LAPORAN PRAKTIKUM
DESAIN DAN PEMROGRAMAN WEB
JOBSHEET 5
Multi Level User

Nama: Osa Mahanani S.
NIM: 1841720066
Kelas: TI-2B

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

Praktikum – Bagian 1: Membuat Login dengan Multi Level User

Langkah	Keterangan
1	Buat tampilan login (views/login/index.php), dan tambahkan code berikut pada views/login/index.php:

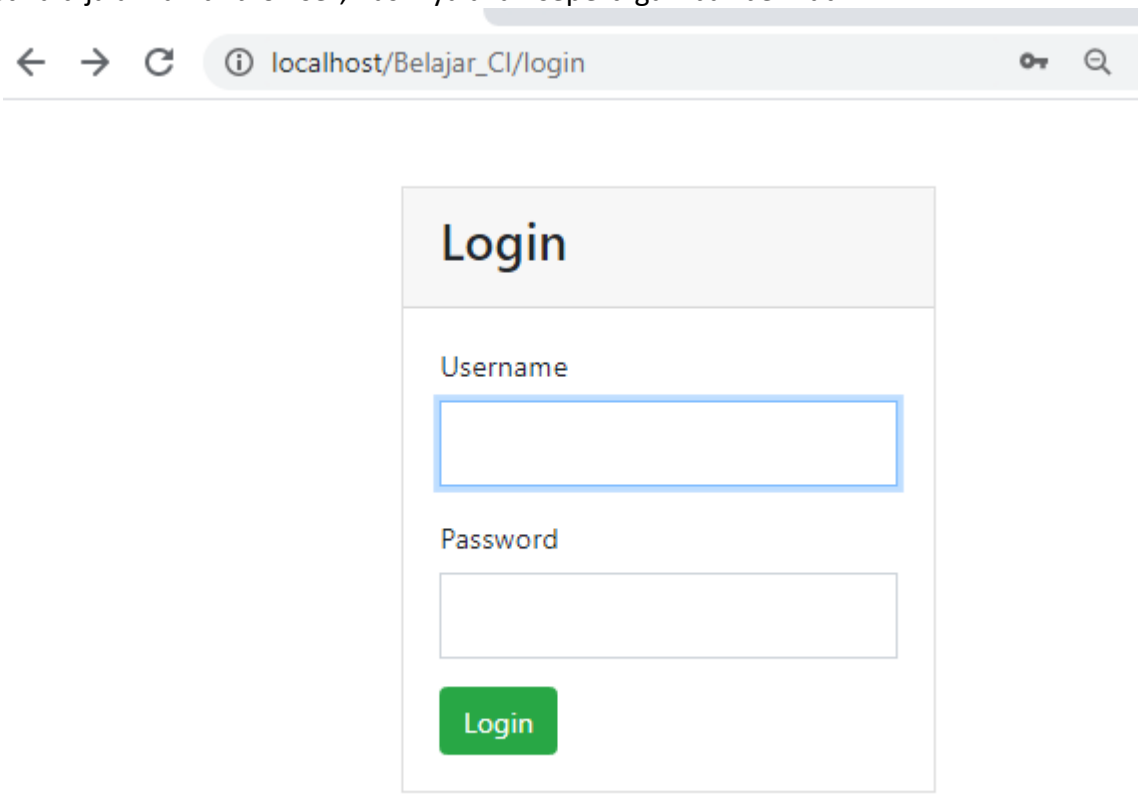
```php
<?=
form_open('login/proses_login');
?>
<div class="container py-5">
    <div class="row">
        <div class="col-md-12">
            <div class="row">
                <div class="col-md-6 mx-auto">

                    <!-- 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" novalidate="" 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>
                        <!--/card-block-->
                    </div>
                    <!-- /form card login -->
                </div>
            </div>
            <!--/row-->
        </div>
        <!--/col-->
    </div>
    <!--/row-->
</div>
<!--/container-->

<?=
form_close();
?>
```

2	Buatlah sebuah ***controller*** dengan nama ***login*** (**controller/login**)

index.php | header_login.php | login.php ×

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

class login extends CI_Controller {

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

}

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

3	Tambahkan sebuah *code* pada **views/template** dengan nama **header_login (views/template/header_login)**
	index.php header_login.php × login.php
	`<!doctype html>`
	`<html lang="en">`
	`  <head>`
	`  <!-- Required meta tags -->`
	`  <meta charset="utf-8">`
	`  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">`
	
