

LAPORAN TUGAS PWL

Jobsheet 5



Disusun Oleh :

ACHMAD RHEZA DARMAWAN / NIM : 1841720115

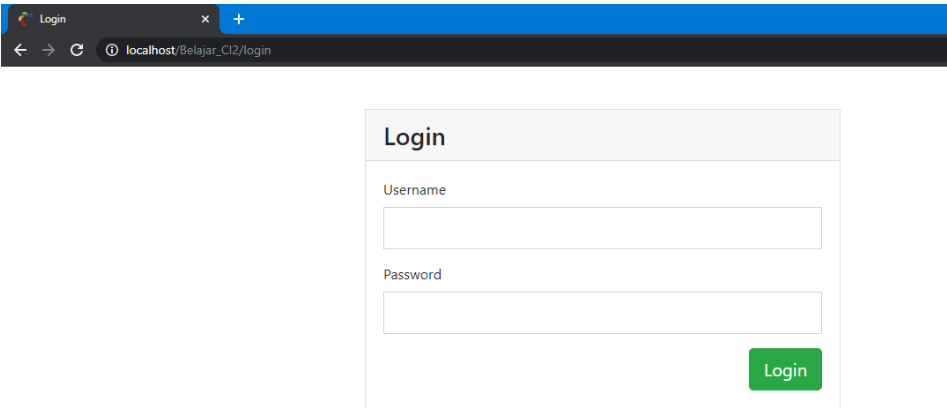
**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG**

**Jl. Candi Panggung No.9, Jatimulyo, Kec. Lowokwaru, Kota Malang,
Jawa Timur 65141**

2020

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> 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	Buatlah sebuah controller dengan nama login (controller/login)

	<div><div>login.php</div><pre>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></div>
3	<div>Tambahkan sebuah <i>code</i> pada views/template dengan nama header_login (views/template/header_login)</div> <div><div>header_login.php</div><pre>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/QSPaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" 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></pre></div>
4	<div>Ketika dijalankan di browser, hasilnya akan seperti gambar berikut:</div> <div></div>
5	<div>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.</div>

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*

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

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Trigger

Struktur tabel Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Terilai	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);//maksudnya adalah selama inputan user yang disimpan pada parameter$user sama dengan username
15
16        $this->db->where('password',$password);
17        $this->db->limit(1);//maksudnya 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

	<p>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 <i>false</i></p>
8	<p>Langkah berikutnya tambahkan form_open dan form_close pada <i>code</i> views/login/index.php</p>  <p>Keterangan:</p> <ol style="list-style-type: none"> 1. Line 2 artinya adalah <i>form</i> yang digunakan untuk membuka halaman proses <i>login</i> pada <i>controller login</i> <p>Line 44 artinya adalah untuk menutup <i>form</i> dikarenakan tadi terdapat perintah <code>form_open</code></p>
9	<p>Buatlah sebuah <i>function</i> dengan nama proses_login pada <i>controller login</i>. Tapi sebelum kita membuat function proses_login maka terlebih dahulu kita tambahkan sebuah function construct</p>  <p>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 <i>function construct</i> tidak perlu ditambahkan lagi.</p>

