

LAPORAN PEMROGRAMAN WEB LANJUT

JOBSHEET 5 MULTI LEVEL LOGIN

Oleh:

KINANTI PERMATA PUTRI NIM. 1841720022

TI-2A



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
MARET 2020**

A. Praktikum

1. Praktikum – Bagian 1: Membuat Login dengan Multi Level User

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

```
application > views > login > index.php > div.container
1  <div class="container">
2    <div class="row mt-3">
3      <div class="col-md-6">
4        <!-- https://getbootstrap.com/docs/4.4/components/card/ -->
5        <div class="card">
6          <div class="card-header">
7            Login
8          </div>
9          <div class="card-body">
10           <form action="" method="post">
11             <!-- https://getbootstrap.com/docs/4.4/components/forms/ -->
12             <div class="form" role="form" autocomplete="off" id="formlogin" novalidate="" method="POST">
13               <div class="form-group">
14                 <label for="username">Username</label>
15                 <input type="text" class="form-control form-control-lg rounded-0" id="username" name="username" required="">
16                 <div class="invalid-feedback">Username salah.</div>
17               </div>
18               <div class="form-group">
19                 <label for="password">Password</label>
20                 <input type="password" class="form-control form-control-lg rounded-0" id="password" name="password" required="" autocomplete="new-password">
21                 <div class="invalid-feedback">Password salah.</div>
22               </div>
23               <button type="submit" class="btn btn-primary float-right" id="btnLogin"> Login </button>
24             </form>
25           </div>
26        </div>
27      </div>
28    </div>
29  </div>
```

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

```
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 index()
8      {
9          $data['title'] = 'Login';
10         $this->load->view('template/header_login', $data);
11         $this->load->view('login/index');
12         $this->load->view('template/footer');
13     }
14
15 }
16
17 /* End of file Controllername.php */
18
19 <?>
```

- 3) Tambahkan sebuah code pada views/template dengan nama header_login (views/template/header_login)

```
application > views > template > header_login.php > html > head
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 <!-- Syle 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:

The screenshot shows a web browser window with the address bar displaying 'localhost/15BelajarCI/code-kelima/login'. The page content is a simple login form with a title 'Login'. It contains two input fields: 'Username' and 'Password'. Below the password field is a blue 'Login' button.

- 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 yaitu admin dan user

Server: localhost » Database: kuliah_ci » Table: user

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	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(10)	utf8mb4_general_ci		No	None			Change Drop More

+ Options

				id	username	password	level
<input type="checkbox"/>	Edit	Copy	Delete	1	admin	admin	admin
<input type="checkbox"/>	Edit	Copy	Delete	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
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          $this->db->select('username, password, level');
9          $this->db->from('user');
10         $this->db->where('username', $username);
11         $this->db->where('password', $password);
12         $this->db->limit(1);
13
14         $query=$this->db->get();
15         if($query->num_rows()==1){
16             return $query->result();
17         } else{
18             return false;
19         }
20     }
21
22 }
23
24 /* End of file ModelName.php */
25
26 >

```

Keterangan:

1. Line 7 sebuah function login dimana terdapat 2 parameter yaitu \$username dan \$password
2. Line 8 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 9 adalah nama tabel-nya (tabel user)
4. Line 10 adalah selama inputan user yang tersimpan pada \$username sama dengan username pada tabel user
5. Line 11 adalah selama inputan user yang tersimpan pada \$password sama dengan password pada tabel user
6. Line 12 adalah data yang diambil cuma 1
7. Line 14 adalah sebuah variabel baru dengan nama \$query yang digunakan untuk mengambil semua data

8. Line 15 – 19 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  <div class="container">
5      <div class="row mt-3">
6          <div class="col-md-6">

32 </div>
33 <!--/container-->
34 <?=
35 form_close();
36 ?>
```

Keterangan:

1. Line 2 artinya adalah form yang digunakan untuk membuka halaman proses login pada controller login
 2. Line 35 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

