



LAPORAN PRAKTIKUM

Web Lanjut

NAMA : Dina Lisuardi
NIM : 1941723004
Kelas : TI 2B
Isi Laporan : Jobsheet-5 : Multi Level User

Bagian 1

Langka h	Keterangan
1	<p>Buat tampilan login (views/login/index.php), dan tambahkan code berikut pada views/login/index.php:</p> <pre>index.php x login.php x header_login.php x <?php form_open('login/proses_login'); ?> <div class="container py-5"> <div class="row"> <div class="col-md-12"> <div class="row"> <div class="col-md-6 mx-auto"> <!-- form card login --> <div class="card rounded-0"> <div class="card-header"> <h3 class="mb-0">Login</h3> </div> <div class="card-body"> <form class="form" role="form" autocomplete="off" id="formLogin" nonvalidate="" method="POST"> <div class="form-group"> <label for="uname1">Username</label> <input type="text" class="form-control form-control-lg rounded-0" name="uname1" id="uname1" required=""> <div class="invalid-feedback">Oops, you missed this one.</div> </div> <div class="form-group"> <label>Password</label> <input type="password" class="form-control form-control-lg rounded-0" name="pwd1" id="pwd1" required="" autocomplete="new-password"> <div class="invalid-feedback">Enter your password too!</div> </div> <button type="submit" class="btn btn-success btn-lg float-right" id="btnLogin">Login</button> </form> </div> </div> <!-- /card-block --> </div> </div> </div> </div> </div> <?php form_close(); ?></pre>
2	<p>Buatlah sebuah controller dengan nama login (controller/login)</p>

	 <pre> 1 <?php 2 defined('BASEPATH') OR exit('No direct script access allowed'); 3 4 class login extends CI_Controller 5 { 6 7 public function index() 8 { 9 \$data['title']='Login'; 10 \$this->load->helper('form'); 11 \$this->load->view('template/header_login',\$data); 12 \$this->load->view('login/index'); 13 \$this->load->view('template/footer'); 14 } 15 } 16 17 /* End of file Controllername.php */ 18 ?> </pre>
3	<p>Tambahkan sebuah <i>code</i> pada views/template dengan nama header_login (views/template/header_login)</p>  <pre> 1 <!doctype html> 2 <html lang="en"> 3 4 <head> 5 <!-- Required meta tags --> 6 <meta charset="utf-8"> 7 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no"> 8 9 <!-- Bootstrap CSS --> 10 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFng68fJT3GxwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdKnLPMO" crossorigin="anonymous"> 11 12 <!-- style css --> 13 <style> 14 .badge { 15 margin-left: 3px; 16 } 17 </style> 18 <!-- <title>Hello, world!</title> --> 19 <title><?= \$title ?></title> 20 </head> 21 <body></body> 22 23 24 <!--<li class="nav-item"> 25 Disabled 26 </li--> 27 28 </div> 29 </div> 30 </nav> </pre>
4	Ketika dijalankan di browser, hasilnya akan seperti gambar berikut:

