

LAPORAN PRAKTIKUM
PEMROGRAMAN WEB LANJUT

Jobsheet-5
Multi Level User

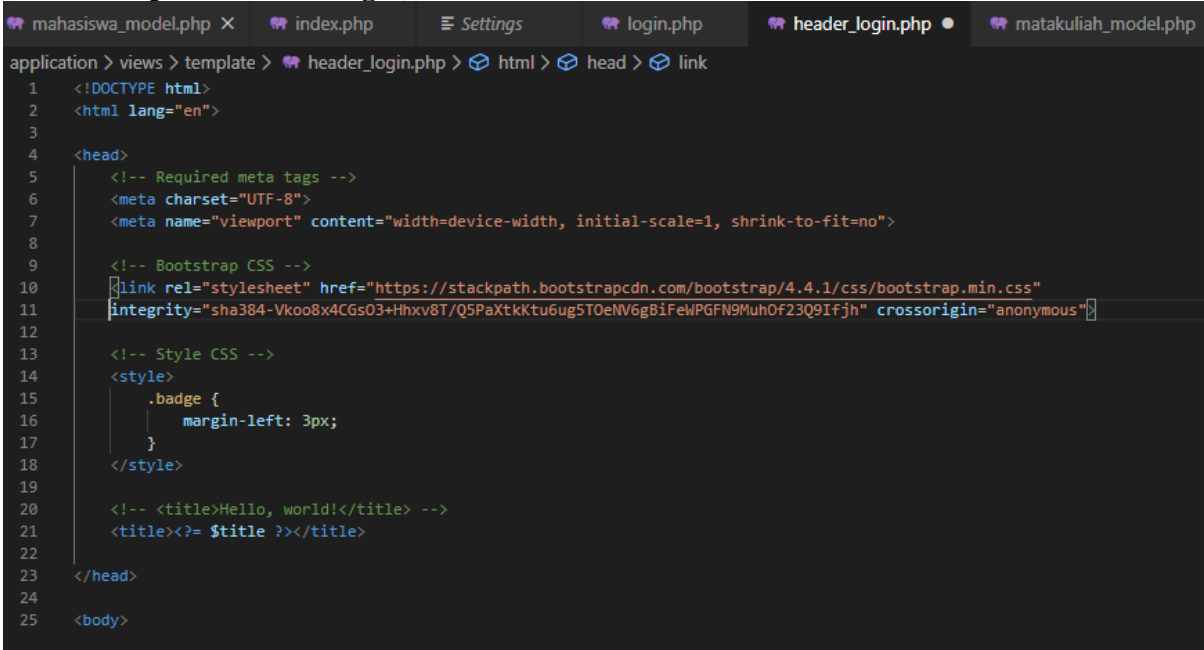
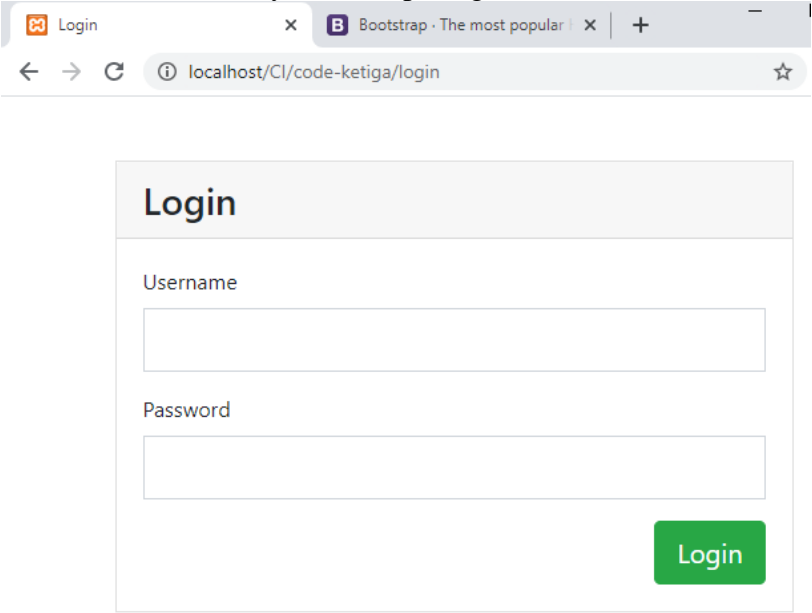


