# **IMPLEMENTASI *SOURCE CODE***

## ***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*, *content* 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>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="stylesheet" href="<?=base\_url()?>assets/dist/css/skins/\_all-skins.min.css"> 12. <link rel="icon" href="<?=base\_url()?>assets/dist/img/analytics.png"> 13. <link rel="stylesheet" href="<?=base\_url()?>assets/bower\_components/datatables.net-bs/css/dataTables.bootstrap.min.css"> 14. <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script> 15. <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script> 16. <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic"> |

1. ***Footer***

|  |
| --- |
| 1. <script src="<?=base\_url()?>assets/bower\_components/jquery/dist/jquery.min.js"></script> 2. <script src="<?=base\_url()?>assets/bower\_components/bootstrap/dist/js/bootstrap.min.js"></script> 3. <script src="<?=base\_url()?>assets/bower\_components/jquery-slimscroll/jquery.slimscroll.min.js"></script> 4. <script src="<?=base\_url()?>assets/dist/js/adminlte.min.js"></script> 5. <script src="<?=base\_url()?>assets/bower\_components/datatables.net/js/jquery.dataTables.min.js"></script> 6. <script src="<?=base\_url()?>assets/bower\_components/datatables.net-bs/js/dataTables.bootstrap.min.js"></script> 7. <script src="<?=base\_url()?>assets/bower\_components/Flot/jquery.flot.js"></script> 8. <script src="<?=base\_url()?>assets/bower\_components/Flot/jquery.flot.pie.js"></script> |

Keterangan :

*Source code* diatas merupakan pembahasan lanjutan dari bab 4 halaman xx tentang bagian *template* bawaaan *header* dan *footer*, lalu pada bagian *source* *code* *header* dan *footer* di bab 5 ini telah disesuai dengan kebutuhan *project* aplikasi yang di bangun. Pada dasar-nya aplikasi yang dibangun tidak memerlukan *template* yang sangat rumit, cukup untuk menarik dan mudah digunakan oleh *user*.

## ***Content***

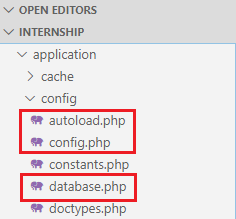
Konten yang dimaksud pada pembahasan ini adalah tentang isi sebuah aplikasi yang mencakup dokumen, data, teks, gambar dll. Jadi hal pertama yang dilakukan oleh *developer* untuk membangun aplikasi yaitu perecanaan pada konsep konten.

1. **Registrasi**
2. **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 :

Pada bagian *source* *code* diatas menjabarkan semua fungsi yang ada pada proses registrasi didalam *controllers* antara lain: *input* data, *edit* data, *detail* data dan *delete* data. Dimana *function* pada *controllers* akan saling berkaitan dengan folder lain seperti *models* dan *views*. Kemudian kita juga harus meng-konfigurasikan file kode seperti *autoload*.php, *config*.php, *database*.php didalam folder *config* agar aplikasi yang dijalankan tidak ada kendala. Karena, file ini digunakan untuk mengatur fungsi-fungsi yang akan dimuat otomatis di awal ketika program dijalankan.

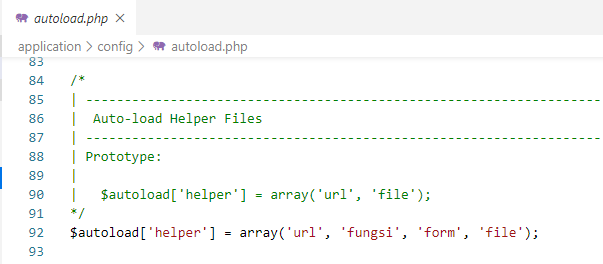


Gambar 5.1 Proses Konfigurasi

Kemudian kita buka *source code* dengan nama *file* *autoload*.php dengan meng-konfigurasikan pada bagian ‘*libraries’*, dan ‘*helper’* seperti berikut.

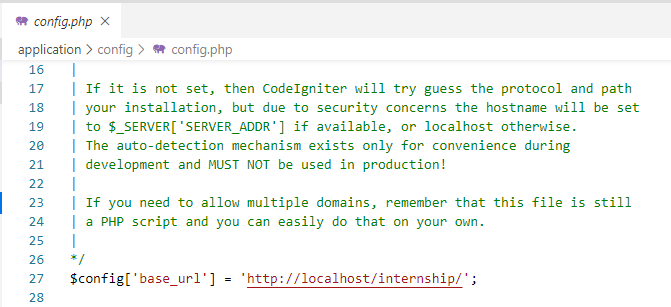


Gambar 5.2 Proses Konfigurasi *autoload* pada *libraries*



Gambar 5.3 Proses Konfigurasi *autoload* pada *helper*

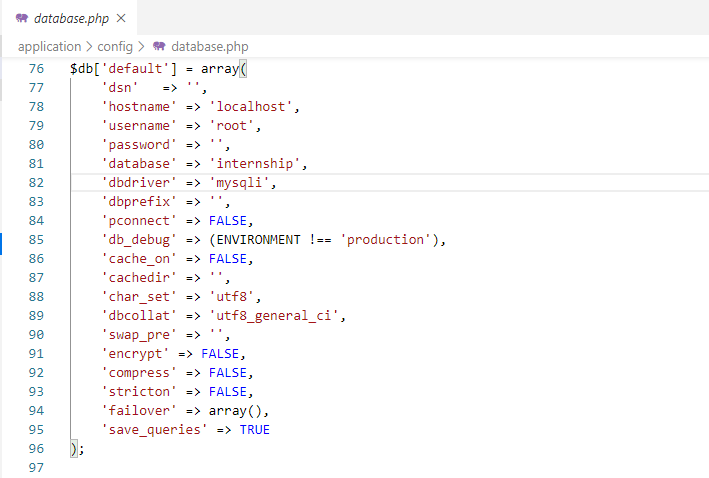
Lalu tahap selanjutnya kita membuka *file* kode dengan nama *config*.php dan memberikan keterangan seperti gambar berikut.



Gambar 5.4 Proses Konfigurasi *config* pada *base\_url*

Penjelasan mengenai gambar diatas adalah pemberian alamat pada bagian *$config[‘base\_url] = ‘......’* . Pada bagian ini disesuaikan dengan nama yang ada didalam folder *‘C:\xampp\htdocs*’ serta menjalankan-nya di *browser*.

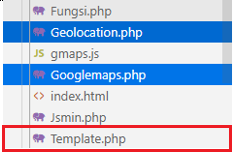
Dan terakhir pada bagian file kode *database*.php dengan memberikan keterangan seperti gambar berikut.



Gambar 5.5 Proses Konfigurasi pada *database*

Sebelum masuk ketahap ini, penulis menyarankan untuk membuat *database* terlebih dahulu seperti contoh pada gambar 4.49 di halaman xx sebagai proses perancangan *databse.* Lalu anda mengakses [*http://localhost/phpmyadmin/*](http://localhost/phpmyadmin/) *link* tersebut untuk membuat *database* sebagai koneksi-nya dan jangan lupa untuk menjalankan terlebih dahulu aplikasi *Xampp* sebagai server. Setelah itu anda baru memberikan keterangan pada *file* *database*.php dengan nama *database* yang anda buat.

Pada tahap ini memerlukan konfigurasi dalam folder ‘*libraries’*, karena untuk tampilan antarmuka proses registrasi menampilkan peta yang memanfaatkan citra *google maps*. Jadi untuk menampilkan fungsi tersebut harus dilakukan penambahan *script* dengan menamai *Googlemaps*.php dan *Jsmin*.php di dalam folder libraries. Contoh gambar seperti berikut :



Gambar 5.6 Proses Penambahan *Script*

Untuk mendapatkan *source code* yang tertera pada gambar diatas, anda bisa mempelajari pada bagian bab 3 halaman xx tentang *Google Maps* atau anda bisa mengakses di [*developers.google.com/maps/documentation/javascript/libraries*](https://developers.google.com/maps/documentation/javascript/libraries)*.* Sedangkan untuk bagian *source code* *template*.php yang berada di dalam folder *libraries* sebagai fungsi untuk mengatur dinamisnya tampilan interaksi yang ada pada konsep konten *website* dengan memanggil fungsi yang ada pada folder *views*.

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 :

Folder *models* adalah folder yang berisikan file berekstensikan *php*, dimana file ini digunakan untuk pengelolaan data ke *database*. Dengan bantuan *controller* yang merujuk ke *model* yang sesuai dan *function* yang sesuai maka sebuah proses yang berkaitan dengan *database* sudah bisa dibuat, sebab sebagai koneksi data yang akan ditampilkan atau dikelola untuk proses selanjutnya.

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> |

* **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> |

* **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> |

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

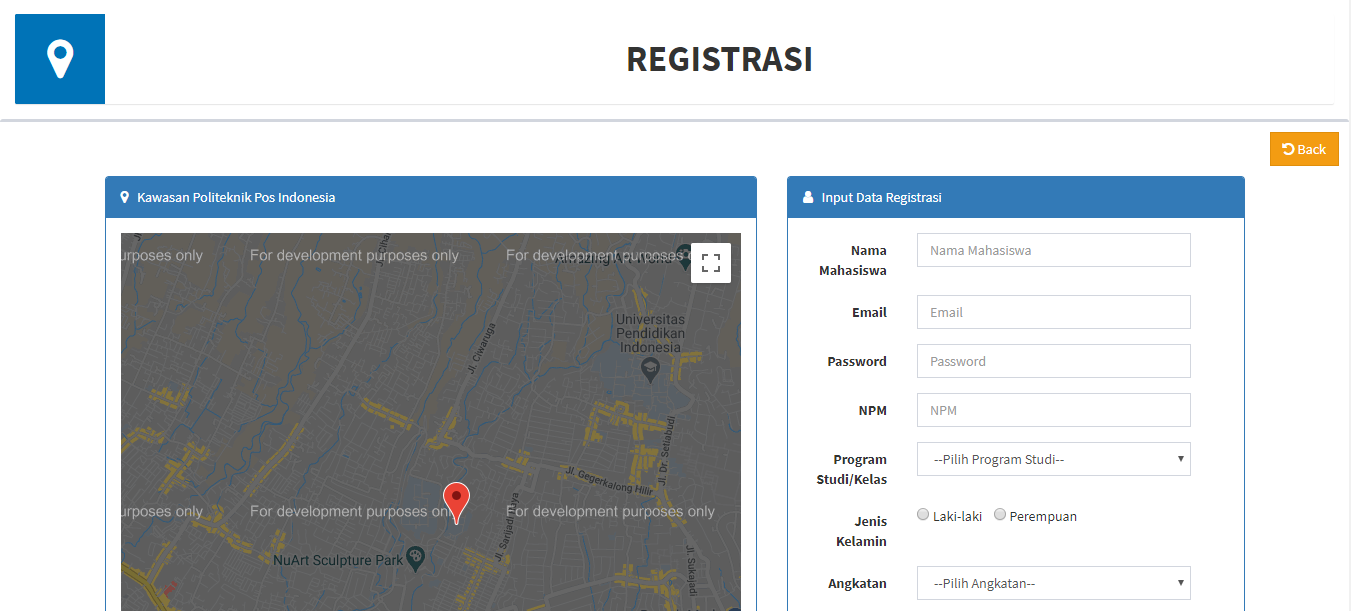
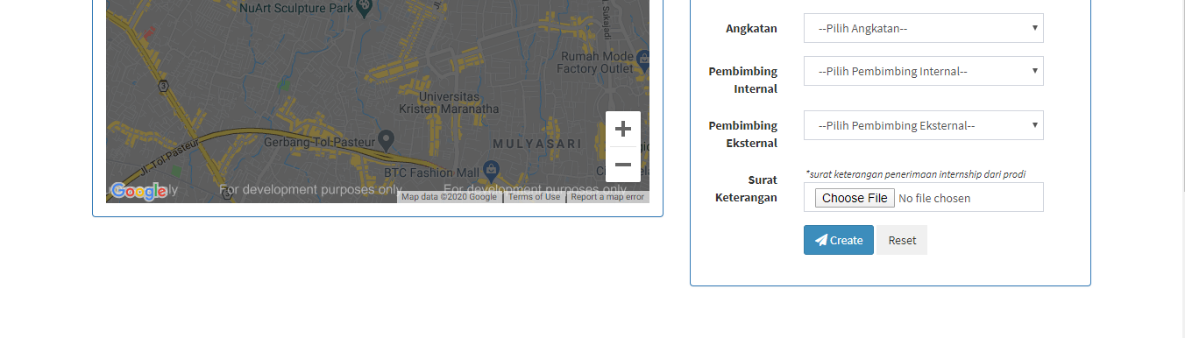
|  |
| --- |
| 1. <!DOCTYPE html> 2. <html> 3. <head> 4. <meta charset="utf-8"> 5. <meta http-equiv="X-UA-Compatible" content="IE=edge"> 6. <title>myINTERNSHIP</title> 7. <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport"> 8. <link rel="stylesheet" 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> |

* **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> |

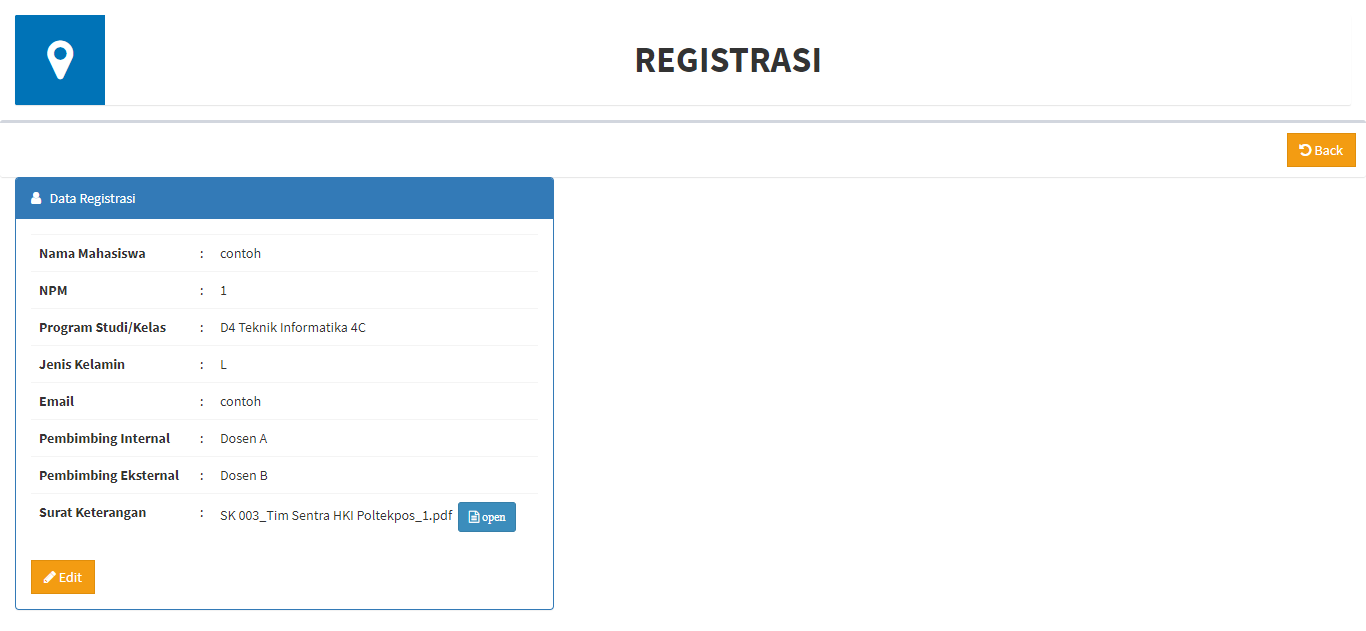
Keterangan :