	<div><div>Login</div><div><div>Username</div><div></div></div><div><div>Password</div><div></div></div><div>Login</div></div>																																																																		
5	Setelah berhasil membuat sebuah form login maka selanjutnya kita akan membuat logic nya. Logic ini digunakan untuk proses validasi data antara inputan user dengan database yang sudah ada.																																																																		
6	<p>Sebelum membuat <i>logic</i>-nya terlebih dahulu buat tabelnya pada database Kuliah_CI (yang pernah dibuat sebelumnya) dengan nama tabel user dan tambahkan 2 data, sebagai contoh yaitudan user</p> <div><div>Server: 127.0.0.1 » Basis data: kuliah_ci » Tabel: user</div><div><div>Jelajahi</div><div>Struktur</div><div>SQL</div><div>Cari</div><div>Tambahkan</div><div>Ekspor</div><div>Impor</div><div>Lainnya</div></div><div><div>Struktur tabel</div><div>Relation view</div></div><table><thead><tr><th>#</th><th>Nama</th><th>Jenis</th><th>Penyortiran</th><th>Atribut</th><th>Kosong</th><th>Bawaan</th><th>Komentar</th><th>Ekstra</th><th>Tindakan</th></tr></thead><tbody><tr><td><input type="checkbox"/></td><td>1 id</td><td>int(11)</td><td></td><td></td><td>Tidak</td><td>Tidak ada</td><td></td><td>AUTO_INCREMENT</td><td> Ubah Hapus</td></tr><tr><td><input type="checkbox"/></td><td>2 username</td><td>varchar(100)</td><td>latin1_swedish_ci</td><td></td><td>Tidak</td><td>Tidak ada</td><td></td><td></td><td> Ubah Hapus</td></tr><tr><td><input type="checkbox"/></td><td>3 password</td><td>varchar(100)</td><td>latin1_swedish_ci</td><td></td><td>Tidak</td><td>Tidak ada</td><td></td><td></td><td> Ubah Hapus</td></tr><tr><td><input type="checkbox"/></td><td>4 level</td><td>varchar(10)</td><td>latin1_swedish_ci</td><td></td><td>Tidak</td><td>Tidak ada</td><td></td><td></td><td> Ubah Hapus</td></tr></tbody></table><div><div>+ Opsi</div><div><div>← T →</div><div></div><div>id</div><div>username</div><div>password</div><div>level</div></div><table><tbody><tr><td><input type="checkbox"/></td><td> Ubah</td><td> Salin</td><td> Hapus</td><td>1</td><td>admin</td><td>admin</td><td>admin</td></tr><tr><td><input type="checkbox"/></td><td> Ubah</td><td> Salin</td><td> Hapus</td><td>2</td><td>user</td><td>user</td><td>user</td></tr></tbody></table></div></div>	#	Nama	Jenis	Penyortiran	Atribut	Kosong	Bawaan	Komentar	Ekstra	Tindakan	<input type="checkbox"/>	1 id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus	<input type="checkbox"/>	2 username	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus	<input type="checkbox"/>	3 password	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus	<input type="checkbox"/>	4 level	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus	<input type="checkbox"/>	Ubah	Salin	Hapus	1	admin	admin	admin	<input type="checkbox"/>	Ubah	Salin	Hapus	2	user	user	user
#	Nama	Jenis	Penyortiran	Atribut	Kosong	Bawaan	Komentar	Ekstra	Tindakan																																																										
<input type="checkbox"/>	1 id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus																																																										
<input type="checkbox"/>	2 username	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus																																																										
<input type="checkbox"/>	3 password	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus																																																										
<input type="checkbox"/>	4 level	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus																																																										
<input type="checkbox"/>	Ubah	Salin	Hapus	1	admin	admin	admin																																																												
<input type="checkbox"/>	Ubah	Salin	Hapus	2	user	user	user																																																												
7	Setelah kita berhasil membuat sebuah tabel untuk menyimpan <i>username</i> , <i>password</i> dan <i>level</i> maka selanjutnya kita akan membuat model dengan nama login_model (models/login_model)																																																																		

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class login_model extends CI_Model {
5
6     public function login($username,$password)
7     {
8         // var_dump($username);
9         // var_dump($password);
10        // die();
11        $this->db->select('username,password,level');
12        $this->db->from('user');
13        $this->db->where('username',$username); //maksudnya adalah
        selama inputan user yang disimpan pada parameter $user
        sama dengan username
14        $this->db->where('password',$password);
15        $this->db->limit(1); //maksudnya data yang diambil cuma 1
16
17        $query=$this->db->get();
18        if ($query->num_rows()==1) //jika data ditemukan
19        {
20            return $query->result();
21        }else{
22            return false;
23        }
24    }
25
26
27
28
29 /*End of file ModelName.php*/
30 ?>

```

Keterangan :

1. Line 6 sebuah function *login* dimana terdapat 2 parameter yaitu \$username dan \$password
2. Line 11 untuk melakukan *select* mana saja yang akan digunakan. Jika kita lihat terdapat 3 parameter yaitu *username*, *password*, dan *level* -> untuk penulisan ini semua harus sesuai dengan nama *field* yang ada pada *tabel*-nya (**tabel user**).
3. Line 12 adalah nama *tabel*-nya (**tabel user**)
4. Line 13 adalah selama inputan *user* yang tersimpan pada \$username sama dengan *username* pada *tabel user*
5. Line 14 adalah selama inputan *user* yang tersimpan pada \$password sama dengan *password* pada *tabel user*
6. Line 15 adalah data yang diambil cuma 1
7. Line 16 adalah sebuah variabel baru dengan nama \$query yang digunakan untuk mengambil semua data

Line 18 – 24 adalah sebuah pengkondisian dimana jika \$query menemukan 1 row pada *tabel* atau dengan kata lain data berhasil ditemukan maka nilai pengembalian querynya akan mencetak isi dari \$query(\$query->result()) dan jika tidak menemukan maka nilai kembaliannya kita set menjadi *false*

8 Langkah berikutnya tambahkan **form_open** dan **form_close** pada *code views/login/index.php*

