

LAPORAN PRATIKUM PEMOGRAMAN WEB LANJUT

JOBSHEET 5 – Multi Level User



DISUSUN OLEH :

NAMA : MARDHIYAH MILLANIA

KELAS : 2A

PRODI : D-4 TEKNIK INFORMATIKA

NIM : 1841720081

ABSEN : 19

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 <!-- form card login --> 10 <div class="card rounded-0"> 11 <div class="card-header"> 12 <h3 class="mb-0">Login</h3> 13 </div> 14 <div class="card-body"> 15 <form class="form" role="form" autocomplete="off" id="formlogin" novalidate="" method="POST"> 16 <div class="form-group"> 17 <label form="uname1">Username</label> 18 <input type="text" class="form-control form-control-lg rounded-0" name="uname1" id="uname1" required=""> 19 <div class="invalid-feedback">Oops, you missed this one.</div> 20 </div> 21 <div class="form-group"> 22 <label Password</label> 23 <input type="password" class="form-control form-control-lg rounded-0" name="pwd1" id="pwd1" required="" autocomplete="new-password"> 24 <div class="invalid-feedback">Enter your password too!</div> 25 </div> 26 <button type="submit" class="btn btn-success btn-lg float-right" id="btnlogin">Login</button> 27 </form> 28 </div> 29 </div> 30 </div> 31 </div> 32 </div> 33 </div> 34 </div> 35 </div> 36 </div> 37 </div> 38 </div> 39 </div> 40 </div> 41 </div> 42 </div> 43 </div> 44 </div> 45 <?= 46 form_close(); 47 ?> </pre>
2	<p>Buatlah sebuah <i>controller</i> dengan nama login (controller/login)</p> <pre> login.php application > controllers > login.php 1 <?php 2 3 defined('BASEPATH') OR exit('No direct script access allowed'); 4 5 class login extends CI_Controller { 6 7 8 public function index() 9 { 10 \$data['title'] = 'Login'; 11 \$this->load->view('template/header_login', \$data); 12 \$this->load->view('login/index'); 13 \$this->load->view('template/footer'); 14 } 15 16 17 18 19 /* End of file login.php */ </pre>
3	<p>Tambahkan sebuah <i>code</i> pada views/template dengan nama header_login (views/template/header_login)</p>

```

header_login.php
application > views > template > header_login.php
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/4.4.1/css/bootstrap.min.css"
10 integrity="sha384-Vkoo8x4CGs03+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">
11
12 <!--style css -->
13 <style>
14 .badge{
15     margin-left:3px;
16 }
17 </style>
18
19 <!-- <title>Hello, world!</title> -->
20 <title><? $title ?></title>
21
22 </head>
23 <body>

```

4

Ketika dijalankan di browser, hasilnya akan seperti gambar berikut:

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

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 yaituadmin dan *user*

Server: localhost » Basis data: kuliah_ci » Tabel: user

Struktur

#	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)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	password	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	level	varchar(10)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

+ Opsi

		id	username	password	level
Ubah Salin Hapus	1	Admin	admin	admin	
Ubah Salin Hapus	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 ×
application > models > login_model.php
1  <?php
2
3  defined('BASEPATH') OR exit('No direct script access allowed');
4
5  class login_model extends CI_Model {
6
7      function login($username, $password){
8
9          // var_dump($username);
10         // var_dump($password);
11         // die();
12         $this->db->select('username,password,level');
13         $this->db->from('user');
14         $this->db->where('username',$username);//maksdnya adalah selama inputan user yang disimpan pada parameter $user sama dengan username
15
16         $this->db->where('password',$password);
17         $this->db->limit(1);//maksdnya data yg diambil cuma 1
18
19         $query=$this->db->get();
20         if($query->num_rows()=1)//jika data ditemukan
21         {
22             return $query->result();
23         }else{
24             return false;
25         }
26     }
27
28
29
30 }
31
32 /* End of file login_model.php */
```

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 kembalinya kita set menjadi *false*

8

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

```
index.php ×
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">
```

```

42 </div>
43 <!--/container-->
44
45 <?=
46 form_close();
47 ?>

```

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*

```

application > controllers > login.php
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class login extends CI_Controller {
6
7     public function __construct(){
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
15    public function index()
16    {
17        $data['title'] = 'Login';
18        $this->load->view('template/header_login', $data);
19        $this->load->view('login/index');
20        $this->load->view('template/footer');
21    }
22 }
23
24 /* End of file login.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

