**BAB V**

**IMPLEMENTASI *SOURCE CODE***

Pada bagian bab ini menjelaskan sumber kode-kode yang berkaitan dengan aplikasi yang dibangun dari tahap penerapan estetika bagian *template* serta *content* yang diperlukan pada aplikasi yang dibangun :

1. **Template**

Adalah pola *layout* dari sebuah *website* yang ingin didesain dan didisiapkan dengan berbagai fitur. Pengguna bisa merubah serta mengembangkan kembali dari desain yang di rencanakan. Dari *template* tersebut terdapat stuktur yang digunakan yaitu *header*, *body* dan *footer*.

1. **Header**

|  |
| --- |
| 1. <!DOCTYPE html> 2. <html> 3. <head> 4. <meta charset="utf-8"> 5. <meta http-equiv="X-UA-Compatible" content="IE=edge"> 6. <title>AdminLTE 2 | Dashboard</title> 7. <!-- Tell the browser to be responsive to screen width --> 8. <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport"> 9. <!-- Bootstrap 3.3.7 --> 10. <link rel="stylesheet" href="bower\_components/bootstrap/dist/css/bootstrap.min.css"> 11. <!-- Font Awesome --> 12. <link rel="stylesheet" href="bower\_components/font-awesome/css/font-awesome.min.css"> 13. <!-- Ionicons --> 14. <link rel="stylesheet" href="bower\_components/Ionicons/css/ionicons.min.css"> 15. <!-- Theme style --> 16. <link rel="stylesheet" href="dist/css/AdminLTE.min.css"> 17. <!-- AdminLTE Skins. Choose a skin from the css/skins 18. folder instead of downloading all of them to reduce the load. --> 19. <link rel="stylesheet" href="dist/css/skins/\_all-skins.min.css"> 20. <!-- Morris chart --> 21. <link rel="stylesheet" href="bower\_components/morris.js/morris.css"> 22. <!-- jvectormap --> 23. <link rel="stylesheet" href="bower\_components/jvectormap/jquery-jvectormap.css"> 24. <!-- Date Picker --> 25. <link rel="stylesheet" href="bower\_components/bootstrap-datepicker/dist/css/bootstrap-datepicker.min.css"> 26. <!-- Daterange picker --> 27. <link rel="stylesheet" href="bower\_components/bootstrap-daterangepicker/daterangepicker.css"> 28. <!-- bootstrap wysihtml5 - text editor --> 29. <link rel="stylesheet" href="plugins/bootstrap-wysihtml5/bootstrap3-wysihtml5.min.css"> 31. <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries --> 32. <!-- WARNING: Respond.js doesn't work if you view the page via file:// --> 33. <!--[if lt IE 9]> 34. <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script> 35. <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script> 36. <![endif]--> 38. <!-- Google Font --> 39. <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic"> 40. </head> |

1. **Footer**

|  |
| --- |
| 1. <!-- jQuery 3 --> 2. <script src="bower\_components/jquery/dist/jquery.min.js"></script> 3. <!-- jQuery UI 1.11.4 --> 4. <script src="bower\_components/jquery-ui/jquery-ui.min.js"></script> 5. <!-- Resolve conflict in jQuery UI tooltip with Bootstrap tooltip --> 6. <script> 7. $.widget.bridge('uibutton', $.ui.button); 8. </script> 9. <!-- Bootstrap 3.3.7 --> 10. <script src="bower\_components/bootstrap/dist/js/bootstrap.min.js"></script> 11. <!-- Morris.js charts --> 12. <script src="bower\_components/raphael/raphael.min.js"></script> 13. <script src="bower\_components/morris.js/morris.min.js"></script> 14. <!-- Sparkline --> 15. <script src="bower\_components/jquery-sparkline/dist/jquery.sparkline.min.js"></script> 16. <!-- jvectormap --> 17. <script src="plugins/jvectormap/jquery-jvectormap-1.2.2.min.js"></script> 18. <script src="plugins/jvectormap/jquery-jvectormap-world-mill-en.js"></script> 19. <!-- jQuery Knob Chart --> 20. <script src="bower\_components/jquery-knob/dist/jquery.knob.min.js"></script> 21. <!-- daterangepicker --> 22. <script src="bower\_components/moment/min/moment.min.js"></script> 23. <script src="bower\_components/bootstrap-daterangepicker/daterangepicker.js"></script> 24. <!-- datepicker --> 25. <script src="bower\_components/bootstrap-datepicker/dist/js/bootstrap-datepicker.min.js"></script> 26. <!-- Bootstrap WYSIHTML5 --> 27. <script src="plugins/bootstrap-wysihtml5/bootstrap3-wysihtml5.all.min.js"></script> 28. <!-- Slimscroll --> 29. <script src="bower\_components/jquery-slimscroll/jquery.slimscroll.min.js"></script> 30. <!-- FastClick --> 31. <script src="bower\_components/fastclick/lib/fastclick.js"></script> 32. <!-- AdminLTE App --> 33. <script src="dist/js/adminlte.min.js"></script> 34. <!-- AdminLTE dashboard demo (This is only for demo purposes) --> 35. <script src="dist/js/pages/dashboard.js"></script> 36. <!-- AdminLTE for demo purposes --> 37. <script src="dist/js/demo.js"></script> |

