat 3 parameter yaitu <i>username</i>, <i>password</i>, dan <i>level</i> -> untuk penulisan ini semua harus sesuai dengan nama <i>field</i> yang ada pada <i>tabel</i>-nya (tabel user). 3. <i>Line 11</i> adalah nama <i>tabel</i>-nya (tabel user) 4. <i>Line 12</i> adalah selama inputan <i>user</i> yang tersimpan pada <i>\$username</i> sama dengan <i>username</i> pada <i>tabel user</i> 5. <i>Line 13</i> adalah selama inputan <i>user</i> yang tersimpan pada <i>\$password</i> sama dengan <i>password</i> pada <i>tabel user</i> 6. <i>Line 14</i> adalah data yang diambil cuma 1 7. <i>Line 16</i> adalah sebuah variabel baru dengan nama <i>\$query</i> yang digunakan untuk mengambil semua data 8. <i>Line 17 – 21</i> adalah sebuah pengkondisian dimana jika <i>\$query</i> menemukan 1 <i>row</i> pada <i>tabel</i> atau dengan kata lain data berhasil ditemukan maka nilai pengembalian querynya akan mencetak isi dari <i>\$query(\$query->result())</i> dan jika tidak menemukan maka nilai kembaliannya kita set menjadi <i>false</i>
8	<p>Langkah berikutnya tambahkan form_open dan form_close pada <i>code views/login/index.php</i></p>  <p>Keterangan:</p> <ol style="list-style-type: none"> 1. <i>Line 2</i> artinya adalah <i>form</i> yang digunakan untuk membuka halaman proses <i>login</i> pada <i>controller login</i> 2. <i>Line 38</i> artinya adalah untuk menutup <i>form</i> dikarenakan tadi terdapat perintah <i>form_open</i>

9

Buatlah sebuah *function* dengan nama **proses_login** pada **controller login**. Tapi sebelum kita membuat *function proses_login* maka terlebih dahulu kita tambahkan sebuah *function construct*

```

Login.php X
application > controllers > Login.php > Login > proses_login
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Login extends CI_Controller {
5
6      public function __construct() {
7          parent::__construct();
8          $this->load->helper('url');
9          $this->load->helper('form');
10         $this->load->model('login_model');
11         // $this->load->library('session');
12     }
13     public function index($name='')
14     {
15         $data['title']='Login';
16         $this->load->view('template/header_login',$data);
17         $this->load->view('Login/index', $data);
18         $this->load->view('template/footer');
19     }

```

Keterangan : Jika pada config/autoload sudah kalian setting maka kita tidak perlu menambahkan lagi seperti pada line 11. Dikarenakan pada config/autoload/library sudah diset maka pada *function construct* tidak perlu ditambahkan lagi.

10

Langkah berikutnya menambahkan **function proses_login** pada **controller login**

```

Login.php X
application > controllers > Login.php > ...
20  public function proses_login() {
21      $username = $this->input->post('uname1');
22      $password = $this->input->post('pwd1');
23
24      $ceklogin = $this->login_model->login($username, $password);
25
26      if($ceklogin) {
27          foreach($ceklogin as $row);
28          // kita set userdata pada session dengan nama user dan isi
29          $this->session->set_userdata('user',$row->username);
30          $this->session->set_userdata('level',$row->level);
31
32          // var_dump($row);
33          if($this->session->userdata('level')=="admin")
34          {
35              redirect('mahasiswa/index');
36          }
37          elseif($this->session->userdata('level')=="user")
38          {
39              redirect('mahasiswa/user');
40          }
41      }
42      else
43      {
44          redirect('login/index', 'refresh');
45      }
46  }
47  }
48
49  /* End of file Controllername.php */
50  >

