



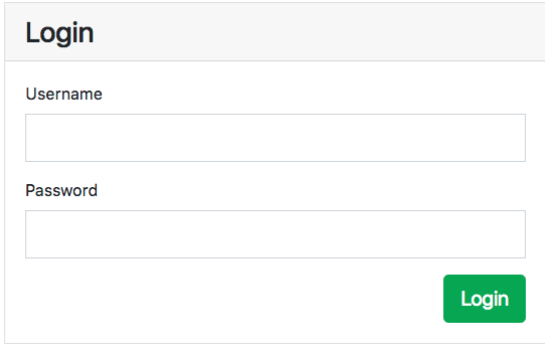
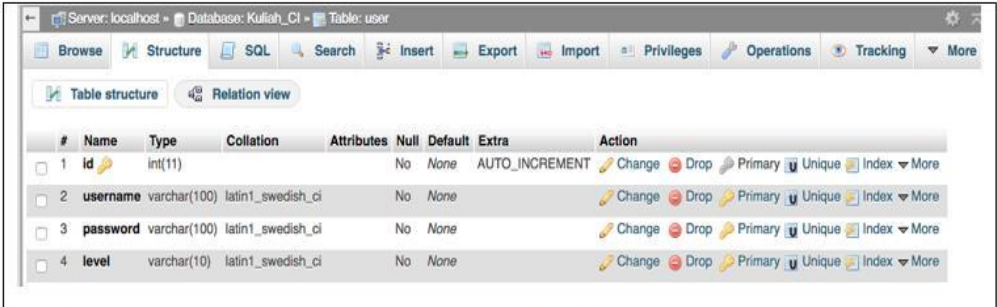
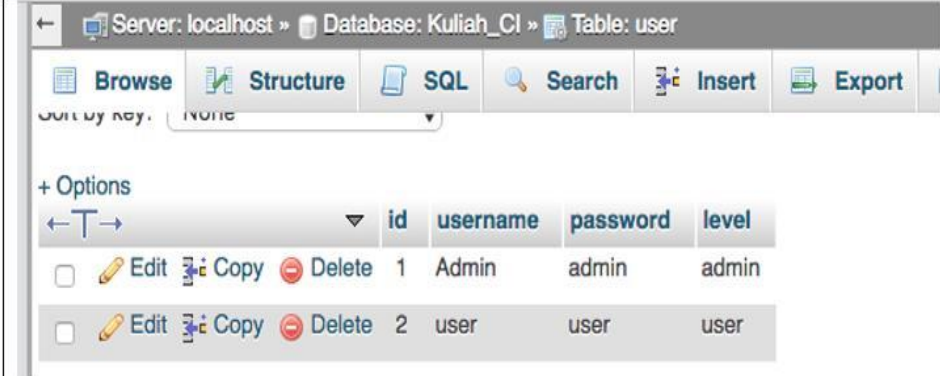
LAPORAN PRAKTIKUM PEMROGRAMAN DASAR WEB

NAMA : NUR HANIFAH
NIM : 1841720062
Kelas : TI 2B
Isi Laporan : Multi Level User

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 open('login/proses_login'); 3 4 s="container py-5" 5 class="row" 6 div class="col-md-12" 7 <div class="row" 8 <div class="col-md-6 max-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" id="uname1" required=""> 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="pwd1" id="pwd1" required="" autoco 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 <!-- form card login --> 33 <!-- rek nesan --><hr></div></pre>