Folder *views* adalah sebuah folder yang digunakan untuk menampung *script-script* yang berisikan desain antarmuka *website*. Anda juga dapat membuat folder *views* untuk mempermudah pengelompokan desain antarmuka yang anda buat. Pada *source code* yang ditampilkan diatas merupakan tampilan antarmuka untuk proses registrasi yang mencakup dari *edit* data, *view* data, *detail* data pada aplikasi yang dibangun seperti gambar berikut :



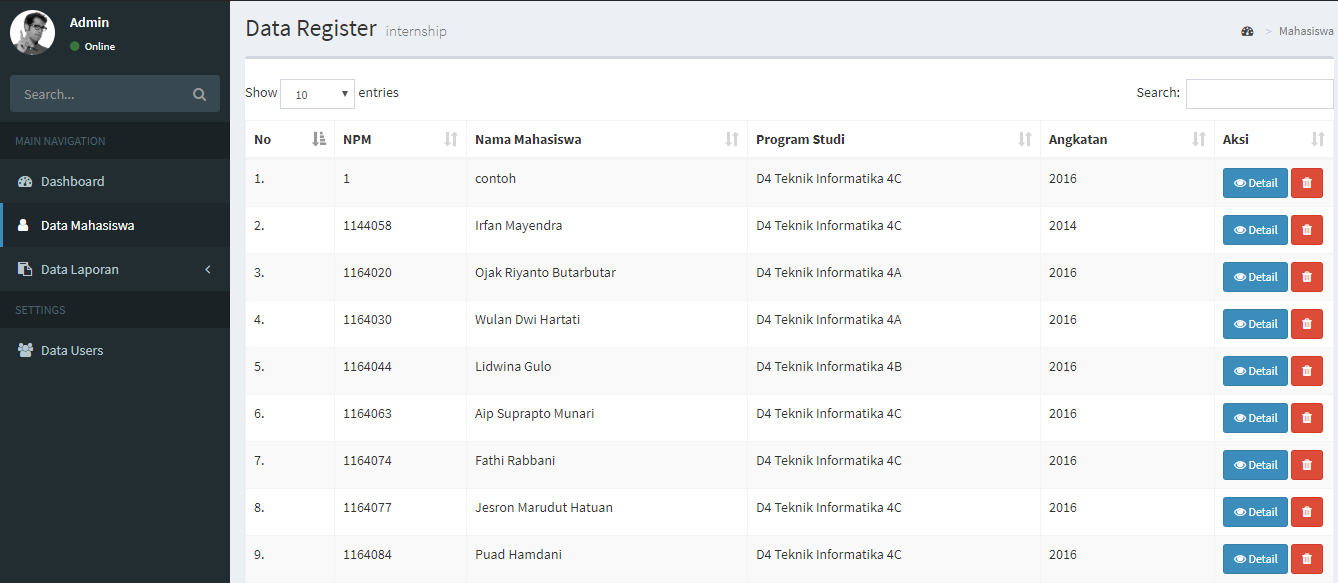
Gambar 5.7 Tampilan Registrasi

Pada gambar diatas terdapat *form-form* untuk mengisi data diri *user* untuk aktor tepatnya adalah mahasiswa/mahasiswa yang akan daftar untuk program *internship*. Sedangkan sebelah kiri terdapat peta posisi kampus yang digambarkan dengan *marker* bawaan dari *template* *Google Maps*. Lalu terdapat 3 tombol yang masing-masing memiliki fungsinya tersendiri antara lain *Create*, *Reset* dan *Back*. Cara penggunaan pada proses ini sangat simple, cukup mengisi semua *form* yang di sediakan sesuai dengan data diri *user*. Apabila semua *form* telah diisi otomatis sistem akan memberikan pesan bahwa proses yang telah dilakukan sukses apabila ada kegagalan sistem akan mengembalikan ke halaman tetap dan memberikan informasi *error*-nya.



Gambar 5.8 *Edit* Data Registrasi

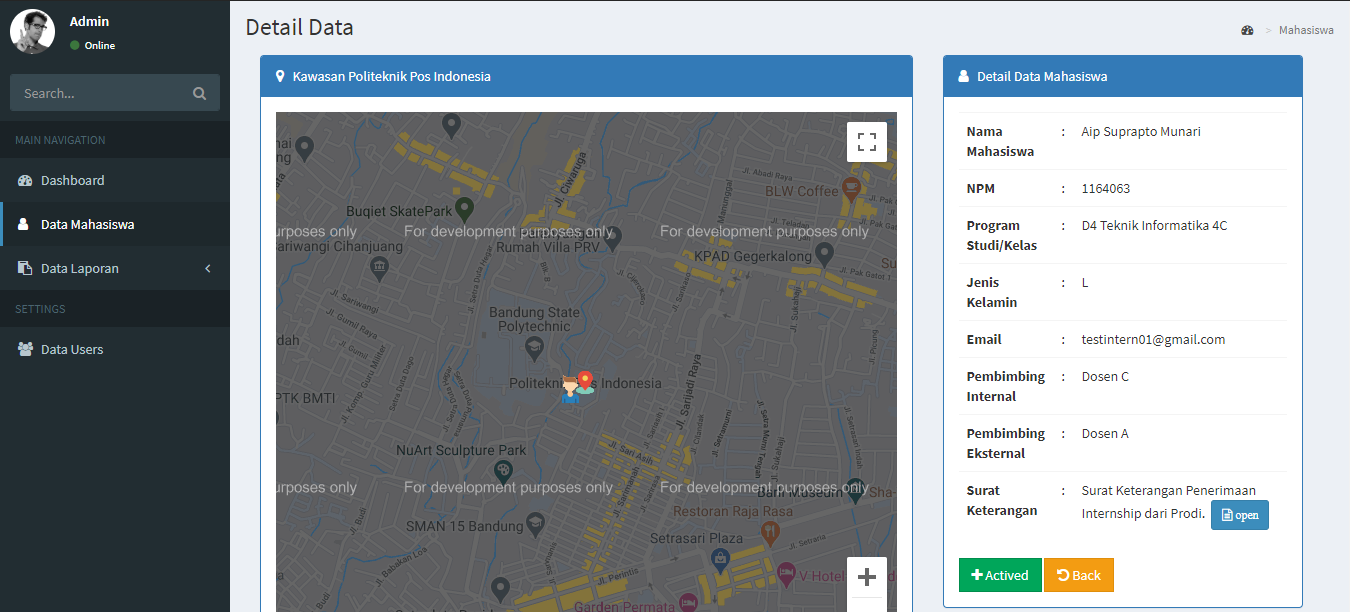
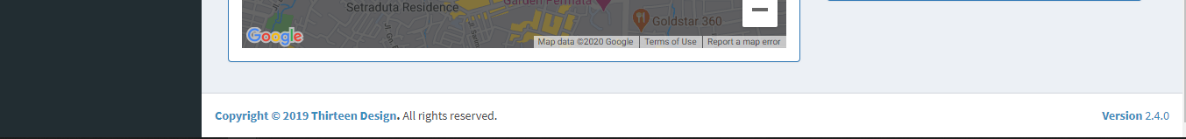
Pada tampilan ini merupakan proses untuk meng-*edit* data apabila *user* mengalami kendala pada saat mengisi data diri. Jadi *developer* memberikan proses tersebut untuk mengurangi kendala tersebut. Bagaimana cara penggunaanya ? cukup meng-klik tombol *edit* dan sistem akan menampilkan form pada saat anda mengisi data diri sebelumnya.



Gambar 5.9 *View* Data Registrasi

Sumber : *Template* AdminLTE versi 2.4.0

Pada proses ini aktor yang bisa melakukan akses untuk melihat data registrasi adalah admin, operator dan koordinator. Cara penggunaanya cukup *simple*, aktor cukup meng-klik menu data mahasiswa lalu sistem akan menampilkan ringkasan data mahasiswa/i yang mendaftar.



Gambar 5.10 *Detail* Data Registrasi

Sumber : *Template* AdminLTE versi 2.4.0

Proses ini adalah tampilan lanjutan dari *view* data registrasi apabila aktor melakukan aksi dengan meng-klik tombol *detail* data. Sistem akan menampilkan keseluruhan data diri mahasiswa.

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 :

Pada bagian *source code* yang tertera diatas merupakan proses untuk mengakses layanan *website* yang dibangun yang berisi *email* dan *password.* Untuk melakukan proses ini membutuhkan konfigurasi dari folder ‘*helpers’* dan ‘*libraries’* bawaan dari instalasi folder-folder CodeIgniter. Langkah-langkah untuk tahap konfigurasi seperti berikut :



Gambar 5.11 Tambah *Script* pada Folder *Helpers*

Penjelasan mengenai gambar diatas adalah *script* tambahan terkait konfigurasi autentikasi pada proses *login*. Jadi anda menambah sebuah *file script* di dalam folder ‘*helpers*’ dengan nama *fungsi\_helper*.php. Selanjutnya menambahkan file script juga pada folder ‘libraries’ dengan nama *Fungsi*.php, sebab konfigurasi pada proses ini akan saling terkait dengan fungsi-fungsi yang diberikan pada folder sebelumnya seperti pada gambar 5.2 dan 5.3.



Gambar 5.12 Tambah *Script* pada Folder *Libraries*

1. **Models/User\_m.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class User\_m extends CI\_Model { 5. public function login($post) 6. { 7. $this->db->select('\*'); 8. $this->db->from('user'); 9. $this->db->where('email', $post['email']); 10. $this->db->where('password', sha1($post['password'])); 11. $query = $this->db->get(); 12. return $query; 13. } 15. public function get($id = null) 16. { 17. $this->db->from('user'); 18. if($id != null) { 19. $this->db->where('id\_user', $id); 20. } 21. $query = $this->db->get(); 22. return $query; 23. } 25. public function add($post) 26. { 27. $params['nama'] = $post['fullname']; 28. $params['email'] = $post['email']; 29. $params['password'] = sha1($post['password']); 30. $params['level'] = $post['level']; 31. $params['status'] = $post['status']; 32. $this->db->insert('user', $params); 33. } 35. public function edit($post) 36. { 37. $params['nama'] = $post['fullname']; 38. $params['email'] = $post['email']; 39. if(!emptyempty($post['password'])) { 40. $params['password'] = sha1($post['password']); 41. } 42. $params['level'] = $post['level']; 43. $params['status'] = $post['status']; 44. $this->db->where('id\_user', $post['id\_user']); 45. $this->db->update('user', $params); 46. } 48. public function del($id) 49. { 50. $this->db->where('id\_user', $id); 51. $this->db->delete('user'); 52. } 53. } |

Keterangan :

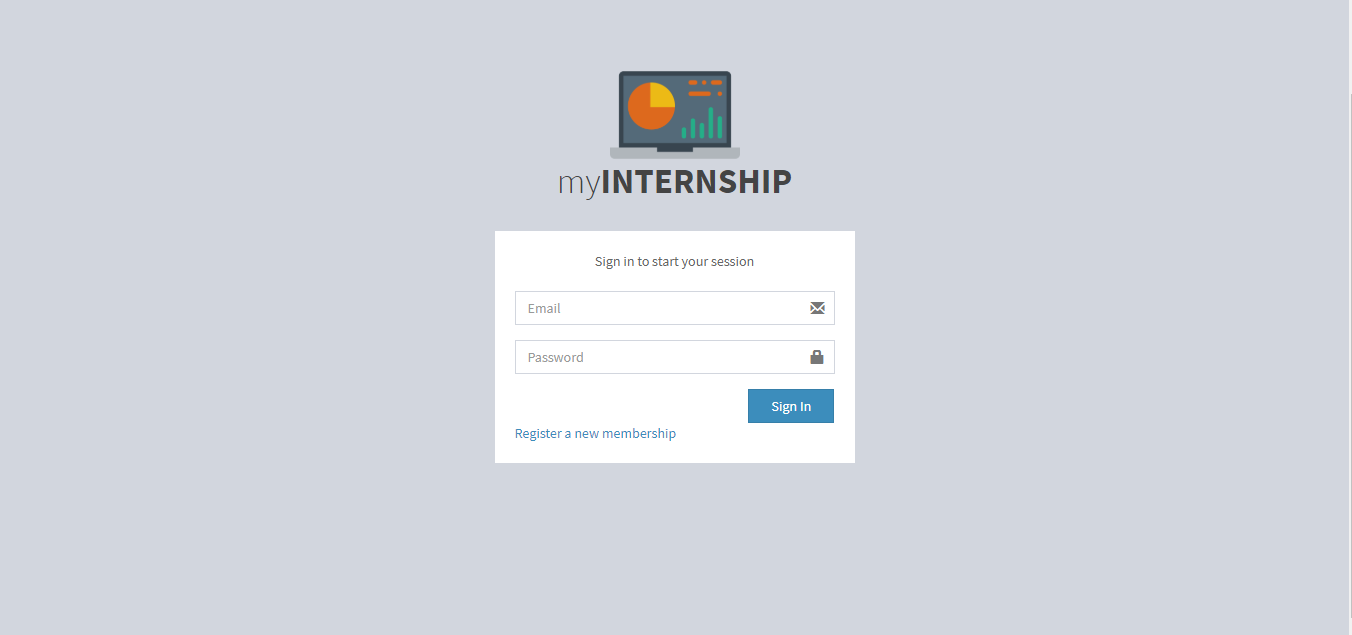
Untuk mengenai folder model yang berektensikan *php,* dimana *file* ini berisikan *script* yang digunakan untuk pengelolaan data ke *database*. Dengan bantuan *controller* ke model yang sesuai *function* yang berkaitan dengan *database* dan yang membedakan dengan *source* *code* *model* sebelumnya adalah terletak dari masing-masing *function* yang ada di *controller* sebagai koneksi ke database.

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 :

Pada penjelasan yang ada pada folder ini adalah *views* digunakan sebagai folder untuk menampung *file script-script* yang berisikan desain antarmuka website. Dimana pada *file* tersebut mempresentasikan *source code* untuk desain *login* diantaranya mencakup logo aplikasi, tombol *sign in*, *form input email* dan *password* serta *link* untuk melanjutkan pendaftaran akun apabila pengguna belum memiliki akses untuk *login*.



Gambar 5.13 Tampilan *Login*

Pada gambar ini merupakan proses pertama untuk mengakses ke dalam sistem bagi mahasiswa/i yang sudah memiliki akun dan telah di aktivasi oleh pihak *operator*. Sedangkan pengguna belum memiliki akses terhadap proses ini, aktor tersebut di arahkan pada *link* “*Register a new membership*” sebagai persyaratan untuk mendapatkan akun. Cara penggunaan pada halaman login ini sangat mudah, tinggal anda masuk *email* dan *password* yang sudah terdaftar. Selanjutnya anda menekan tombol sign-in dan sistem akan memvalidasi serta membaca level akun masuk ke sistem dan menampilkan-nya sesuai akses levelnya.