1. **Content**
2. **Registrasi**
3. **Controller/Register.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Register extends CI\_Controller { 5. function \_\_construct() 6. { 7. parent::\_\_construct(); 8. $this->load->model('register\_m'); 9. $this->load->library('form\_validation'); 10. $this->load->library(array('googlemaps')); 12. } 14. public function index() 15. { 16. $data = array('' => ''); 17. $this->load->view('register/register\_form\_input', $data, FALSE); 18. } 20. public function inputdataregistrasi() 21. { 22. $this->load->library('googlemaps'); 23. $config['center'] = '-6.873074, 107.575958'; 24. $config['zoom'] = '15'; 25. $this->googlemaps->initialize($config);  28. $marker['position'] = '-6.873074, 107.575958'; 29. // $marker['draggable'] = true; 30. // $marker['ondragend'] = 'setMapToForm(event.latLng.lat(),event.latLng.lng());'; 31. $this->googlemaps->add\_marker($marker); 33. $valid=$this->form\_validation; 34. $valid->set\_rules('nama','Nama Mahasiswa','required',array('required' =>'%s Harus Diisi')); 36. if($valid->run()==FALSE) 37. { 38. $data['map'] = $this->googlemaps->create\_map(); 39. $data['angkatan'] = $this->register\_m->tampilangkatan(); 40. $data['kelas'] = $this->register\_m->tampilkelas(); 41. $data['p\_internal'] = $this->register\_m->tampilp\_internal(); 42. $data['p\_eksternal'] = $this->register\_m->tampilp\_ekstenal(); 44. $this->load->view('register/register\_form\_input', $data, FALSE); 45. $this->load->view('template'); 46. } 47. else 48. { 49. $email = $this->input->post('email'); 50. $nama = $this->input->post('nama'); 51. $password = SHA1($this->input->post('password')); 52. $data1 = array( 53. 'email'=>$email, 54. 'password'=>$password, 55. 'nama'=>$nama, 56. 'level'=>2, 57. 'status'=>0 58. ); 59. $i=$this->input; 60. $npm = $this->register\_m->create('user',$data1); 61. $data = array( 63. 'email' => $i->post('email'), 64. 'password' => SHA1 ($i->post('password')), 65. 'npm' => $i->post('npm'), 66. 'nama' => $i->post('nama'), 67. 'j\_kelamin' => $i->post('j\_kelamin'), 68. 'kelas\_id' => $i->post('kelas\_id'), 69. 'angkatan\_id' => $i->post('angkatan\_id'), 70. 'internal\_id' => $i->post('internal\_id'), 71. 'eksternal\_id' => $i->post('eksternal\_id'), 72. 'latitude' => -6.873776, 73. 'longitude' => 107.575639, 74. 'berkas' => $i->post('berkas'), 75. ); 76. // $insert = $this->register\_m->create('user',$data1); 77. $insert1 = $this->register\_m->create('tb\_mhs',$data); 78. // $this->db->select("MAX(id\_user)+1 AS id"); 79. // $this->db->from("tb\_mhs"); 80. // $query = $this->db->get(); 81. // return $query->row()->id; 82. // $this->register\_m->add($data); 83. $this->session->set\_flashdata('sukses', 'Data Registrasi Berhasil di Tambahkan'); 84. redirect(base\_url('register/viewdataregistrasi'), 'refresh'); 85. } 86. } 88. public function viewdataregistrasi() 89. { 90. $register=$this->register\_m->view\_data\_registrasi(); 91. $data = array( 92. 'register'    =>$register 93. ); 94. $this->load->view('register/view\_data\_registrasi', $data, FALSE); 96. } 98. public function register\_data() 99. { 100. check\_not\_login(); 101. $register=$this->register\_m->register\_data(); 102. $map= $this->googlemaps->create\_map(); 103. $data = array( 104. 'map'     =>$map, 105. 'register'    =>$register 106. ); 107. $this->template->load('template', 'register/register\_data', $data, FALSE); 108. } 110. public function detail($id) 111. { 112. check\_not\_login(); 113. $this->load->library('googlemaps'); 114. $config['center'] = '-6.873074, 107.575958'; 115. $config['zoom'] = '15'; 116. $this->googlemaps->initialize($config); 118. $lokasi=$this->register\_m->tampildata(); 119. foreach ($lokasi as $key => $value) { 120. $marker = array(); 121. $marker['animation'] = 'DROP'; 122. $marker['position'] = "$value->latitude, $value->longitude"; 123. $marker['infowindow\_content'] = '<div class="media" style="width:300px">'; 124. $marker['infowindow\_content'] .= '<div class="media-left">'; 125. $marker['infowindow\_content'] .= '</div>'; 126. $marker['infowindow\_content'] .= '<div class="media-body">'; 127. // $marker['infowindow\_content'] .= '<a>'.$value->lokasi.'</a><br>'; 128. $marker['infowindow\_content'] .= '<a>'.$value->latitude.'</a><br>'; 129. $marker['infowindow\_content'] .= '<a>'.$value->longitude.'</a><br>'; 130. $marker['infowindow\_content'] .= '</div>'; 131. $marker['infowindow\_content'] .= '</div>'; 132. $marker['icon'] = base\_url('assets/icon/location.png'); 134. $this->googlemaps->add\_marker($marker); 135. } 136. $this->googlemaps->initialize($config); 137. $ID = $this->uri->segment('3'); 138. $detail=$this->register\_m->register\_data\_1($ID); 139. $map= $this->googlemaps->create\_map(); 140. $data = array( 141. 'map' =>$map, 142. 'detail' =>$detail 143. ); 144. $this->template->load('template', 'register/detail', $data, FALSE); 146. } 148. public function delete($id) 149. { 150. $data= array('id\_register' => $id); 151. $this->register\_m->delete($data); 152. $this->session->set\_flashdata('sukses', 'Data Berhasil Dihapus'); 153. redirect(base\_url('register/inputdataregistrasi'), 'refresh'); 154. } 156. public function edit($id) 157. { 158. $this->form\_validation->set\_rules('fullname', 'Nama', 'required|min\_length[5]|callback\_nama\_check'); 159. $this->form\_validation->set\_rules('email', 'Email', 'required'); 160. if($this->input->post('password')) { 161. $this->form\_validation->set\_rules('password', 'Password', 'min\_length[5]'); 162. $this->form\_validation->set\_rules('passconf', 'Konfirmasi Password', 'matches[password]', 163. array('matches' => '%s tidak sesuai dengan password') 164. ); 165. } 166. if($this->input->post('passconf')) { 167. $this->form\_validation->set\_rules('passconf', 'Konfirmasi Password', 'matches[password]', 168. array('matches' => '%s tidak sesuai dengan password') 169. ); 170. } 171. $this->form\_validation->set\_rules('level', 'Level', 'required'); 173. $this->form\_validation->set\_message('required', '%s masih kosong, silakan isi'); 174. $this->form\_validation->set\_message('min\_length', '%s minimal 5 karakter'); 175. $this->form\_validation->set\_message('is\_unique', '%s ini sudah digunakan !'); 177. $this->form\_validation->set\_error\_delimiters('<span class="help-block">', '</span>'); 179. if ($this->form\_validation->run() == FALSE) { 180. $query = $this->register\_m->get($id); 181. if($query->num\_rows() > 0) { 182. $data['row'] = $query->row(); 183. $this->template->load('template', 'register/register\_form\_edit', $data); 184. } else { 185. echo "<script>alert('Data tidak ditemukan');"; 186. echo "window.location='".site\_url('register/register\_data')."';</script>"; 187. } 188. } else { 189. $post = $this->input->post(null, TRUE); 190. $this->register\_m->edit($post); 191. if($this->db->affected\_rows() > 0) { 192. echo "<script>alert('Data berhasil disimpan');</script>"; 193. } 194. echo "<script>window.location='".site\_url('register\_data')."';</script>"; 195. } 197. } 199. } |