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
40                 redirect('mahasiswa/index');
41             }
42             elseif($this->session->userdata('level')=="user")
43             {
44                 redirect("mahasiswa/user");
45             }
46         }
47         else
48         {
49             redirect('login/index','refresh');
50         }
51     }

```

Keterangan :

1. Line 25 dan 26 adalah sebuah variabel baru dimana isinya adalah inputan dari *user*. *User* akan mengirimkan data yang terkirim dengan metode **POST** yang tersimpan sementara pada **uname1** dan **pwd1** (**uname1** dan **pwd1** adalah **name** pada inputan *username* dan *password* pada page login/index pada *tag input*).
2. Line 28 adalah variabel baru(**\$cek_login**) isinya adalah sebuah model(**login_model**) dimana model(**login_model**) tersebut akan menjalankan sebuah *function*(**function login**) dengan nama login(yang telah kita buat sebelumnya) dan mempunyai 2 paramater yaitu *\$username* dan *\$password* dimana isinya adalah isi dari *line* 25 dan 26, yang nanti akan dikirimkan ke *function* login pada *login_model*.
3. Line 30-51 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 *user* sedangkan isinya adalah *username* (*line* 34) dan kita juga *set_userdata* menggunakan *session* dengan nama variabelnya *level* (*line* 35) 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*(*line* 38-41). Apabila *levelnya* berupa *user* maka akan meredirect ke halaman *user* (*line* 42-45). Jika *\$cek_login* bernilai salah maka akan me-*redirect* ke halaman *login* kembali (*line* 47 - 51).

11

Pada *line* 25 dan 26 tambahkan **htmlspecialchars** agar *user* tidak dapat memasukkan *char* yang special.

```

25
26     $username=htmlspecialchars($this->input->post('uname1'));
27     $password=htmlspecialchars($this->input->post('pwd1'));
28

```

12

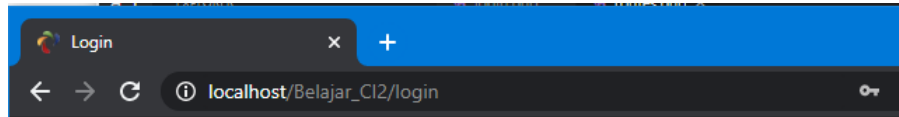
Agar halaman *login* yang pertama kali di-*load* maka kita harus ubah konfigurasinya pada **application/config/routes**