NAMA : DINA RISKY ALIN SAPUTRI
NIM : 1841720016
KELAS : 2B

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020

Praktikum – Bagian 1: Membuat Login dengan Multi Level User

Langka h	Keterangan
1	<p>Buat tampilan login (views/login/index.php), dan tambahkan code berikut pada views/login/index.php:</p> <pre data-bbox="337 428 1539 1234"><?= 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 to 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> </div> </div> </div> </div> </div> <?> form_close(); ?></pre>
2	<p>Buatlah sebuah <i>controller</i> dengan nama <i>login</i> (controller/login)</p> <pre data-bbox="337 1310 1192 1875">mahasiswa_model.php index.php Settings login.php X application > controllers > login.php > login > index 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 /* End of file login.php */ 18</pre>

3	<p>Tambahkan sebuah <i>code</i> pada views/template dengan nama header_login (views/template/header_login)</p>  <pre> 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/4.4.1/css/bootstrap.min.css" 11 integrity="sha384-Vkoo8x4CGs03+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous"> 12 13 <!-- Style CSS --> 14 <style> 15 .badge { 16 margin-left: 3px; 17 } 18 </style> 19 20 <!-- <title>Hello, world!</title> --> 21 <title><?= \$title ?></title> 22 23 </head> 24 25 <body> </pre>
4	<p>Ketika dijalankan di browser, hasilnya akan seperti gambar berikut:</p> 
5	<p>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.</p>
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>

Server: 127.0.0.1 » 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)	latin1_swedish_ci		No	None			Change Drop More
3	password	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
4	level	varchar(10)	latin1_swedish_ci		No	None			Change Drop More

Showing rows 0 - 1 (2 total, Query took 0.0012 seconds.)

SELECT * FROM `user`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP]