Keterangan

Cara Penggunaan

1. **Model/Register\_m.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Register\_m extends CI\_Model { 5. public function create($table,$data) 6. { 7. $query = $this->db->insert($table, $data); 8. return $this->db->insert\_id(); 10. } 12. public function register\_data\_1($ID) 13. { 14. return $this->db->query("SELECT \* FROM tb\_mhs paa 15. left join tb\_kelas pme 16. on paa.kelas\_id = pme.id 17. left join tb\_p\_internal pin 18. on paa.internal\_id = pin.id 19. left join tb\_p\_eksternal pen 20. on paa.eksternal\_id = pen.id 21. left join tb\_angkatan akt 22. on paa.angkatan\_id = akt.id where paa.npm='$ID'")->result(); 23. } 25. public function register\_data() 26. { 27. return $this->db->query("SELECT \* FROM tb\_mhs paa 28. left join tb\_kelas pme 29. on paa.kelas\_id = pme.id 30. left join tb\_angkatan akt 31. on paa.angkatan\_id = akt.id")->result(); 32. } 34. public function tampildata() 35. { 36. $this->db->select('\*'); 37. $this->db->from('tb\_mhs'); 38. $query=$this->db->get(); 39. return $query->result(); 40. } 42. public function tampilangkatan() 43. { 44. $this->db->select('\*'); 45. $this->db->from('tb\_angkatan'); 46. $query=$this->db->get(); 47. return $query->result(); 48. } 50. public function tampilkelas() 51. { 52. $this->db->select('\*'); 53. $this->db->from('tb\_kelas'); 54. $query=$this->db->get(); 55. return $query->result(); 56. } 58. public function tampilp\_internal() 59. { 60. $this->db->select('\*'); 61. $this->db->from('tb\_p\_internal'); 62. $query=$this->db->get(); 63. return $query->result(); 64. } 66. public function tampilp\_ekstenal() 67. { 68. $this->db->select('\*'); 69. $this->db->from('tb\_p\_eksternal'); 70. $query=$this->db->get(); 71. return $query->result(); 72. } 74. public function edit($post) 75. { 76. $params['nama'] = $post['fullname']; 77. $params['email'] = $post['email']; 78. if(!emptyempty($post['password'])) { 79. $params['password'] = sha1($post['password']); 80. } 81. $params['alamat'] = $post['alamat'] != "" ? $post['alamat'] : null; 82. $params['level'] = $post['level']; 83. $this->db->where('npm', $post['npm']); 84. $this->db->update('tb\_mhs', $params); 85. } 87. public function get($id = null) 88. { 89. $this->db->from('tb\_mhs'); 90. if($id != null) { 91. $this->db->where('npm', $id); 92. } 93. $query = $this->db->get(); 94. return $query; 95. } 97. public function view\_data\_registrasi() 98. { 99. $this->db->select('a.\*, b.kelas, c.p\_eksternal, d.p\_internal'); 100. $this->db->from('tb\_mhs a'); 101. $this->db->join('tb\_kelas b', 'b.id = a.kelas\_id', 'left'); 102. $this->db->join('tb\_p\_internal d', 'd.id = a.internal\_id', 'left'); 103. $this->db->join('tb\_p\_eksternal c', 'c.id = a.eksternal\_id', 'left'); 104. $this->db->where('id\_user', 0); 105. $query = $this->db->get(); 106. return $query->result(); 107. } 109. public function tampilnpm($ID) 110. { 111. $this->db->select('\*'); 112. $query = $this->db->get\_where('tb\_mhs', array('id\_user' => $ID)); 113. return $query->result(); 114. } 115. } |