	`  <!-- Bootstrap CSS -->`
	`  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO" crossorigin="anonymous">`
	
	`  <!-- style css  -->`
	`  <style>`
	`  .badge{`
	`  margin-left:3px;`
	`  }`
	`  </style>`
	
	`  <!-- <title>Hello, world!</title> -->`
	`  <title><?= $title ?></title>`
	
	`  </head>`
	`  <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*

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	**id**	int(11)			No	*None*			Change Drop More
2	**username**	varchar(100)	utf8mb4_general_ci		No	*None*			Change Drop More
3	**password**	varchar(100)	utf8mb4_general_ci		No	*None*			Change Drop More
4	**level**	varchar(100)	utf8mb4_general_ci		No	*None*			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext

+ Options

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)**

application > models > login_model.php

```
1   <?php
2   defined('BASEPATH') OR exit('No direct script access allowed');
3
4   class login_model extends CI_Model {
5
6       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);
14          $this->db->where('password',$password);
15          $this->db->limit(1); //maksudnya adlh data yg diambil cuma 1
16
17          $query=$this->db->get();
18          if($query->num_rows()==1)
19          {
20              return $query->result();
21          }else{
22              return false;
23          }
24      }
25  }
26
27
28
29  /* End of file Controllername.php */
30
31  ?>
```

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 8. *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** application > views > login > index.php 1 <?= 2 form_open('login/proses_login'); 3 ?> 4 5 6 7 8 41 42 <!--container--> 43 44 <?= 45 form_close(); 46 ?> 47 Keterangan: 1. *Line* 2 artinya adalah *form* yang digunakan untuk membuka halaman proses *login* pada *controller login* 2. *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**

```
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
15      public function index()
16      {
17           $data['title']='Login';
18           $this->load->helper('form');
19           $this->load->view('template/header_login',$data);
20           $this->load->view('login/index');
21           $this->load->view('template/footer');
22      }
23
24  }
```

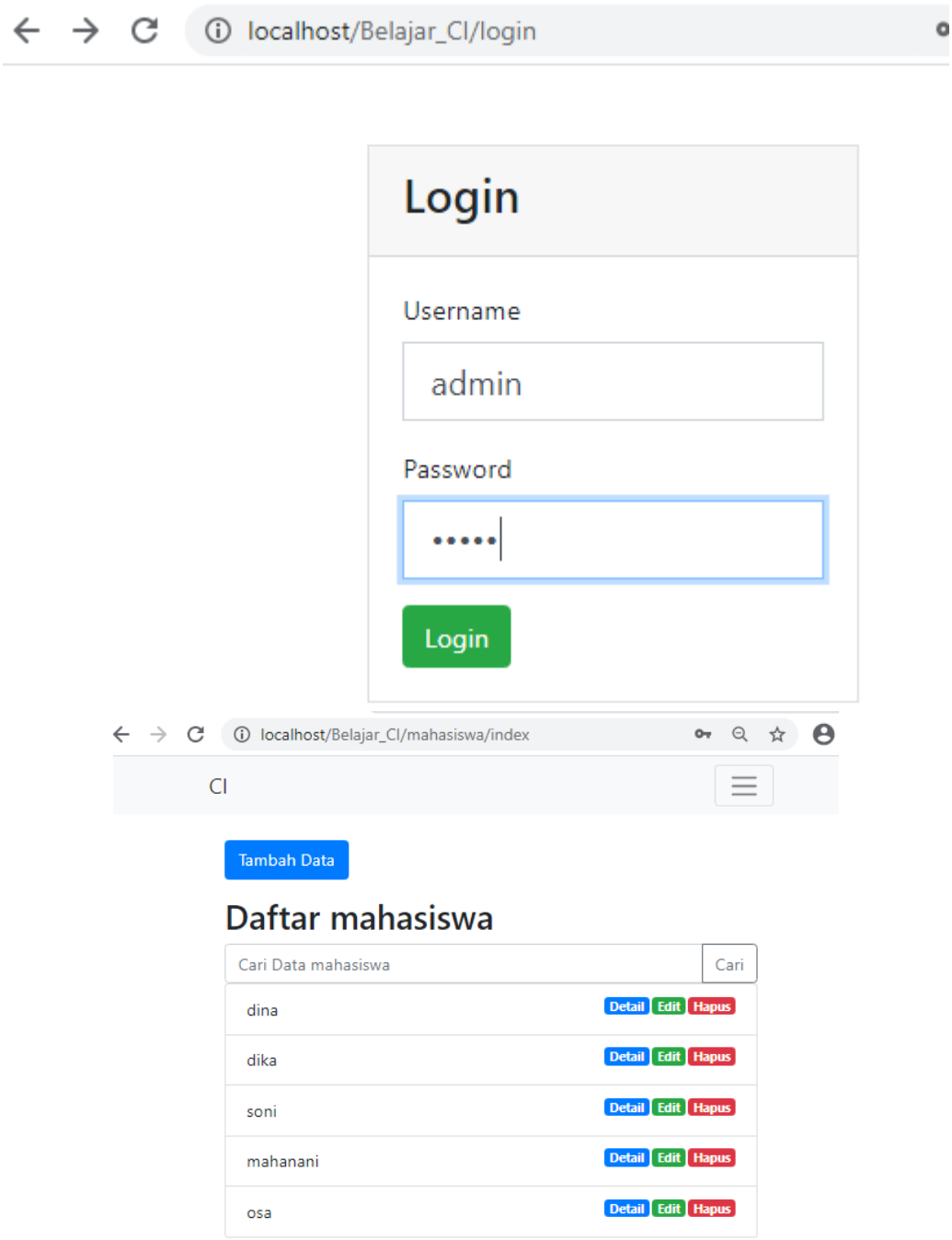
Keterangan : Jika pada config/autoload sudah kalian setting maka kita tidak perlu menambahkan lagi seperti pada line 11. Dikarenakan pada config/autoload/library sudah diset maka pada *function* construct tidak perlu ditambahkan lagi.