Show all | Number of rows: 25

+ Options

	id	username	password	level
1	Admin	admin	admin	admin
2	user	user	user	user

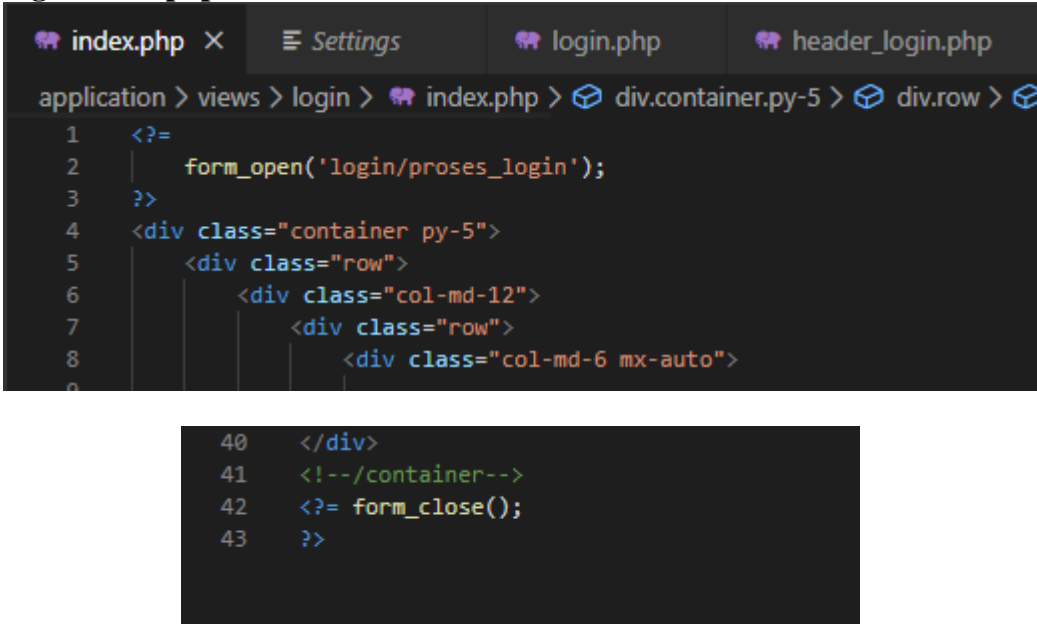
7

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.php login_model.php X
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     public 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        $this->db->where('password', $password);
16        $this->db->limit(1); //maksudnya data yg diambil cuma 1
17
18        $query = $this->db->get();
19        if ($query->num_rows() == 1) //jika data ditemukan
20        {
21            return $query->result();
22        } else {
23            return false;
24        }
25    }
26 }
27
28 /* End of file login_model.php */
29
```

Keterangan :

1. Line 6 sebuah function *login* dimana terdapat 2 parameter yaitu *\$username* dan *\$password*

	<ol style="list-style-type: none"> 2. <i>Line 11</i> untuk melakukan <i>select</i> mana saja yang akan digunakan. Jika kita lihat terdapat 3 parameter yaitu <i>username</i>, <i>password</i>, dan <i>level</i> -> untuk penulisan ini semua harus sesuai dengan nama <i>field</i> yang ada pada <i>tabel</i>-nya (tabel user). 3. <i>Line 12</i> adalah nama <i>tabel</i>-nya (tabel user) 4. <i>Line 13</i> adalah selama inputan <i>user</i> yang tersimpan pada <i>\$username</i> sama dengan <i>username</i> pada <i>tabel user</i> 5. <i>Line 14</i> adalah selama inputan <i>user</i> yang tersimpan pada <i>\$password</i> sama dengan <i>password</i> pada <i>tabel user</i> 6. <i>Line 15</i> adalah data yang diambil cuma 1 7. <i>Line 16</i> adalah sebuah variabel baru dengan nama <i>\$query</i> yang digunakan untuk mengambil semua data 8. <i>Line 18 – 24</i> adalah sebuah pengkondisian dimana jika <i>\$query</i> menemukan 1 <i>row</i> pada <i>tabel</i> atau dengan kata lain data berhasil ditemukan maka nilai pengembalian <i>query</i>-nya akan mencetak isi dari <i>\$query(\$query->result())</i> dan jika tidak menemukan maka nilai kembaliannya kita set menjadi <i>false</i>
8	<p>Langkah berikutnya tambahkan form_open dan form_close pada <i>code</i> 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 40 </div> 41 <!--/container--> 42 <?= 43 form_close(); 44 ?> </pre> <p>Keterangan:</p> <ol style="list-style-type: none"> 1. <i>Line 2</i> artinya adalah <i>form</i> yang digunakan untuk membuka halaman proses <i>login</i> pada <i>controller login</i> 2. <i>Line 44</i> artinya adalah untuk menutup <i>form</i> dikarenakan tadi terdapat perintah <i>form_open</i>
9	<p>Buatlah sebuah <i>function</i> dengan nama proses_login pada <i>controller login</i>. Tapi sebelum kita membuat <i>function proses_login</i> maka terlebih dahulu kita tambahkan sebuah <i>function construct</i></p>

```

k.php X login.php X header_login.php login_model.php
application > controllers > login.php > login > __construct
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      {
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     }
15
16     public function index()
17     {
18         $data['title'] = 'Login';
19         $this->load->view('template/header_login', $data);
20         $this->load->view('login/index');
21         $this->load->view('template/footer');
22     }
23 }
24
25 /* End of file login.php */
26

```

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**