Keterangan

Cara Penggunaan

1. **Views/register**

* **register\_form\_input.php**

|  |
| --- |
| 1. <!DOCTYPE html> 2. <html> 3. <head> 4. <meta charset="utf-8"> 5. <meta http-equiv="X-UA-Compatible" content="IE=edge"> 6. <title>myINTERNSHIP</title> 7. <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport"> 8. <link rel="stylesheet" href="<?=base\_url()?>assets/bower\_components/bootstrap/dist/css/bootstrap.min.css"> 9. <link rel="stylesheet" href="<?=base\_url()?>assets/bower\_components/font-awesome/css/font-awesome.min.css"> 10. <link rel="stylesheet" href="<?=base\_url()?>assets/dist/css/AdminLTE.min.css"> 11. <link rel="icon" href="<?=base\_url()?>assets/dist/img/analytics.png"> 12. <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic"> 13. </head> 14. <!-- Content Header (Page header) --> 15. <section class="content-header"> 17. <div class="row"> 18. <div class="col-md-12 col-sm-10 col-xs-12"> 19. <div class="info-box"> 20. <span class="info-box-icon bg-blue"><i class="fa fa-map-marker"></i></span> 22. <div class="info-box-content"> 23. <center><h1><b>REGISTRASI</b></h1></center> 24. </div> 25. <!-- /.info-box-content --> 26. </div> 27. <!-- /.info-box --> 28. </div> 30. </section> 32. <div class="box"> 33. <div class="box-header"> 34. <div class="pull-right"> 35. <a href="<?=site\_url('auth/login')?>" class="btn btn-warning btn-flat"> 36. <i class="fa fa-undo"></i> Back 37. </a> 38. </div> 39. </div> 40. <div class="container"> 41. <div class="row"> 42. <div class="col-md-7"> 43. <div class="panel panel-primary"> 44. <div class="panel-heading"><i class="fa fa-map-marker"></i>   Kawasan Politeknik Pos Indonesia</div> 45. <div class="panel-body"> 47. <?php echo $map['html']; ?> 48. <?php echo $map['js']; ?> 49. </div> 50. </div> 51. </div> 53. <div class="col-md-5"> 54. <div class="panel panel-primary"> 55. <div class="panel-heading"><i class="fa fa-user"></i>   Input Data Registrasi</div> 56. <div class="panel-body"> 58. <?php 59. echo validation\_errors('<div class="alert alert-warning" id=""warning-alert>','</div>'); 60. if ($this->session->flashdata('sukses')) { 61. echo '<div class="alert alert-success" id="success-alert">'; 62. echo $this->session->flashdata('sukses'); 63. echo '</div>'; 64. } 65. echo form\_open(base\_url('register/inputdataregistrasi'), 'class="form-horizontal" entype="multipart/formdata"'); 66. ?> 68. <form class="form-horizontal"> 69. <div class="form-group"> 70. <label class="col-sm-3 control-label">Nama Mahasiswa</label> 71. <div class="col-sm-8"> 72. <input type="text" name="nama" class="form-control" placeholder="Nama Mahasiswa" 73. value="<?php echo set\_value('nama') ?>" required> 74. </div> 75. </div> 77. <div class="form-group"> 78. <label class="col-sm-3 control-label">Email</label> 79. <div class="col-sm-8"> 80. <input type="text" name="email" class="form-control" placeholder="Email" 81. value="<?php echo set\_value('email') ?>" required> 82. </div> 83. </div> 85. <div class="form-group"> 86. <label class="col-sm-3 control-label">Password</label> 87. <div class="col-sm-8"> 88. <input type="password" name="password" class="form-control" placeholder="Password" 89. value="<?php echo set\_value('password') ?>" required> 90. </div> 91. </div> 93. <div class="form-group"> 94. <label class="col-sm-3 control-label">NPM</label> 95. <div class="col-sm-8"> 96. <input type="npm" name="npm" class="form-control" placeholder="NPM" 97. value="<?php echo set\_value('npm') ?>" required> 98. </div> 99. </div> 101. <div class="form-group"> 102. <label class="col-sm-3 control-label">Program Studi/Kelas</label> 103. <div class="col-sm-8"> 104. <select required name="kelas\_id" class="form-control"> 105. <option>--Pilih Program Studi--</option> 106. <?php 107. foreach ($kelas as $row) { ?> 108. <option value="<?php echo $row->id ?>"><?php echo $row->kelas ?></option>; 109. <?php } ?> 110. </select> 111. </div> 112. </div>  115. <div class="form-group"> 116. <label class="col-sm-3 control-label">Jenis Kelamin</label> 117. <div class="col-sm-8"> 118. <input type="radio" name="j\_kelamin" value="L"> Laki-laki 119. <input type="radio" name="j\_kelamin" value="P"> Perempuan 120. </div> 121. </div> 123. <div class="form-group"> 124. <label class="col-sm-3 control-label">Angkatan</label> 125. <div class="col-sm-8"> 126. <select required name="angkatan\_id" class="form-control"> 127. <option>--Pilih Angkatan--</option> 128. <?php 129. foreach ($angkatan as $row) { ?> 130. <option value="<?php echo $row->id ?>"><?php echo $row->angkatan ?></option>; 131. <?php } ?> 132. </select> 133. </div> 134. </div> 136. <div class="form-group"> 137. <label class="col-sm-3 control-label">Pembimbing Internal</label> 138. <div class="col-sm-8"> 139. <select required name="internal\_id" class="form-control"> 140. <option>--Pilih Pembimbing Internal--</option> 141. <?php 142. foreach ($p\_internal as $row) { ?> 143. <option value="<?php echo $row->id ?>"><?php echo $row->p\_internal ?></option>; 144. <?php } ?> 145. </select> 146. </div> 147. </div> 149. <div class="form-group"> 150. <label class="col-sm-3 control-label">Pembimbing Eksternal</label> 151. <div class="col-sm-8"> 152. <select required name="eksternal\_id" class="form-control"> 153. <option>--Pilih Pembimbing Eksternal--</option> 154. <?php 155. foreach ($p\_eksternal as $row) { ?> 156. <option value="<?php echo $row->id ?>"><?php echo $row->p\_eksternal ?></option>; 157. <?php } ?> 158. </select> 159. </div> 160. </div> 161. <div class="form-group"> 162. <label class="col-sm-3 control-label">Surat Keterangan</label> 163. <small><i>\*surat keterangan penerimaan internship dari prodi</i></small> 164. <div class="col-sm-8"> 165. <input type="file" name="berkas" class="form-control" placeholder="Surat Keterangan" 166. value="<?php echo set\_value('berkas') ?>" required> 167. </div> 168. </div> 170. <div class="form-group"> 171. <label class="col-sm-3 control-label"></label> 172. <label class="col-sm-8"> 173. <button class="btn btn-primary" type="submit" > 174. <i class="fa fa-paper-plane"></i> Create</button> 175. <button class="btn btn-flat" type="reset">Reset</button> 176. <?php echo form\_close(); ?> 177. </div> 178. </div> 179. </form> 180. </div> 181. </div> 182. </html> |

Keterangan

Cara Penggunaan

* **register\_form\_edit.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Users 3. <small>Pengguna</small> 4. </h1> 5. <ol class="breadcrumb"> 6. <li><a href="#"><i class="fa fa-dashboard"></i></a></li> 7. <li class="active">Users</li> 8. </ol> 9. </section> 11. <!-- Main content --> 12. <section class="content"> 14. <div class="box"> 15. <div class="box-header"> 16. <h3 class="box-title">Edit User</h3> 17. <div class="pull-right"> 18. <a href="<?=site\_url('register/register\_data')?>" class="btn btn-warning btn-flat"> 19. <i class="fa fa-undo"></i> Back 20. </a> 21. </div> 22. <div class="box-body"> 23. <div class="row"> 24. <div class="col-md-4 col-md-offset-4"> 25. <form action="" method="post"> 26. <div class="form-group <?=form\_error('fullname') ? 'has-error' : null?>"> 27. <label>Nama \*</label> 28. <input type="hidden" name="nama" value="<?=$row->nama?>"> 29. <input type="text" name="fullname" value="<?=$this->input->post('fullname') ? $this->input->post('fullname') : $row->nama?>" class="form-control"> 30. <?=form\_error('fullname')?> 31. </div> 32. <div class="form-group <?=form\_error('email') ? 'has-error' : null?>"> 33. <label>Email \*</label> 34. <input type="text" name="email" value="<?=$this->input->post('email') ? $this->input->post('email') : $row->email?>" class="form-control"> 35. <?=form\_error('email')?> 36. </div> 37. <div class="form-group <?=form\_error('password') ? 'has-error' : null?>"> 38. <label>Password</label> <small>(Biarkan kosong jika tidak ingin diubah) </small> 39. <input type="password" name="password" value="<?=$this->input->post('password')?>" class="form-control"> 40. <?=form\_error('password')?> 41. </div> 42. <div class="form-group <?=form\_error('passconf') ? 'has-error' : null?>"> 43. <label>Password Confirmation</label> 44. <input type="password" name="passconf" value="<?=$this->input->post('passconf')?>" class="form-control"> 45. <?=form\_error('passconf')?> 46. </div> 47. <div class="form-group"> 48. <button type="submit" class="btn btn-primary btn-flat"> 49. <i class="fa fa-paper-plane"></i> Simpan 50. </button> 51. <button type="Reset" class="btn btn-flat">Reset</button> 52. </div> 53. </form> 54. </div> 55. </div> 56. </div> 57. </div> 58. </section> |