```

	<p>Keterangan :</p> <ol style="list-style-type: none"> 1. <i>Line 21 dan 22</i> adalah sebuah variabel baru dimana isinya adalah inputan dari <i>user</i>. <i>User</i> akan mengirimkan data yang terkirim dengan metode POST yang tersimpan sementara pada uname1 dan pwd1 (uname1 dan pwd1 adalah name pada inputan <i>username</i> dan <i>password</i> pada page login/index pada <i>tag input</i>). 2. <i>Line 24</i> adalah variabel baru(\$cek_login) isinya adalah sebuah model(login_model) dimana model(login_model) tersebut akan menjalankan sebuah <i>function</i>(function login) dengan nama login(yang telah kita buat sebelumnya) dan mempunyai 2 paramater yaitu \$username dan \$password dimana isinya adalah isi dari <i>line 21</i> dan <i>22</i>, yang nanti akan dikirimkan ke <i>function</i> login pada <i>login_model</i>. 3. <i>Line 26-33</i> adalah sebuah pengkondisian jika \$cek_login bernilai benar maka cek_login akan dipecah menjadi beberapa <i>row</i> (\$cek_login as \$row). setelah itu kita <i>set_userdata</i> menggunakan <i>session</i> dengan nama variabel barunya adalah <i>user</i> sedangkan isinya adalah <i>username</i> (<i>line 33</i>) dan kita juga <i>set_userdata</i> menggunakan <i>session</i> dengan nama variabelnya <i>level</i> (<i>line 37</i>) sedangkan isinya adalah <i>level</i>. Setelah kita pecah menjadi beberapa <i>row</i> dan mempunyai isi yang berbeda-beda maka langkah berikutnya kita buat pengkondisian jika <i>session</i> <i>userdata('level')</i> berisi <i>admin</i> maka akan me-<i>redirect</i> pada halaman <i>mahasiswa/index</i>(<i>line 33-35</i>). Apabila <i>levelnya</i> berupa <i>user</i> maka akan meredirect ke halaman <i>user</i> (<i>line 37-39</i>). Jika \$cek_login bernilai salah maka akan me-<i>redirect</i> ke halaman <i>login</i> kembali (<i>line 42 - 44</i>).
11	<p>Pada <i>line 25</i> dan <i>26</i> tambahkan htmlspecialchars agar <i>user</i> tidak dapat memasukkan <i>char</i> yang special.</p> <pre> Login.php X application > controllers > Login.php > Login > proses_login 20 public function proses_login() { 21 \$username = htmlspecialchars(\$this->input->post('uname1')); 22 \$password = htmlspecialchars(\$this->input->post('pwd1')); </pre>
12	<p>Agar halaman <i>login</i> yang pertama kali di-<i>load</i> maka kita harus ubah konfigurasinya pada application/config/routes</p> <pre> routes.php X application > config > routes.php > ... 52 \$route['default_controller'] = 'Login'; 53 \$route['404_override'] = ''; 54 \$route['translate_uri_dashes'] = FALSE; </pre>

13

Hasilnya jika kita masuk menggunakan admin

The screenshot shows two browser windows. The top window is titled 'Login' and shows the login page with the URL 'localhost/Belajar_CI2/'. It has a navigation bar with 'CI', 'Home', 'Data Mahasiswa', and 'About'. The login form has fields for 'Username' (containing 'admin') and 'Password' (containing '*****'), and a green 'Login' button.

The bottom window is titled 'List Mahasiswa' and shows the list of students page with the URL 'localhost/Belajar_CI2/mahasiswa'. It has the same navigation bar. There is a blue 'Tambah Data' button, a search bar with 'Cari Data Mahasiswa' and a blue 'Cari' button. Below is a table titled 'Daftar Mahasiswa' with 6 rows of student data, each with 'Detail', 'Edit', and 'Hapus' buttons.