```

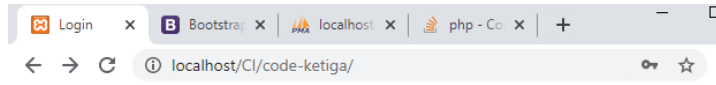
3      public function proses_login()
4      {
5          $username = $this->input->post('uname1');
6          $password = $this->input->post('pwd1');
7
8          $ceklogin = $this->login_model->login($username, $password);
9
10         if ($ceklogin) {
11             foreach ($ceklogin as $row) {
12                 //kita set userdata pada session dengan nama user dan isi uername kita isikan username yang ada pada $row
13                 $this->session->set_userdata('user', $row->username);
14                 $this->session->set_userdata('level', $row->level);
15
16                 //var_dump($row);
17                 if ($this->session->userdata('level') == "admin") {
18                     redirect('mahasiswa/index');
19                 } elseif ($this->session->userdata('level') == "user") {
20                     redirect('mahasiswa/user');
21                 }
22             }
23         } else {
24             redirect('login/index', 'refresh');
25         }
26     }
27 }
28
29 /* End of file login.php */
30

```

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

	<p>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>).</p> <ol style="list-style-type: none"> 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</i> dan <i>26</i>, yang nanti akan dikirimkan ke <i>function</i> login pada <i>login_model</i>. 3. <i>Line 30-51</i> adalah sebuah pengkondisian jika \$cek_login bernilai benar maka cek_login akan dipecah menjadi beberapa <i>row</i> (\$cek_login as \$row). setelah itu kita set <i>userdata</i> menggunakan <i>session</i> dengan nama variabel barunya adalah <i>user</i> sedangkan isinya adalah <i>username</i> (<i>line 34</i>) dan kita juga set <i>userdata</i> menggunakan <i>session</i> dengan nama variabelnya <i>level</i> (<i>line 35</i>) sedangkan isinya adalah <i>level</i>. Setelah kita pecah menjadi beberapa <i>row</i> dan mempunyai isi yang berbeda-beda maka langkah berikutnya kita buat pengkondisian jika <i>session userdata('level')</i> berisi <i>admin</i> maka akan me-<i>redirect</i> pada halaman <i>mahasiswa/index</i>(<i>line 38-41</i>). Apabila <i>levelnya</i> berupa <i>user</i> maka akan meredirect ke halaman <i>user</i> (<i>line 42-45</i>). Jika \$cek_login bernilai salah maka akan me-<i>redirect</i> ke halaman <i>login</i> kembali (<i>line 47 - 51</i>).
11	<p>Pada <i>line 25</i> dan <i>26</i> tambahkan htmlspecialchars agar <i>user</i> tidak dapat memasukkan <i>char</i> yang special.</p> <pre> public function proses_login() { \$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'; \$route['404_override'] = ''; \$route['translate_uri_dashes'] = FALSE; </pre>