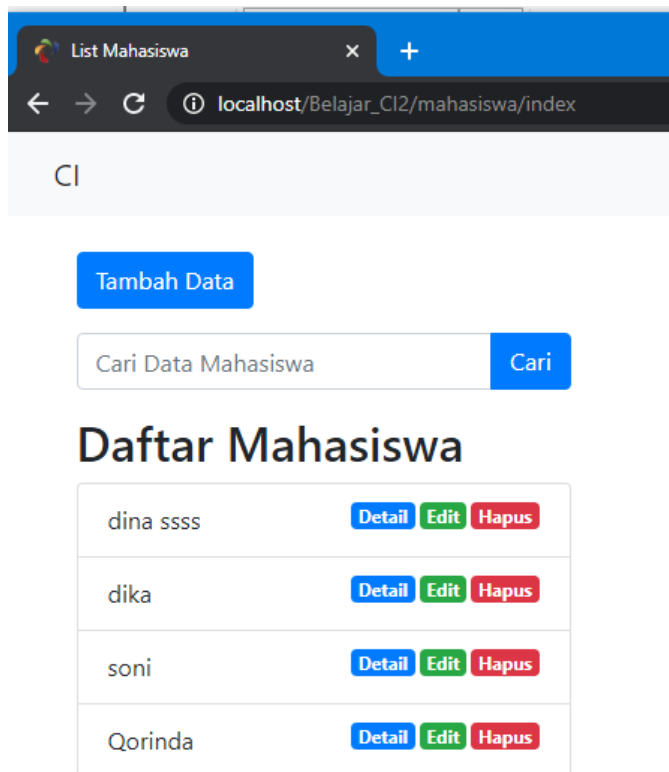
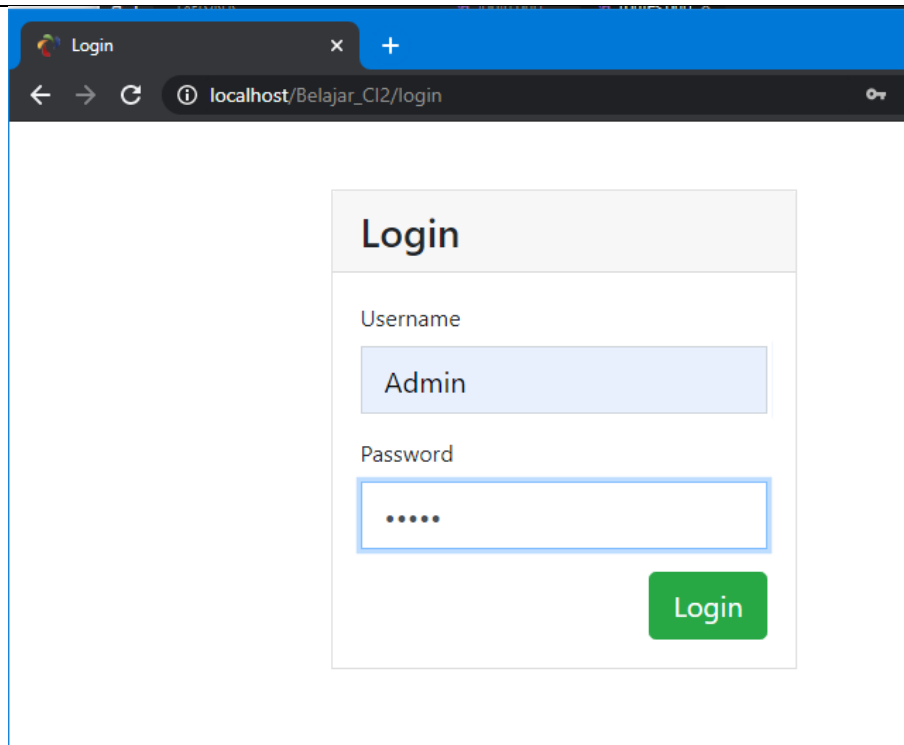
Langkah berikutnya menambahkan **function proses_login** pada *controller login*

```

24 public function proses_login(){
25
26
27     $ceklogin=$this->login_model->login($username,$password);
28
29     if($ceklogin){
30
31         foreach($ceklogin as $row);
32         // kita set userdata pada session dengan nama user dan isi username kita isikan username yg ada pada $row
33         $this->session->set_userdata('user',$row->username);
34         $this->session->set_userdata('level',$row->level);
35
36         // var_dump($row);
37         if($this->session->userdata('level')=="admin")
38         {
39             redirect('mahasiswa/index');
40         }
41         elseif($this->session->userdata('level')=="user")
42         {
43             redirect('mahasiswa/user');
44         }
45     }
46     else
47     {
48         redirect('login/index','refresh');
49     }
50 }
51

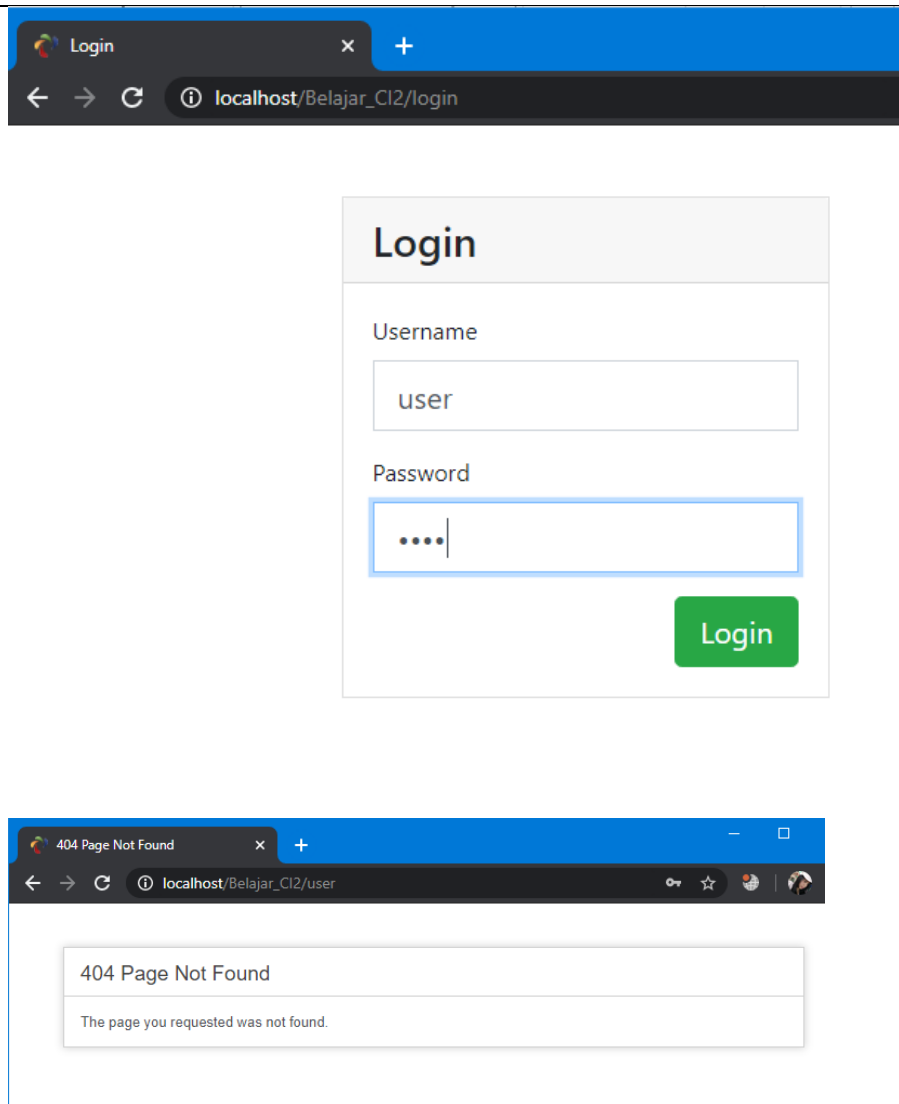
```