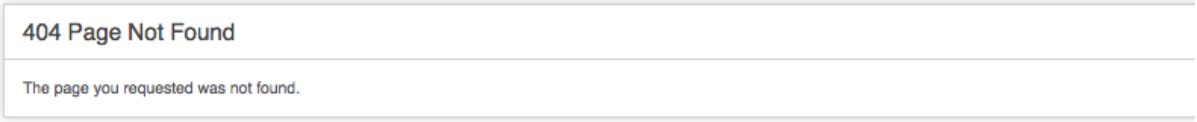
10 Langkah berikutnya menambahkan **function proses_login** pada **controller login**

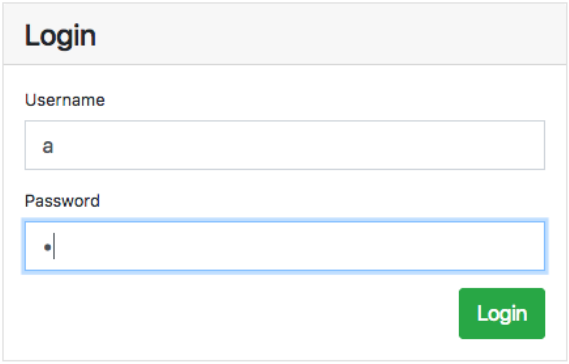
application > controllers > login.php

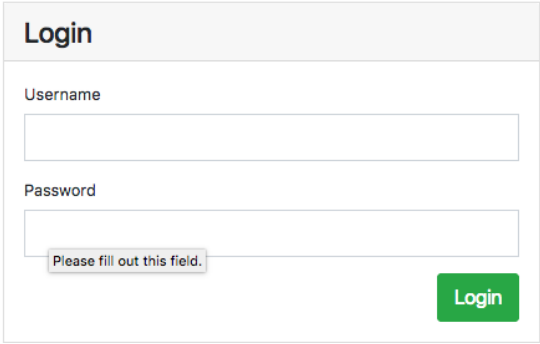
```
24      public function proses_login(){

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

29          $ceklogin=$this->login_model->login($username,$password);

31          if($ceklogin){
32              foreach($ceklogin as $row);

34              $this->session->set_userdata('user', $row->username);
35              $this->session->set_userdata('level',$row->level);

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              redirect('login/index','refresh');
48          }
49      }
50  }
51  /* End of file login.php */
52  ?>
```