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 = array( 37. 'npm' => $this->laporan\_m->tampilnpm($where), 38. 'lokasi' => $this->laporan\_m->tampilplokasi\_id(), 39. 'map' => $this->googlemaps->create\_map(), 40. 'dosen' => $this->laporan\_m->getdosen() 41. ); 42. $this->template->load('template', 'laporan/laporan\_form\_input', $data, FALSE); 43. } 44. else 45. { 47. $config['upload\_path']          = './upload/'; 48. $config['allowed\_types']        = 'pdf|doc|docx'; 49. $config['max\_size']             = 0; 50. $this->load->library('upload', $config); 51. $this->upload->initialize($config); 53. if ( ! $this->upload->do\_upload('upload')){ 54. $error = array('error' => $this->upload->display\_errors()); 55. $this->load->view('welcome',$error); 57. }else{  60. $asd = $this->upload->data('file\_name'); 61. $d = $asd; 63. $u\_kegiatan=$this->input->post('u\_kegiatan'); 64. $npm=$this->input->post('npm'); 65. $latitude2=$this->input->post('latitude2'); 66. $longitude2=$this->input->post('longitude2'); 67. $lokasi\_id=$this->input->post('lokasi\_id'); 68. $hasil=$this->input->post('hasil'); 69. $komentar=$this->input->post('komentar'); 70. $dosen=$this->input->post('dosen'); 72. $data = array( 73. 'u\_kegiatan' => $u\_kegiatan, 74. 'npm' => $npm, 75. 'latitude2' => $latitude2, 76. 'longitude2' => $longitude2, 77. 'lokasi\_id' => $lokasi\_id, 78. 'hasil' => $hasil, 79. 'absensi' => 1, 80. 'status' => 1, 81. 'komentar' => $komentar, 82. 'nilai' => 0, 83. 'dosen' => $dosen, 84. 'upload' => $d 85. ); 86. $this->laporan\_m->add($data); 87. $this->session->set\_flashdata('sukses', 'Data Laporan Berhasil di Tambahkan'); 88. redirect(base\_url('laporan/inputlokasi'), 'refresh'); 89. } 91. } 92. } 93. function haversiine($id) 94. { 95. $explodeVal = explode('~', $id); 96. // dari prodi 97. $lat1 = -6.873776; 98. $lon1 = 107.575639; 99. // target 100. $lat2 = $explodeVal[0]; 101. $lon2 = $explodeVal[1]; 102. // distance between latitudes 103. // and longitudes 104. $dLat = ($lat2 - $lat1) \* 105. M\_PI / 180.0; 106. $dLon = ($lon2 - $lon1) \* 107. M\_PI / 180.0; 109. // convert to radians 110. $lat1 = ($lat1) \* M\_PI / 180.0; 111. $lat2 = ($lat2) \* M\_PI / 180.0; 113. // apply formulae 114. $a = pow(sin($dLat / 2), 2) + 115. pow(sin($dLon / 2), 2) \* 116. cos($lat1) \* cos($lat2); 117. $rad = 6371; 118. $c = 2 \* asin(sqrt($a)); 119. $SimpanRad = $rad \* $c; 120. // Driver code 121. $penjumlahan = $SimpanRad\*1000; 122. $data['jsonarray'] = $penjumlahan.' METER'; 123. echo json\_encode($data); 124. } 126. public function datareportkegiatan() 127. { 128. $laporan=$this->laporan\_m->datareportkegiatan(); 129. $map= $this->googlemaps->create\_map(); 130. $data = array( 131. 'map'     =>$map, 132. 'laporan' =>$laporan 133. ); 134. $this->template->load('template', 'mahasiswa/data\_laporan', $data, FALSE); 135. } 137. public function datareportkegiatan1() 138. { 139. $laporan=$this->laporan\_m->datareportkegiatan1(); 140. $map= $this->googlemaps->create\_map(); 141. $data = array( 142. 'map'     =>$map, 143. 'laporan' =>$laporan 144. ); 145. $this->template->load('template', 'mahasiswa/data\_laporan1', $data, FALSE); 146. } 148. public function detail($id) 149. { 150. check\_not\_login(); 151. $ID = $this->uri->segment('3'); 152. $logbook=$this->laporan\_m->tampil\_data\_1($ID); 153. $data = array( 154. 'logbook' =>$logbook 155. ); 156. $this->template->load('template', 'laporan/tampil\_logbook', $data, FALSE); 158. } 160. public function lihat($id) 161. { 162. check\_not\_login(); 163. $ID = $this->uri->segment('3'); 164. $logbook=$this->laporan\_m->muncul\_data\_1($ID); 165. $data = array( 166. 'logbook' =>$logbook 167. ); 168. $this->template->load('template', 'laporan/tampil\_logbook1', $data, FALSE); 170. } 172. public function detail1($id) 173. { 174. $this->load->library('googlemaps'); 175. $config['center'] = '-6.873074, 107.575958'; 176. $config['zoom'] = '15'; 177. $this->googlemaps->initialize($config); 179. $lokasi=$this->laporan\_m->tampil\_data\_2(); 180. foreach ($lokasi as $key => $value) { 181. $marker = array(); 182. $marker['animation'] = 'DROP'; 183. $marker['position'] = "$value->latitude2, $value->longitude2"; 184. $marker['infowindow\_content'] = '<div class="media" style="width:300px">'; 185. $marker['infowindow\_content'] .= '<div class="media-left">'; 186. $marker['infowindow\_content'] .= '</div>'; 187. $marker['infowindow\_content'] .= '<div class="media-body">'; 188. $marker['infowindow\_content'] .= '<a>'.$value->latitude2.'</a><br>'; 189. $marker['infowindow\_content'] .= '<a>'.$value->longitude2.'</a><br>'; 190. $marker['infowindow\_content'] .= '</div>'; 191. $marker['infowindow\_content'] .= '</div>'; 192. $marker['icon'] = base\_url('assets/icon/location.png'); 194. $this->googlemaps->add\_marker($marker); 196. } 197. $this->googlemaps->initialize($config); 198. $map= $this->googlemaps->create\_map(); 199. $ID = $this->uri->segment('3'); 200. $detail=$this->laporan\_m->tampil\_data\_2(); 201. $data = array( 202. 'map' =>$map, 203. 'detail' => $detail 204. ); 205. $this->template->load('template', 'laporan/detail\_logbook\_1', $data, FALSE); 206. } 208. public function delete($id) 209. { 210. $data= array('id\_lokasi' => $id); 211. $this->laporan\_m->delete($data); 212. $this->session->set\_flashdata('sukses', 'Data Berhasil Dihapus'); 213. redirect(base\_url('laporan/datareportkegiatan'), 'refresh'); 214. } 216. public function update($id) 217. { 218. $this->load->library('googlemaps'); 220. $config['center'] = '-6.873074, 107.575958'; 221. $config['zoom'] = '15'; 222. $this->googlemaps->initialize($config);  225. $marker['position'] = '-6.873074, 107.575958'; 226. $marker['draggable'] = true; 227. $marker['ondragend'] = 'setMapToForm(event.latLng.lat(),event.latLng.lng());'; 228. $this->googlemaps->add\_marker($marker); 230. $laporan=$this->laporan\_m->detail($id); 231. $valid=$this->form\_validation; 232. $valid->set\_rules('nama\_lokasi','Nama Lokasi','required',array('required' =>'%s Harus Diisi')); 234. if($valid->run()==FALSE) 235. { 236. $map= $this->googlemaps->create\_map(); 237. $data = array( 238. 'map'     =>$map, 239. 'laporan' =>$laporan 240. ); 241. $this->template->load('template', 'mahasiswa/data\_laporan\_edit', $data, FALSE); 243. } 244. else 245. { 246. $i=$this->input; 247. $data = array( 248. 'id\_lokasi' => $id, 249. 'nama' => $i->post('nama') , 250. 'nama\_lokasi' => $i->post('nama\_lokasi'), 251. 'deskripsi' => $i->post('deskripsi'), 252. 'latitude' => $i->post('latitude'), 253. 'longitude' => $i->post('longitude'), 254. 'nilai' => 0, 255. 'waktu' => date('Y-m-d H:i:s') 256. ); 257. $this->laporan\_m->update($data); 258. $this->session->set\_flashdata('sukses', 'Data Laporan Berhasil diubah'); 259. redirect(base\_url('laporan/datareportkegiatan'), 'refresh'); 260. } 262. } 264. public function tambah() 265. { 266. $this->form\_validation->set\_rules('nilai', 'nilai', 'required'); 267. if ($this->form\_validation->run() == FALSE) { 268. $query = $this->laporan\_m->get1($id); 269. if($query->num\_rows() > 0) { 270. $data['row'] = $query->row(); 271. $this->template->load('template', 'laporan/nilai', $data); 272. } else { 273. echo "<script>alert('Data tidak ditemukan');"; 274. echo "window.location='".site\_url('laporan/tampil\_data')."';</script>"; 275. } 276. } else { 277. $post = $this->input->post(null, TRUE); 278. $this->laporan\_m->edit($post); 279. if($this->db->affected\_rows() > 0) { 280. echo "<script>alert('Data berhasil disimpan');</script>"; 281. } 282. echo "<script>window.location='".site\_url('laporan/tampil\_data')."';</script>"; 283. } 284. } 285. } |

Keterangan :

Folder *controllers* merupakan yang berfungsi sebagai *file-file* untuk mengatur alur proses aliran data atau fungsi dibuat. *Controller* adalah salah satu bagian yang sangat berperan dalam *website* berbasis-kan MVC. Tanpa *controller* *view* dan *model* tidak akan bisa tampil atau berjalan semestinya. Didalam sebuah *controller* terdapat suatu *class* yang disesuaikan dengan nama *file* yang dibuat dan beberapa fungsi yang disesuaikan dengan kebutahan yang ada, diantaranya class untuk ‘Laporan’ dan fungsi *input* lokasi, *view* data, *detail* data dan *update* data. Untuk proses ini tidak ada konfigurasi tambahan, dikarena-kan sudah diwakili dari beberapa *controller* sebelumnya.

1. **Models/Laporan\_m.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Laporan\_m extends CI\_Model { 5. public function get($id = null) 6. { 7. $this->db->from('tbl\_lokasi'); 8. if($id != null) { 9. $this->db->where('id\_lokasi', $id); 10. } 11. $query = $this->db->get(); 12. return $query; 13. } 15. public function del($id) 16. { 17. $this->db->where('id\_lokasi', $id); 18. $this->db->delete('tbl\_lokasi'); 19. } 21. public function add($data) 22. { 23. $this->db->insert('tb\_kegiatan', $data); 24. }  27. public function datareportkegiatan() 28. { 29. return $this->db->query("SELECT \* FROM tb\_mhs paa 30. left join tb\_kelas pme 31. on paa.kelas\_id = pme.id 32. left join tb\_angkatan akt 33. on paa.angkatan\_id = akt.id")->result(); 34. } 36. public function datareportkegiatan1() 37. { 38. return $this->db->query("SELECT \* FROM tb\_mhs paa 39. left join tb\_kelas pme 40. on paa.kelas\_id = pme.id 41. left join tb\_angkatan akt 42. on paa.angkatan\_id = akt.id")->result(); 43. } 45. public function delete($data) 46. { 47. $this->db->where('id\_lokasi', $data['id\_lokasi']); 48. $this->db->delete('tbl\_lokasi'); 49. } 51. public function detail($id\_lokasi) 52. { 53. $this->db->select('\*'); 54. $this->db->from('tbl\_lokasi'); 55. $this->db->where('id\_lokasi', $id\_lokasi); 56. $query=$this->db->get(); 57. return $query->row(); 58. } 60. public function update($data) 61. { 62. $this->db->where('id\_lokasi', $data['id\_lokasi']); 63. $this->db->update('tbl\_lokasi',$data); 64. } 66. public function tampildata() 67. { 68. $this->db->select('\*'); 69. $this->db->from('tb\_mhs'); 70. $query=$this->db->get(); 71. return $query->result(); 72. } 74. public function tampilnpm($ID) 75. { 76. $this->db->select('\*'); 77. $query = $this->db->get\_where('tb\_mhs', array('id\_user' => $ID)); 78. return $query->result(); 79. } 81. public function tbl\_mhs($ID) 82. { 83. $this->db->select('\*'); 84. $query = $this->db->get\_where('tb\_mhs', array('id\_user' => $ID)); 85. return $query; 86. } 88. public function tampilplokasi\_id() 89. { 90. $this->db->select('\*'); 91. $this->db->from('tb\_lokasi'); 92. $query=$this->db->get(); 93. return $query->result(); 94. } 96. public function tampil\_data\_1($ID) 97. { 98. return $this->db->query("SELECT \* FROM tb\_kegiatan paa 99. left join tb\_lokasi pme 100. on paa.lokasi\_id = pme.id 101. left join tb\_p\_internal a 102. on paa.dosen = a.id where paa.npm='$ID'")->result(); 103. } 104. public function tampil\_data\_2() 105. { 106. $this->db->select('a.\*, b.npm, b.latitude as lat1, b.longitude as long1, c.lokasi'); 107. $this->db->from('tb\_kegiatan a'); 108. $this->db->join('v\_login b', 'a.npm = b.npm', 'left'); 109. $this->db->join('tb\_lokasi c', 'c.id = a.lokasi\_id', 'left'); 110. $query=$this->db->get(); 111. return $query->result(); 112. } 114. public function muncul\_data\_1($ID) 115. { 116. return $this->db->query("SELECT \* FROM tb\_kegiatan paa 117. left join tb\_lokasi pme 118. on paa.lokasi\_id = pme.id 119. left join tb\_p\_internal a 120. on paa.dosen = a.id where paa.npm='$ID,'")->result(); 122. } 124. public function get1($id = null) 125. { 126. $this->db->from('tb\_kegiatan'); 127. if($id != null) { 128. $this->db->where('id', $id); 129. } 130. $query = $this->db->get(); 131. return $query; 132. } 134. public function edit($post) 135. { 136. $params['nilai'] = $post['nilai']; 137. // $this->db->where('id\_user', $post['id\_user']); 138. $this->db->update('tb\_kegiatan', $params); 139. } 141. public function getDosen() 142. { 143. return $this->db->get('tb\_p\_internal')->result(); 144. } 146. } |

Keterangan :

Pada *source code* diatas merupakan *file* yang ada didalam folder *models* dengan berekstensikan php. Dimana *file* ini berisikan *script* yang digunakan untuk pengelolaan data ke *database* sebagai jembatan koneksi. Tanpa peranan *model*, *controller* atau *view* tidak akan berjalan semestinya. Sebab *model* adalah salah satu bagian sangat berperan dalam website berbasis-kan MVC.