	<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 login/index pada tag input). 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</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 25 dan 26</i>, yang nanti akan dikirimkan ke <i>function</i> login pada login_model. 3. <i>Line 30-51</i> adalah sebuah pengkondisian jika \$cek_login bernilai benar maka cek_login akan dipecah menjadi beberapa row (\$cek_login as \$row). setelah itu kita set_userdata menggunakan session dengan nama variabel barunya adalah <i>user</i> sedangkan isinya adalah <i>username</i> (<i>line 34</i>) dan kita juga set_userdata menggunakan session dengan nama variabelnya level (<i>line 35</i>) sedangkan isinya adalah level. Setelah kita pecah menjadi beberapa row dan mempunyai isi yang berbeda-beda maka langkah berikutnya kita buat pengkondisian jika session_userdata('level') berisi admin maka akan me-redirect pada halaman mahasiswa/index(<i>line 38-41</i>). Apabila levelnya berupa user maka akan meredirect ke halaman <i>user</i> (<i>line 42-45</i>). Jika \$cek_login bernilai salah maka akan me-redirect ke halaman <i>login</i> kembali (<i>line 47 - 51</i>).
11	<p>Pada <i>line 25 dan 26</i> tambahkan htmlspecialchars agar <i>user</i> tidak dapat memasukkan <i>char</i> yang special.</p> <pre> 25 26 \$username=htmlspecialchars(\$this->input->post('uname1')); 27 \$password=htmlspecialchars(\$this->input->post('pwd1')); 28 </pre>
12	<p>Agar halaman <i>login</i> yang pertama kali di-load maka kita harus ubah konfigurasinya pada application/config/routes</p> <pre> 52 \$route['default_controller'] = 'login'; 53 \$route['404_override'] = ''; 54 \$route['translate_uri_dashes'] = FALSE; 55 </pre> <p>Default_controller di ubah menjadi login.</p>
13	<p>Hasilnya jika kita masuk menggunakan admin</p>