<table>
<tr><td></td><td>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 menjalan 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 $cek_login bernilai salah maka akan me-redirect ke halaman login kembali (line 47 - 51).</td></tr>
<tr><td>11</td><td>Pada line 25 dan 26 tambahkan htmlspesialchar agar user tidak dapat memasukkan char yang special.
<pre>$username=htmlspecialchars($this->input->post('uname1'));
$password=htmlspecialchars($this->input->post('pwd1'));</pre></td></tr>
<tr><td>12</td><td>Agar halaman login yang pertama kali di-load maka kita harus ubah konfigurasinya pada application/config/routes
<pre>$route['default_controller'] = 'login'; //rubah menjadi
$route['404_override'] = '';
$route['translate_uri_dashes'] = FALSE;</pre></td></tr>
<tr><td>13</td><td>Hasilnya jika kita masuk menggunakan admin</td></tr>
</table>

<table>
<tr>
<td></td>
<td>


</td>
</tr>
<tr>
<td>14</td>
<td>Hasilnya jika kita masuk menggunakan user (jika kita lihat maka hasilnya page not found hal ini dikarenakan kita belum membuat halaman tersebut)
 
</td>
</tr>
<tr>
<td>15</td>
<td>Hasilnya jika kita salah memasukkan password dan username (jika salah maka redirect ke halaman login tanpa ada pemberitahuan jika username dan password salah)</td>
</tr>
</table>





16

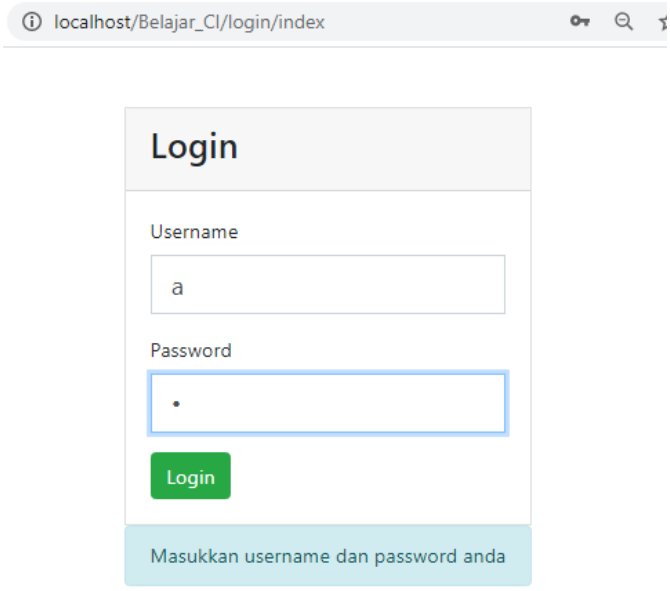
Kita tambahkan ***alert*** agar pada saat ***user*** salah memasukkan ***username*** dan ***password***, maka akan muncul pemberitahuan bahwa ***username*** dan ***password*** anda salah. Tambahkan *code* berikut pada **login/index**

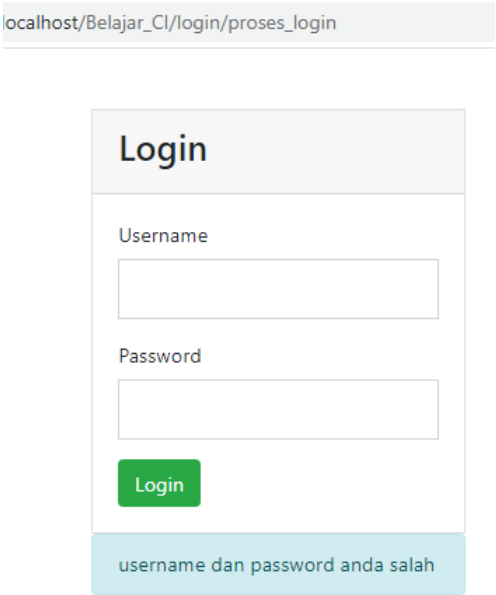
```
<div class="alert alert-info" role="alert">
    <?php
    if(isset($pesan))
    {
        echo $pesan;
    }else{
        echo"Masukkan username dan password anda";
    }
    ?>
</div>
```

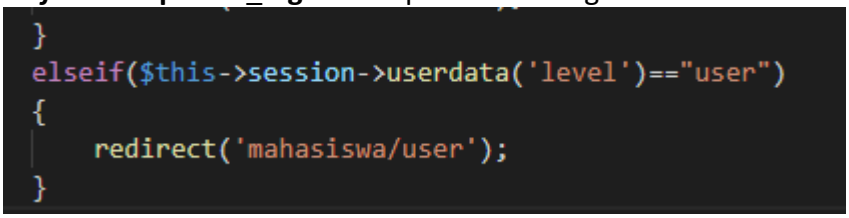
dan pada bagian **controller/login/proses_login** kita ubah pada *else*-nya