13	<p>Hasilnya jika kita masuk menggunakan admin</p>



Login

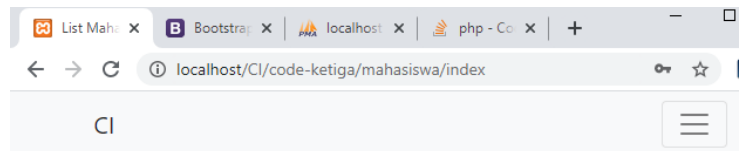
Username

admin

Password

.....

Login



Tambah Data

Cari Data mahasiswa

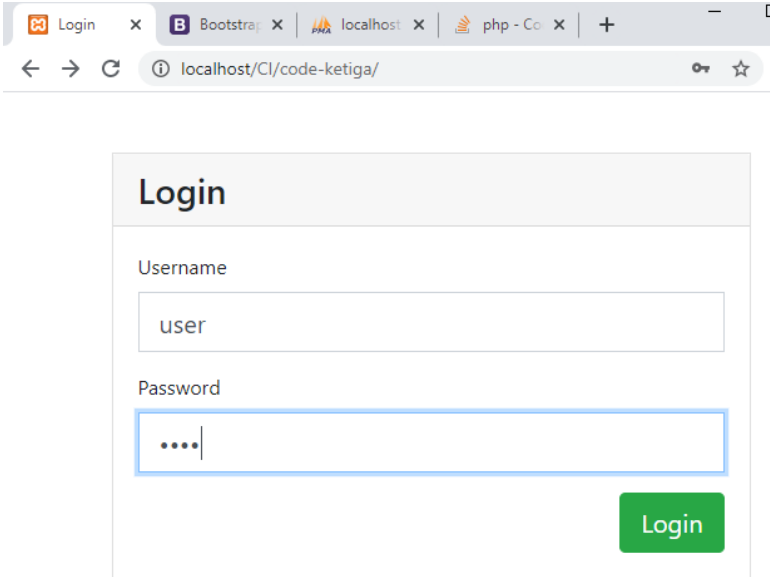
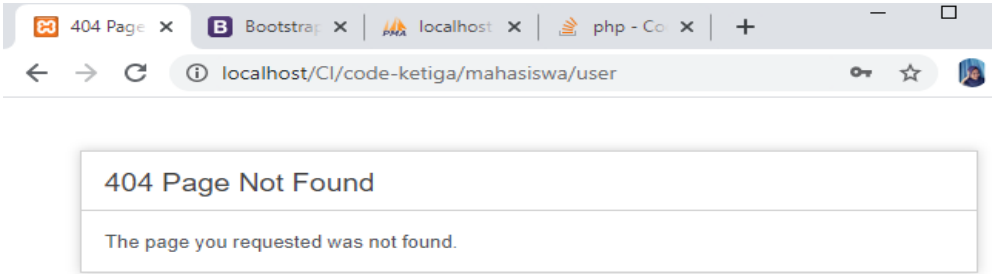
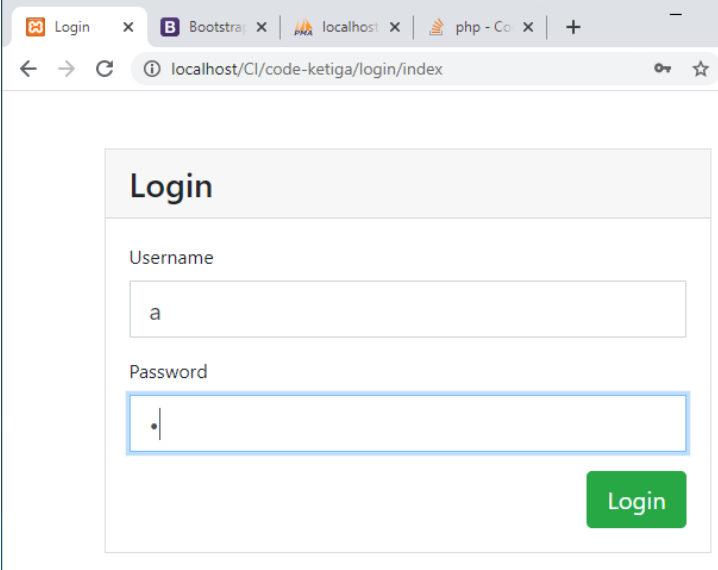
Cari

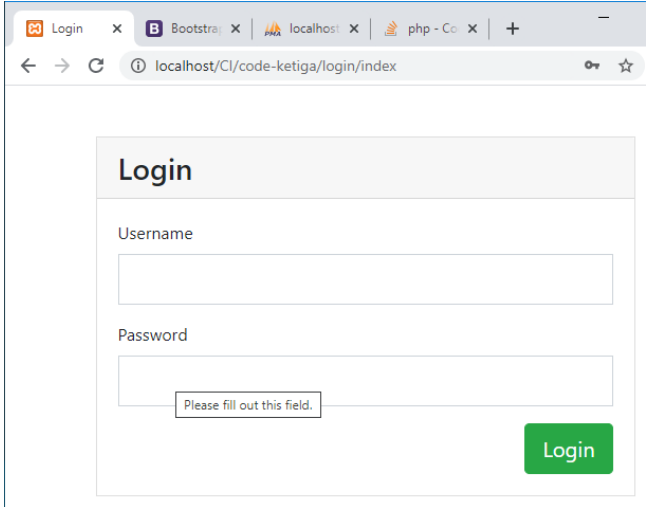
Daftar Mahasiswa

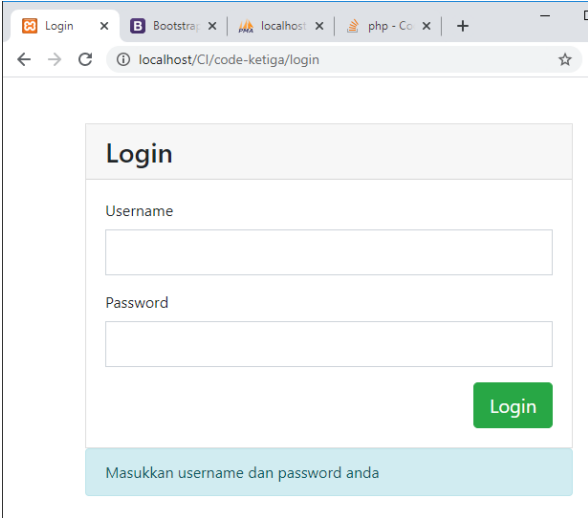
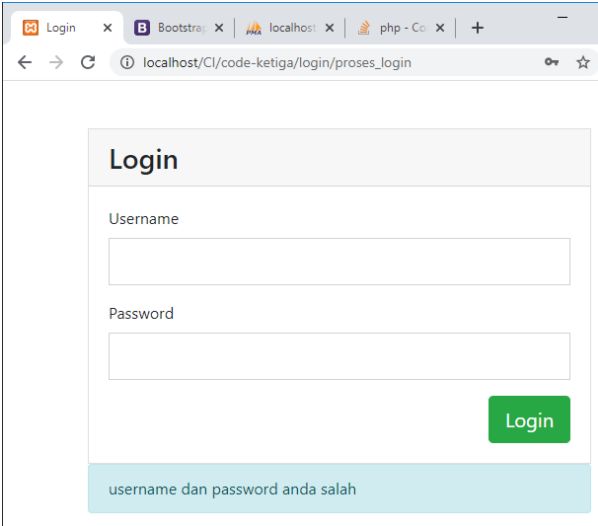
soni atalia	<div>Detail Edit Hapus</div>
Agit Ari	<div>Detail Edit Hapus</div>
Silvi	<div>Detail Edit Hapus</div>
Agung	<div>Detail Edit Hapus</div>
Ani	<div>Detail Edit Hapus</div>

14

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

	 
15	<p>Hasilnya jika kita salah memasukkan <i>password</i> dan <i>username</i> (jika salah maka <i>redirect</i> ke halaman login tanpa ada pemberitahuan jika <i>username</i> dan <i>password</i> salah)</p> 

	
16	<p>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</p> <pre data-bbox="391 842 1472 1163"><!--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 else-nya</p> <pre data-bbox="469 1199 1390 1499">} 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>Sehingga hasilnya akan tampak seperti berikut</p>

	 
17	Jika kita lihat pada percobaan sebelumnya, pada saat kita <i>login</i> dengan menggunakan level <i>user</i> maka halaman tidak akan tampil atau <i>page not found</i> . Untuk kita akan membuat halaman <i>user</i> sederhana
18	<p>Jika dilihat pada function proses_login terdapat <i>code</i> sebagai berikut:</p> <pre> } elseif (\$this->session->userdata('level') == "user") { redirect('mahasiswa/index'); } </pre> <p>Untuk lebih mudahnya pada line 44 ganti <i>redirect</i> ('mahasiswa/index') menjadi <i>redirect</i>('user');</p> <pre> } elseif (\$this->session->userdata('level') == "user") { redirect('user'); } </pre>
19	Setelah mengganti redirect , maka yang harus dilakukan adalah membuat sebuah view dengan nama <i>user</i> 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:

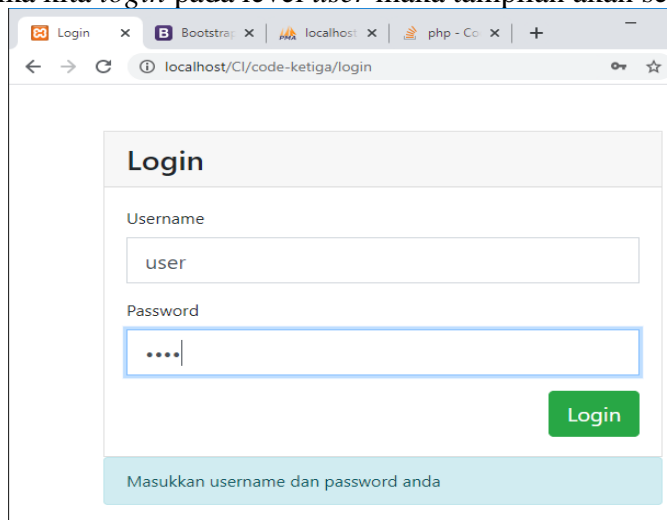
```
login.php  user.php  login_model.php  index.php
application > controllers > user.php > user
1  <?php
2
3  defined('BASEPATH') or exit('No direct script access allowed');
4
5  class user extends CI_Controller
6  {
7      public function __construct()
8      {
9          parent::__construct();
10 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