1. **Views/laporan**

* **laporan\_form\_input.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Reports Activity 3. <small>Harian</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="container"> 12. <div class="container"> 13. <div class="row"> 14. <div class="col-md-8 col-md-offset-2"> 15. <div class="panel panel-primary"> 16. <div class="panel-heading"><i class="fa fa-map-marker"></i>&nbsp;&nbsp;Input Report Activity</div> 17. <div class="panel-body"> 18. <div class="form-group"> 19. <?php 20. echo validation\_errors('<div class="alert alert-warning" id=""warning-alert>','</div>'); 21. if ($this->session->flashdata('sukses')) { 22. echo '<div class="alert alert-success" id="success-alert">'; 23. echo $this->session->flashdata('sukses'); 24. echo '</div>'; 25. } 26. echo form\_open(base\_url('laporan/inputlokasi'), 'class="form-horizontal" entype="multipart/formdata"'); 27. ?> 28. <form class="form-horizontal"> 29. <div class="form-group"> 30. <label class="control-label">Kegiatan :</label> 31. <select required name="u\_kegiatan" class="form-control"> 32. <option value>--Pilih--</option> 33. <option>Internship 1</option> 34. <option>Internship 2</option> 35. </select> 36. </div> 37. <div class="form-group"> 38. <?php 39. foreach ($npm as $row) { ?> 40. <label class="control-label">NPM :</label> 41. <input type="text" name="npm" class="form-control" readonly="readonly" placeholder="npm" 42. value="<?php echo $row->npm ?>" required> 43. </div> 44. <?php } ?> 45. </div> 46. <div class="form-group"> 47. <html> 48. <body> 49. <script> 50. if ('geolocation' in navigator) { 51. console.log('geolocation available'); 52. navigator.geolocation.getCurrentPosition(position => { 53. const lat = position.coords.latitude; 54. const lon = position.coords.longitude; 55. document.getElementById('latitude2').textContent = lat; 56. document.getElementById('longitude2').textContent = lon; 57. // console.log(position); 58. }); 59. } else { 60. console.log('geolocation not available'); 61. } 62. </script> 63. <p> 64. <textarea id="latitude2" type="text" name="latitude2" readonly="readonly" class="form-control" placeholder="Latitude" 65. value="<?php echo set\_value('latitude2') ?>"></textarea> 66. <textarea id="longitude2" type="text" name="longitude2" readonly="readonly" class="form-control" placeholder="Longitude" 67. value="<?php echo set\_value('longitude2') ?>"></textarea> 68. <textarea id='showhasil' type="text" name='hasil' readonly="readonly" class="form-control" 69. value="<?php echo set\_value('hasil') ?>'"></textarea> 70. </p> 71. <a href='#' class='btn btn-primary' id='coba11'>Klil Hasil</a> 72. </body> 73. </html> 74. </div> 75. <div class="form-group"> 76. <label class="control-label">Keterangan Tempat :</label> 77. <select required name="lokasi\_id" class="form-control"> 78. <option>--Pilih--</option> 79. <?php 80. foreach ($lokasi as $row) { ?> 81. <option value="<?php echo $row->id ?>"><?php echo $row->lokasi?></option>; 82. <?php } ?> 83. </select> 84. </div> 85. <div class="form-group"> 86. <label class="control-label">Pembimbing :</label> 87. <select name="dosen" id="dosen" class="form-control"> 88. <option value="#">-- Pilih Dosen --</option> 89. <?php foreach($dosen as $d) { ?> 90. <option value="<?= $d->id;?>"><?= $d->p\_internal;?></option> 91. <?php } ?> 92. </select> 93. </div> 94. <div class="form-group"> 95. <input type="hidden" name="komentar" class="form-control" 96. value="<?php echo set\_value('tidak ada komentar') ?>" > 97. </div> 98. <div class="form-group"> 99. <label class="control-label">Upload Berkas (pdf|doc|docx)</label> 100. <input type="file" name="upload" class="form-control" placeholder="upload file" 101. value="<?php echo set\_value('upload') ?>"><br> 102. </div> 103. <div class="form-group"> 104. <button class="btn btn-success" type="submit" > 105. <i class="fa fa-paper-plane"></i> Simpan</button> 106. <button class="btn btn-flat" type="reset">Reset</button> 108. </div> 109. </div> 110. </div> 111. </div> 112. </form> 113. </div> 114. </div> 115. </div> 116. </section> 117. <script> 118. var map, infoWindow; 119. function initMap() { 120. map = new google.maps.Map(document.getElementById('map'), { 121. center: {lat: -6.873074, lng: 107.575958}, 122. zoom: 15 123. }); 124. infoWindow = new google.maps.InfoWindow; 125. if (navigator.geolocation) { 126. navigator.geolocation.getCurrentPosition(function(position) { 127. var pos = { 128. lat: position.coords.latitude, 129. lng: position.coords.longitude 130. }; 131. infoWindow.setPosition(pos); 132. infoWindow.setContent('Posisi Kamu Disini.'); 133. infoWindow.open(map); 134. map.setCenter(pos); 135. }, function() { 136. handleLocationError(true, infoWindow, map.getCenter()); 137. }); 138. } else { 139. // Browser doesn't support Geolocation 140. handleLocationError(false, infoWindow, map.getCenter()); 141. } 142. } 143. function handleLocationError(browserHasGeolocation, infoWindow, pos) { 144. infoWindow.setPosition(pos); 145. infoWindow.setContent(browserHasGeolocation ? 146. 'Error: The Geolocation service failed.' : 147. 'Error: Your browser doesn\'t support geolocation.'); 148. infoWindow.open(map); 149. } 150. </script> 151. <script async defer 152. src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDuba8i0GLvJRNcwAd5fedoAZczm0Yribg&callback=initMap"> 153. </script> |

* **laporan\_data.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Reports Activity Harian 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> 11. <!-- Main content --> 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">Lokasi Mahasiswa</div> 18. <div class="panel-body"> 20. </div> 21. <div class="panel-footer"></div> 22. </div> 23. </div> 25. <div class="col-md-5"> 26. <div class="panel panel-primary"> 27. <div class="panel-heading">Input Data Laporan</div> 28. <div class="panel-body"> 29. <form action="<?=site\_url('laporan/process')?>" method="post"> 30. <div class="form-group"> 31. <label>Nama \*</label> 32. <input type="text" name="nama" value="" class="form-control" required> 33. </div> 34. <div class="form-group"> 35. <button type="submit" class="btn btn-success btn-flat"> 36. <i class="fa fa-paper-plane"></i> Save 37. </button> 38. <button type="Reset" class="btn btn-flat">Reset</button> 39. </div> 40. </form> 41. </div> 42. </div> 43. <div class="panel-footer"></div> 44. </div> 45. </div>  48. <div class="box-body table-responsive"> 49. <table class="table table-bordered table-striped"> 50. <thead> 51. <tr> 52. <th>#</th> 53. <th>Nama</th> 54. <th>Nama\_lokasi</th> 55. <th>Deskripsi</th> 56. <th>Latitude</th> 57. <th>Longitude</th> 58. <th>Waktu</th> 59. <th>Aksi</th> 60. </tr> 61. </thead> 62. <tbody> 63. <?php $no = 1; 64. foreach($row->result() as $key => $data) { ?> 65. <tr> 66. <td><?=$no++?>.</td> 67. <td><?=$data->npm?></td> 68. <td><?=$data->date?></td> 69. <td><?=$data->u\_kegiatan?></td> 70. <td><?=$data->lokasi\_id?></td> 71. <td><?=$data->longitude2?></td> 72. <td><?=$data->latitude2?></td> 73. <td><?=$data->upload?></td> 74. <td class="text-center" width="160px"> 75. <a href="<?=site\_url('laporan/del/'.$data->id)?>" onclick="return confirm('Apakah anda yakin hapus data?')" class="btn btn-danger btn-xs"> 76. <i class="fa fa-trash"></i> Delete 77. </a> 78. </td> 79. </tr> 80. <?php 81. } ?> 82. </tbody> 83. </table> 84. </div> 85. </div> 86. </section> |

* **tampil\_logbook.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Reports Activity Harian 3. <small>Internship 1</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>No</th> 23. <th>NPM</th> 24. <th>Kegiatan</th> 25. <th>Nama lokasi</th> 26. <th>Latitude</th> 27. <th>Longitude</th> 28. <th>Hasil Jarak</th> 29. <th>Pembimbing</th> 30. <th>Absensi</th> 31. <th>Status</th> 32. <th>Nilai</th> 33. <th>Tanggal/Waktu</th> 34. <th>Berkas</th> 35. <th>Aksi</th> 36. </tr> 37. </thead> 38. <tbody> 39. <?php $no=1 ; 40. foreach($logbook as $logbook) { ?> 41. <tr> 42. <td><?=$no++?>.</td> 43. <td><?=$logbook->npm?></td> 44. <td><?=$logbook->u\_kegiatan?></td> 45. <td><?=$logbook->lokasi?></td> 46. <td><?=$logbook->latitude2?></td> 47. <td><?=$logbook->longitude2?></td> 48. <td><?=$logbook->hasil?></td> 49. <td><?=$logbook->p\_internal?></td> 50. <td><?=$logbook->absensi?></td> 51. <td><?=$logbook->status?></td> 52. <td><?=$logbook->nilai?></td> 53. <td><?=$logbook->date?></td> 54. <!-- <td><?=$logbook->upload?></td> --> 55. <td> 56. <a href="<?=base\_url().'./upload/'.$logbook->upload;?>" target="\_blank"> 57. <center><i class="fa fa-download"></center></i> 58. </a> 59. </td> 60. <td> 61. <!-- <a href="<?php echo base\_url('laporan/tambah/'.$logbook->id); ?>" class="btn  --> 62. <!-- btn-success btn-sm" type="button"><i class="fa fa-plus"> Tambah</i></a> --> 63. <a href="<?php echo base\_url('laporan/detail1/'.$logbook->npm); ?>" class="btn 64. btn-primary btn-sm" type="button"><i class="fa fa-eye"> Detail</i></a> 65. </td> 66. </tr> 67. <?php 68. } ?> 69. </tbody> 70. </table> 71. </div> 72. </div> 73. </section> |

* **tampil\_logbook1.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Reports Activity Harian 3. <small>Internship 2</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>No</th> 23. <th>NPM</th> 24. <th>Kegiatan</th> 25. <th>Nama lokasi</th> 26. <th>Latitude</th> 27. <th>Longitude</th> 28. <th>Hasil Jarak</th> 29. <th>Pembimbing</th> 30. <th>Absensi</th> 31. <th>Status</th> 32. <th>Nilai</th> 33. <th>Tanggal/Waktu</th> 34. <th>Upload</th> 35. <th>Aksi</th> 36. </tr> 37. </thead> 38. <tbody> 39. <?php $no=1 ; 40. foreach($logbook as $logbook) { ?> 41. <tr> 42. <td><?=$no++?>.</td> 43. <td><?=$logbook->npm?></td> 44. <td><?=$logbook->u\_kegiatan?></td> 45. <td><?=$logbook->lokasi?></td> 46. <td><?=$logbook->latitude2?></td> 47. <td><?=$logbook->longitude2?></td> 48. <td><?=$logbook->hasil?></td> 49. <td><?=$logbook->p\_internal?></td> 50. <td><?=$logbook->absensi?></td> 51. <td><?=$logbook->status?></td> 52. <td><?=$logbook->nilai?></td> 53. <td><?=$logbook->date?></td> 54. <!-- <td><?=$logbook->upload?></td> --> 55. <td> 56. <a href="<?=base\_url().'./upload/'.$logbook->upload;?>" target="\_blank"> 57. <center><i class="fa fa-download"></center></i> 58. </a> 59. </td> 60. <td> 61. <!-- <a href="<?php echo base\_url('laporan/tambah/'.$logbook->id); ?>" class="btn  --> 62. <!-- btn-success btn-sm" type="button"><i class="fa fa-plus"> Tambah</i></a> --> 63. <a href="<?php echo base\_url('laporan/lihat1/'.$logbook->npm); ?>" class="btn 64. btn-primary btn-sm" type="button"><i class="fa fa-eye"> Detail</i></a> 65. </td> 66. </tr> 67. <?php 68. } ?> 69. </tbody> 70. </table> 71. </div> 72. </div> 73. </section> |

* **detail\_logbook1.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">Pemetaan Marker Posisi</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">Detail Data Kegiatan</div> 27. <div class="panel-body"> 29. <table class="table"> 30. <?php $no=1 ; 31. foreach($detail as $detail) { ?> 32. <tr> 33. <!-- <td> 34. <img src="<?php echo base\_url(); ?>assets/img/<?php echo $detail->foto; ?> 35. "width="90" height="110"> 36. </td> 37. <td></td> --> 38. </tr> 39. <tr> 40. <th>NPM</th> 41. <td>:</td> 42. <td><?php echo $detail->npm; ?></td> 43. </tr> 44. <tr> 45. <th>Uraian Kegiatan</th> 46. <td>:</td> 47. <td><?php echo $detail->u\_kegiatan; ?></td> 48. </tr> 49. <tr> 50. <th>Latitude</th> 51. <td>:</td> 52. <td><?php echo $detail->lat1; ?></td> 53. </tr> 54. <tr> 55. <th>Longitude</th> 56. <td>:</td> 57. <td><?php echo $detail->long1; ?></td> 58. </tr> 59. <tr> 60. <th>Latitude2</th> 61. <td>:</td> 62. <td><?php echo $detail->latitude2; ?></td> 63. </tr> 64. <tr> 65. <th>Longitude2</th> 66. <td>:</td> 67. <td><?php echo $detail->longitude2; ?></td> 68. </tr> 69. <tr> 70. <th>Tanggal/Waktu</th> 71. <td>:</td> 72. <td><?php echo $detail->date; ?></td> 73. </tr> 74. <tr> 75. <th>Hasil Jarak</th> 76. <td>:</td> 77. <td><?php echo $detail->hasil; ?></td> 78. </tr> 79. <tr> 80. <th>Berkas</th> 81. <td>:</td> 82. <td><?php echo $detail->upload; ?>  <button class="btn 83. btn-primary btn-sm" type="button"><i class="fa fa-file-text-o"> open</i></button></td> 84. </tr> 85. <?php } ?> 86. </table> 87. <tr> 88. <!-- <button class="btn btn-success" type="submit" > 89. <i class="fa fa-check-square-o"></i> Approve</button> --> 90. <a href="<?=site\_url('laporan/datareportkegiatan')?>"> 91. <button class="btn btn-warning btn-flat"><i class="fa fa-undo"></i> Back</button> 92. </tr> 93. </section> |