```
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('session');
14 }
```

Keterangan: Jika pada config/autoload sudah kalian setting maka kita tidak perlu menambahkan lagi seperti pada line 13. 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
- Keterangan:

```

24 public function proses_login()
25 {
26     $username = $this->input->post('uname1');
27     $password = $this->input->post('pwd1');
28
29     $ceklogin = $this->login_model->login($username,$password);
30
31     if($ceklogin){
32         foreach($ceklogin as $row);
33         $this->session->set_userdata('user', $row->username);
34         $this->session->set_userdata('level', $row->level);
35
36         if($this->session->set_userdata('level') == "admin"){
37             redirect('mahasiswa/index');
38         }
39         elseif ($this->session->set_userdata('level') == "user"){
40             redirect('mahasiswa/user');
41         }
42     }
43     else{
44         redirect('login/index', 'refresh');
45     }
46 }

```

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).

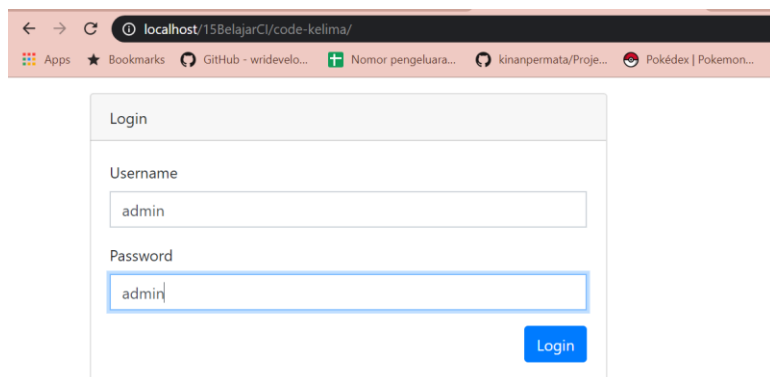
- 11) Pada line 25 dan 26 tambahkan htmlspecialchars agar user tidak dapat memasukkan char yang special.

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

- 12) Agar halaman login yang pertama kali di-load maka kita harus ubah konfigurasi pada application/config/routes

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

- 13) Hasilnya jika kita masuk menggunakan admin

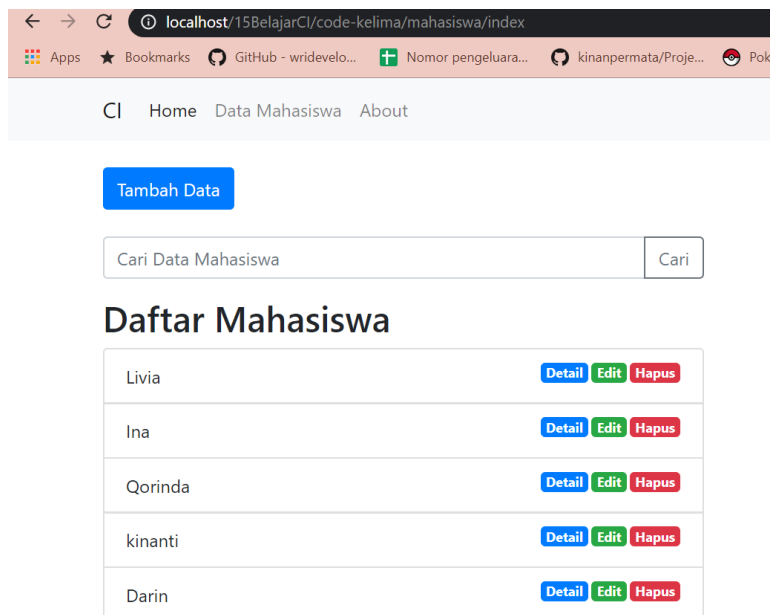


login

Username
admin

Password
admin

Login



CI Home Data Mahasiswa About

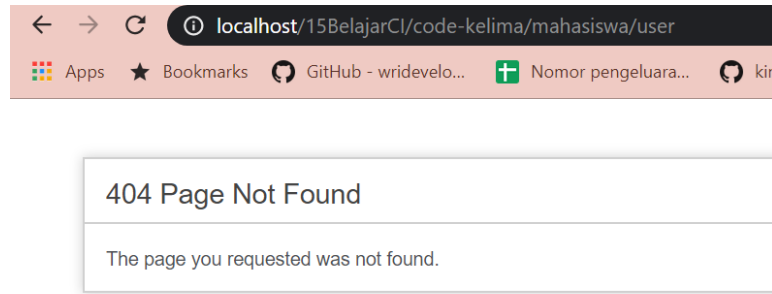
Tambah Data

Cari Data Mahasiswa Cari

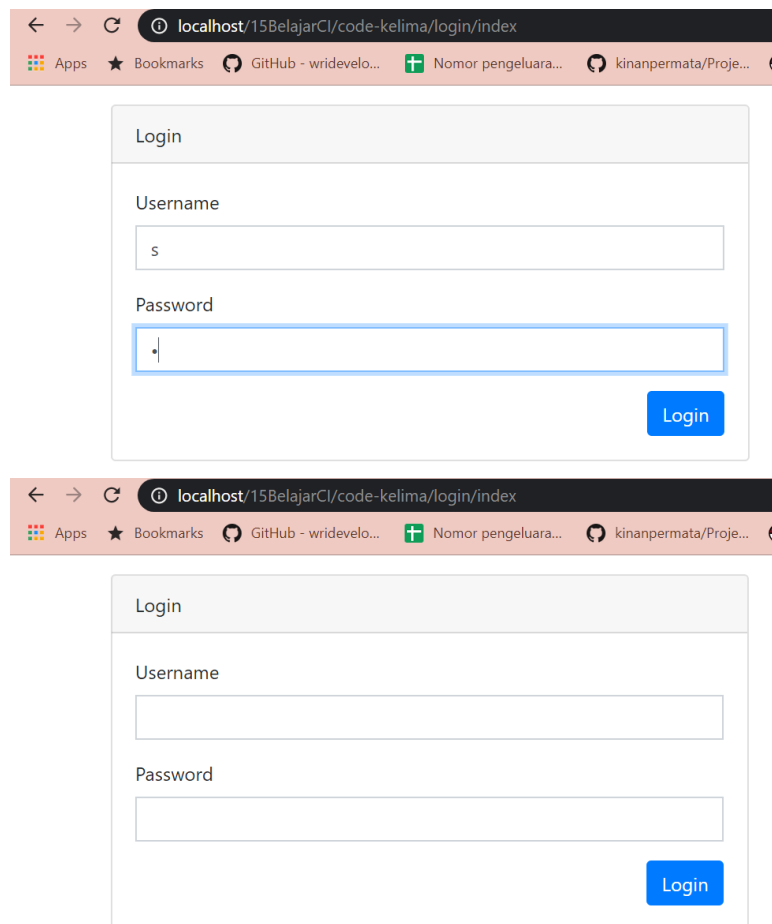
Daftar Mahasiswa

Livia	Detail Edit Hapus
Ina	Detail Edit Hapus
Qorinda	Detail Edit Hapus
kinanti	Detail Edit Hapus
Darin	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)



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