Username	Action
Reza Maulana	Detail Edit Hapus
Baul	Detail Edit Hapus
Roni	Detail Edit Hapus
Septianda Reza Maulana	Detail Edit Hapus
Agsal Mirza	Detail Edit Hapus
Alan Perdana Timor	Detail Edit Hapus

14

Hasilnya jika kita masuk menggunakan *user* (jika kita lihat maka hasilnya *page not found* hal ini dikarenakan kita belum membuat halaman tersebut)

The screenshot shows two browser windows. The top window is titled 'Login' and shows the login page with the URL 'localhost/Belajar_CI2/'. It has a navigation bar with 'CI', 'Home', 'Data Mahasiswa', and 'About'. The login form has fields for 'Username' (containing 'user') and 'Password' (containing '*****'), and a green 'Login' button.

The bottom window is titled '404 Page Not Found' and shows the error page with the URL 'localhost/Belajar_CI2/mahasiswa/user'. It has the same navigation bar. The error message is '404 Page Not Found' and 'The page you requested was not found.'

15

Hasilnya jika kita salah memasukkan *password* dan *username* (jika salah maka **redirect** ke halaman login tanpa ada pemberitahuan jika *username* dan *password* salah)

16

Kita tambahkan **alert** agar pada saat *user* salah memasukkan *username* dan *password*, maka akan muncul pemberitahuan bahwa *username* dan *password* anda salah. Tambahkan *code* berikut pada **login/index**

```

index.php X
application > views > login > index.php > div.container.py-5 > div.row > div.col-md-12 > div.row > div.col-md-6.mx-auto > div.card.rou
10      <!-- Form Card Login -->
11      <!-- Cek Pesan -->
12      <div class="alert alert-info" role="alert">
13          <?php
14              if(isset($pesan)) {
15                  echo $pesan;
16              }else {
17                  echo "Masukkan username dan password anda";
18              }
19          ?>
20      </div>

```

dan pada bagian **controller/login/proses_login** kita ubah pada *else*-nya

```

Login.php X
application > controllers > Login.php > ...
42      else
43      {
44          $data['pesan'] = "Username dan Password anda salah!!";
45          $data['title'] = 'Login';
46          $this->load->view('template/header_login', $data);
47          $this->load->view('Login/index', $data);
48          $this->load->view('template/footer');
49          //redirect('login/index', 'refresh');
50      }
51  }
52  }

```


Sehingga hasilnya akan tampak seperti berikut

The first screenshot shows a web browser at the URL `localhost/Belajar_CI2/login/index`. The page has a navigation bar with links: `CI`, `Home`, `Data Mahasiswa`, and `About`. The main content area contains a `Login` form with the following fields and elements:

- `Username` input field containing the text `septiandareza`.
- `Password` input field (empty).
- A green `Login` button.
- A light blue message box at the bottom containing the text `Masukkan username dan password anda`.

The second screenshot shows the same web browser at the URL `localhost/Belajar_CI2/login/proses_login`. The page has the same navigation bar. The `Login` form is identical to the first screenshot, but the light blue message box at the bottom now contains the text `Username dan Password anda salah!!`.

- 17 Jika kita lihat pada percobaan sebelumnya, pada saat kita *login* dengan menggunakan level *user* maka halaman tidak akan tampil atau *page not found*. Untuk kita akan membuat halaman *user* sederhana

- 18 Jika dilihat pada **function proses_login** terdapat **code** sebagai berikut:

```
Login.php X
application > controllers > Login.php > ...
37 elseif($this->session->userdata('level')=="user")
38 {
39     redirect('mahasiswa/user');
40 }
41 }
```

Untuk lebih mudahnya pada line 44 ganti `redirect('mahasiswa/index')` menjadi `redirect('user');`

```
Login.php X
application > controllers > Login.php > ...
37 elseif($this->session->userdata('level')=="user")
38 {
39     redirect('user');
40 }
41 }
```