Keterangan

Cara Penggunaan

* **register\_data.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Data Register 3. <small>internship</small> 4. </h1> 5. <ol class="breadcrumb"> 6. <li><a href="#"><i class="fa fa-dashboard"></i></a></li> 7. <li class="active">Mahasiswa</li> 8. </ol> 9. </section> 10. <section class="content"> 11. <div class="row"> 12. <div class="col-xs-12"> 13. <div class="box"> 14. <div class="box-header"> 16. </div> 17. <!-- /.box-header --> 18. <div class="table-responsive"> 19. <table id="example1" class="table table-bordered table-striped"> 20. <thead> 21. <tr> 22. <th style="width : 50px">No</th> 23. <th>NPM</th> 24. <th>Nama Mahasiswa</th> 25. <!-- <th>Email</th> --> 26. <th>Program Studi</th> 27. <th>Angkatan</th> 28. <th style="width : 80px">Aksi</th> 29. </tr> 30. </thead> 31. <?php $no=1 ; foreach($register as $register) { ?> 32. <tr> 33. <td><?php echo $no++; ?>.</td> 34. <td><?php echo $register->npm; ?></td> 35. <td><?php echo $register->nama; ?></td> 36. <!-- <td><?php echo $register->email; ?></td> --> 37. <td><?php echo $register->kelas; ?></i></td> 38. <td><?php echo $register->angkatan; ?></td> 39. <td> 40. <!-- <a href="<?php echo base\_url('register/edit/'.$register->npm); ?>" class="btn 41. btn-success btn-sm" type="button"><i class="fa fa-check-square-o"> Approval</i></a> --> 42. <a href="<?php echo base\_url('register/detail/'.$register->npm); ?>" class="btn 43. btn-primary btn-sm" type="button"><i class="fa fa-eye"></i> Detail</a> 44. <a href="<?php echo base\_url('register/delete/'.$register->npm); ?>" class="btn 45. btn-danger btn-sm" type="button" onclick="return confirm('Yakin data ini dihapus. ?')"><i class="fa fa-trash"></i></a> 46. </td> 47. </tr> 48. <?php } ?> 50. </table> 51. </div> 52. </div> 53. </div> 54. </div> 55. </section> |

Keterangan

Cara Penggunaan

* **view\_data\_register.php**

|  |
| --- |
| 1. <!DOCTYPE html> 2. <html> 3. <head> 4. <meta charset="utf-8"> 5. <meta http-equiv="X-UA-Compatible" content="IE=edge"> 6. <title>myINTERNSHIP</title> 7. <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport"> 8. <link rel="stylesheet" href="<?=base\_url()?>assets/bower\_components/bootstrap/dist/css/bootstrap.min.css"> 9. <link rel="stylesheet" href="<?=base\_url()?>assets/bower\_components/font-awesome/css/font-awesome.min.css"> 10. <link rel="stylesheet" href="<?=base\_url()?>assets/dist/css/AdminLTE.min.css"> 11. <link rel="icon" href="<?=base\_url()?>assets/dist/img/analytics.png"> 12. <!--[if lt IE 9]> 13. <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script> 14. <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script> 15. <![endif]--> 16. <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic"> 17. </head> 18. <!-- Content Header (Page header) --> 19. <section class="content-header"> 21. <div class="row"> 22. <div class="col-md-12 col-sm-10 col-xs-12"> 23. <div class="info-box"> 24. <span class="info-box-icon bg-blue"><i class="fa fa-map-marker"></i></span> 26. <div class="info-box-content"> 27. <center><h1><b>REGISTRASI</b></h1></center> 28. </div> 29. <!-- /.info-box-content --> 30. </div> 31. <!-- /.info-box --> 32. </div> 34. </section> 36. <div class="box"> 37. <div class="box-header"> 38. <div class="pull-right"> 39. <a href="<?=site\_url('auth/login')?>" class="btn btn-warning btn-flat"> 40. <i class="fa fa-undo"></i> Back 41. </a> 42. </div> 43. </div> 45. <div class="col-md-5"> 46. <div class="panel panel-primary"> 47. <div class="panel-heading"><i class="fa fa-user"></i>   Data Registrasi</div> 48. <div class="panel-body"> 50. <table class="table"> 51. <?php $no=1 ; foreach($register as $register) { ?> 52. <tr> 53. <!-- <td> 54. <img src="<?php echo base\_url(); ?>assets/img/<?php echo $register->foto; ?> 55. "width="90" height="110"> 56. </td> 57. <td></td> --> 58. </tr> 59. <tr> 60. <th>Nama Mahasiswa</th> 61. <td>:</td> 62. <td><?php echo $register->nama; ?></td> 63. </tr> 64. <tr> 65. <th>NPM</th> 66. <td>:</td> 67. <td><?php echo $register->npm; ?></td> 68. </tr> 69. <tr> 70. <th>Program Studi/Kelas</th> 71. <td>:</td> 72. <td><?php echo $register->kelas; ?></td> 73. </tr> 74. <tr> 75. <th>Jenis Kelamin</th> 76. <td>:</td> 77. <td><?php echo $register->j\_kelamin; ?></td> 78. </tr> 79. <tr> 80. <th>Email</th> 81. <td>:</td> 82. <td><?php echo $register->email; ?></td> 83. </tr> 84. <tr> 85. <tr> 86. <th>Pembimbing Internal</th> 87. <td>:</td> 88. <td><?php echo $register->p\_internal; ?></td> 89. </tr> 90. <th>Pembimbing Eksternal</th> 91. <td>:</td> 92. <td><?php echo $register->p\_eksternal; ?></td> 93. </tr> 94. <tr> 95. <th>Surat Keterangan</th> 96. <td>:</td> 97. <td><?php echo $register->berkas; ?>  <button class="btn 98. btn-primary btn-sm" type="button"><i class="fa fa-file-text-o"> open</i></button></td> 99. </tr> 100. <?php } ?> 101. </table> 102. <tr> 103. <a href="<?=site\_url('')?>"> 104. <button class="btn btn-warning btn-flat"><i class="fa fa-pencil"></i> Edit</button> 105. </tr> 106. </html> |