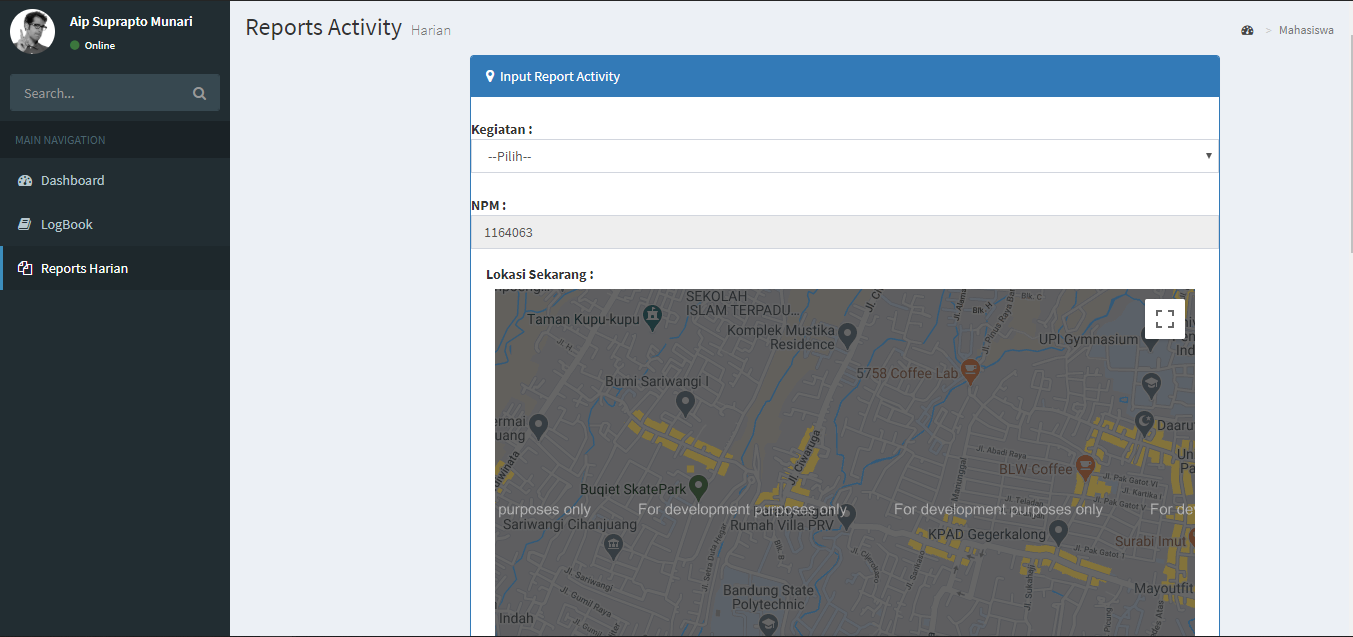
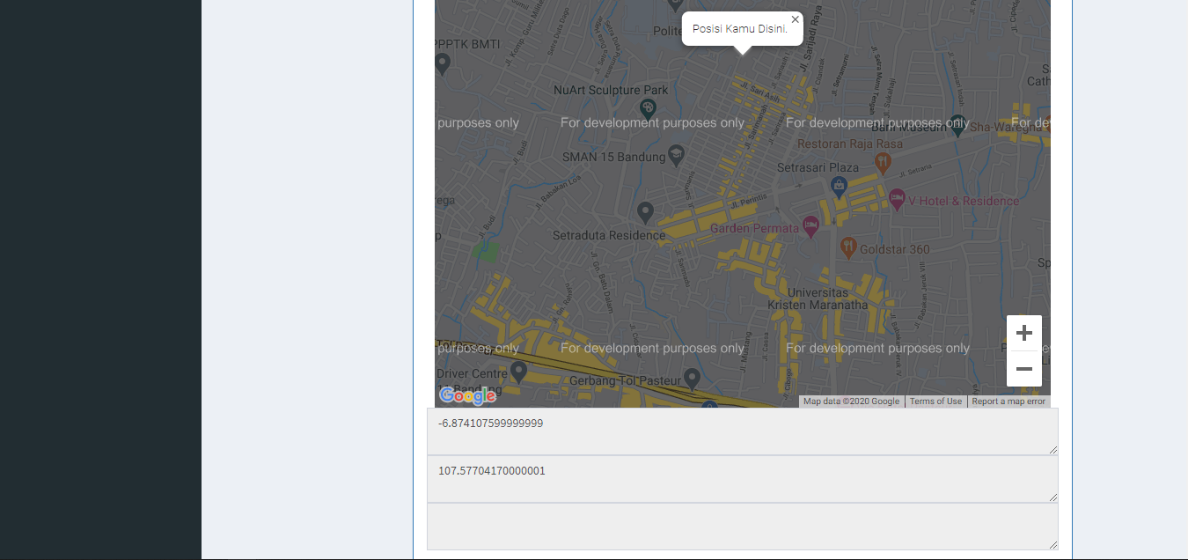
* **nilai.php**

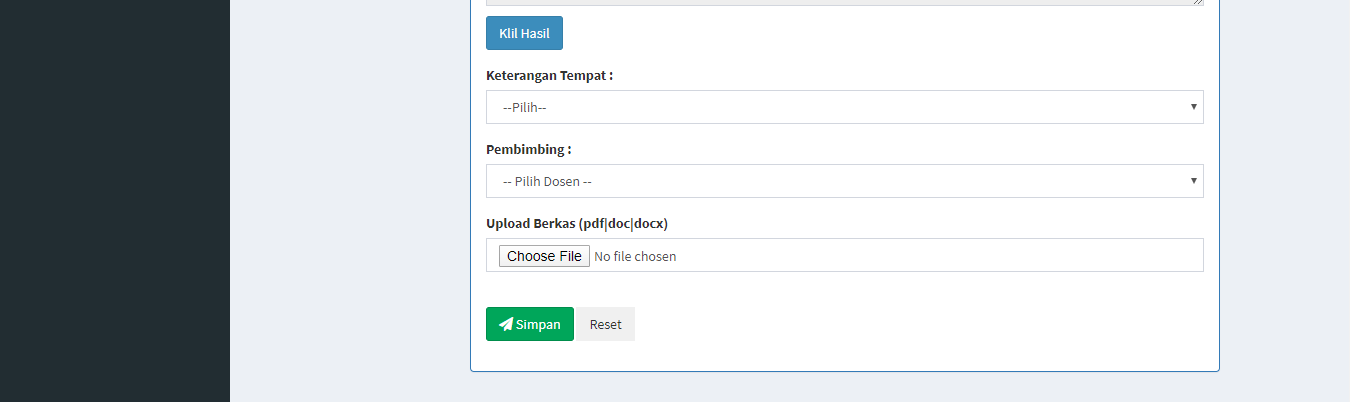
|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Users 3. <small>Pengguna</small> 4. </h1> 5. <ol class="breadcrumb"> 6. <li><a href="#"><i class="fa fa-dashboard"></i></a></li> 7. <li class="active">Users</li> 8. </ol> 9. </section> 11. <!-- Main content --> 12. <section class="content"> 14. <div class="box"> 15. <div class="box-header"> 16. <h3 class="box-title">Tambah Nilai</h3> 17. <div class="pull-right"> 18. <a href="<?=site\_url('laporan/datareportkegiatan')?>" 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('nilai') ? 'has-error' : null?>"> 27. <label>Nilai \*</label> 28. <input type="hidden" name="nilai" value="<?=$row->nilai?>"> 29. <input type="text" name="nilai" value="<?=$this->input->post('nilai') ? $this->input->post('nilai') : $row->nilai?>" class="form-control"> 30. <?=form\_error('nilai')?> 31. </div> 32. <div class="form-group"> 33. <button type="submit" class="btn btn-primary btn-flat"> 34. <i class="fa fa-paper-plane"></i> Simpan 35. </button> 36. <button type="Reset" class="btn btn-flat">Reset</button> 37. </div> 38. </form> 39. </div> 40. </div> 41. </div> 42. </div> 43. </section> |

Keterangan :

Pada folder *views* merupakan isi *file*-*file* untuk desain antarmuka *website*.

Untuk digunakan-nya folder *views* supaya mempermudah pengelompokan desain antarmuka yang anda buat. *Source* *code* diatas adalah tampilan untuk interkasi aplikasi yang dibangun yaitu diantara-nya proses untuk menampilkan *input* *laporan*, *view* laporan dan *detail* laporan. Dan untuk tampilan yang interaksi-nya sebagai berikut :

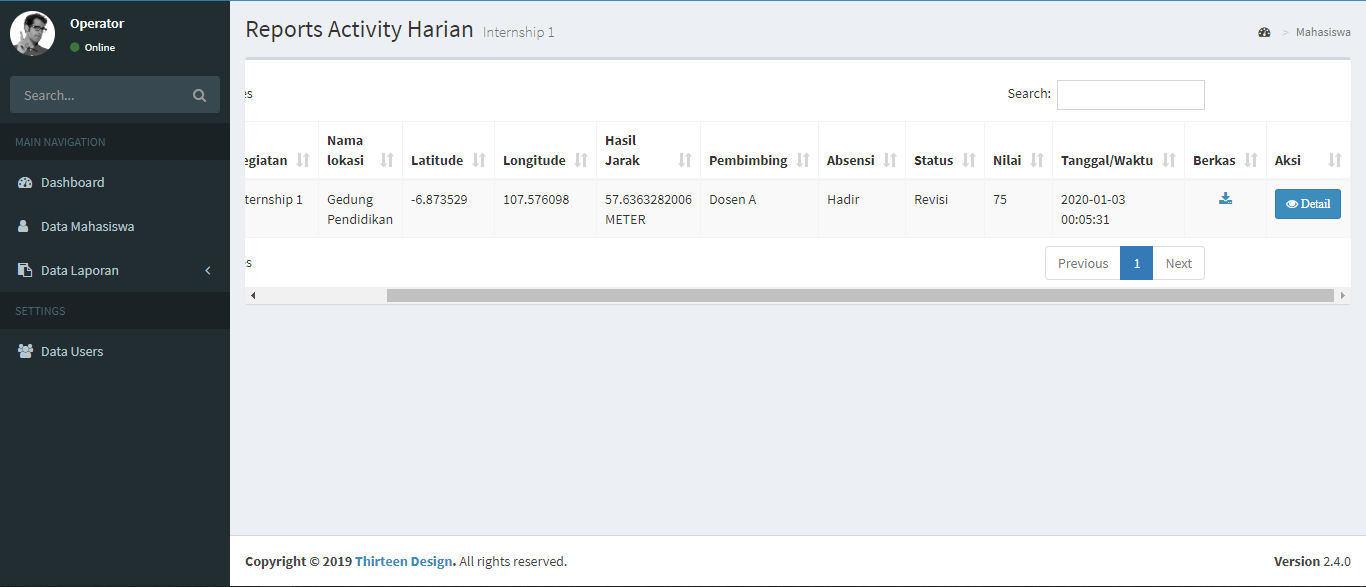




Gambar 5.14 Tampilan *Input* Laporan

Sumber : *Template* AdminLTE versi 2.4.0

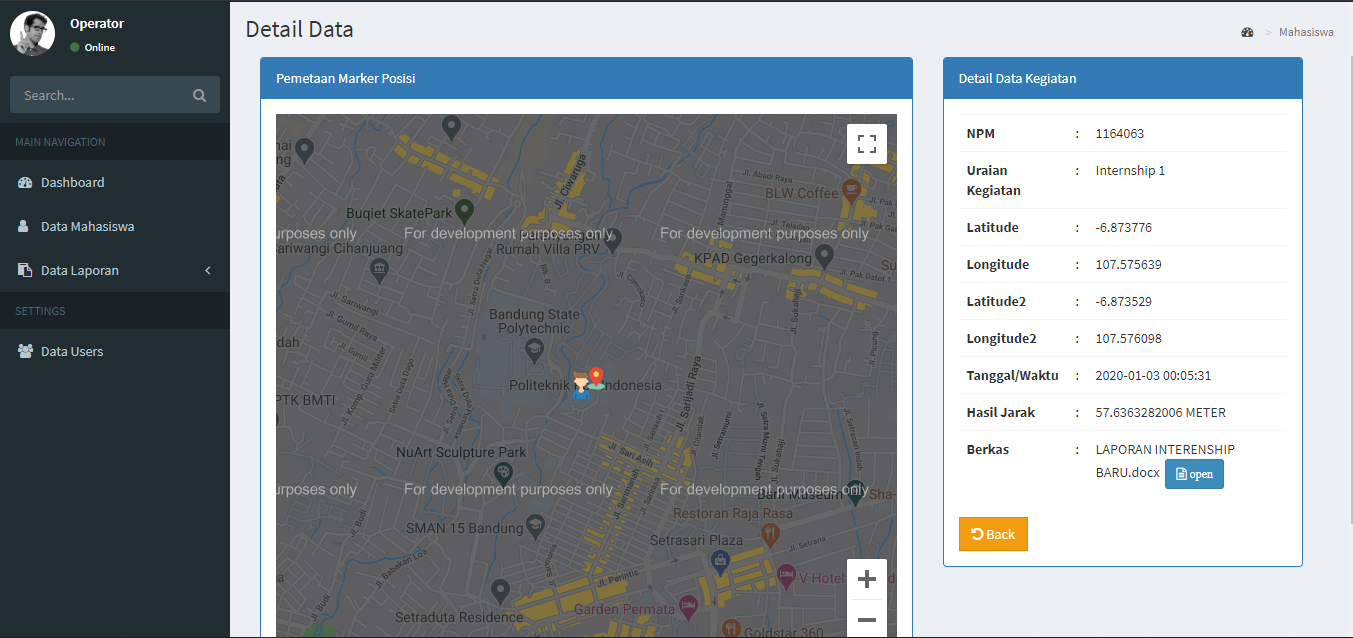
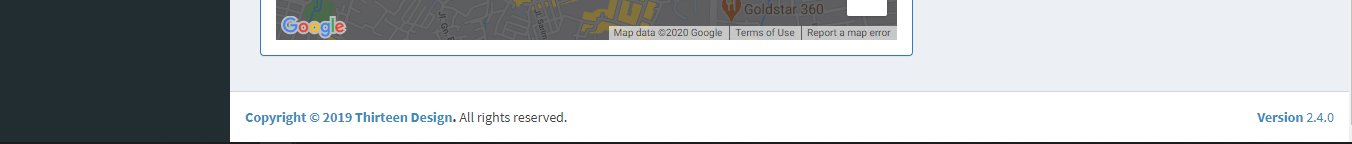
Pada gambar tersebut menjelaskan halaman *form* sebagai laporan kegiatan yang harus diisi oleh mahasiswa/i dalam proses kegiatan *internship* perhari-nya. Karena, laporan tersebut dibutuhkan sebagai bukti rekapan aktivitas kegiatan atau disebut *logbook*. Cara penggunaan pada halaman ini, aktor atau mahasiswa/i tinggal mengisi form yang tersedia. Mulai dari pengisian keterangan kegiatan serta *upload* berkas yang ditujukan untuk pembimbing. Sistem akan mengirimkan data-data yang telah diberi aksi dan menampilkan-nya sesuai *user* yang dipilih. Selanjutnya perhitungan jarak yang ditampilkan pada *form* laporan adalah sebagai jarak mahasiswa/i pada saat posisi titik awal ke titik posisi saat mengirimkan berkas. Jadi dengan bantuan dari algoritma *harvesine formula*, sistem akan menghitung jarak tersebut dan menampilkan hasil-nya dibawah *form* titik koordinat *longitude* apabila dilakukan aksi oleh aktor dengan menekan tombol klik hasil.



Gambar 5.15 *View* Data Laporan

Sumber : *Template* AdminLTE versi 2.4.0

Pada halaman ini adalah tampilan untuk data laporan kegiatan *internship* yang telah diinput oleh mahasiswa. Aktor yang bisa mengakses pada halaman ini adalah admin, operator dan koordinator. Cara penggunaan untuk masuk ke halaman tersebut, langkah pertama aktor yang bersangkutan memilih menu data laporan dan sistem akan menampilkan opsi pilihan seperti memunculkan opsi *internship* I dan *internship* II. Setelah memilih opsi salah satu contohnya memilih opsi *internship* I maka sistem akan menampilkan data seperti pada gambar 5.15.