19

Setelah mengganti **redirect**, maka yang harus dilakukan adalah membuat sebuah **view** dengan nama **user** pada **folder mahasiswa**. sebelum kita membuat **view** dengan nama **user** kita terlebih dahulu membuat sebuah **controller** dengan nama **user**. Buat **controller** user dengan **code** sebagai berikut:

```

user.php X
application > controllers > user.php > ...
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class user extends CI_Controller {
5
6      public function __construct()
7      {
8          parent::__construct();
9          $this->load->helper('url');
10         $this->load->helper('form');
11         $this->load->model('login_model');
12         // $this->load->library('session');
13     }
14     public function index() {
15         $this->load->view('mahasiswa/user');
16     }
17 }
18
19 /* End of file Controllername.php */
20 ?>

```

20

Membuat tampilan pada **view** dengan nama **user**

```

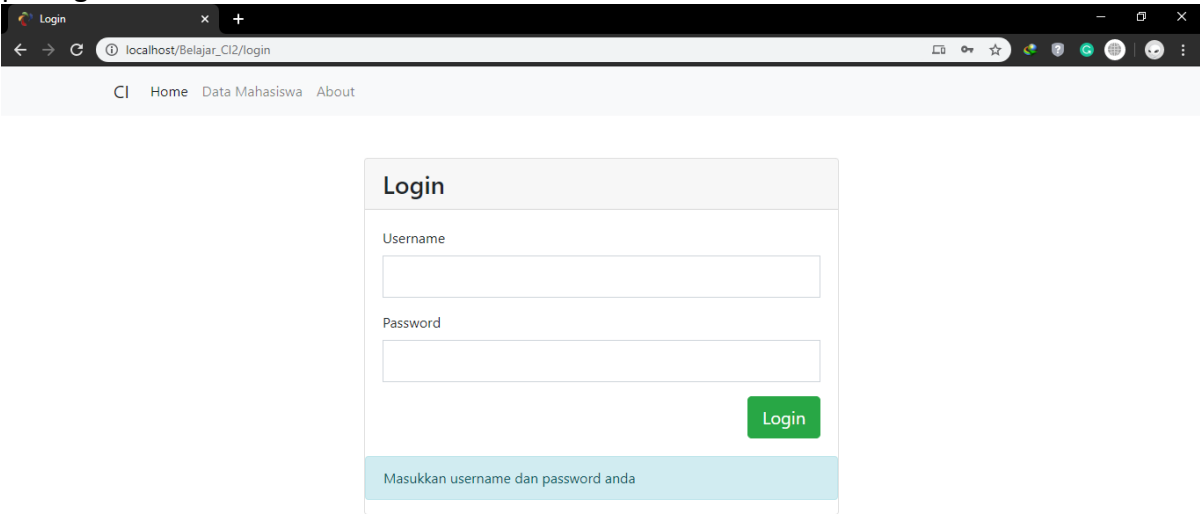
user.php X
application > views > mahasiswa > user.php > h1
1  <h1> Ini halaman user</h1>

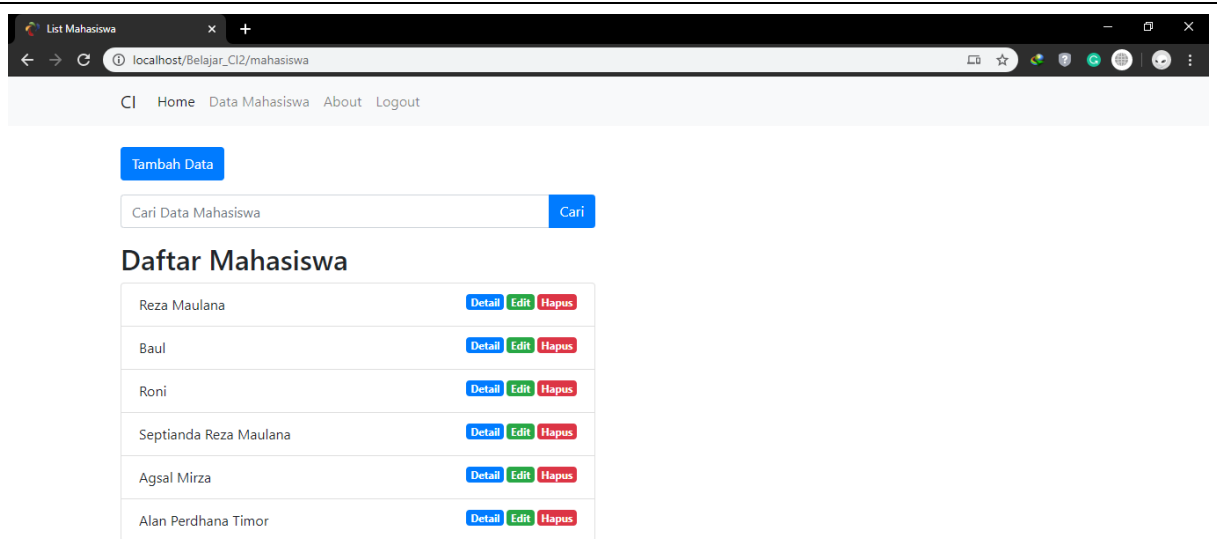
```