```
52 $route['default_controller'] = 'login';  
53 $route['404_override'] = '';  
54 $route['translate_uri_dashes'] = FALSE;  
55
```

Default_controller di ubah menjadi login.

13

Hasilnya jika kita masuk menggunakan admin

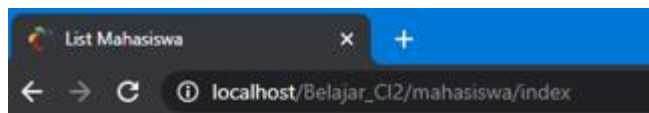


Login

Username

Password

Login



CI

Tambah Data

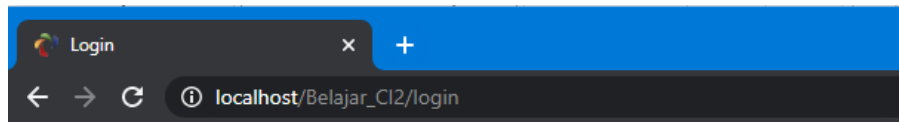
Cari

Daftar Mahasiswa

dina ssss	Detail	Edit	Hapus
dika	Detail	Edit	Hapus
soni	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)

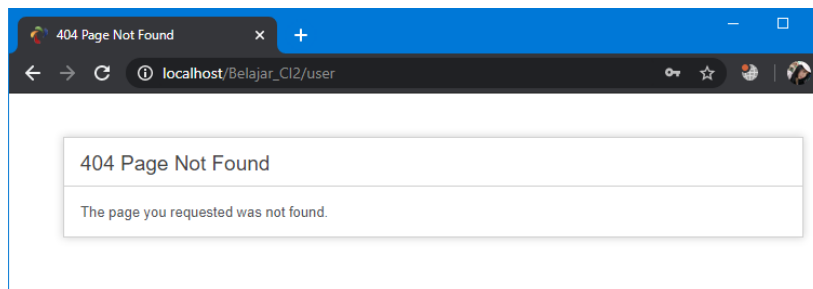


Login

Username

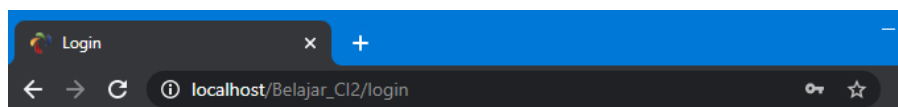
Password

Login



15

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

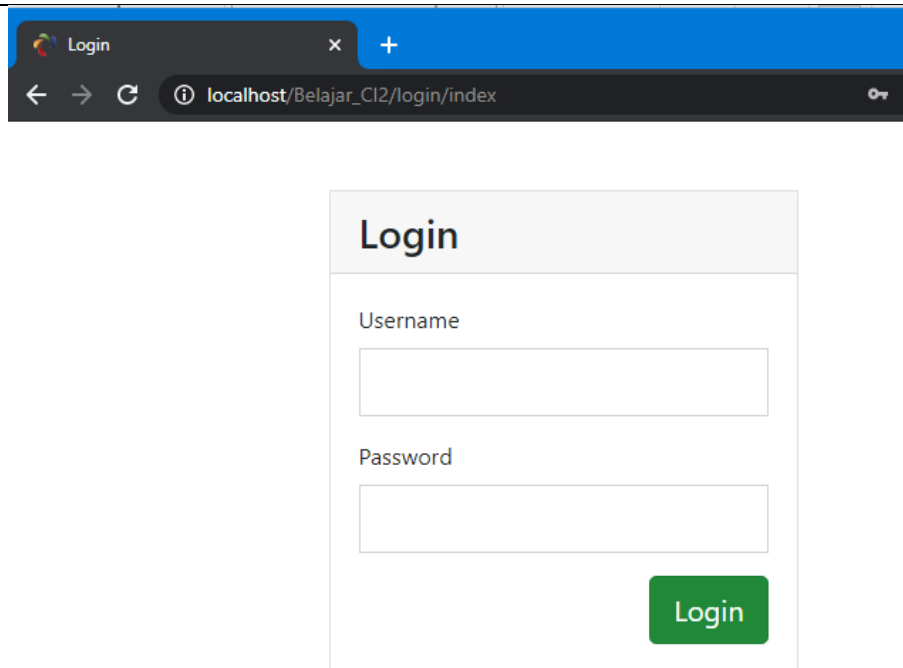


Login

Username

Password

Login



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
44 </div>