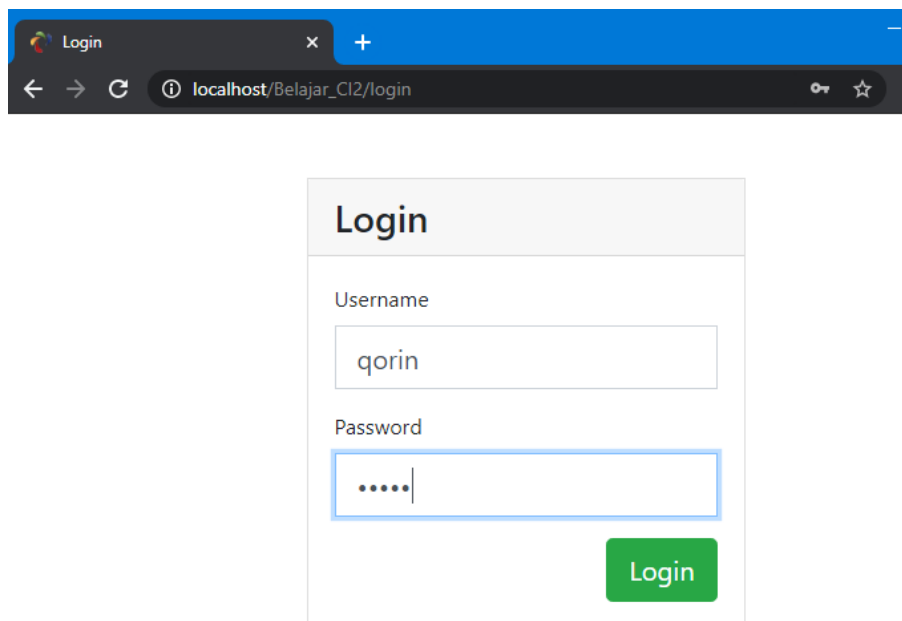
14

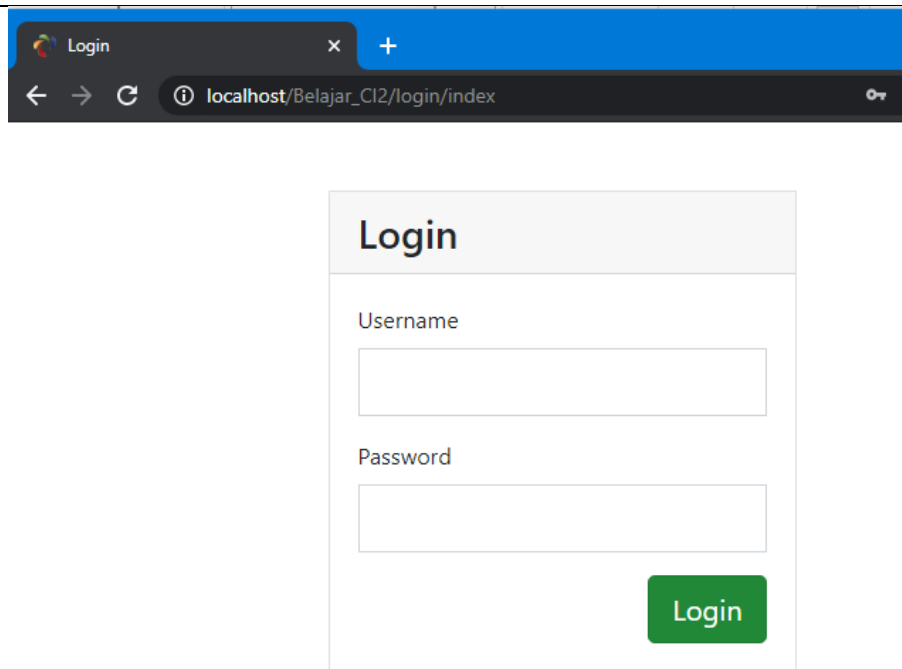
Hasilnya jika kita masuk menggunakan *user* (jika kita lihat maka hasilnya *page not found* hal ini dikarenakan kita belum membuat halaman tersebut)



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
application > views > login > index.php
30      <!--/card-block-->
31    </div>
32    <!--/form card login -->
33    <!-- cek pesan -->
34    <div class="alert alert-info" role="alert">
35      <?php
36        if(isset($pesan))
37        {
38          echo $pesan;
39        }else{
40          echo "Masukkan username dan Password anda";
41        }
42      <?>
43    </div>
44  </div>

```

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