Sehingga hasilnya jika kita **login** pada level **user** maka tampilan akan seperti gambar berikut

The image shows two browser screenshots. The top screenshot is of a login page at `localhost/Belajar_CI2/login`. It features a 'Login' form with 'Username' and 'Password' fields. The username field contains 'user' and the password field contains four dots. A green 'Login' button is at the bottom right. Below the form is a light blue message box that says 'Masukkan username dan password anda'. The bottom screenshot shows the user page at `localhost/Belajar_CI2/user`, displaying the text 'Ini halaman user'.

Ini halaman user

21	<p>Setelah berhasil membuat halaman <i>user</i>, maka langkah berikutnya kita membuat menu <i>logout</i>. Tambahkan <i>code</i> berikut pada views/template/header line 36</p> <pre> header.php X application > views > template > header.php > html > body > nav.navbar.navbar-expand-lg.navbar-light.bg-light 34 <div class="navbar-nav"> 35 <a class="nav-item nav-link active" href="<?= base_url(); ?>">Home (current) 36 <a class="nav-item nav-link" href="<?= base_url(); ?>">mahasiswaData Mahasiswa 37 <a class="nav-item nav-link" href="<?= base_url(); ?>">About 38 <a class="nav-item nav-link" href="<?= base_url(); ?>">login/logoutLogout </pre> <p>line 38 diubah menjadi seperti pada gambar</p>
22	<p>Tambahkan function <i>logout</i> pada controller <i>login</i> <i>code</i>-nya sebagai berikut</p> <pre> Login.php X application > controllers > Login.php > ... 52 public function logout() { 53 \$this->session->sess_destroy(); 54 redirect('login', 'refresh'); 55 } 56 } </pre>
23	<p>Hasilnya jika kita klik <i>logout</i> maka akan muncul halaman <i>login</i>. Pada halaman <i>user</i> kita juga tambahkan <i>code</i> untuk proses <i>logout</i> sederhana</p> <pre> user.php X application > views > mahasiswa > user.php > ... 1 <h1> Ini halaman user</h1> 2 <a href=" <?= base_url(). 'login/logout'; ?>">Logout </pre>
24	<p>Jika kita lakukan perubahan langsung pada url http://localhost/Belajar_CI2/login menjadi http://localhost/Belajar_CI2/mahasiswa seperti pada gambar dibawah ini:</p> 



maka kita akan dapat masuk secara langsung tanpa perlu *login* terlebih dahulu. Untuk itu kita harus menambahkan beberapa *code* pada *controller/mahasiswa* seperti pada gambar dibawah ini (*line 13-15*)

```

mahasiswa.php x
application > controllers > mahasiswa.php > Mahasiswa > __construct
6   public function __construct(){
7       parent::__construct();
8       $this->load->model('mahasiswa_model');
9       $this->load->library("form_validation");
10      //$this->load->database();
11
12      // validasi level
13      if($this->session->userdata('level')!="admin") {
14          |   redirect('login', 'refresh');
15      }