Gambar 5.16 *Detail* Data Laporan

Sumber : *Template* AdminLTE versi 2.4.0

Proses ini adalah lanjutan dari gambar 5.15 apabila aktor ingin melihat data secara terperinci. Cara penggunaan-nya, aktor yang mengakses halaman ini cukup menekan tombol detail dan sistem akan memprosesnya serta menampilkan rincian data-data yang sebelumnya telah diinput.

1. **Logbook**
2. **controllers/Logbook.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Logbook extends CI\_Controller { 5. function \_\_construct() 6. { 7. parent::\_\_construct(); 8. check\_not\_login(); 9. $this->load->model('logbook\_m'); 10. $this->load->model('laporan\_m'); 11. $this->load->library(array('googlemaps')); 13. } 15. public function index() 16. { 18. $data = array('' => ''); 19. $this->template->load('template', 'logbook/logbook\_data', $data, FALSE); 20. } 22. public function tampildata() 23. { 24. $where = $this->session->userdata('iduser'); 25. $get\_data = $this->laporan\_m->tbl\_mhs($where); 26. if ($get\_data->num\_rows() > 0) { 27. $data = $get\_data->row(); 28. $npm = $data->npm; 29. } 30. $data = array( 31. 'logbook'=> $this->logbook\_m->tampildata($npm), 32. ); 34. $this->template->load('template', 'logbook/logbook\_data', $data); 35. } 37. public function detail($id) 38. { 39. $where = $this->session->userdata('iduser'); 40. $get\_data = $this->laporan\_m->tbl\_mhs($where); 41. if ($get\_data->num\_rows() > 0) { 42. $data = $get\_data->row(); 43. $npm = $data->npm; 44. } 46. check\_not\_login(); 47. $this->load->library('googlemaps'); 48. $config['center'] = '-6.873074, 107.575958'; 49. $config['zoom'] = '15'; 50. $this->googlemaps->initialize($config); 51. $ID = $this->uri->segment('3'); 52. $lokasi=$this->logbook\_m->logbook\_data\_11($npm); 53. foreach ($lokasi as $key => $value) { 54. $marker = array(); 55. $marker['animation'] = 'DROP'; 56. $marker['position'] = "$value->latitude2, $value->longitude2"; 57. $marker['infowindow\_content'] = '<div class="media" style="width:300px">'; 58. $marker['infowindow\_content'] .= '<div class="media-left">'; 59. $marker['infowindow\_content'] .= '</div>'; 60. $marker['infowindow\_content'] .= '<div class="media-body">'; 61. $marker['infowindow\_content'] .= '<a>'.$value->lokasi.'</a><br>'; 62. $marker['infowindow\_content'] .= '<a>'.$value->latitude2.'</a><br>'; 63. $marker['infowindow\_content'] .= '<a>'.$value->longitude2.'</a><br>'; 64. $marker['infowindow\_content'] .= '</div>'; 65. $marker['infowindow\_content'] .= '</div>'; 66. $marker['icon'] = base\_url('assets/icon/location.png'); 68. $this->googlemaps->add\_marker($marker); 70. } 71. $this->load->library('googlemaps'); 72. $config['center'] = '-6.873074, 107.575958'; 73. $config['zoom'] = '15'; 74. $this->googlemaps->initialize($config); 75. $ID = $this->uri->segment('3'); 76. $lokasi=$this->logbook\_m->logbook\_data\_2(); 77. foreach ($lokasi as $key => $value) { 78. $marker = array(); 79. $marker['animation'] = 'DROP'; 80. $marker['position'] = "$value->latitude, $value->longitude"; 81. $marker['infowindow\_content'] = '<div class="media" style="width:300px">'; 82. $marker['infowindow\_content'] .= '<div class="media-left">'; 83. $marker['infowindow\_content'] .= '</div>'; 84. $marker['infowindow\_content'] .= '<div class="media-body">'; 85. $marker['infowindow\_content'] .= '<a>'.$value->lokasi.'</a><br>'; 86. $marker['infowindow\_content'] .= '<a>'.$value->latitude.'</a><br>'; 87. $marker['infowindow\_content'] .= '<a>'.$value->longitude.'</a><br>'; 88. $marker['infowindow\_content'] .= '</div>'; 89. $marker['infowindow\_content'] .= '</div>'; 90. $marker['icon'] = base\_url('assets/icon/location.png'); 92. $this->googlemaps->add\_marker($marker); 93. } 94. $this->googlemaps->initialize($config); 95. $map= $this->googlemaps->create\_map(); 96. $where = $this->session->userdata('iduser'); 97. $get\_data = $this->laporan\_m->tbl\_mhs($where); 98. if ($get\_data->num\_rows() > 0) { 99. $data = $get\_data->row(); 100. $npm = $data->npm; 101. $detail=$this->logbook\_m->logbook\_data\_11($npm); 102. $data = array( 103. 'map' =>$map, 104. 'detail' => $detail 105. ); 106. $this->template->load('template', 'logbook/detail\_logbook', $data, FALSE); 108. } 109. } 111. public function cetak() 112. { 113. check\_not\_login(); 114. $detail = $this->logbook\_m->logbook\_data\_1(); 115. $data = array( 116. 'detail' => $detail 117. ); 118. $this->template->load('template', 'logbook/print\_logbook', $data, FALSE); 119. } 121. public function edit($id) 122. { 123. $this->form\_validation->set\_rules('id', 'id'); 124. $this->form\_validation->set\_rules('upload', 'Upload', 'required'); 125. $this->form\_validation->set\_error\_delimiters('<span class="help-block">', '</span>'); 127. if ($this->form\_validation->run() == FALSE) { 128. $query = $this->logbook\_m->get($id); 129. if($query->num\_rows() > 0) { 130. $data['row'] = $query->row(); 131. $this->template->load('template', 'logbook/logbook\_form\_edit', $data); 132. } else { 133. echo "<script>alert('Data tidak ditemukan');"; 134. echo "window.location='".site\_url('logbook/logbook\_form\_edit')."';</script>"; 135. } 136. } else { 137. $post = $this->input->post(null, TRUE); 138. $this->logbook\_m->edit($post); 139. if($this->db->affected\_rows() > 0) { 140. echo "<script>alert('Data berhasil disimpan');</script>"; 141. } 142. echo "<script>window.location='".site\_url('logbook/tampildata')."';</script>"; 143. } 145. } 146. } |

Keterangan :

Untuk bagian konten ke-4 tidak beda jauh dengan konfigurasi penjelasan sebelumnya dikarena-kan anatara konten ke-3 hampir sama dalam mengeksekusi pengolahaan datanya. Hanya berbeda sedikit untuk penambahan fungsi seperti *edit* data, dan *print out*.

1. **models/Logbook\_m.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Logbook\_m extends CI\_Model { 5. public function tampildata($ID) 6. { 7. $this->db->select('k.lokasi\_id, k.id, k.npm, k.date, k.u\_kegiatan, k.latitude2, k.longitude2, k.hasil, k.absensi, k.status, k.nilai, k.upload ,l.lokasi'); 8. $this->db->from('tb\_kegiatan k'); 9. $this->db->join('tb\_lokasi l', 'l.id = k.lokasi\_id'); 10. $this->db->where('k.npm', $ID); 11. $query = $this->db->get(); 12. return $query->result(); 13. } 15. public function logbook\_data\_11($a) 16. { 17. $this->db->select('a.\*, b.npm, b.latitude as lat1, b.longitude as long1, c.lokasi'); 18. $this->db->from('tb\_kegiatan a'); 19. $this->db->join('v\_login b', 'a.npm = b.npm', 'left'); 20. $this->db->join('tb\_lokasi c', 'c.id = a.lokasi\_id', 'left'); 21. $this->db->where('b.npm', $a); 22. $query=$this->db->get(); 23. return $query->result(); 24. } 26. public function logbook\_data\_1() 27. { 28. $this->db->select('\*'); 29. $this->db->from ('tb\_mhs'); 30. $this->db->join('tb\_kegiatan', 'tb\_kegiatan.npm=tb\_mhs.npm'); 31. $query = $this->db->get(); 32. return $query->result(); 33. } 35. public function logbook\_data\_2() 36. { 37. $this->db->select('\*'); 38. $this->db->from('tb\_mhs'); 39. $query=$this->db->get(); 40. return $query->result(); 41. } 43. public function get($id = null) 44. { 45. $this->db->from('tb\_kegiatan'); 46. if($id != null) { 47. $this->db->where('id', $id); 48. } 49. $query = $this->db->get(); 50. return $query; 51. } 53. public function edit($post) 54. { 55. $params['upload'] = $post['upload']; 56. $this->db->where('id', $post['id']); 57. $this->db->update('tb\_kegiatan', $params); 58. } 59. } |

Keterangan :

Pada folder *models* terdapat *file*-*file* yang mempresentasikan fungsi-fungsi *script* yang dimana alur tersebut digunakan untuk pengelolaan data ke *database*.

1. **views/logbook**

* **logbook\_data.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">Pemetaan Marker Posisi</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">Detail Data Kegiatan</div> 27. <div class="panel-body"> 29. <table class="table"> 30. <?php $no=1 ; 31. foreach($detail as $detail): ?> 32. <tr> 33. <!-- <td> 34. <img src="<?php echo base\_url(); ?>assets/img/<?php echo $detail->foto; ?> 35. "width="90" height="110"> 36. </td> 37. <td></td> --> 38. </tr> 39. <tr> 40. <th>NPM</th> 41. <td>:</td> 42. <td><?php echo $detail->npm; ?></td> 43. </tr> 44. <tr> 45. <th>Uraian Kegiatan</th> 46. <td>:</td> 47. <td><?php echo $detail->u\_kegiatan; ?></td> 48. </tr> 49. <tr> 50. <th>Latitude</th> 51. <td>:</td> 52. <td><?php echo $detail->lat1; ?></td> 53. </tr> 54. <tr> 55. <th>Longitude</th> 56. <td>:</td> 57. <td><?php echo $detail->long1; ?></td> 58. </tr> 59. <tr> 60. <th>Latitude2</th> 61. <td>:</td> 62. <td><?php echo $detail->latitude2; ?></td> 63. </tr> 64. <tr> 65. <th>Longitude2</th> 66. <td>:</td> 67. <td><?php echo $detail->longitude2; ?></td> 68. </tr> 69. <tr> 70. <th>Tanggal/Waktu</th> 71. <td>:</td> 72. <td><?php echo $detail->date; ?></td> 73. </tr> 74. <tr> 75. <th>Hasil Jarak</th> 76. <td>:</td> 77. <td><?php echo $detail->hasil; ?></td> 78. </tr> 79. <tr> 80. <th>Berkas</th> 81. <td>:</td> 82. <td><?php echo $detail->upload; ?></td> 83. </tr> 84. <tr> 85. <th>Nilai</th> 86. <td>:</td> 87. <td><?php echo $detail->nilai; ?></td> 88. </tr> 89. <?php endforeach;?> 90. </table> 91. <tr> 92. <!-- <button class="btn btn-success" type="submit" > 93. <i class="fa fa-check-square-o"></i> Approve</button> --> 94. <a href="<?=site\_url('logbook/tampildata')?>"> 95. <button class="btn btn-warning btn-flat"><i class="fa fa-undo"></i> Back</button> 96. <a href="<?=base\_url('logbook/cetak')?>"> 97. <button class="btn btn-success btn-flat"><i class="fa fa-print"></i> Print</button> 98. </tr> 99. <./section> |

* **detail\_logbook.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">Pemetaan Marker Posisi</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">Detail Data Kegiatan</div> 27. <div class="panel-body"> 29. <table class="table"> 30. <?php $no=1 ; 31. foreach($detail as $detail): ?> 32. <tr> 33. <!-- <td> 34. <img src="<?php echo base\_url(); ?>assets/img/<?php echo $detail->foto; ?> 35. "width="90" height="110"> 36. </td> 37. <td></td> --> 38. </tr> 39. <tr> 40. <th>NPM</th> 41. <td>:</td> 42. <td><?php echo $detail->npm; ?></td> 43. </tr> 44. <tr> 45. <th>Uraian Kegiatan</th> 46. <td>:</td> 47. <td><?php echo $detail->u\_kegiatan; ?></td> 48. </tr> 49. <tr> 50. <th>Latitude</th> 51. <td>:</td> 52. <td><?php echo $detail->lat1; ?></td> 53. </tr> 54. <tr> 55. <th>Longitude</th> 56. <td>:</td> 57. <td><?php echo $detail->long1; ?></td> 58. </tr> 59. <tr> 60. <th>Latitude2</th> 61. <td>:</td> 62. <td><?php echo $detail->latitude2; ?></td> 63. </tr> 64. <tr> 65. <th>Longitude2</th> 66. <td>:</td> 67. <td><?php echo $detail->longitude2; ?></td> 68. </tr> 69. <tr> 70. <th>Tanggal/Waktu</th> 71. <td>:</td> 72. <td><?php echo $detail->date; ?></td> 73. </tr> 74. <tr> 75. <th>Hasil Jarak</th> 76. <td>:</td> 77. <td><?php echo $detail->hasil; ?></td> 78. </tr> 79. <tr> 80. <th>Berkas</th> 81. <td>:</td> 82. <td><?php echo $detail->upload; ?></td> 83. </tr> 84. <tr> 85. <th>Nilai</th> 86. <td>:</td> 87. <td><?php echo $detail->nilai; ?></td> 88. </tr> 89. <?php endforeach;?> 90. </table> 91. <tr> 92. <!-- <button class="btn btn-success" type="submit" > 93. <i class="fa fa-check-square-o"></i> Approve</button> --> 94. <a href="<?=site\_url('logbook/tampildata')?>"> 95. <button class="btn btn-warning btn-flat"><i class="fa fa-undo"></i> Back</button> 96. <a href="<?=base\_url('logbook/cetak')?>"> 97. <button class="btn btn-success btn-flat"><i class="fa fa-print"></i> Print</button> 98. </tr> 99. </section> |