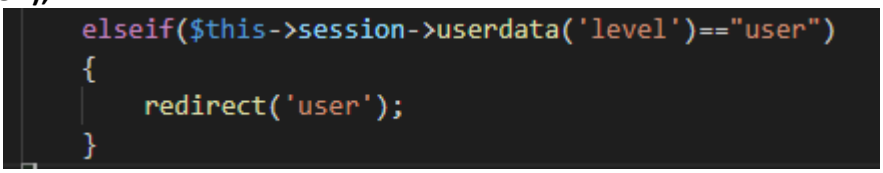
```
else
{
    $data['pesan']="username dan password anda salah";
    $data['title']="Login";
    $this->load->view('template/header_login',$data);
    $this->load->view('login/index',$data);
    $this->load->view('template/footer');
    //redirect('login/index','refresh');
}
```

Sehingga hasilnya akan tampak seperti berikut

<table>
<tr><td></td><td>




</td></tr>
<tr><td>17</td><td>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</td></tr>
<tr><td>18</td><td>Jika dilihat pada function proses_login terdapat code sebagai berikut:


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


</td></tr>
</table>

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:

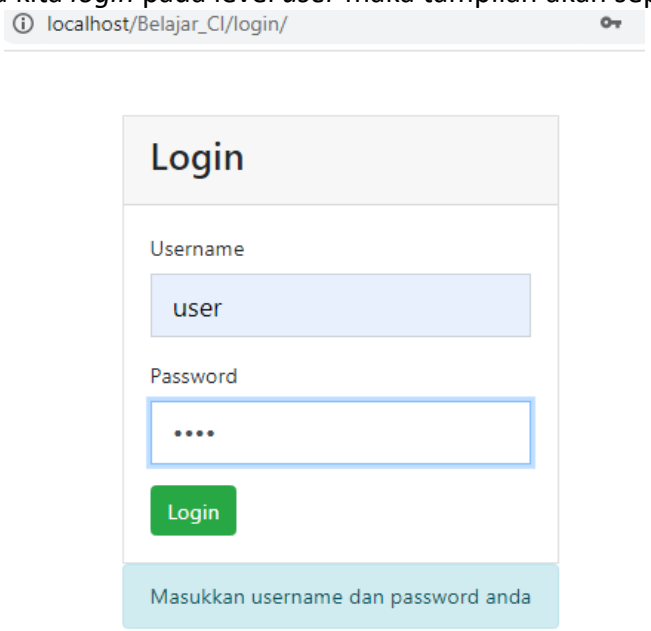
```php
<?php

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

class Controllername 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');
    }

}

/* End of file Controllername.php */

?>
```

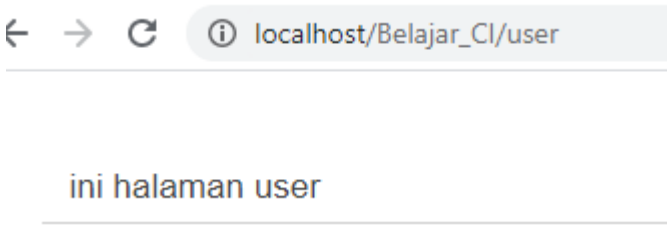
20	Membuat tampilan pada *view* dengan nama *user*

application > controllers > user.php

```
1   <h1> ini halaman user</h1>
```

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



<table>
<tr><td></td><td> 
</td></tr>
<tr><td>21</td><td>Setelah berhasil membuat halaman user, maka langkah berikutnya kita membuat menu logout. Tambahkan code berikut pada views/template/header line 36

```
<div class="collapse navbar-collapse" id="navbarNavAltMarkup">
  <div class="navbar-nav">
    <a class="nav-item nav-link" href="<?= base_url(); ?>Home">Home</a>
    <a class="nav-item nav-link" href="<?= base_url(); ?>mahasiswa">Data Mahasiswa</a>
    <a class="nav-item nav-link" href="<?= base_url(); ?>">About</a>
    <a class="nav-item nav-link" href="<?= base_url(); ?>login/logout">Logout</a>
    <!-- <a class="nav-item nav-link disabled" href="#">Disabled</a> -->
  </div>
```