Keterangan

Cara Penggunaan

* **detail.php**

|  |
| --- |
| 1. <script src="http://maps.google.com/maps/api/js?sensor=false"></script> 2. <section class="content-header"> 3. <h1>Detail Data 4. <small></small> 5. </h1> 6. <ol class="breadcrumb"> 7. <li><a href="#"><i class="fa fa-dashboard"></i></a></li> 8. <li class="active">Mahasiswa</li> 9. </ol> 10. </section> 11. <?php echo $map['js']; ?> 12. <section class="content"> 13. <div class="container"> 14. <div class="row"> 15. <div class="col-md-7"> 16. <div class="panel panel-primary"> 17. <div class="panel-heading"><i class="fa fa-map-marker"></i>   Kawasan Politeknik Pos Indonesia</div> 18. <div class="panel-body"> 19. <?php echo $map['html']; ?> 20. </div> 21. </div> 22. </div> 24. <div class="col-md-4"> 25. <div class="panel panel-primary"> 26. <div class="panel-heading"><i class="fa fa-user"></i>   Detail Data Mahasiswa</div> 27. <div class="panel-body"> 29. <table class="table"> 30. <?php $no=1 ; foreach($detail as $detail) { ?> 31. <tr> 32. <!-- <td> 33. <img src="<?php echo base\_url(); ?>assets/img/<?php echo $detail->foto; ?> 34. "width="90" height="110"> 35. </td> 36. <td></td> --> 37. </tr> 38. <tr> 39. <th>Nama Mahasiswa</th> 40. <td>:</td> 41. <td><?php echo $detail->nama; ?></td> 42. </tr> 43. <tr> 44. <th>NPM</th> 45. <td>:</td> 46. <td><?php echo $detail->npm; ?></td> 47. </tr> 48. <tr> 49. <th>Program Studi/Kelas</th> 50. <td>:</td> 51. <td><?php echo $detail->kelas; ?></td> 52. </tr> 53. <tr> 54. <th>Jenis Kelamin</th> 55. <td>:</td> 56. <td><?php echo $detail->j\_kelamin; ?></td> 57. </tr> 58. <tr> 59. <th>Email</th> 60. <td>:</td> 61. <td><?php echo $detail->email; ?></td> 62. </tr> 63. <tr> 64. <tr> 65. <th>Pembimbing Internal</th> 66. <td>:</td> 67. <td><?php echo $detail->p\_internal; ?></td> 68. </tr> 69. <th>Pembimbing Eksternal</th> 70. <td>:</td> 71. <td><?php echo $detail->p\_eksternal; ?></td> 72. </tr> 73. <tr> 74. <th>Surat Keterangan</th> 75. <td>:</td> 76. <td><?php echo $detail->berkas; ?>  <button class="btn 77. btn-primary btn-sm" type="button"><i class="fa fa-file-text-o"> open</i></button></td> 78. </tr> 79. <?php } ?> 80. </table> 81. <tr> 82. <a href="<?=site\_url('user')?>"> 83. <button class="btn btn-success btn-flat"><i class="fa fa-plus"></i> Actived</button> 84. <a href="<?=site\_url('register/register\_data')?>"> 85. <button class="btn btn-warning btn-flat"><i class="fa fa-undo"></i> Back</button> 86. </tr> 87. </section> |

Keterangan

Cara Penggunaan

1. **Login**
2. **Controller/Auth.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Auth extends CI\_Controller { 5. public function \_\_construct() 6. { 7. parent::\_\_construct(); 8. $this->load->library('form\_validation'); 9. } 11. public function login() 12. { 13. check\_already\_login(); 14. $this->load->view('login'); 15. } 16. public function process() 17. { 18. $post = $this->input->post(null, TRUE); 19. if(isset($post['login'])) { 20. $this->load->model('user\_m'); 21. $query = $this->user\_m->login($post); 22. if($query->num\_rows() > 0) { 23. $row = $query->row(); 24. $params = array( 25. 'iduser' => $row->id\_user, 26. 'level' => $row->level, 27. 'status' => $row->status 28. ); 29. if($params['status'] == 1) { 30. $this->session->set\_userdata($params); 31. echo "<script> 32. alert('Selamat, login berhasil'); 33. window.location='".site\_url('dashboard')."'; 34. </script>"; 35. } else { 36. echo "<script> 37. alert('Login gagal, email atau password salah'); 38. window.location='".site\_url('auth/login')."'; 39. </script>"; 40. } 41. } 42. } 43. } 45. public function logout() 46. { 47. $params = array('iduser', 'level'); 48. $this->session->unset\_userdata($params); 49. redirect('auth/login'); 50. } 51. } |

Keterangan

Cara Penggunaan

1. **Views/login.php**

|  |
| --- |
| 1. <!DOCTYPE html> 2. <html> 3. <head> 4. <meta charset="utf-8"> 5. <meta http-equiv="X-UA-Compatible" content="IE=edge"> 6. <title>myINTERNSHIP</title> 7. <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport"> 8. <link rel="stylesheet" type="text/css" href="<?php echo base\_url('assets/css/style.css') ?>"> 9. <link rel="stylesheet" href="<?=base\_url()?>assets/bower\_components/bootstrap/dist/css/bootstrap.min.css"> 10. <link rel="stylesheet" href="<?=base\_url()?>assets/bower\_components/font-awesome/css/font-awesome.min.css"> 11. <link rel="stylesheet" href="<?=base\_url()?>assets/dist/css/AdminLTE.min.css"> 12. <link rel="icon" href="<?=base\_url()?>assets/dist/img/analytics.png">   16. <!--[if lt IE 9]> 17. <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script> 18. <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script> 19. <![endif]--> 20. <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic"> 21. </head> 22. <body class="hold-transition login-page"> 23. <div class="login-box"> 24. <div class="login-logo"> 25. <img src="<?php echo base\_url('assets/dist/img/analytics.png') ?> "> 26. </div> 27. <div class="login-box"> 28. <div class="login-logo"> 29. <a>my<b>INTERNSHIP</b></a> 30. </div> 31. <div class="login-box-body"> 32. <p class="login-box-msg">Sign in to start your session</p> 33. <form action="<?=site\_url('auth/process')?>" method="post"> 34. <div class="form-group has-feedback"> 35. <input type="email" name="email" class="form-control" placeholder="Email" value="<?= set\_value('email'); ?>"> 36. <span class="glyphicon glyphicon-envelope form-control-feedback"></span> 37. <?= form\_error('email', '<small class="text-danger pl-3">', '</small>'); ?> 38. </div> 39. <div class="form-group has-feedback"> 40. <input type="password" name="password" class="form-control" placeholder="Password" required> 41. <span class="glyphicon glyphicon-lock form-control-feedback"></span> 42. <?= form\_error('password', '<small class="text-danger pl-3">', '</small>'); ?> 43. </div> 44. <div class="row"> 45. <div class="col-xs-8"> 46. </div> 47. <div class="col-xs-4"> 48. <button type="submit" name="login" class="btn btn-primary btn-block btn-flat">Sign In</button> 49. </div> 50. </div> 51. <a href="<?=base\_url('register/inputdataregistrasi')?>" class="text-center">Register a new membership</a> 52. </div> 53. </div> 54. </div> 55. </form> 56. </div> 57. </div> |