```

<?=
form_open('login/proses_login');
?>
<div class="container py-5">
....<div class="row">
.....<div class="col-md-12">
.....<div class="row">
.....<div class="col-md-6 mx-auto">

```

```

<?=
form_close();
?>

```

Keterangan:

1. Line 2 artinya adalah *form* yang digunakan untuk membuka halaman proses *login* pada *controller login*

Line 44 artinya adalah untuk menutup *form* dikarenakan tadi terdapat perintah *form_open*

- 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**

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class login extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load->helper('url');
        $this->load->helper('form');
        $this->load->helper('login_model');
        //$this->load->library('session');
    }

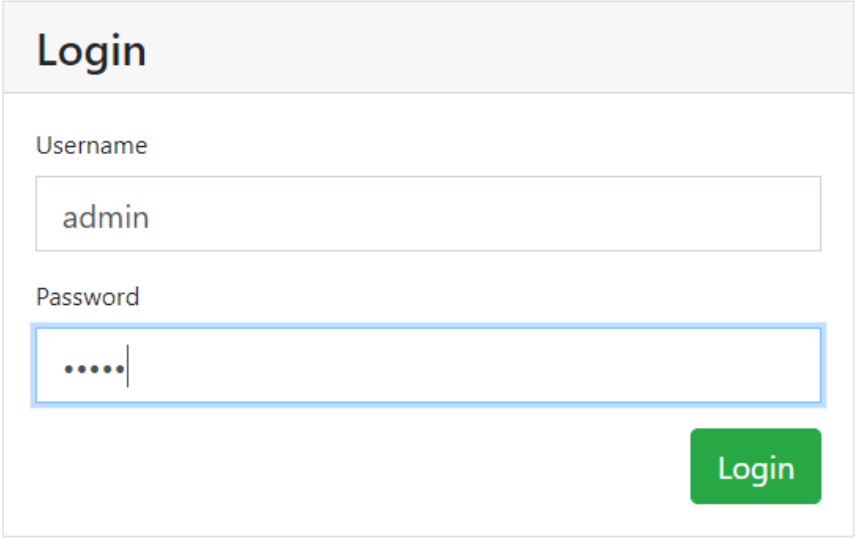
    public function index()
    {
        $data['title']='Login';
        $this->load->helper('form');
        $this->load->view('template/header_login',$data);
        $this->load->view('login/index');
        $this->load->view('template/footer');
    }
}

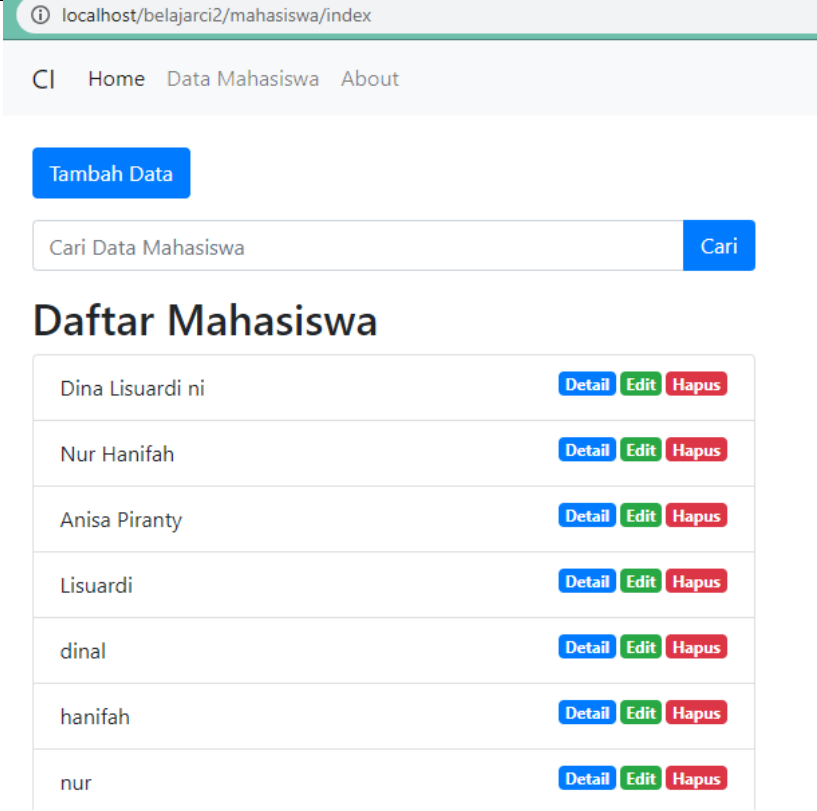
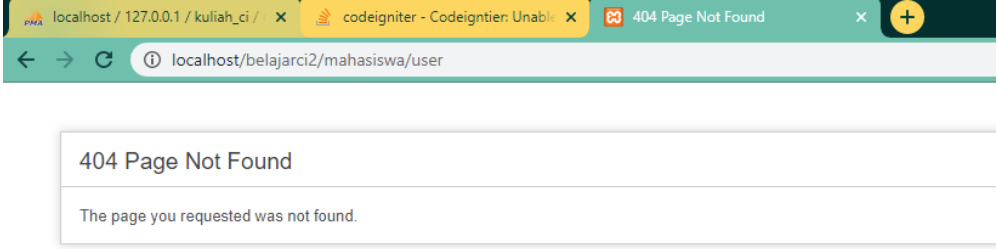
/* End of file Controllername.php */
?>