2	<p>Buatlah sebuah <i>controller</i> dengan nama login (controller/login)</p> <pre> 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 { 8 parent::__construct(); 9 \$this->load->helper('url'); 10 \$this->load->helper('form'); 11 \$this->load->model('login_model'); 12 } 13 public function index() 14 { 15 \$data['title'] = 'Login'; 16 \$this->load->helper('form'); 17 \$this->load->view('template/header_login', \$data); 18 \$this->load->view('login/index'); 19 \$this->load->view('template/footer'); 20 } 21 22 public function proses_login() 23 { 24 \$username = htmlspecialchars(\$this->input->post('uname1')); 25 \$password = htmlspecialchars(\$this->input->post('pwd1')); 26 27 \$ceklogin = \$this->login_model->login(\$username, \$password); 28 29 if (\$ceklogin) { 30 foreach (\$ceklogin as \$row); 31 //kita set semua data pada session dengan nama user dan isi username kita isikan username 32 \$this->session->set_userdata('user', \$row->username); 33 \$this->session->set_userdata('level', \$row->level); </pre>
3	<p>Tambahkan sebuah <i>code</i> pada views/template dengan nama header_login (views/template/header_login)</p> <pre> application > views > template > header_login.php > ... 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=" 11 12 <!-- style css --> 13 <style> 14 .badge { 15 margin-left: 3px; 16 } 17 </style> 18 <title><?= \$title ?></title> 19 </head> 20 21 <body> </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. 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 yaitu <i>admin</i> dan <i>user</i></p>  
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)

```

application > models > login_model.php > login_model > login
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          $this->db->select('username,password,level');
8          $this->db->from('user');
9          $this->db->where('username',$username);//ini maksudnya selama inputan user
10         $this->db->where('password',$password);
11         $this->db->limit(1);//maksudnya data yang diambil cuma 1
12
13         $query=$this->db->get();
14         if($query->num_rows()==1)//jika di temukan
15         {
16             return $query->result();
17         }else{
18             return false;
19         }
20     }
21 }
22 }
23 }
24 ?>

```

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**

```

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">

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

```

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*

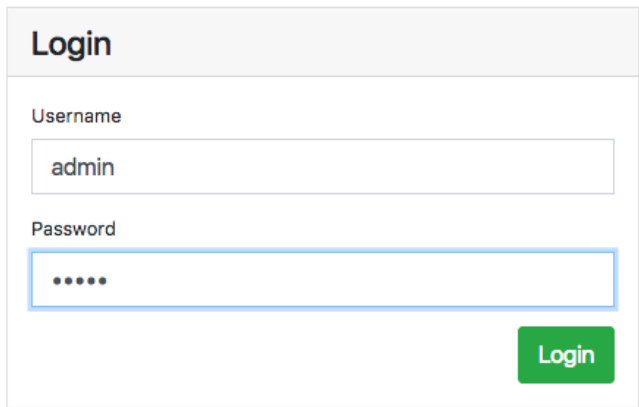
```

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      {
8          parent::__construct();
9          $this->load->helper('url');
10         $this->load->helper('form');
11         $this->load->model('login_model');
12     }
13     public function index()
14     {
15         $data['title'] = 'Login';
16         $this->load->helper('form');
17         $this->load->view('template/header_login', $data);
18         $this->load->view('login/index');
19         $this->load->view('template/footer');
20     }
21

```

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 = htmlspecialchars(\$this->input->post('uname1')); \$password = htmlspecialchars(\$this->input->post('pwd1')); \$ceklogin = \$this->login_model->login(\$username, \$password); if (\$ceklogin) { foreach (\$ceklogin as \$row); //kita set semua data pada session dengan nama user dan isi username kita \$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('user'); } } 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'); } } </pre> <p>Keterangan :</p> <ol style="list-style-type: none"> 1. Line 25 dan 26 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. Line 28 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 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 <i>function login</i> 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 <i>user</i> sedangkan isinya adalah <i>username</i> (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

	<p>maka langkah berikutnya kita buat pengkondisian jika <i>session</i> <i>userdata('level')</i> berisi admin maka akan me-<i>redirect</i> pada halaman mahasiswa/index(line 38-41). Apabila levelnya berupa user maka akan meredirect ke halaman <i>user</i> (line 42-45). Jika <i>\$cek_login</i> bernilai salah maka akan me-<i>redirect</i> ke halaman <i>login</i> kembali (line 47 - 51).</p>
11	<p>Pada <i>line</i> 25 dan 26 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-<i>load</i> maka kita harus ubah konfigurasinya pada application/config/routes</p> <pre>\$route['default_controller'] = 'login'; //rubah menjadi login yg sblnnya home \$route['404_override'] = ''; \$route['translate_uri_dashes'] = FALSE;</pre>
13	<p>Hasilnya jika kita masuk menggunakan admin</p> 