```

49      else
50      {
51        $data['pesan']="username dan passwrod anda salah";
52        $data['title']='Login';
53        $this->load->view('template/header_login',$data);
54        $this->load->view('login/index',$data);
55        $this->load->view('template/footer');
56        // redirect('login/index','refresh');
57      }

```

Sehingga hasilnya akan tampak seperti berikut

Login

Username

qorin

Password

.....

Login

Masukkan username dan Password anda

Login

Username

Password

Login

username dan passwrod anda salah

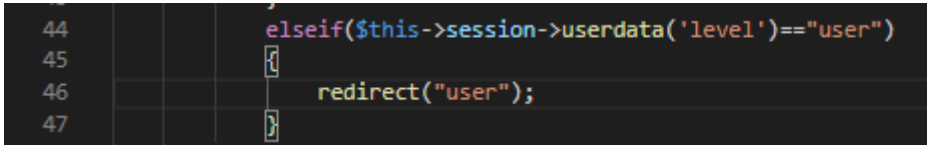
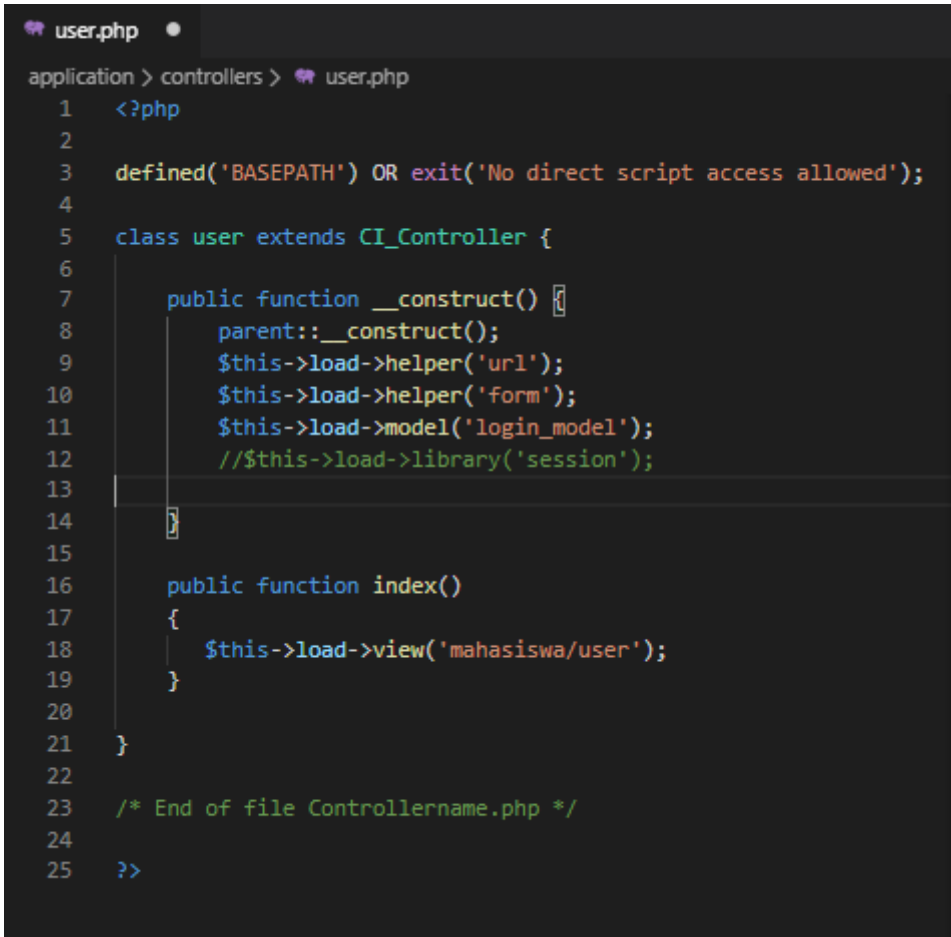
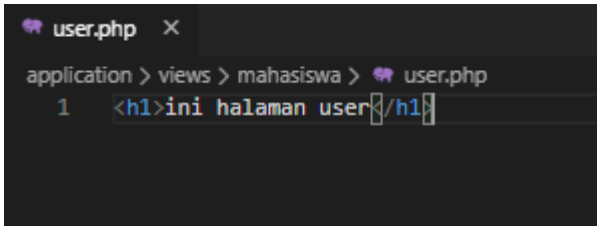
17 Jika kita lihat pada percobaan sebelumnya, pada saat kita *login* dengan menggunakan level *user* maka halaman tidak akan tampil atau *page not found*. Untuk kita akan membuat halaman *user* sederhana

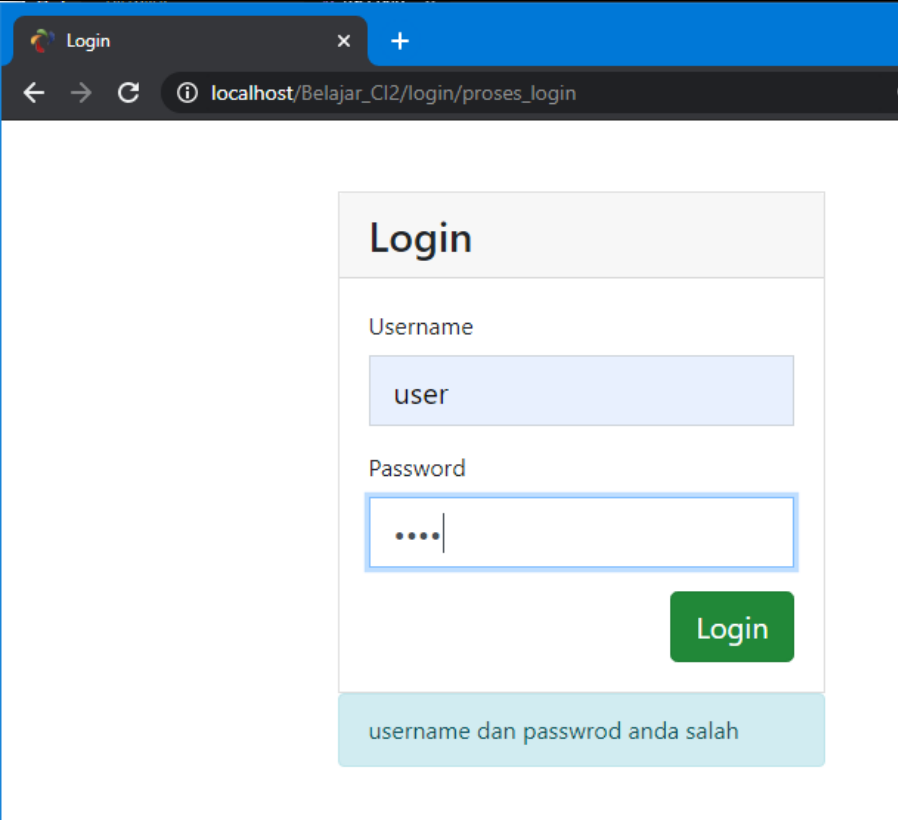
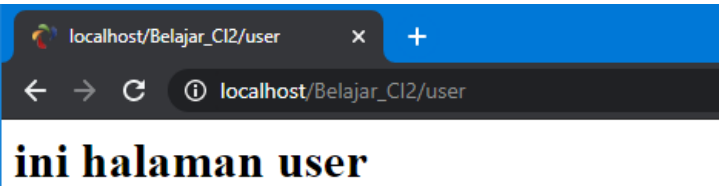
18 Jika dilihat pada **function proses_login** terdapat *code* sebagai berikut:

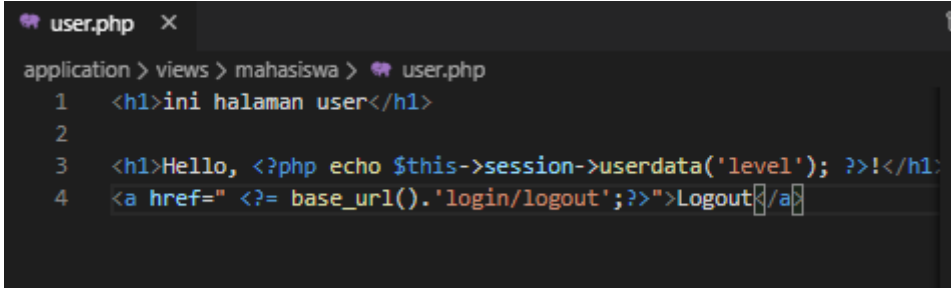
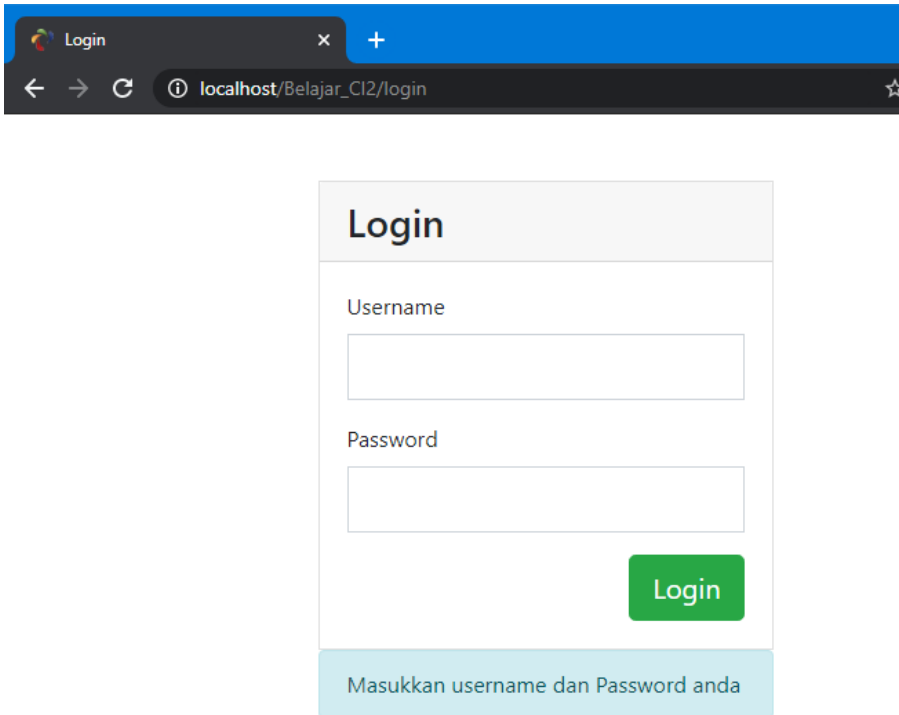
```