20

Membuat tampilan pada *view* dengan nama *user*

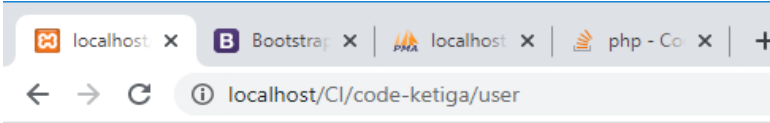
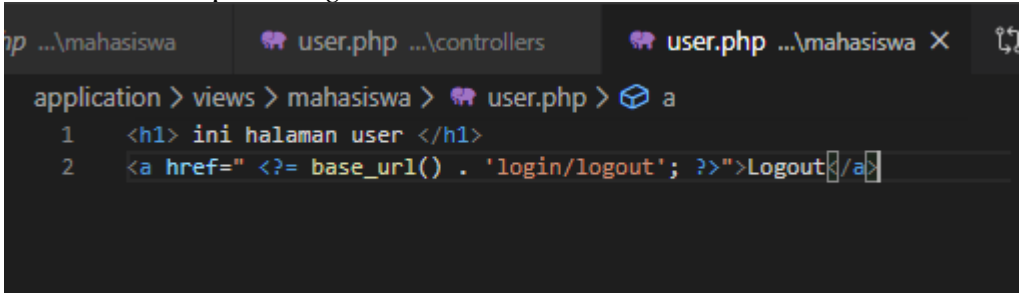
```
user.php ...mahasiswa  login_model.php
application > views > mahasiswa > user.php > h1
1  <h1> ini halaman user </h1>
```