Tambah Data

Cari Data Mahasiswa

Cari

Daftar Mahasiswa

Hanifah cantik

Detail Edit Hapus

Silvi

Detail Edit Hapus

hanifah

Detail Edit Hapus

Nur Hanifah

Detail Edit Hapus

NingHanifah

Detail Edit Hapus

violet

Detail Edit Hapus

violet

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

user

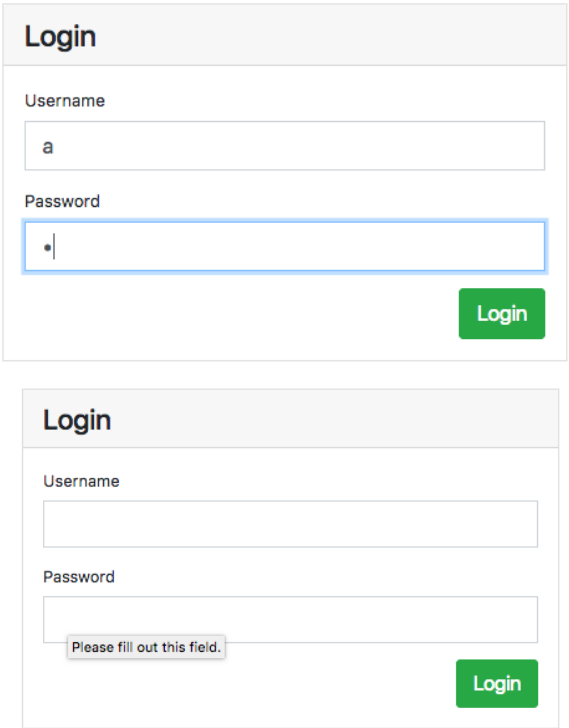
Password

....

Login

404 Page Not Found

The page you requested was not found.

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> <div data-bbox="609 333 1174 1056">  </div>
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 code berikut pada login/index</p> <pre data-bbox="542 1188 1252 1575"> <!-- /form card login --> <!-- cek pesan --> <div class="alert alert-info" role="alert"> <?php if(isset(\$pesan)) { echo \$pesan; }else{ echo "Masukkan username dan password anda"; } ?> </div> </pre> <p>dan pada bagian controller/login/proses_login kita ubah pada <i>else</i>-nya</p>

```

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

The image shows two screenshots of a login form. The top screenshot shows a successful login attempt with the username 'a' and a password field containing a dot. The bottom screenshot shows an unsuccessful login attempt with the message 'username dan password anda salah' displayed at the bottom.

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:

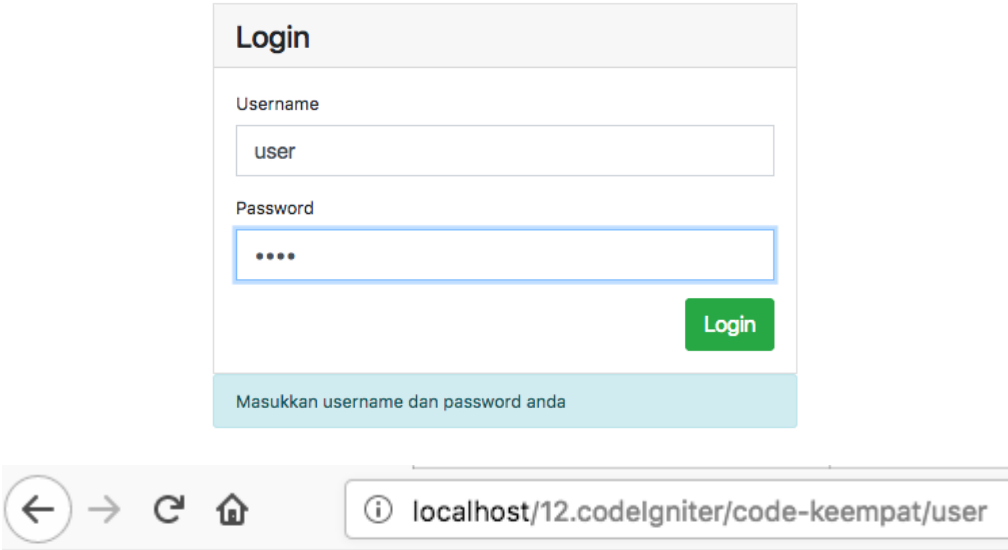
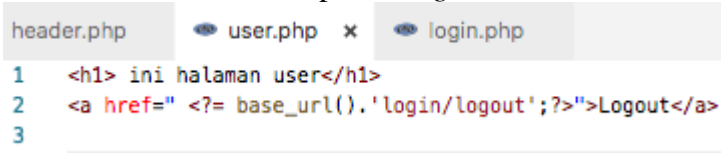
```

