**LAPORAN PRAKTIKUM PEMROGRAMAN**

**WEBSITE LANJUT**

**TUGAS PRAKTIKUM**

****

**Oleh :**

**M. WAHYU PRASETYO / NIM : 1841720177**

**KELAS TI 2B**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2020**

|  |  |
| --- | --- |
| 1 | Buat tampilan login (views/login/index.php), dan tambahkan code berikut pada views/login/index.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" no validate="" 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="text" 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>  </div> |
| 2 | Buatlah sebuah ***controller*** dengan nama ***login*** (**controller/login**)  <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class login extends CI\_Controller{      public function Index()      {          # code...          $data['title']='Login';          $this->load->view('template/header\_login',$data);          $this->load->view('login/index');          $this->load->view('template/footer');        }  }  ?> |
| 3 | Tambahkan sebuah *code* pada **views**/**template** dengan nama **header**\_**login (views/template/header\_login)**  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">      <style>          .badge{              margin-left: 3px;          }      </style>      <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* |

|  |  |
| --- | --- |
| 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)**  <?php        defined('BASEPATH') OR exit('No direct script access allowed');        class login\_model extends CI\_Model {            public function login($username, $password)          {              # code...              $this->db->select('username,password,level');              $this->db->form('user');              //pengecekan username pada database              $this->db->where('username',$username);              $this->db->where('password', $password);              //mengambil 1 data saja              $this->db->limit(1);                $query = $this->db->get();                  if($query->num\_rows()==1){                      return $query->result();                  }else{                      return false;                  }          }        }        /\* End of file login\_model.php \*/    ?> |
| 8 | Langkah berikutnya tambahkan **form**\_**open** dan **form**\_**close** pada *code* **views/login/index.php** |
| 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**  <?php  defined('BASEPATH') OR exit('No direct script access allowed');  class login extends CI\_Controller{      public function \_\_construct()      {          # code...          parent::\_\_construct();          $this->load->helper('url');          $this->load->helper('form');          $this->load->model('login\_model');        }      public function Index()      {          # code...          $data['title']='Login';          $this->load->view('template/header\_login',$data);          $this->load->view('login/index');          $this->load->view('template/footer');        }  }  ?> |
| 10 | Langkah berikutnya menambahkan **function proses\_login** pada **controller login**  public function proses\_login()      {          # code...          $username = $this->input->post('uname1');          $password = $this->input->post('pwd1');          $ceklogin = $this->login\_model->login($username, $password);          if ($ceklogin) {              # code...              foreach ($ceklogin as $row);              $this->session->userdata('user',$row->username);              $this->session->userdata('level',$row->level);              if ($this->session->userdata('level')=="admin"){                  redirect('mahasiswa/index');              }elseif($this->session->userdata('level')=="user"){                  redirect('mahasiswa/user');              }else{                  redirect('login/index','refresh');              }          }      } |
| 11 | Pada *line* 25 dan 26 tambahkan **htmlspesialchar** 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 konfigurasinya pada **application/config/routes**  $route['default\_controller'] = 'login';  $route['404\_override'] = '';  $route['translate\_uri\_dashes'] = FALSE; |
| 13 | Hasilnya jika kita masuk menggunakan admin |

|  |  |
| --- | --- |
| 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 c*ode* berikut pada **login/index**  <!-- cek pesan -->  <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);    $this->load->view('template/footer');  }  Sehingga hasilnya akan tampak seperti berikut |
| 17 | Jika kita lihat pada percobaan sebelumnya, pada saat kita *login* dengan menggunakan level *user* maka halaman tidak akan tampil atau *page not found*. Untuk kita akan membuat halaman *user* sederhana |
| 18 | Jika dilihat pada ***function* proses**\_**login** terdapat *code* sebagai berikut:  elseif($this->session->userdata('level')=="user"){      redirect('mahasiswa/user');  }  **Untuk lebih mudahnya pada line 44 ganti *redirect* (‘mahasiswa/index’) menjadi redirect(‘user’);**  elseif($this->session->userdata('level')=="user"){      redirect('user');  } |
| 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        defined('BASEPATH') OR exit('No direct script access allowed');        class user extends CI\_Controller {            public function \_\_construct()          {              parent::\_\_construct();              $this->load->helper('url');              $this->load->helper('form');              $this->load->model('login\_model');          }            public function index()          {              $this->load->view('mahasiswa/user');            }        }        /\* End of file user.php \*/    ?> |
| 20 | Membuat tampilan pada *view* dengan nama *user*  <h1> Ini halaman User </h1>  Sehingga hasilnya jika kita *login* pada level *user* maka tampilan akan seperti gambar berikut |
| 21 | 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="navbarNav">      <div class="navbar-nav">        <a class="nav-item nav-link" href="<?= base\_url(); ?>">Home <span class="sr-only">[current]</span></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>      </div>    </div> |
| 22 | Tambahkan ***function logout* pada *controller login*** *code*-nya sebagai berikut  public function logout()      {          # code...          $this->session->sess\_destroy();          redirect('login','refresh');      } |
| 23 | Hasilnya jika kita klik *logout* maka akan muncul halaman *login.* Pada halaman *user* kita juga tambahkan c*ode* untuk proses *logout* sederhana  <h1> Ini halaman User </h1>  <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:    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()          {              //untuk menjalankan fungsi class 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');              }          } |
| 25 | Lakukan hal yang sama pada ***user*** (***controller* user**)      class user extends CI\_Controller {            public function \_\_construct()          {              parent::\_\_construct();              $this->load->helper('url');              $this->load->helper('form');              $this->load->model('login\_model');                // validasi level              if($this->session->userdata('level')!="user"){                 redirect('login','refresh');              }          } |
| 26 | Lakukan hal yang sama pada ***home*** (***controller* home**)          public function index($name='')          {              $data['title']='Home';              //echo "Selamat Datang di Halaman HOME";              //data adalah array yang disi dengan nama              //$data['name']=$name;              $this->load->view('template/header',$data);              //tambah data pada home/index              $this->load->view('home/index',$data);              $this->load->view('template/footer');              // validasi level              if($this->session->userdata('level')!="admin"){                  redirect('login','refresh');               }          } |
| 27 | Tambahkan *code* berikut agar pada halaman admin tampil level *user*nya (**views/home/index**)  <h1>Hello, <?= $this->session->userdata('level'); ?>!</h1> |
| 28 | ***Comment* beberapa *code* pada *controller home* (*line* 12 dan hapus parameter name pada index)**          public function index($name='')          {              $data['title']='Home';              //echo "Selamat Datang di Halaman HOME";              //data adalah array yang disi dengan nama              //$data['name']=$name;              $this->load->view('template/header',$data);              //tambah data pada home/index              $this->load->view('home/index',$data);              $this->load->view('template/footer');              // validasi level              if($this->session->userdata('level')!="admin"){                  redirect('login','refresh');               }          }  Sehingga hasilnya adalah sebagai berikut: |
| 29 | Lakukan hal yang sama pada *user*  <h1> Ini halaman User </h1>  <a href="<?= base\_url().'login/logout';?>">Logout</a> |