* **logbook\_form\_edit.php**

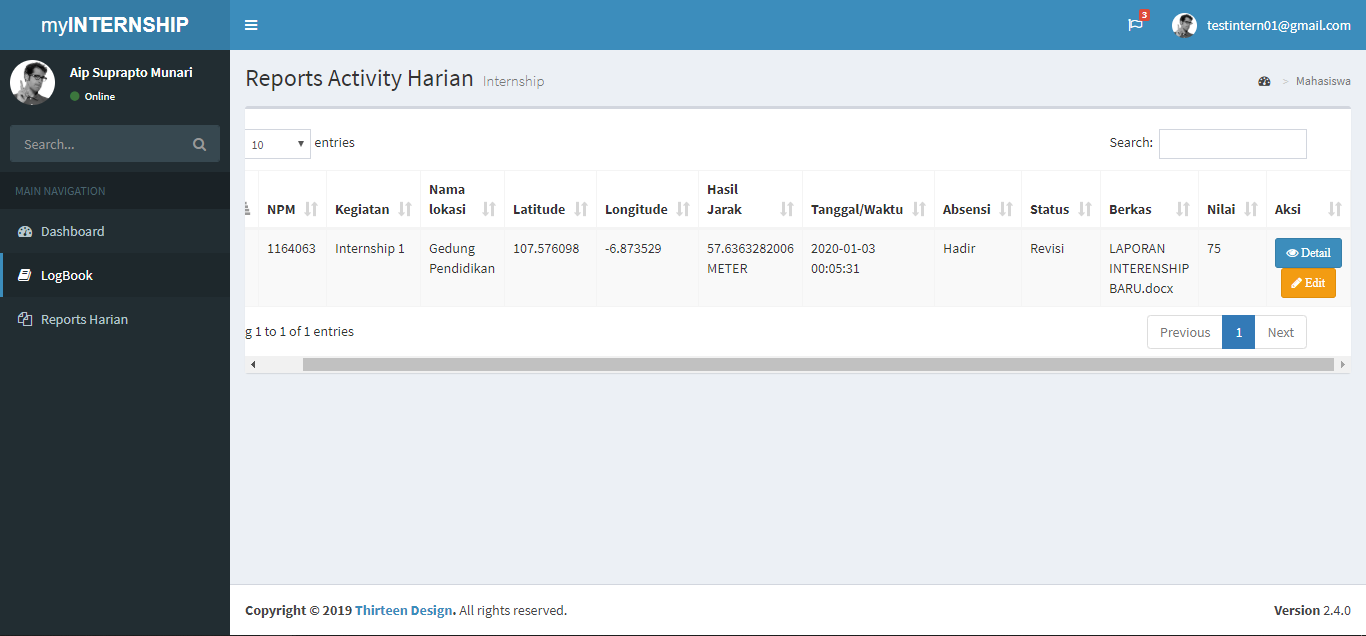
|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Logbook 3. <small>Internship 1</small> 4. </h1> 5. <ol class="breadcrumb"> 6. <li><a href="#"><i class="fa fa-dashboard"></i></a></li> 7. <li class="active">Mahasiswa</li> 8. </ol> 9. </section> 11. <!-- Main content --> 12. <section class="content"> 14. <div class="box"> 15. <div class="box-header"> 16. <div class="pull-right"> 17. <a href="<?=site\_url('logbook/tampildata')?>" class="btn btn-warning btn-flat"> 18. <i class="fa fa-undo"></i> Back 19. </a> 20. </div> 21. <div class="box-body"> 22. <div class="row"> 23. <div class="col-md-4 col-md-offset-4"> 24. <form action="" method="post"> 25. <div class="form-group <?=form\_error('id') ? 'has-error' : null?>"> 26. <input type="hidden" name="id" value="<?=$row->id?>"> 27. </div> 29. <div class="form-group <?=form\_error('upload') ? 'has-error' : null?>"> 30. <label>Upload \*</label> 31. <input type="file" name="upload"  value="<?=$this->input->post('upload') ? $this->input->post('upload') : $row->upload?>" class="form-control"> 32. <?=form\_error('upload')?> 33. </div> 35. <div class="form-group"> 36. <button type="submit" class="btn btn-primary btn-flat"> 37. <i class="fa fa-paper-plane"></i> Simpan 38. </button> 39. <button type="Reset" class="btn btn-flat">Reset</button> 40. </div> 41. </form> 42. </div> 43. </div> 44. </div> 45. </div> 46. </section> |

* **print\_logbook.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-4 col-md-offset-4"> 16. <div class="panel panel-primary"> 17. <div class="panel-heading">Detail Data Kegiatan</div> 18. <div class="panel-body"> 19. <table class="table"> 20. </section> 21. <script type="text/javascript"> 22. window.print(); 23. </script> 24. <?php $no=1 ; 25. foreach($detail as $detail):  ?> 26. </tr> 27. <tr> 28. <th>NPM</th> 29. <td>:</td> 30. <td><?php echo $detail->npm; ?></td> 31. </tr> 32. <tr> 33. <th>Uraian Kegiatan</th> 34. <td>:</td> 35. <td><?php echo $detail->u\_kegiatan; ?></td> 36. </tr> 37. <tr> 38. <th>Latitude</th> 39. <td>:</td> 40. <td><?php echo $detail->latitude; ?></td> 41. </tr> 42. <tr> 43. <th>Longitude</th> 44. <td>:</td> 45. <td><?php echo $detail->longitude; ?></td> 46. </tr> 47. <tr> 48. <th>Latitude2</th> 49. <td>:</td> 50. <td><?php echo $detail->latitude2; ?></td> 51. </tr> 52. <tr> 53. <th>Longitude2</th> 54. <td>:</td> 55. <td><?php echo $detail->longitude2; ?></td> 56. </tr> 57. <tr> 58. <th>Tanggal/Waktu</th> 59. <td>:</td> 60. <td><?php echo $detail->date; ?></td> 61. </tr> 62. <tr> 63. <th>Hasil Jarak</th> 64. <td>:</td> 65. <td><?php echo $detail->hasil; ?></td> 66. </tr> 67. <tr> 68. <th>Berkas</th> 69. <td>:</td> 70. <td><?php echo $detail->upload; ?></td> 71. </tr> 72. <?php endforeach; ?> 73. </table> 74. </section> |

Keterangan :

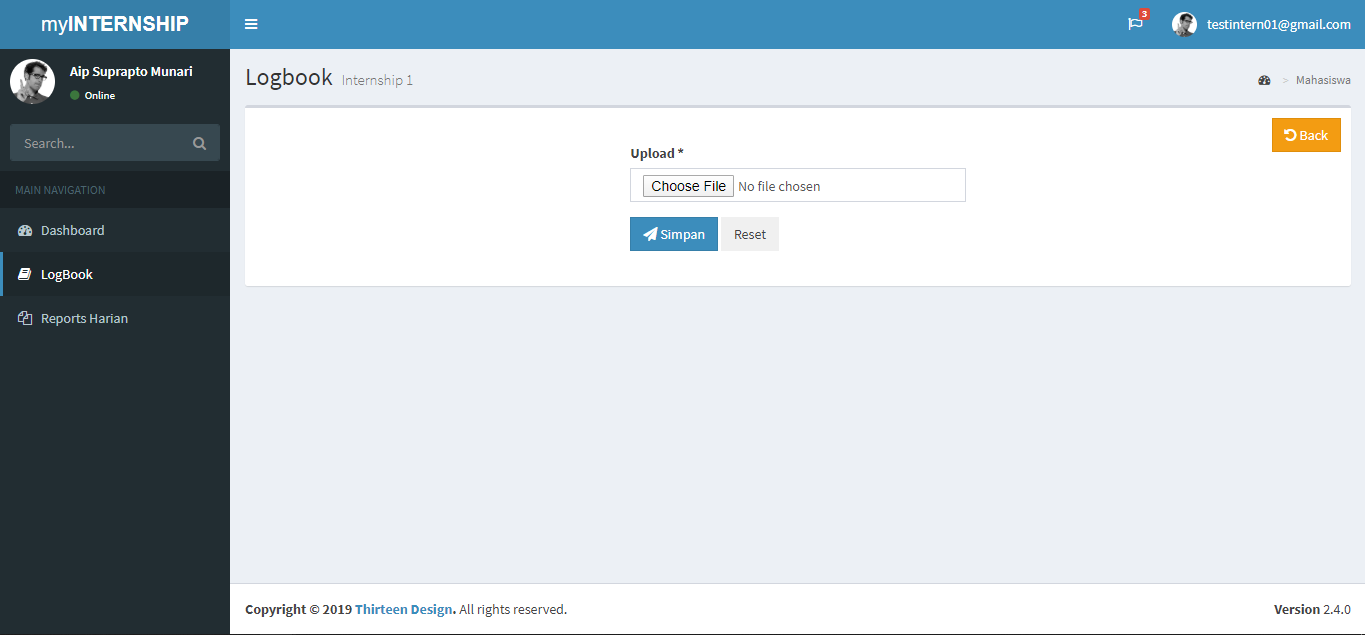
*Source code* yang dicantumkan pada folder *views* adalah berisikan *script-script* desain antarmuka *website* yang dibangun. Desain yang digunakan seperti gambar berikut :



Gambar 5.17 *View* Data Laporan

Sumber : *Template* AdminLTE versi 2.4.0

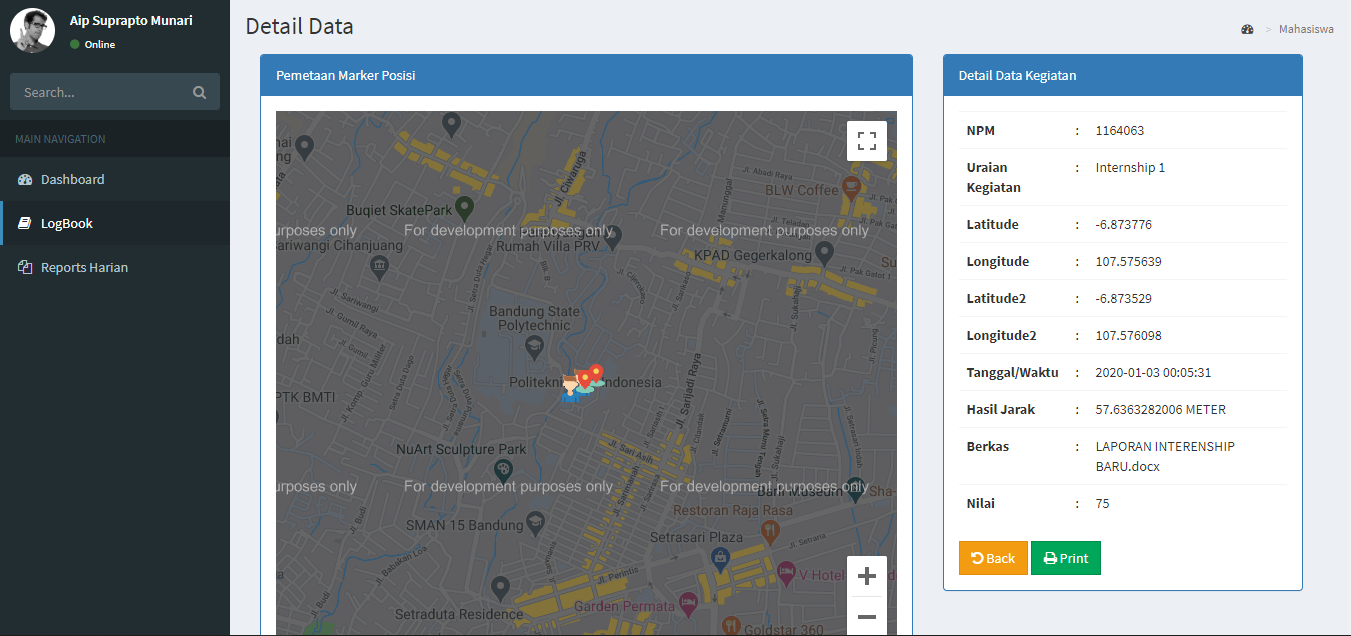
Pada halaman ini adalah sebagai ulasan untuk mahasiswa/i melihat data rekapan laporan atau pemberitahuan apabila data laporan kegiatan sudah dilakukan pengecekan oleh aktor pembimbing. Cara penggunaan untuk proses halaman ini cukup memilih menu yang ber-*icon* gambar buku atau tulisan menu *Logbook*.



Gambar 5.18 *Edit* Data Laporan

Sumber : *Template* AdminLTE versi 2.4.0

Untuk masuk ke halaman tersebut, apabila data laporan sudah dilakukan pengecekan atau status dalam laporan berubah menjadi revisi. Jadi aktor disini adalah mahasiswa mengirimkan kembali berkas kegiatan ke dosen pembimbing untuk dilakukan-nya pengecekan ulang. Cara penggunaan, mahasiswa/i cukup memilih menu *Logbook* dan sistem akan menampilkan data. Lalu mahasiswa terlebih dahulu melihat status data laporan apabila direvisi maka dilakukan-nya aksi *edit* data dengan menekan tombol berwarna kuning yang digambarkan dengan *icon* pensil pada gambar 5.17.



Gambar 5.19 *Detail* Data Laporan

Sumber : *Template* AdminLTE versi 2.4.0

Proses ini adalah lanjutan dari gambar 5.15 apabila mahasiswa/i ingin melihat data secara terperinci. Cara penggunaan-nya, aktor yang mengakses halaman ini cukup menekan tombol detail dan sistem akan memprosesnya serta menampilkan rincian data-data yang sebelumnya telah diinput. Selanjutnya apabila mahasiswa/i membutuhkan tanda bukti dalam bentuk kertas, aktor tinggal menekan tombol *print* untuk mencetaknya.

1. **Penilaian**
2. **controllers/Penilaian.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Penilaian extends CI\_Controller { 5. function \_\_construct() 6. { 7. parent::\_\_construct(); 8. check\_not\_login(); 9. $this->load->model('penilaian\_m'); 10. $this->load->library('form\_validation'); 11. $this->load->library(array('googlemaps')); 12. } 14. public function index() 15. { 16. check\_not\_login(); 17. $ID = $this->uri->segment('3'); 18. $penilaian=$this->penilaian\_m->getreport($ID); 19. $data = array( 20. 'penilaian' =>$penilaian 21. ); 22. $this->template->load('template', 'penilaian/data\_penilaian', $data, FALSE); 23. } 25. public function index1() 26. { 27. check\_not\_login(); 28. $ID = $this->uri->segment('3'); 29. $penilaian=$this->penilaian\_m->getreport1($ID); 30. $data = array( 31. 'penilaian' =>$penilaian 32. ); 33. $this->template->load('template', 'penilaian/data\_penilaian1', $data, FALSE); 34. } 36. public function tampil() 37. { 38. $where = $this->session->userdata('iduser'); 39. $get\_data = $this->penilaian\_m->siswa($where); 40. if ($get\_data->num\_rows() > 0) { 41. $data = $get\_data->row(); 42. $npm = $data->id\_user; 43. } 44. $penilaian=$this->penilaian\_m->tampil($where); 45. $data = array( 46. 'penilaian'   =>$penilaian 47. ); 48. $this->template->load('template', 'penilaian/tampil\_data', $data, FALSE); 49. } 51. public function history() 52. { 53. $where = $this->session->userdata('iduser'); 54. $get\_data = $this->penilaian\_m->siswa1($where); 55. if ($get\_data->num\_rows() > 0) { 56. $data = $get\_data->row(); 57. $npm = $data->id\_user; 58. } 59. $penilaian=$this->penilaian\_m->history($where); 60. $data = array( 61. 'penilaian'   =>$penilaian 62. ); 63. $this->template->load('template', 'penilaian/data\_history', $data, FALSE); 64. } 66. public function edit($id) 67. { 68. $this->form\_validation->set\_rules('id', 'id'); 69. $this->form\_validation->set\_rules('absensi', 'Absensi'); 70. $this->form\_validation->set\_rules('status', 'Status', 'required'); 71. $this->form\_validation->set\_rules('nilai', 'Nilai'); 72. $this->form\_validation->set\_rules('komentar', 'Komentar'); 73. $this->form\_validation->set\_error\_delimiters('<span class="help-block">', '</span>'); 75. if ($this->form\_validation->run() == FALSE) { 76. $query = $this->penilaian\_m->get($id); 77. if($query->num\_rows() > 0) { 78. $data['row'] = $query->row(); 79. $this->template->load('template', 'penilaian/penilaian\_form\_edit', $data); 80. } else { 81. echo "<script>alert('Data tidak ditemukan');"; 82. echo "window.location='".site\_url('penilaian/tampil')."';</script>"; 83. } 84. } else { 85. $post = $this->input->post(null, TRUE); 86. $this->penilaian\_m->edit($post); 87. if($this->db->affected\_rows() > 0) { 88. echo "<script>alert('Data berhasil disimpan');</script>"; 89. } 90. echo "<script>window.location='".site\_url('penilaian/index')."';</script>"; 91. } 93. } 95. public function detail() 96. { 97. check\_not\_login(); 98. $this->load->library('googlemaps'); 99. $config['center'] = '-6.873074, 107.575958'; 100. $config['zoom'] = '15'; 101. $this->googlemaps->initialize($config); 102. $lokasi=$this->penilaian\_m->tampil\_marker(); 103. foreach ($lokasi as $key => $value) { 104. $marker = array(); 105. $marker['animation'] = 'DROP'; 106. $marker['position'] = "$value->latitude2, $value->longitude2"; 107. $marker['infowindow\_content'] = '<div class="media" style="width:300px">'; 108. $marker['infowindow\_content'] .= '<div class="media-left">'; 109. $marker['infowindow\_content'] .= '</div>'; 110. $marker['infowindow\_content'] .= '<div class="media-body">'; 111. $marker['infowindow\_content'] .= '<a>'.$value->lokasi.'</a><br>'; 112. $marker['infowindow\_content'] .= '<a>'.$value->latitude2.'</a><br>'; 113. $marker['infowindow\_content'] .= '<a>'.$value->longitude2.'</a><br>'; 114. $marker['infowindow\_content'] .= '</div>'; 115. $marker['infowindow\_content'] .= '</div>'; 116. $marker['icon'] = base\_url('assets/icon/location.png'); 118. $this->googlemaps->add\_marker($marker); 119. } 120. $this->googlemaps->initialize($config); 121. $ID = $this->uri->segment('3'); 122. $detail=$this->penilaian\_m->tampil\_chart($ID); 123. $detail=$this->penilaian\_m->tampil\_marker\_1($ID); 124. $map= $this->googlemaps->create\_map(); 125. $data = array( 126. 'map' =>$map, 127. 'detail' =>$detail 128. ); 129. $this->template->load('template', 'penilaian/detail\_data', $data, FALSE); 130. } 131. } |