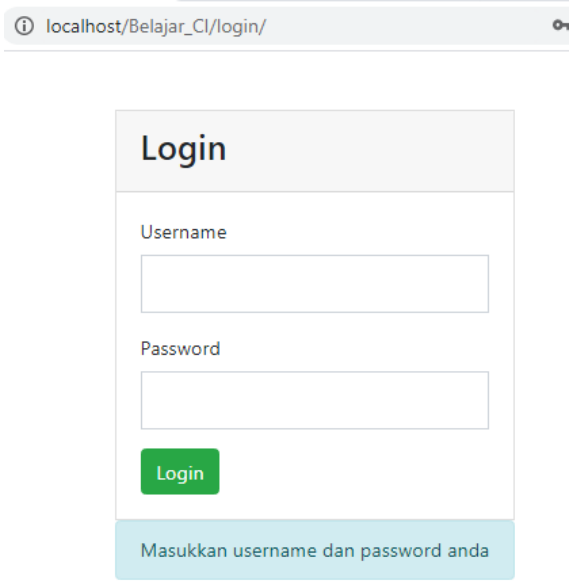
line 33 diubah menjadi seperti pada gambar</td></tr>
<tr><td>22</td><td>Tambahkan function logout pada controller login code-nya sebagai berikut

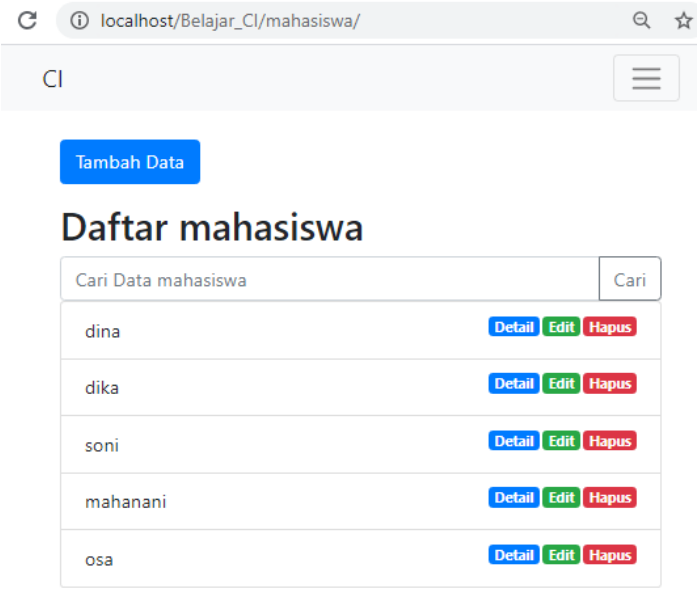
```
public function logout(){
    $this->session->sess_destroy();
    redirect('login','refresh');
}
```

</td></tr>
<tr><td>23</td><td>Hasilnya jika kita klik logout maka akan muncul halaman login. Pada halaman user kita juga tambahkan code untuk proses logout sederhana

```
ation > controllers > user.php
<h1> ini halaman user</h1>
<a href=" <?= base_url(). 'login/logout';?>">Logout</a>
```

</td></tr>
<tr><td>24</td><td>Jika kita lakukan perubahan langsung pada url <u>http://localhost/12.codeIgniter/code-keempat/login/</u> menjadi <u>http://localhost/12.codeIgniter/codekeempat/mahasiswa/</u> seperti pada gambar dibawah ini:


</td></tr>
</table>



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 c
    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**)

```php
class Controllername 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');

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

    public function index()
    {
        $this->load->view('mahasiswa/user');
    }

}

/* End of file Controllername.php */

?>
```

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

```php

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

class Home extends CI_Controller {

    //tambahkan parameter name pada index
    public function index($name='')
    {
        $data['title']='Home';
        //$data adalah sebuah array dengan isi arraynya adalah
        $data['name']=$name;
        $this->load->view('template/header',$data);
        //echo "Selamat datang di halaman home";
        $this->load->view('home/index',$data);
        $this->load->view('template/footer');

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

    }

}

/* End of file Home.php */

?>
```

27 Tambahkan *code* berikut agar pada halaman admin tampil level *user*nya (**views/home/index**)

```
<h1>Hello, <?= $this->session->userdata('level'); ?>!</h1>
```

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

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         //echo "Selamat datang di halaman home";
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
28 /* End of file Home.php */
29
30 ?>
```

Sehingga hasilnya adalah sebagai berikut:

Hello, admin!

29 Lakukan hal yang sama pada *user*

application > controllers > 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>
5
```

	ini halaman user **Hello, user!** Logout