Keterangan

Cara Penggunaan

1. **Laporan**
2. **Controllers/Laporan.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Laporan extends CI\_Controller { 5. function \_\_construct() 6. { 7. parent::\_\_construct(); 8. check\_not\_login(); 9. $this->load->model('laporan\_m'); 10. $this->load->library('form\_validation'); 11. $this->load->library(array('googlemaps')); 12. $this->load->library(array('geolocation')); 13. $this->load->config('geolocation', true);  16. } 18. public function index() 19. { 20. $data = array('' => ''); 21. $this->template->load('template', 'laporan/laporan\_form\_input', $data, FALSE); 22. } 24. public function inputlokasi() 25. { 26. $this->load->library('googlemaps'); 27. $config['center'] = '-6.873074, 107.575958'; 28. $config['zoom'] = '15'; 29. $this->googlemaps->initialize($config); 30. $valid=$this->form\_validation; 31. $valid->set\_rules('u\_kegiatan','Uraian Kegiatan','required',array('required' =>'%s Harus Diisi')); 33. if($valid->run()==FALSE) 34. { 35. $where  = $this->session->userdata('iduser'); 36. // $data['npm'] = $this->laporan\_m->tampilnpm(); 37. // $data['npm'] = $this->laporan\_m->tampilnpm($where); 38. // $data['lokasi'] = $this->laporan\_m->tampilplokasi\_id(); 39. // $data['map'] = $this->googlemaps->create\_map(); 40. $data = array( 41. 'npm' => $this->laporan\_m->tampilnpm($where), 42. 'lokasi' => $this->laporan\_m->tampilplokasi\_id(), 43. 'map' => $this->googlemaps->create\_map(), 44. 'dosen' => $this->laporan\_m->getdosen() 45. ); 46. $this->template->load('template', 'laporan/laporan\_form\_input', $data, FALSE); 47. } 48. else 49. { 51. $config['upload\_path']          = './upload/'; 52. $config['allowed\_types']        = 'pdf|doc|docx'; 53. $config['max\_size']             = 0; 54. // $config['max\_width']            = 1024; 55. // $config['max\_height']           = 768 56. $this->load->library('upload', $config); 57. $this->upload->initialize($config); 59. if ( ! $this->upload->do\_upload('upload')){ 60. $error = array('error' => $this->upload->display\_errors()); 61. $this->load->view('welcome',$error); 63. }else{  66. $asd = $this->upload->data('file\_name'); 67. $d = $asd; 69. $u\_kegiatan=$this->input->post('u\_kegiatan'); 70. $npm=$this->input->post('npm'); 71. $latitude2=$this->input->post('latitude2'); 72. $longitude2=$this->input->post('longitude2'); 73. $lokasi\_id=$this->input->post('lokasi\_id'); 74. $hasil=$this->input->post('hasil'); 75. $komentar=$this->input->post('komentar'); 76. $dosen=$this->input->post('dosen'); 77. // $upload=$this->input->post('upload'); 79. $data = array( 80. 'u\_kegiatan' => $u\_kegiatan, 81. 'npm' => $npm, 82. 'latitude2' => $latitude2, 83. 'longitude2' => $longitude2, 84. 'lokasi\_id' => $lokasi\_id, 85. 'hasil' => $hasil, 86. 'absensi' => 1, 87. 'status' => 1, 88. 'komentar' => $komentar, 89. 'nilai' => 0, 90. 'dosen' => $dosen, 91. 'upload' => $d 92. ); 93. $this->laporan\_m->add($data); 94. $this->session->set\_flashdata('sukses', 'Data Laporan Berhasil di Tambahkan'); 95. redirect(base\_url('laporan/inputlokasi'), 'refresh'); 96. } 98. } 99. } 100. function haversine($id) 101. { 102. $explodeVal = explode('~', $id); 103. // dari prodi 104. $lat1 = -6.873776; 105. $lon1 = 107.575639; 106. // target 107. $lat2 = $explodeVal[0]; 108. $lon2 = $explodeVal[1]; 109. // distance between latitudes 110. // and longitudes 111. $dLat = ($lat2 - $lat1) \* 112. M\_PI / 180.0; 113. $dLon = ($lon2 - $lon1) \* 114. M\_PI / 180.0; 116. // convert to radians 117. $lat1 = ($lat1) \* M\_PI / 180.0; 118. $lat2 = ($lat2) \* M\_PI / 180.0; 120. // apply formulae 121. $a = pow(sin($dLat / 2), 2) + 122. pow(sin($dLon / 2), 2) \* 123. cos($lat1) \* cos($lat2); 124. $rad = 6371; 125. $c = 2 \* asin(sqrt($a)); 126. $SimpanRad = $rad \* $c; 127. // Driver code 128. $penjumlahan = $SimpanRad\*1000; 129. $data['jsonarray'] = $penjumlahan.' METER'; 130. echo json\_encode($data); 131. } 133. public function datareportkegiatan() 134. { 135. $laporan=$this->laporan\_m->datareportkegiatan(); 136. $map= $this->googlemaps->create\_map(); 137. $data = array( 138. 'map'     =>$map, 139. 'laporan' =>$laporan 140. ); 141. $this->template->load('template', 'mahasiswa/data\_laporan', $data, FALSE); 142. } 144. public function datareportkegiatan1() 145. { 146. $laporan=$this->laporan\_m->datareportkegiatan1(); 147. $map= $this->googlemaps->create\_map(); 148. $data = array( 149. 'map'     =>$map, 150. 'laporan' =>$laporan 151. ); 152. $this->template->load('template', 'mahasiswa/data\_laporan1', $data, FALSE); 153. } 155. public function detail($id) 156. { 157. check\_not\_login(); 158. $ID = $this->uri->segment('3'); 159. $logbook=$this->laporan\_m->tampil\_data\_1($ID); 160. $data = array( 161. 