```

29         </div>
30         <div class="alert alert-info" role="alert">
31             <?php
32                 if(isset($pesan))
33                 {
34                     echo $pesan;
35                 }else{
36                     echo "Masukkan Username dan Password anda";
37                 }
38             ?>
39         </div>
40     </div>

```

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

```

40     } else{
41         $data['pesan']="Username dan Password anda salah";
42         $data['title'] = 'Login';
43         $this->load->view('template/header_login', $data);
44         $this->load->view('login/index', $data);
45         $this->load->view('template/footer', $data);
46     }

```

Sehingga hasilnya akan tampak seperti berikut

The screenshot shows a web browser window with the address bar displaying 'localhost/15BelajarCI/code-kelima/login/proses_login'. The browser's address bar also shows several tabs: 'Apps', 'Bookmarks', 'GitHub - wrdevelo...', 'Nomor pengeluaran...', 'kinanpermata/Proje...', and 'Poké'. The main content of the browser is a login form. The form has a title 'Login' at the top. Below the title are two input fields: 'Username' and 'Password'. To the right of the 'Password' field is a blue button labeled 'Login'. At the bottom of the form, there is a light blue alert box with the text 'Username dan Password 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:

```

36     } else if($this->session->userdata('level')=="user"){
37         redirect('mahasiswa/user');
38     }

```

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

```

36     } else if($this->session->userdata('level')=="user"){
37         redirect(['user']);
38     }

```

- 19) Setelah mengganti redirect, maka yang harus dilakukan adalah membuat sebuah view

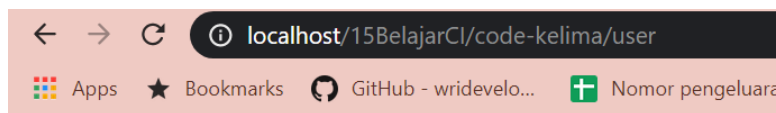
dengan nama user pada folder mahasiswa. sebelum kita membuat view dengan nama user kita terlebih dahulu membuat sebuah controller dengan nama user. Buat controller user dengan code sebagai berikut:

```
application > controllers > user.php > user > index
1  <?php
2
3
4  defined('BASEPATH') OR exit('No direct script access allowed');
5
6  class user extends CI_Controller {
7
8      public function __construct()
9      {
10         parent::__construct();
11         $this->load->helper('url');
12         $this->load->helper('form');
13         $this->load->model('login_model');
14     }
15
16     public function index()
17     {
18         $this->load->view('mahasiswa/user');
19     }
20
21 }
22
23 /* End of file Controllername.php */
24
25
26 ?>
```

20) Membuat tampilan pada view dengan nama user

```
application > views > mahasiswa > user.php > h1
1  <h1> ini Halaman user </h1>
```

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



ini Halaman user

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

```
32  <div class="navbar-nav">
33      <a class="nav-item nav-link active" href="= base_url(); ?&gt;Home"&gt;Home&lt;/a&gt;
34      &lt;a class="nav-item nav-link" href="<?= base_url(); ?&gt;mahasiswa"&gt;Data Mahasiswa&lt;/a&gt;
35      &lt;a class="nav-item nav-link" href="<?= base_url(); ?&gt;"&gt;About&lt;/a&gt;
36      &lt;a class="nav-item nav-link" href="<?= base_url(); ?&gt;login/logout"&gt;Logout&lt;/a&gt;
37  &lt;/div&gt;</pre
```

line 33 diubah menjadi seperti pada gambar

22) Tambahkan function logout pada controller login code-nya sebagai berikut