```

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	<p>Langkah berikutnya menambahkan function proses_login pada controller login</p> <pre> public function proses_login() { \$username=\$this->input->post('uname1'); \$password=\$this->input->post('pwd1'); \$ceklogin=\$this->login_model->login(\$username,\$password); if (\$ceklogin) { foreach (\$ceklogin as \$row); //kita set userdata pada session dengan nama user dan isi username kita isikan username yang ada pada \$row \$this->session->set_userdata('user',\$row->username); \$this->session->set_userdata('level',\$row->level); //var_dump(\$row); if (\$this->session->userdata('level')=="admin") { redirect('mahasiswa/index'); } elseif (\$this->session->userdata('level')=="user") { redirect('mahasiswa/user'); } else { redirect('login/index','refresh'); } } } </pre> <p>Keterangan :</p> <ol style="list-style-type: none"> 1. <i>Line 25 dan 26</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 <i>login/index</i> pada <i>tag input</i>). 2. <i>Line 28</i> adalah variabel baru(\$cek_login) isinya adalah sebuah model(login_model) dimana model(login_model) tersebut akan menjalankan sebuah <i>function(function login)</i> dengan nama <i>login</i>(yang telah kita buat sebelumnya) dan mempunyai 2 paramater yaitu <i>\$username</i> dan <i>\$password</i> dimana isinya adalah isi dari <i>line 25</i> dan <i>26</i>, yang nanti akan dikirimkan ke <i>function login</i> pada <i>login_model</i>. 3. <i>Line 30-51</i> adalah sebuah pengkondisian jika <i>\$cek_login</i> bernilai benar maka <i>cek_login</i> akan dipecah menjadi beberapa <i>row</i> (<i>\$cek_login as \$row</i>). 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 34</i>) dan kita juga <i>set_userdata</i> menggunakan <i>session</i> dengan nama variabelnya <i>level</i> (<i>line 35</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 userdata('level')</i> berisi <i>admin</i> maka akan me-<i>redirect</i> pada halaman <i>mahasiswa/index</i>(<i>line 38-41</i>). Apabila <i>levelnya</i> berupa <i>user</i> maka akan meredirect ke halaman <i>user</i> (<i>line 42-45</i>). Jika <i>\$cek_login</i> bernilai salah maka akan me-<i>redirect</i> ke halaman <i>login</i> kembali (<i>line 47 - 51</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>\$username=htmlspecialchars(\$this->input->post('uname1')); \$password=htmlspecialchars(\$this->input->post('pwd1'));</pre>
12	<p>Agar halaman <i>login</i> yang pertama kali di-load maka kita harus ubah konfigurasinya pada application/config/routes</p> <pre>\$route['default_controller'] = 'login';//rubah menjadi login yang sebelumnya home \$route['404_override'] = ''; \$route['translate_uri_dashes'] = FALSE;</pre>
13	<p>Hasilnya jika kita masuk menggunakan admin</p> 