```

Keterangan :

Line 13-15 adalah pengkondisian dimana jika levelnya bukan admin maka halaman akan me-*redirect* ke halaman *login*

25	<p>Lakukan hal yang sama pada user (<i>controller user</i>)</p> <pre> user.php × application > controllers > user.php > ... 1 <?php 2 defined('BASEPATH') OR exit('No direct script access allowed'); 3 4 class user extends CI_Controller { 5 6 public function __construct() 7 { 8 parent::__construct(); 9 \$this->load->helper('url'); 10 \$this->load->helper('form'); 11 \$this->load->model('login_model'); 12 // \$this->load->library('session'); 13 14 // validasi level 15 if(\$this->session->userdata('level')!="user") { 16 redirect('login', 'refresh'); 17 } 18 } </pre>
26	<p>Lakukan hal yang sama pada home (<i>controller home</i>)</p> <pre> Home.php × application > controllers > Home.php > ... 18 // validasi level 19 if(\$this->session->userdata('level')!="admin") { 20 redirect('login', 'refresh'); 21 } </pre>
27	<p>Tambahkan <i>code</i> berikut agar pada halaman admin tampil level <i>user</i>nya (views/home/index)</p> <pre> index.php × application > views > home > index.php > h1 1 <h1>Hello Septianda Reza Maulana, <?= \$this->session->userdata('level'); ?>!</h1> </pre>

28

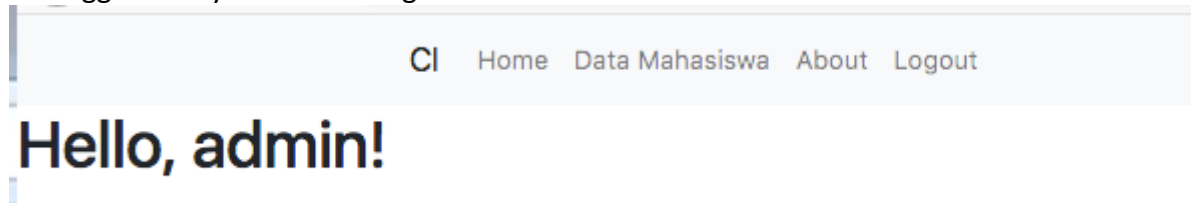
Comment beberapa *code* pada *controller home* (line 11 dan hapus parameter *name* pada *index*)

```

Home.php X
application > controllers > Home.php > ...
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Home extends CI_Controller {
5
6      //tambahkan parameter name pada index
7      public function index()
8      {
9          $data['title']='Home';
10         // $data adalah sebuah array dengan isi arraynya adalah name dan diisi $name
11         // $data['name']=$name;
12         $this->load->view('template/header',$data);
13         //tambahkan $data pada home/index
14         $this->load->view('home/index', $data);
15         $this->load->view('template/footer');
16         $this->load->library('session');
17
18         // validasi level
19         if($this->session->userdata('level')!="admin") {
20             redirect('login', 'refresh');
21         }
22     }
23
24 }
25
26 /* End of file Controllername.php */
27 <?

```

Sehingga hasilnya adalah sebagai berikut:



29

Lakukan hal yang sama pada *user*

```

user.php X
application > views > mahasiswa > user.php > ...
1  <h1> Ini halaman user</h1>
2
3  <h1>Hello Septianda Reza Maulana, <?php echo $this->session->userdata('level'); ?></h1>
4  <a href=" <?= base_url().'login/logout';?>">Logout</a>

```

Ini halaman user

Hello Septianda Reza Maulana, user!

[Logout](#)