'logbook' =>$logbook 162. ); 163. $this->template->load('template', 'laporan/tampil\_logbook', $data, FALSE); 165. } 167. public function lihat($id) 168. { 169. check\_not\_login(); 170. $ID = $this->uri->segment('3'); 171. $logbook=$this->laporan\_m->muncul\_data\_1($ID); 172. $data = array( 173. 'logbook' =>$logbook 174. ); 175. $this->template->load('template', 'laporan/tampil\_logbook1', $data, FALSE); 177. } 179. public function detail1($id) 180. { 181. $this->load->library('googlemaps'); 182. $config['center'] = '-6.873074, 107.575958'; 183. $config['zoom'] = '15'; 184. $this->googlemaps->initialize($config); 186. $lokasi=$this->laporan\_m->tampil\_data\_2(); 187. foreach ($lokasi as $key => $value) { 188. $marker = array(); 189. $marker['animation'] = 'DROP'; 190. $marker['position'] = "$value->latitude2, $value->longitude2"; 191. $marker['infowindow\_content'] = '<div class="media" style="width:300px">'; 192. $marker['infowindow\_content'] .= '<div class="media-left">'; 193. $marker['infowindow\_content'] .= '</div>'; 194. $marker['infowindow\_content'] .= '<div class="media-body">'; 195. // $marker['infowindow\_content'] .= '<h5 class="media-heading">'.$value->nama.'</h5>'; 196. // $marker['infowindow\_content'] .= '<a>'.$value->nama\_lokasi.'</a><br>'; 197. // $marker['infowindow\_content'] .= '<a>'.$value->deskripsi.'</a><br>'; 198. $marker['infowindow\_content'] .= '<a>'.$value->latitude2.'</a><br>'; 199. $marker['infowindow\_content'] .= '<a>'.$value->longitude2.'</a><br>'; 200. // $marker['infowindow\_content'] .= '<a>'.$value->waktu.'</a>'; 201. $marker['infowindow\_content'] .= '</div>'; 202. $marker['infowindow\_content'] .= '</div>'; 203. $marker['icon'] = base\_url('assets/icon/location.png'); 205. $this->googlemaps->add\_marker($marker); 207. } 208. $this->googlemaps->initialize($config); 209. $map= $this->googlemaps->create\_map(); 210. $ID = $this->uri->segment('3'); 211. // $detail = $this->logbook\_m->logbook\_data\_1($ID); 212. $detail=$this->laporan\_m->tampil\_data\_2(); 213. $data = array( 214. 'map' =>$map, 215. 'detail' => $detail 216. ); 217. $this->template->load('template', 'laporan/detail\_logbook\_1', $data, FALSE); 218. } 220. public function delete($id) 221. { 222. $data= array('id\_lokasi' => $id); 223. $this->laporan\_m->delete($data); 224. $this->session->set\_flashdata('sukses', 'Data Berhasil Dihapus'); 225. redirect(base\_url('laporan/datareportkegiatan'), 'refresh'); 226. } 228. public function update($id) 229. { 230. $this->load->library('googlemaps'); 232. $config['center'] = '-6.873074, 107.575958'; 233. $config['zoom'] = '15'; 234. $this->googlemaps->initialize($config);  237. $marker['position'] = '-6.873074, 107.575958'; 238. $marker['draggable'] = true; 239. $marker['ondragend'] = 'setMapToForm(event.latLng.lat(),event.latLng.lng());'; 240. $this->googlemaps->add\_marker($marker); 242. $laporan=$this->laporan\_m->detail($id); 243. $valid=$this->form\_validation; 244. $valid->set\_rules('nama\_lokasi','Nama Lokasi','required',array('required' =>'%s Harus Diisi')); 246. if($valid->run()==FALSE) 247. { 248. $map= $this->googlemaps->create\_map(); 249. $data = array( 250. 'map'     =>$map, 251. 'laporan' =>$laporan 252. ); 253. $this->template->load('template', 'mahasiswa/data\_laporan\_edit', $data, FALSE); 255. } 256. else 257. { 258. $i=$this->input; 259. $data = array( 260. 'id\_lokasi' => $id, 261. 'nama' => $i->post('nama') , 262. 'nama\_lokasi' => $i->post('nama\_lokasi'), 263. 'deskripsi' => $i->post('deskripsi'), 264. 'latitude' => $i->post('latitude'), 265. 'longitude' => $i->post('longitude'), 266. 'nilai' => 0, 267. 'waktu' => date('Y-m-d H:i:s') 268. ); 269. $this->laporan\_m->update($data); 270. $this->session->set\_flashdata('sukses', 'Data Laporan Berhasil diubah'); 271. redirect(base\_url('laporan/datareportkegiatan'), 'refresh'); 272. } 274. } 276. public function tambah() 277. { 278. $this->form\_validation->set\_rules('nilai', 'nilai', 'required'); 279. if ($this->form\_validation->run() == FALSE) { 280. $query = $this->laporan\_m->get1($id); 281. if($query->num\_rows() > 0) { 282. $data['row'] = $query->row(); 283. $this->template->load('template', 'laporan/nilai', $data); 284. } else { 285. echo "<script>alert('Data tidak ditemukan');"; 286. echo "window.location='".site\_url('laporan/tampil\_data')."';</script>"; 287. } 288. } else { 289. $post = $this->input->post(null, TRUE); 290. $this->laporan\_m->edit($post); 291. if($this->db->affected\_rows() > 0) { 292. echo "<script>alert('Data berhasil disimpan');</script>"; 293. } 294. echo "<script>window.location='".site\_url('laporan/tampil\_data')."';</script>"; 295. } 296. } 297. } |

Keterangan

Cara Penggunaan

1. **Models/Laporan\_m.php**
2. **Views/laporan**
3. **Logbook**
4. **Penilaian**