	
14	<p>Hasilnya jika kita masuk menggunakan <i>user</i> (jika kita lihat maka hasilnya <i>page not found</i> hal ini dikarenakan kita belum membuat halaman tersebut)</p> 
15	<p>Hasilnya jika kita salah memasukkan <i>password</i> dan <i>username</i> (jika salah maka redirect ke halaman login tanpa ada pemberitahuan jika <i>username</i> dan <i>password</i> salah)</p>
16	<p>Kita tambahkan alert agar pada saat <i>user</i> salah memasukkan <i>username</i> dan <i>password</i>, maka akan muncul pemberitahuan bahwa <i>username</i> dan <i>password</i> anda salah. Tambahkan <i>code</i> berikut pada login/index</p>


```
<div class="alert alert-info" role="alert">
<?php
    if (isset($pesan))
    {
        echo $pesan;
    }else{
        echo "Masukkan username dan password
        anda";
    }
?>
</div>
```

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

```
$data['pesan']="username dan password anda salah";
$data['title']='Login';
$this->load->view('template/header_login',$data);
$this->load->view('login/index');
$this->load->view('template/footer');
//redirect('login/index','refresh');
```

Sehingga hasilnya akan tampak seperti berikut

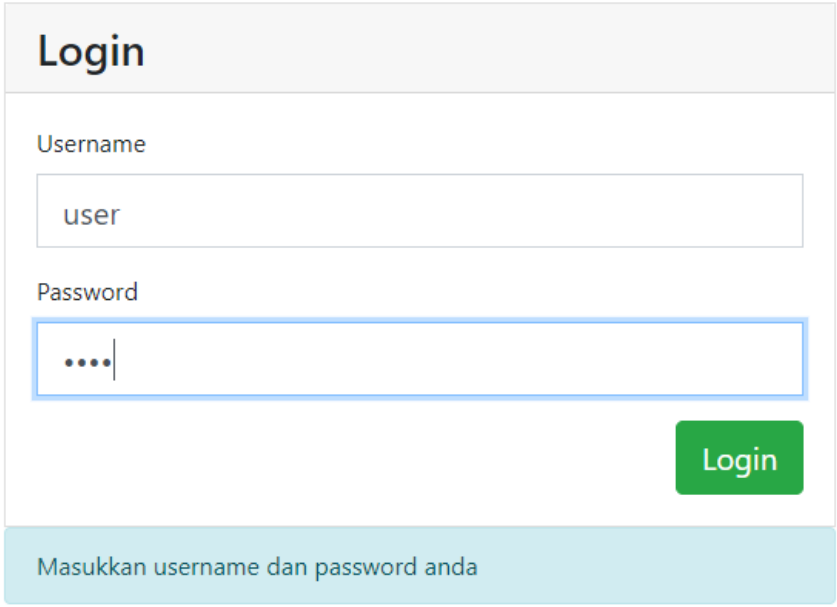
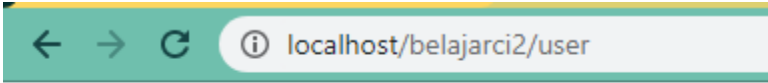
17 Jika kita lihat pada percobaan sebelumnya, pada saat kita *login* dengan menggunakan level *user* maka halaman tidak akan tampil atau *page not found*. Untuk kita akan membuat halaman *user* sederhana