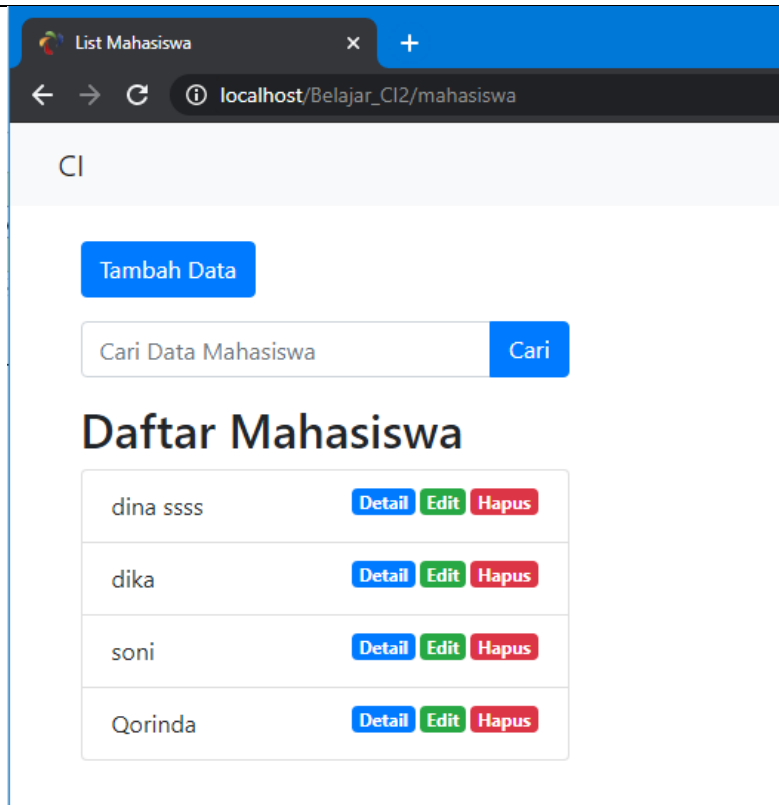
44         elseif($this->session->userdata('level')=="user")
45         {
46             redirect(["mahasiswa/user"]);
47         }

```

	<p>Untuk lebih mudahnya pada line 44 ganti <i>redirect</i> ('mahasiswa/index') menjadi <i>redirect</i>('user');</p> 
19	<p>Setelah mengganti <i>redirect</i>, maka yang harus dilakukan adalah membuat sebuah <i>view</i> dengan nama <i>user</i> pada <i>folder mahasiswa</i>. sebelum kita membuat <i>view</i> dengan nama <i>user</i> kita terlebih dahulu membuat sebuah <i>controller</i> dengan nama <i>user</i>. Buat <i>controller</i> <i>user</i> dengan <i>code</i> sebagai berikut:</p> 
20	<p>Membuat tampilan pada <i>view</i> dengan nama <i>user</i></p>  <p>Sehingga hasilnya jika kita <i>login</i> pada level <i>user</i> maka tampilan akan seperti gambar berikut</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 data-bbox="316 1429 1493 1798"><div class="collapse navbar-collapse" id="navbarNavAltMarkup"> <div class="navbar-nav"> Home mahasiswa">Data Mahasiswa</ ">About login/logout"> <!-- Disabled --> </div> </div> </div> </nav></pre> <p><i>line 33</i> diubah menjadi seperti pada gambar</p>
22	<p>Tambahkan <i>function logout</i> pada <i>controller login</i> <i>code</i>-nya sebagai berikut</p>

	<pre> 60 public function logout(){ 61 \$this->session->sess_destroy(); 62 redirect('login','refresh'); 63 } 64 } 65 66 67 /* End of file login.php */ </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 3 <h1>Hello, <?php echo \$this->session->userdata('level'); ?>!</h1> 4 <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> 



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 15-19*)