}
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> application > controllers > user.php > user > index 1 <?php 2 defined('BASEPATH') or exit('No direct script access allowed'); 3 4 class user extends CI_Controller 5 { 6 7 public function __construct() 8 { 9 parent::__construct(); 10 \$this->load->helper('url'); 11 \$this->load->helper('form'); 12 \$this->load->model('login_model'); 13 \$this->load->library('form_validation'); 14 15 //validasi level 16 if (\$this->session->userdata('level') != "user") { 17 redirect('login', 'refresh'); 18 } 19 } 20 21 public function index() 22 { 23 \$this->load->view('mahasiswa/user'); 24 } 25 </pre>
20	<p>Membuat tampilan pada view dengan nama user</p>  <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> 31 <div class="collapse navbar-collapse" id="navbarNavAltMarkup"> 32 <div class="navbar-nav"> 33 <a class="nav-item nav-link" href="<?= base_url(); ?>Home">Home 34 <a class="nav-item nav-link" href="<?= base_url(); ?>mahasiswa">Data Mahasi 35 <a class="nav-item nav-link" href="<?= base_url(); ?>">About 36 <a class="nav-item nav-link" href="<?= base_url(); ?>login/logout">Logout 37 </div> 38 </div> </pre> <p>line 33 diubah menjadi seperti pada gambar</p> </pre>
22	<p>Tambahkan function <i>logout</i> pada controller <i>login</i> <i>code</i>-nya sebagai berikut</p> <pre> public function logout() { \$this->session->sess_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> 
24	<p>Jika kita lakukan perubahan langsung pada url http://localhost/12.codeIgniter/code-keempat/login/ menjadi</p>

<http://localhost/12.codeIgniter/codekeempat/mahasiswa/> seperti pada gambar dibawah ini:

The screenshot shows a web application interface. At the top, there is a browser address bar with the URL `localhost/12.codeIgniter/code-keempat/login`. Below the address bar is a login form titled "Login". The form has two input fields: "Username" and "Password". A green "Login" button is located to the right of the password field. Below the input fields is a light blue box with the text "Masukkan username dan password anda".

Below the login form is a navigation bar with links: "CI", "Home", "Data Mahasiswa", "About", and "Logout".

Below the navigation bar is a blue button labeled "Tambah Data".

Below the button is a search bar with the placeholder text "Cari Data Mahasiswa" and a blue "Cari" button.

Below the search bar is a section titled "Daftar Mahasiswa". It contains a table with student names and action buttons.

Hanifah cantik	Detail	Edit	Hapus
Silvi	Detail	Edit	Hapus
hanifah	Detail	Edit	Hapus
Nur Hanifah	Detail	Edit	Hapus
NingHanifah	Detail	Edit	Hapus
violet	Detail	Edit	Hapus
violet	Detail	Edit	Hapus
NingHanifah	Detail	Edit	Hapus

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*)

```
//fungsi yang akan dijalankan saat classnya diinstansiasi
public function __construct()
{
    //digunakan untuk menjalankan fungsi constrauct pada class parrent(ci_controller)
    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)**