```

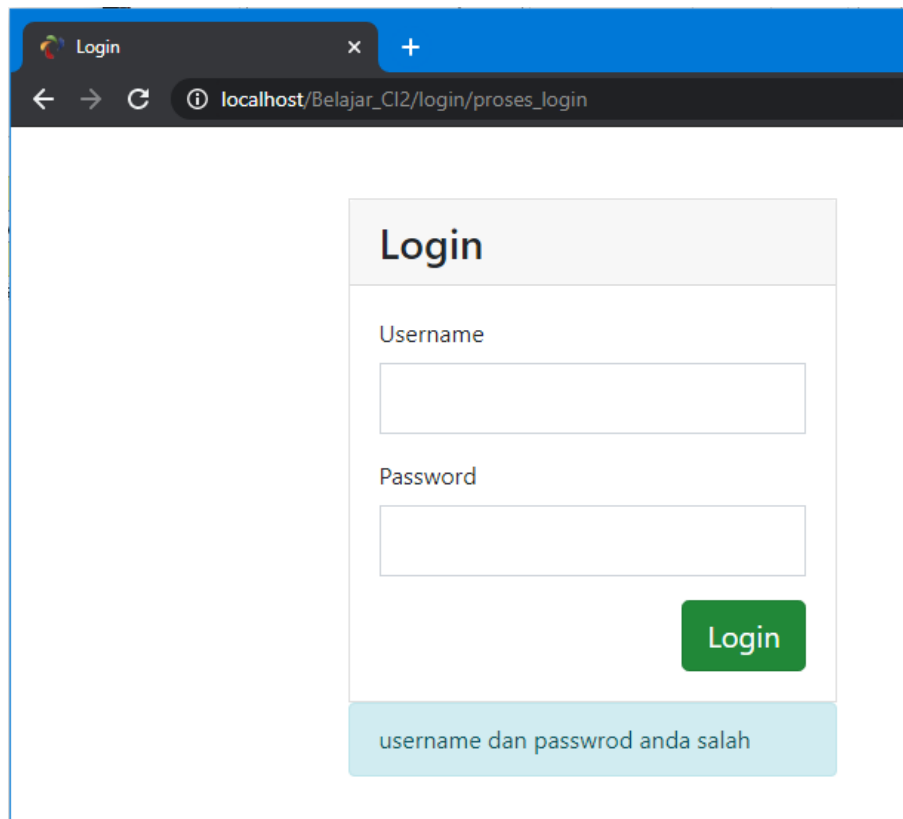
dan pada bagian **controller/login/proses_login** kita ubah pada *else*-nya

```

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



17

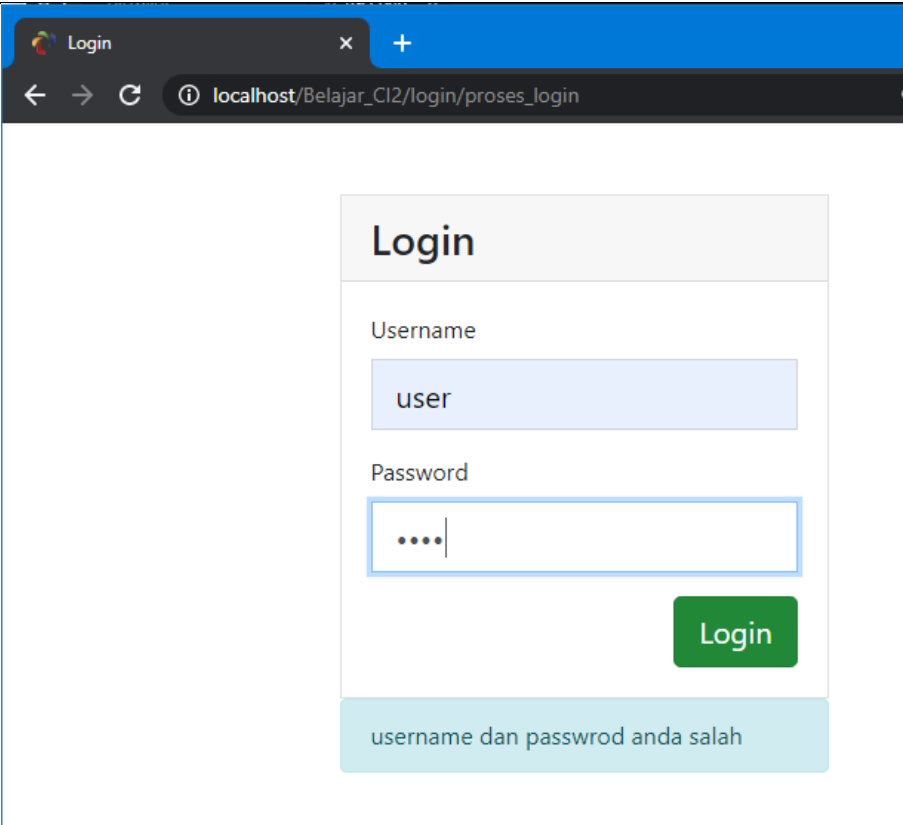
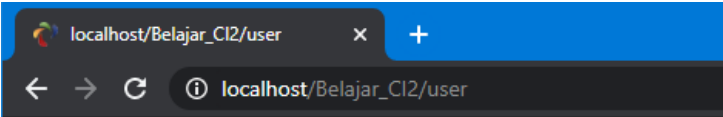
Jika kita lihat pada percobaan sebelumnya, pada saat kita *login* dengan