```
public function __construct()
{
    //digunakan untuk menjalankan fungsi construct pada class
    parent::__construct();
    $this->load->model('mahasiswa_model');
    $this->load->library('form_validation');

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

Keterangan :

Line 15-17 adalah pengkondisian dimana jika levelnya bukan admin maka halaman akan me-*redirect* ke halaman *login*

25

Lakukan hal yang sama pada *user (controller user)*

```

user.php x
application > controllers > user.php
1  <?php
2
3  defined('BASEPATH') OR exit('No direct script access allowed');
4
5  class user extends CI_Controller {
6
7      public function __construct() {
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     }
19
20     public function index()
21     {
22         $this->load->view('mahasiswa/user');
23     }
24 }
25
26 /* End of file Controllername.php */
27
28 ?>

```

26

Lakukan hal yang sama pada *home* (*controller home*)

```

Home.php x
application > controllers > Home.php
1  <?php
2
3  defined('BASEPATH') OR exit('No direct script access allowed');
4
5  class Home extends CI_Controller {
6
7      //tambahkan parameter name pada index
8      public function index($name= '')
9      {
10         $data['title']='Home';
11         // $data adalah sebuah array dengan isi arraynya adalah r
12         $data['name']=$name;
13         $this->load->view('template/header',$data);
14         // tambahkan $data pada home/index
15         $this->load->view('home/index',$data);
16         $this->load->view('template/footer');
17
18         //validasi level
19         if($this->session->userdata('level')!="admin") {
20             redirect('login','refresh');
21         }
22     }
23
24 }
25
26 /* End of file Controllername.php */
27
28 ?>

```

27

Lakukan hal yang sama pada *home* (controller home)

```

index.php x
application > views > home > index.php
1
2 <h1>Hello, <?= $this->session->userdata('level'); ?>!</h1>
3

```

28

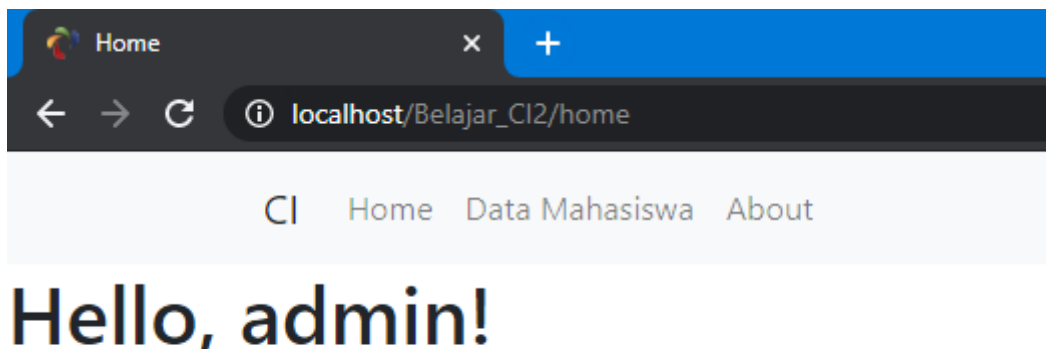
Comment beberapa code pada controller home (line 12 dan hapus parameter name pada index)

```

Home.php x
application > controllers > Home.php
1 <?php
2
3 defined('BASEPATH') OR exit('No direct script access allowed');
4
5 class Home extends CI_Controller {
6
7     //tambahkan parameter name pada index
8     public function index()
9     {
10         $data['title']='Home';
11         // $data adalah sebuah array dengan isi arraynya adalah
12         // $data['name']=$name;
13         $this->load->view('template/header',$data);
14         // tambahkan $data pada home/index
15         $this->load->view('home/index',$data);
16         $this->load->view('template/footer');
17         $this->load->library('session');
18
19         //validasi level
20         if($this->session->userdata('level')!="admin") {
21             redirect('login','refresh');
22         }
23     }
24
25 }
26
27 /* End of file Controllername.php */
28
29 ?>

```

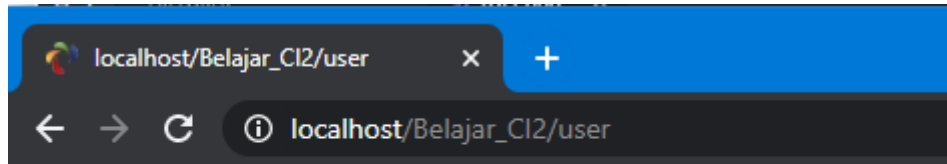
Sehingga hasilnya adalah sebagai berikut:



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, <?php echo $this->session->userdata('level'); ?>!</h1>
4 <a href=" <?= base_url(). 'login/logout'; ?>">Logout</a>
```



ini halaman user

Hello, user!

[Logout](#)