```
application > controllers > user.php > user > index
1  <?php
2  defined('BASEPATH') or exit('No direct script access allowed');
3
4  class user extends CI_Controller
5  {
6
7      public function __construct()
8      {
9          parent::__construct();
10         $this->load->helper('url');
11         $this->load->helper('form');
12         $this->load->model('login_model');
13         $this->load->library('form_validation');
14
15         //validasi level
16         if ($this->session->userdata('level') != "user") {
17             redirect('login', 'refresh');
18         }
19     }
20
21     public function index()
22     {
23         $this->load->view('mahasiswa/user');
24     }
25 }
26
```

26

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

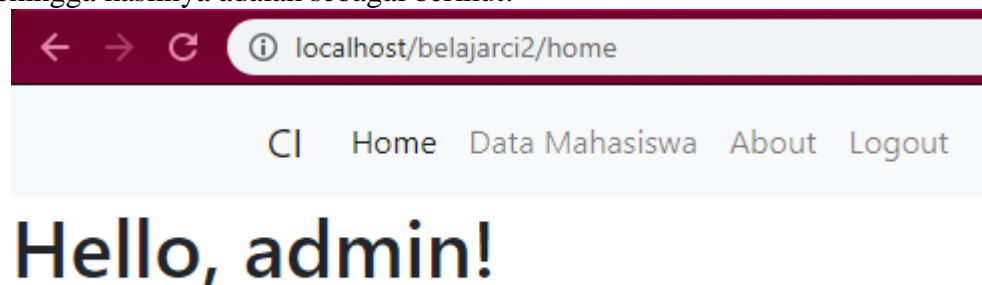
	<pre> application > controllers > Home.php > Home > index 1 <?php 2 3 defined('BASEPATH') or exit('No direct script access allowed'); 4 5 class Home extends CI_Controller 6 { 7 public function index() 8 { 9 //echo "Selamat datang di halaman home"; 10 \$data['title'] = 'Home'; 11 //\$data adl array dg isi arraynya adl nama dan diisi \$name 12 // \$data['name'] = \$name; 13 \$this->load->view('template/header', \$data); 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 ?> </pre>
27	<p>Tambahkan <i>code</i> berikut agar pada halaman admin tampil level <i>user</i>nya (views/home/index)</p> <div> <div>Home.php</div> <div>index.php ×</div> <div>login.php</div> <div>user.php</div> </div> <pre> 1 2 <h1>Hello, <?= \$this->session->userdata('level'); ?>!</h1> 3 4 </pre>
28	<p>Comment beberapa <i>code</i> pada <i>controller home</i> (line 12 dan hapus parameter name pada index)</p>

```

application > controllers > Home.php > Home > index
1  <?php
2
3  defined('BASEPATH') or exit('No direct script access allowed');
4
5  class Home extends CI_Controller
6  {
7      public function index()
8      {
9          //echo "Selamat datang di halaman home";
10         $data['title'] = 'Home';
11         //$data adl array dg isi arraynya adl nama dan diisi $name
12         // $data['name'] = $name;
13         $this->load->view('template/header', $data);
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 ?>

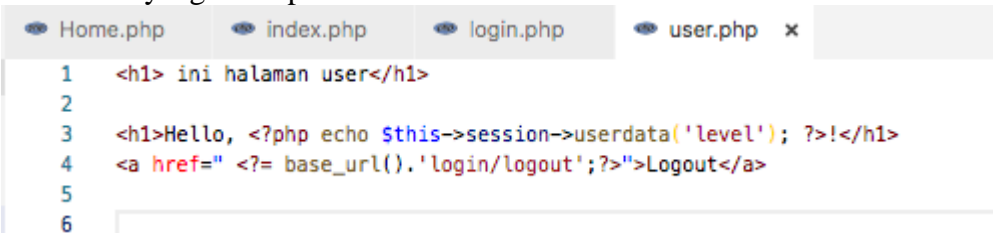
```


Sehingga hasilnya adalah sebagai berikut:



29

Lakukan hal yang sama pada *user*



	 <p>ini adalah halaman user</p> <p>Hello, user!</p> <p>Logout</p>
--	---

-- Selamat Mengerjakan --