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

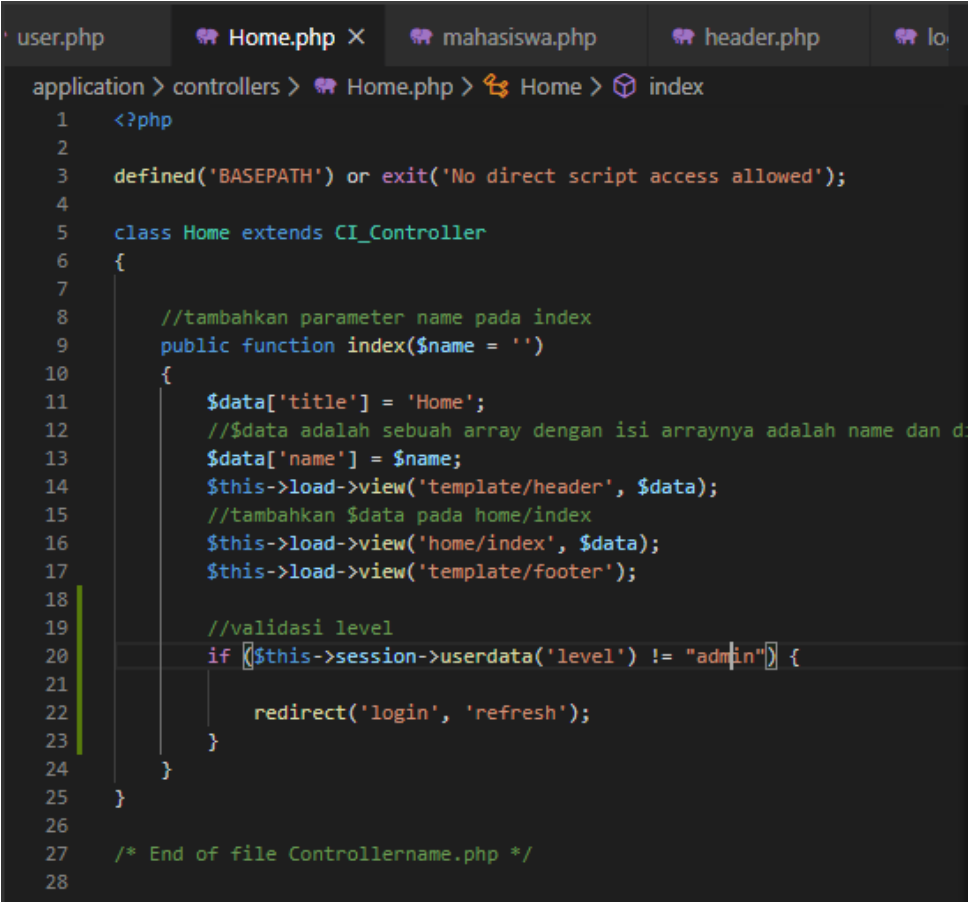
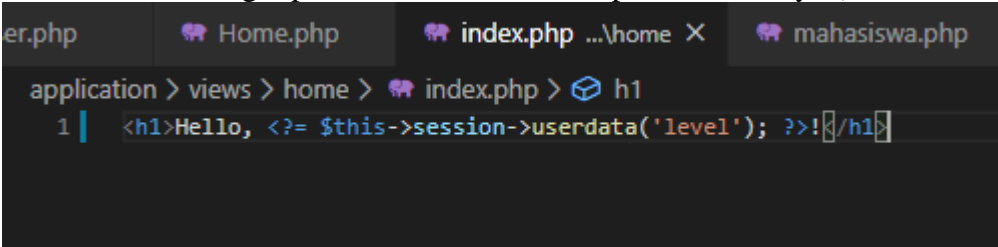


The screenshot shows a web browser window with the address bar displaying 'localhost/CI/code-ketiga/login'. The page contains a login form with the following elements:

- Login** (Section Header)
- Username** label above a text input field containing the text 'user'.
- Password** label above a password input field containing four dots '....'.
- A green **Login** button.
- A light blue footer area with the text 'Masukkan username dan password anda'.