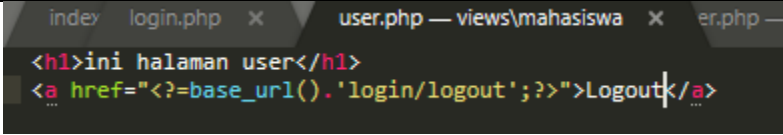
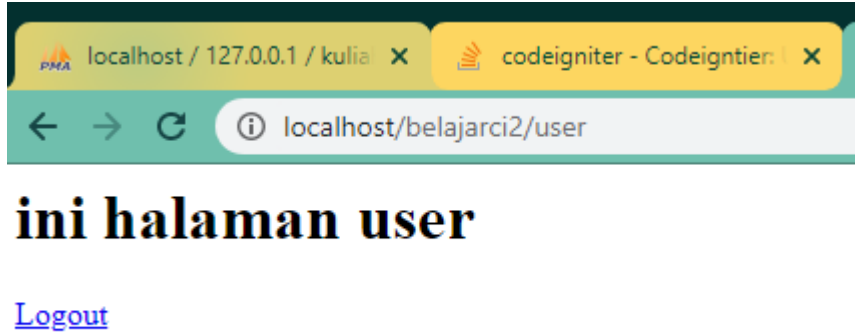
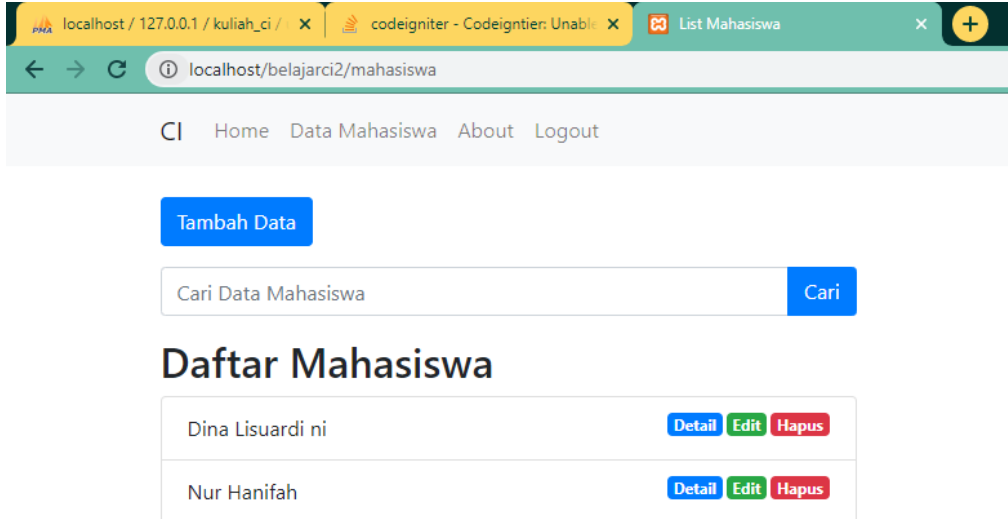
18 Jika dilihat pada **function proses_login** terdapat *code* sebagai berikut:

```
elseif ($this->session->userdata('level')=="user")
{
    redirect('mahasiswa/user');
}
```

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

	<pre>elseif (\$this->session->userdata('level')== "user") { redirect('user'); }</pre>
19	<p>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:</p>  <pre><?php defined('BASEPATH') OR exit('No direct script access allowed'); class user extends CI_Controller { public function __construct() { parent::__construct(); \$this->load->helper('url'); \$this->load->helper('form'); \$this->load->model('login_model'); //\$this->load->library('session'); } public function index() { \$this->load->view('mahasiswa/user'); } }</pre>
20	<p>Membuat tampilan pada view dengan nama user</p>  <pre><h1>ini halaman user</h1></pre> <p>Sehingga hasilnya jika kita login pada level user maka tampilan akan seperti gambar berikut</p>

	  <p>ini halaman user</p>
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><div class="collapse navbar-collapse" id="navbarNavAltMarkup"> <div class="navbar-nav"> home"> Home mahasiswa">Data Mahasiswa ">About login/ logout">Logout </div> </div></pre>
22	<p>Tambahkan function <i>logout</i> pada controller login <i>code</i>-nya sebagai berikut</p> <pre>public function logout() { \$this->session->session_destroy(); redirect('login','refresh'); }</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> index login.php x user.php — views\mahasiswa x er.php — <h1>ini halaman user</h1> <a href="<?base_url().\'login/logout\';?>">Logout </pre>
24	<p>Jika kita lakukan perubahan langsung pada url http://localhost/12.codeigniter/code-keempat/login/ menjadi http://localhost/12.codeigniter/codekeempat/mahasiswa/ seperti pada gambar dibawah ini:</p>  <p>ini halaman user</p> <p>Logout</p>  <p>maka kita akan dapat masuk secara langsung tanpa perlu <i>login</i> terlebih dahulu. Untuk itu kita harus menambahkan beberapa <i>code</i> pada <i>controller/mahasiswa</i> seperti pada gambar dibawah ini (<i>line 15-19</i>)</p>