```
50     public function logout(){
51         $this->session->sess_destroy();
52         redirect('login', 'refresh');
53     }
```

23) Hasilnya jika kita klik logout maka akan muncul halaman login. Pada halaman user kita juga tambahkan code untuk proses logout sederhana

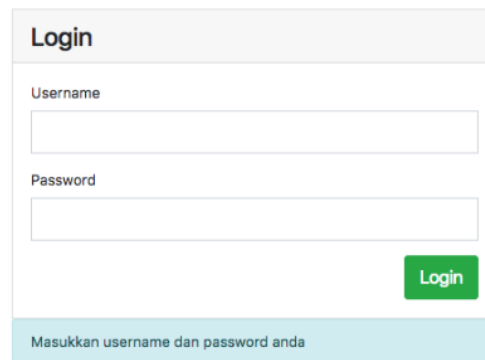
```
application > views > mahasiswa > user.php > a
1     <h1> ini Halaman user </h1>
2     <a href="<? base_url(). 'login/logout';?>">Logout </a>
```

24) 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:

localhost/12.codeigniter/code-keempat/login



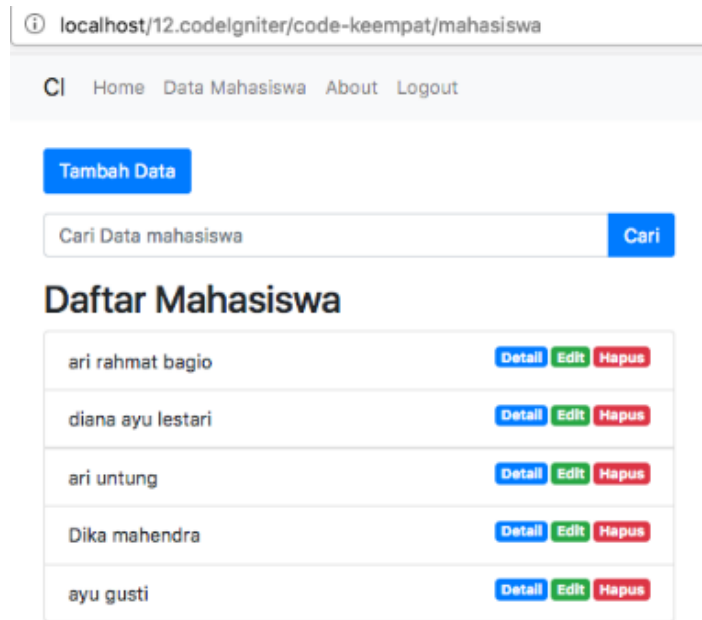
Login

Username

Password

Login

Masukkan username dan password anda



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)

```
8      public function __construct()
9      {
10         parent::__construct();
11         $this->load->model('mahasiswa_model');
12         $this->load->helper('url', 'form');
13         $this->load->library('form_validation');
14
15         if($this->session->userdata('level')!="admin"){
16             redirect('login', 'refresh');
17         }
18     }
```

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)

```
8      public function __construct()
9      {
10         parent::__construct();
11         $this->load->helper('url');
12         $this->load->helper('form');
13         $this->load->model('login_model');
14
15         if($this->session->userdata('level')!="user"){
16             redirect('login', 'refresh');
17         }
18     }
19
20     public function index()
21     {
22         $this->load->view('mahasiswa/user');
23     }
```

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

```
7      public function index($name='')
8      {
9         $data['title']='Home';
10        $data['name']=$name;
11        $this->load->view('template/header', $data);
12        $this->load->view('home/index', $data);
13        $this->load->view('template/footer');
14
15        if($this->session->userdata('level')!="admin"){
16            redirect('login', 'refresh');
17        }
18    }
```

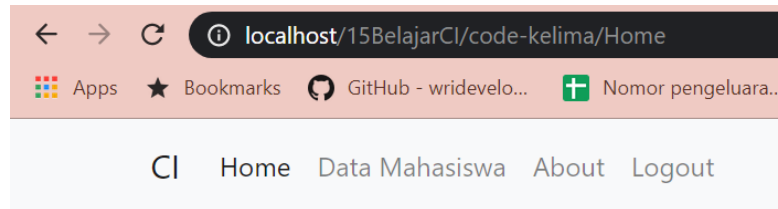
27) Tambahkan code berikut agar pada halaman admin tampil level usernya (views/home/index)

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

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

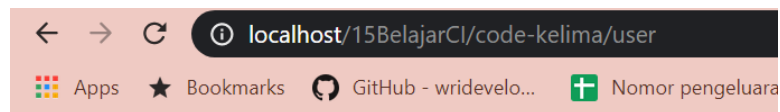
```
5 class Home extends CI_Controller {
6
7     public function index($name='')
8     {
9         $data['title']='Home';
10        //$data['name']=$name;
11        $this->load->view('template/header', $data);
12        $this->load->view('home/index', $data);
13        $this->load->view('template/footer');
14        $this->load->library(['session']);
15
16
17        if($this->session->userdata('level')!="admin"){
18            redirect('login', 'refresh');
19        }
20    }
```

Sehingga hasilnya adalah sebagai berikut:



29) Lakukan hal yang sama pada user

```
application > views > mahasiswa > user.php > h1
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](#)