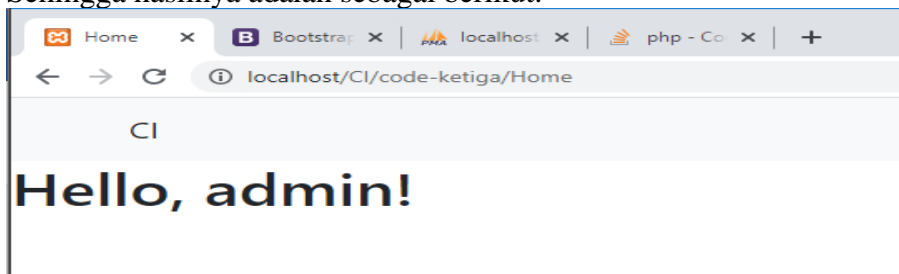
	 <p>ini halaman user</p>
21	<p>Setelah berhasil membuat halaman <i>user</i>, maka langkah berikutnya kita membuat menu <i>logout</i>. Tambahkan <i>code</i> berikut pada views/template/header line 36</p> <pre> <div class="collapse navbar-collapse" id="navbarNavAltMarkup"> <div class="navbar-nav"> Home mahasiswa About login/logout <!-- Disabled --> </div> </div> </pre> <p>line 33 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> 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>Menambahkan beberapa <i>code</i> pada <i>controller/mahasiswa</i> seperti pada gambar dibawah ini (line 15-19)</p>

	<pre> //fungsi yang akan dijalankan saat classnya diinstansiasi public function __construct() { //digunakan untuk menjalankan fungsi constrauct pada class parrent(parent::__construct(); \$this->load->model('mahasiswa_model'); \$this->load->library('form_validation'); //validasi level if (\$this->session->userdata('level') != "admin") { redirect('login', 'refresh'); } } </pre> <p>Keterangan :</p> <p>Line 15-17 adalah pengkondisian dimana jika levelnya bukan admin maka halaman akan me-<i>redirect</i> ke halaman <i>login</i></p>
25	<p>Lakukan hal yang sama pada <i>user (controller user)</i></p>  <pre> user.php x mahasiswa.php header.php login_model.php application > controllers > user.php > user > __construct 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 { 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 15 //validasi level 16 if (\$this->session->userdata('level') != "user") { 17 18 redirect('login', 'refresh'); 19 } 20 } 21 </pre>
26	<p>Lakukan hal yang sama pada <i>home (controller home)</i></p>

	 <pre> user.php Home.php X mahasiswa.php header.php lo 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 8 //tambahkan parameter name pada index 9 public function index(\$name = '') 10 { 11 \$data['title'] = 'Home'; 12 //\$data adalah sebuah array dengan isi arraynya adalah name dan d 13 \$data['name'] = \$name; 14 \$this->load->view('template/header', \$data); 15 //tambahkan \$data pada home/index 16 \$this->load->view('home/index', \$data); 17 \$this->load->view('template/footer'); 18 19 //validasi level 20 if (\$this->session->userdata('level') != "admin") { 21 22 redirect('login', 'refresh'); 23 } 24 } 25 } 26 27 /* End of file Controllername.php */ 28 </pre>
27	<p>Tambahkan <i>code</i> berikut agar pada halaman admin tampil level <i>user</i>nya (views/home/index)</p>  <pre> er.php Home.php index.php ...\home X mahasiswa.php application > views > home > index.php > h1 1 <h1>Hello, <?= \$this->session->userdata('level'); ?>!</h1> </pre>
28	<p>Comment beberapa <i>code</i> pada <i>controller home</i> (line 12 dan hapus parameter name pada index)</p>

```
ser.php Home.php X index.php ...\home mahasiswa.php
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
8     //tambahkan parameter name pada index
9     public function index()
10    {
11        $data['title'] = 'Home';
12        //$data adalah sebuah array dengan isi arraynya adalah name dan di
13        // $data['name'] = $name;
14        $this->load->view('template/header', $data);
15        //tambahkan $data pada home/index
16        $this->load->view('home/index', $data);
17        $this->load->view('template/footer');
18        $this->load->library('session');
19
20        //validasi level
21        if ($this->session->userdata('level') != "admin") {
22
23            redirect('login', 'refresh');
24        }
25    }
26 }
27
28 /* End of file Controllername.php */
29
```

Sehingga hasilnya adalah sebagai berikut:



29

Lakukan hal yang sama pada *user*


```
rollers user.php ...\mahasiswa X Home.php index.php ...\hor
application > views > mahasiswa > user.php > a
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>
```