menggunakan level *user* maka halaman tidak akan tampil atau *page not found*. Untuk kita akan membuat halaman *user* sederhana

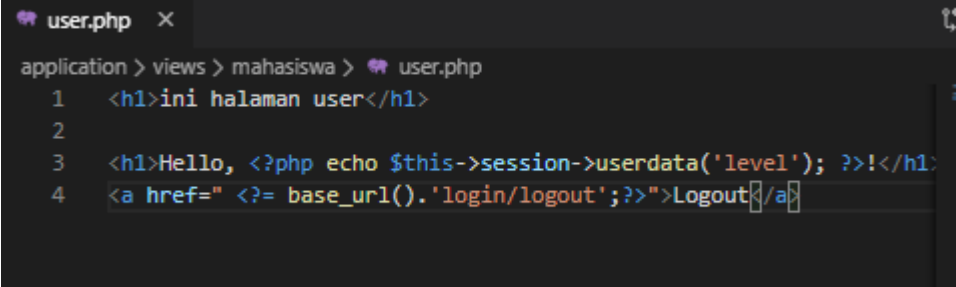
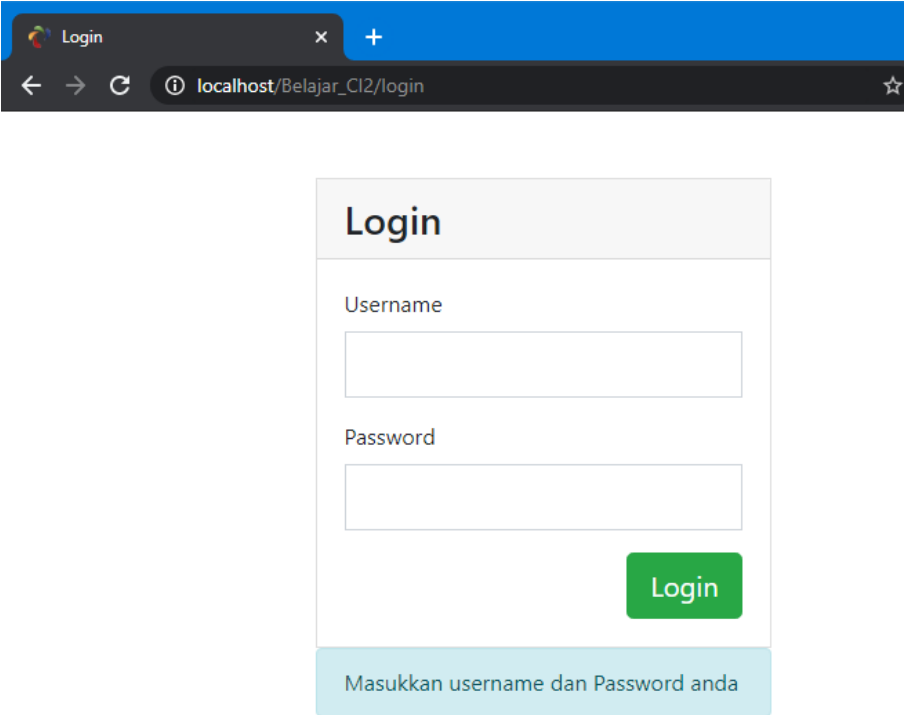
18

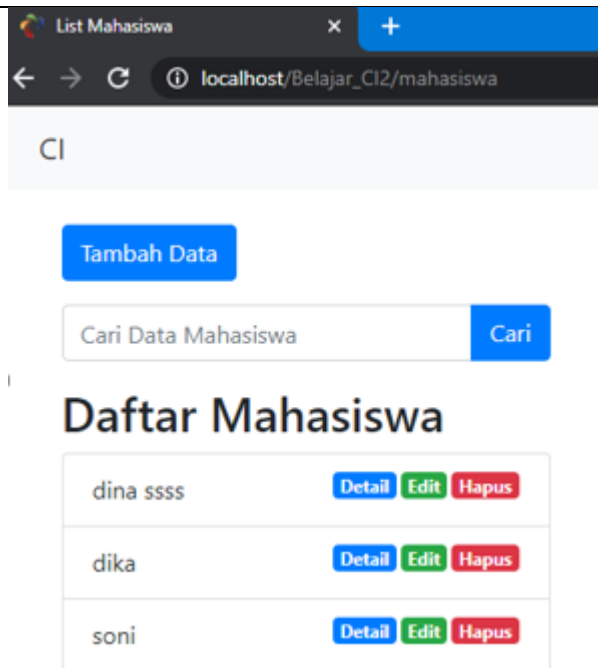
Jika dilihat pada **function proses_login** terdapat *code* sebagai berikut:

```
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> <pre> 44 elseif(\$this->session->userdata('level')== "user") 45 { 46 redirect("user"); 47 } </pre>
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> user dengan <i>code</i> sebagai berikut:</p> <pre> user.php 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 15 public function index() 16 { 17 \$this->load->view('mahasiswa/user'); 18 } 19 20 } 21 22 23 /* End of file Controllername.php */ 24 25 ?> </pre>
20	<p>Membuat tampilan pada <i>view</i> dengan nama <i>user</i></p> <pre> user.php application > views > mahasiswa > user.php 1 <h1>ini halaman user</h1> </pre> <p>Sehingga hasilnya jika kita <i>login</i> pada level <i>user</i> 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 data-bbox="316 1433 1493 1803"><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>  <p>The screenshot shows a web browser window with the title 'Login' and the address bar displaying 'localhost/Belajar_CI2/login'. The page content includes a 'Login' header, a 'Username' label with an input field, a 'Password' label with an input field, a green 'Login' button, and a footer text 'Masukkan username dan Password anda'.</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)

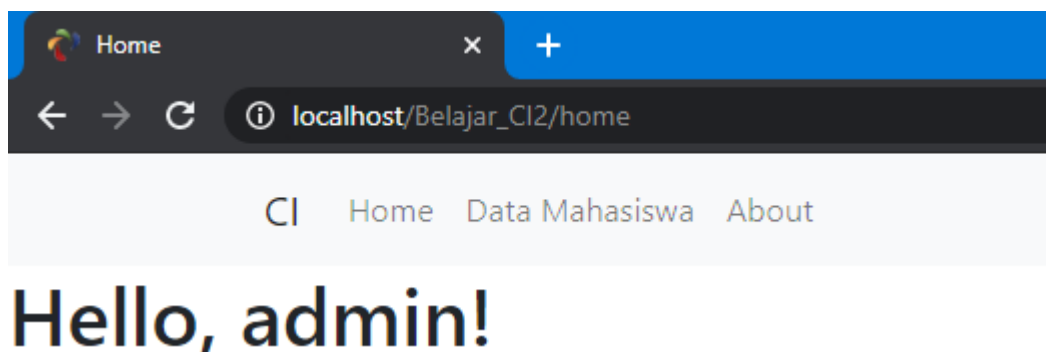
```
index.php X
application > views > home > index.php
1
2 <h1>Hello, <?= $this->session->userdata('level'); ?></h1>
3
```

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 ?>
```

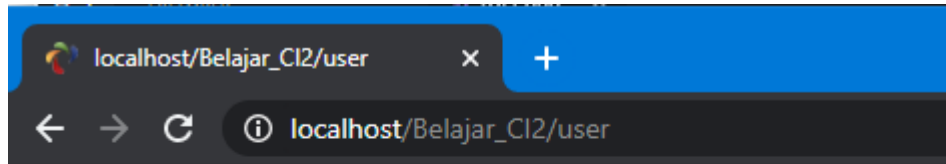
Sehingga hasilnya adalah sebagai berikut:



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](#)