	<pre> public function __construct() { parent::__construct(); \$this->load->model('mahasiswa_model'); \$this->load->library('form_validation'); //validasi level if (\$this->session->userdata('level')!="admin") { redirect('login','refresh'); } } </pre> <p>Keterangan :</p> <p>Line 15-17 adalah pengkondisian dimana jika levelnya bukan admin maka halaman akan me-redirect ke halaman login</p>
25	<p>Lakukan hal yang sama pada user (controller user)</p> <pre> public function __construct() { parent::__construct(); \$this->load->helper('url'); \$this->load->helper('form'); \$this->load->model('login_model'); //\$this->load->library('session'); if (\$this->session->userdata('level')!="user") { redirect('login','refresh'); } } </pre>
26	<p>Lakukan hal yang sama pada home (controller home)</p> <pre> //validasi level if (\$this->session->userdata('level')!="admin") { redirect('login','refresh'); } </pre>
27	<p>Tambahkan code berikut agar pada halaman admin tampil level usernya (views/home/index)</p> 
28	<p>Comment beberapa code pada controller home (line 12 dan hapus parameter name pada index)</p>

```

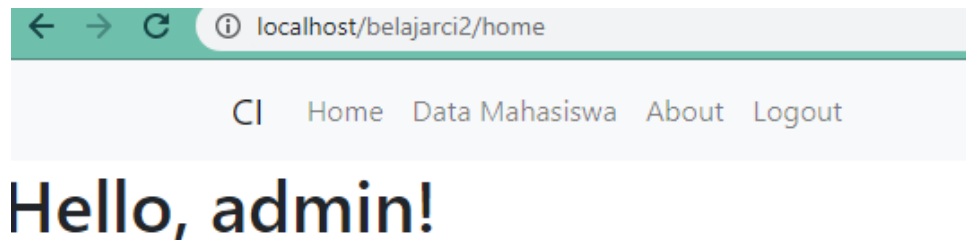
class Home extends CI_Controller {
    public function index($name='')
    {
        //echo "Selamat datang di halaman home";
        $data['title'] = 'Home';
        //$data adl array dg isi arraynya adl nama dan diisi $name
        // $data['name'] = $name;
        $this->load->view('template/header',$data);
        $this->load->view('home/index',$data);
        $this->load->view('template/footer');
        $this->load->library('session');

        //validasi level
        if ($this->session->userdata('level')!="admin") {
            redirect('login','refresh');
        }
    }
}

/* End of file Controllername.php */

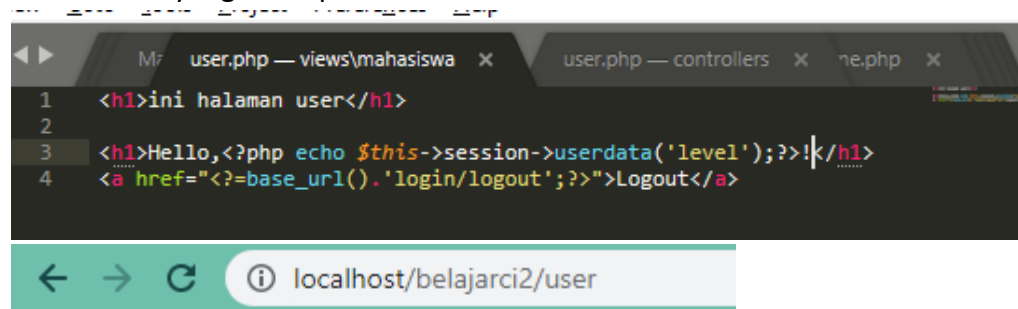
```

Sehingga hasilnya adalah sebagai berikut:



29

Lakukan hal yang sama pada *user*



ini halaman user

Hello,user!

[Logout](#)