Keterangan :

Sebagaimana untuk folder *controllers* adalah *file*-*file* untuk mengatur alur fungsi yang dibuat. Jadi penjelasan pada bagian ini cukup singkat. Sebab mempresentasikan teori atau alur kerjanya MVC.

1. **models/Penilaian\_m.php**

|  |
| --- |
| 1. <?php defined('BASEPATH') OR exit('No direct script access allowed'); 3. class Penilaian\_m extends CI\_Model { 5. public function getreport($ID) 6. { 7. $this->db->select('b.nama, a.u\_kegiatan, c.lokasi, a.absensi, a.latitude2, a.status, a.longitude2, a.dosen, a.hasil, a.date, a.upload, a.id'); 8. $this->db->from('tb\_kegiatan a'); 9. $this->db->join('tb\_mhs b', 'b.npm = a.npm', 'left'); 10. $this->db->join('tb\_lokasi c', 'c.id = a.lokasi\_id', 'left'); 11. $this->db->where('a.status',1); 12. $this->db->where('a.npm', $ID); 13. return $this->db->get()->result(); 14. } 16. public function getreport1($ID) 17. { 18. $this->db->select('b.nama, a.u\_kegiatan, c.lokasi, a.absensi, a.status, a.nilai, a.latitude2, a.longitude2, a.dosen, a.hasil, a.date, a.upload, a.id'); 19. $this->db->from('tb\_kegiatan a'); 20. $this->db->join('tb\_mhs b', 'b.npm = a.npm', 'left'); 21. $this->db->join('tb\_lokasi c', 'c.id = a.lokasi\_id', 'left'); 22. $this->db->where('a.npm', $ID); 23. $this->db->where('a.status', 'Diterima'); 24. return $this->db->get()->result(); 25. } 27. public function dosen($ID) 28. { 29. $this->db->select('\*'); 30. $query = $this->db->get\_where('user', array('id\_user' => $ID)); 31. return $query; 32. } 34. public function get($id = null) 35. { 36. $this->db->from('tb\_kegiatan'); 37. if($id != null) { 38. $this->db->where('id', $id); 39. } 40. $query = $this->db->get(); 41. return $query; 42. } 44. public function edit($post) 45. { 46. $params['absensi'] = $post['absensi']; 47. $params['nilai'] = $post['nilai']; 48. $params['komentar'] = $post['komentar']; 49. $params['hasil'] = $post['hasil']; 50. $params['status'] = $post['status']; 51. $this->db->where('id', $post['id']); 52. $this->db->update('tb\_kegiatan', $params); 53. } 55. public function tampil($ID) 56. { 57. $this->db->select('b.angkatan, c.kelas, a.npm, a.nama, a.kelas\_id, a.internal\_id, a.angkatan\_id, d.p\_internal, a.id\_user'); 58. $this->db->from('tb\_mhs a'); 59. $this->db->join('tb\_angkatan b', 'b.id = a.angkatan\_id', 'left'); 60. $this->db->join('tb\_kelas c', 'c.id = a.kelas\_id', 'left'); 61. $this->db->join('tb\_p\_internal d', 'd.id = a.internal\_id', 'left'); 62. $this->db->where('a.internal\_id', $ID); 63. return $this->db->get()->result(); 64. } 66. public function siswa($ID) 67. { 68. $this->db->select('\*'); 69. $query = $this->db->get\_where('tb\_mhs', array('npm' => $ID)); 70. return $query; 71. } 73. public function history($ID) 74. { 75. $this->db->select('b.angkatan, c.kelas, a.npm, a.nama, a.kelas\_id, a.internal\_id, a.angkatan\_id, d.p\_internal, a.id\_user'); 76. $this->db->from('tb\_mhs a'); 77. $this->db->join('tb\_angkatan b', 'b.id = a.angkatan\_id', 'left'); 78. $this->db->join('tb\_kelas c', 'c.id = a.kelas\_id', 'left'); 79. $this->db->join('tb\_p\_internal d', 'd.id = a.internal\_id', 'left'); 80. $this->db->where('a.internal\_id', $ID); 81. return $this->db->get()->result(); 82. } 84. public function siswa1($ID) 85. { 86. $this->db->select('\*'); 87. $query = $this->db->get\_where('tb\_mhs', array('npm' => $ID)); 88. return $query; 89. } 91. public function tampil\_marker() 92. { 93. $this->db->select('\*, b.lokasi'); 94. $this->db->from('tb\_kegiatan a'); 95. $this->db->join('tb\_lokasi b', 'b.id = a.lokasi\_id', 'left'); 96. $query=$this->db->get(); 97. return $query->result(); 98. } 100. public function tampil\_marker\_1($ID) 101. { 102. $this->db->select('b.nama, a.u\_kegiatan, a.absensi, a.status, a.nilai, a.dosen, c.lokasi, a.latitude2, a.longitude2, a.dosen, a.hasil, a.date, a.upload, a.id'); 103. $this->db->from('tb\_kegiatan a'); 104. $this->db->join('tb\_mhs b', 'b.npm = a.npm', 'left'); 105. $this->db->join('tb\_lokasi c', 'c.id = a.lokasi\_id', 'left'); 106. $this->db->join('tb\_p\_internal d', 'd.id = a.dosen', 'left'); 107. $this->db->where('a.id', $ID); 108. return $this->db->get()->result(); 109. } |

Keterangan :

Bagian *file* yang berada dalam folder *models* adalah berisi *script-script* yang menghubungkan untuk digunakan sebagai pengelolaan data dari *database* dengan konsep berbasis-kan MVC.

1. **views/penilaian**

* **tampil\_data.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Data Bimbingan 3. <small>internship 1</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. <div class="table-responsive"> 18. <table id="example1" class="table table-bordered table-striped"> 19. <thead> 20. <tr> 21. <th style="width : 50px">No</th> 22. <th>NPM</th> 23. <th>Nama Mahasiswa</th> 24. <th>Program Studi</th> 25. <th>Angkatan</th> 26. <th style="width : 80px">Aksi</th> 27. </tr> 28. </thead> 29. <?php $no=1 ; foreach($penilaian as $penilaian) { ?> 30. <tr> 31. <td><?php echo $no++; ?>.</td> 32. <td><?php echo $penilaian->npm; ?></td> 33. <td><?php echo $penilaian->nama; ?></td> 34. <td><?php echo $penilaian->kelas; ?></i></td> 35. <td><?php echo $penilaian->angkatan; ?></td> 36. <td> 37. <a href="<?php echo base\_url('penilaian/index/'.$penilaian->npm); ?>" class="btn 38. btn-primary btn-sm" type="button"><i class="fa fa-eye"></i> Detail</a> 39. </td> 40. </tr> 41. <?php } ?> 43. </table> 44. </div> 45. </div> 46. </div> 47. </div> 48. </section> |

* **data\_penilaian.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Reports Activity Harian 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 class="table-responsive"> 17. <table id="example2" class="table table-bordered table-striped"> 18. <thead> 19. <tr> 20. <th>No</th> 21. <th>Nama</th> 22. <th>Kegiatan</th> 23. <th>Nama lokasi</th> 24. <th>Latitude</th> 25. <th>Longitude</th> 26. <th>Hasil Jarak</th> 27. <th>Tanggal/Waktu</th> 28. <th>Berkas</th> 29. <th>Aksi</th> 30. </tr> 31. </thead> 32. <tbody> 33. <?php $no=1 ; 34. foreach($penilaian as $penilaian) { ?> 35. <tr> 36. <td><?=$no++?>.</td> 37. <td><?=$penilaian->nama?></td> 38. <td><?=$penilaian->u\_kegiatan?></td> 39. <td><?=$penilaian->lokasi?></td> 40. <td><?=$penilaian->longitude2?></td> 41. <td><?=$penilaian->latitude2?></td> 42. <td><?=$penilaian->hasil?></td> 43. <td><?=$penilaian->date?></td> 44. <td> 45. <a href="<?=base\_url().'./upload/'.$penilaian->upload;?>" target="\_blank"> 46. <center><i class="fa fa-download"></center></i> 47. </a> 48. </td> 49. <td class="text-center" width="160px"> 50. <a href="<?php echo base\_url('penilaian/edit/'.$penilaian->id); ?>" class="btn 51. btn-warning btn-sm" type="button"><i class="fa fa-pencil"> Edit</i></a> 52. <a href="<?php echo base\_url('penilaian/detail/'.$penilaian->id); ?>" class="btn 53. btn-primary btn-sm" type="button"><i class="fa fa-eye"> Detail</i></a> 54. <a href="<?=site\_url('laporan/delete/'.$penilaian->id)?>" onclick="return confirm('Apakah anda yakin hapus data?')" class="btn btn-danger btn-sm"> 55. <i class="fa fa-trash"></i> Delete 56. </a> 57. </td> 58. </tr> 59. <?php 60. } ?> 61. </tbody> 62. </table> 63. </div> 64. </div> 65. <./section> |

* **data\_penilaian1.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Data Diterima 3. <small>Laporan </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. <div class="table-responsive"> 18. <table id="example2" class="table table-bordered table-striped"> 19. <thead> 20. <tr> 21. <th>No</th> 22. <th>Nama</th> 23. <th>Kegiatan</th> 24. <th>Nama lokasi</th> 25. <th>Latitude</th> 26. <th>Longitude</th> 27. <th>Hasil Jarak</th> 28. <th>Tanggal/Waktu</th> 29. <th>Absensi</th> 30. <th>Status</th> 31. <th>Nilai</th> 32. <th>Aksi</th> 33. </tr> 34. </thead> 35. <tbody> 36. <?php $no=1 ; 37. foreach($penilaian as $penilaian) { ?> 38. <tr> 39. <td><?=$no++?>.</td> 40. <td><?=$penilaian->nama?></td> 41. <td><?=$penilaian->u\_kegiatan?></td> 42. <td><?=$penilaian->lokasi?></td> 43. <td><?=$penilaian->longitude2?></td> 44. <td><?=$penilaian->latitude2?></td> 45. <td><?=$penilaian->hasil?></td> 46. <td><?=$penilaian->date?></td> 47. <td><?=$penilaian->absensi?></td> 48. <td><?=$penilaian->status?></td> 49. <td><?=$penilaian->nilai?></td> 50. <td class="text-center" width="160px"> 51. <a href="<?php echo base\_url('penilaian/edit/'.$penilaian->id); ?>" class="btn 52. btn-warning btn-sm" type="button"><i class="fa fa-pencil"> Edit</i></a> 53. </td> 54. </tr> 55. <?php 56. } ?> 57. </tbody> 58. </table> 59. </div> 60. </div> 61. </section> |

* **penilaian\_form\_edit.php**

|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Penilaian 3. <small>Laporan 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> 11. <section class="content"> 13. <div class="box"> 14. <div class="box-header"> 15. <div class="pull-right"> 16. <a href="<?=site\_url('penilaian/tampil')?>" class="btn btn-warning btn-flat"> 17. <i class="fa fa-undo"></i> Back 18. </a> 19. </div> 20. <div class="box-body"> 21. <div class="row"> 22. <div class="col-md-4 col-md-offset-4"> 23. <form action="" method="post"> 24. <div class="form-group <?=form\_error('id') ? 'has-error' : null?>"> 25. <input type="hidden" name="id" value="<?=$row->id?>"> 26. </div> 28. <div class="form-group <?=form\_error('u\_kegiatan') ? 'has-error' : null?>"> 29. <label>Kegiatan \*</label> 30. <input type="text" name="u\_kegiatan" readonly="readonly" value="<?=$this->input->post('u\_kegiatan') ? $this->input->post('u\_kegiatan') : $row->u\_kegiatan?>" class="form-control"> 31. <?=form\_error('u\_kegiatan')?> 32. </div> 34. <div class="form-group <?=form\_error('hasil') ? 'has-error' : null?>"> 35. <label>Hasil Jarak \*</label> 36. <input type="text" name="hasil" readonly="readonly" value="<?=$this->input->post('hasil') ? $this->input->post('hasil') : $row->hasil?>" class="form-control"> 37. <?=form\_error('hasil')?> 38. </div> 40. <div class="form-group <?=form\_error('absensi') ? 'has-error' : null?>"> 41. <label>Absensi \*</label> 42. <select name="absensi" class="form-control"> 43. <?php $absensi = $this->input->post('absensi') ? $this->input->post('absensi') : $row->absensi ?> 44. <option value="1">Belum ada aksi</option> 45. <option value="Alfa"<?=$absensi == 2 ? 'selected' : null?>>Alfa</option> 46. <option value="Hadir"<?=$absensi == 3 ? 'selected' : null?>>Hadir</option> 47. </select> 48. <?=form\_error('status')?> 49. </div> 51. <div class="form-group <?=form\_error('status') ? 'has-error' : null?>"> 52. <label>Status \*</label> 53. <select name="status" class="form-control"> 54. <?php $status = $this->input->post('level') ? $this->input->post('level') : $row->status ?> 55. <option value="1">Belum dicek</option> 56. <option value="Revisi"<?=$status == 2 ? 'selected' : null?>>Revisi</option> 57. <option value="Diterima"<?=$status == 3 ? 'selected' : null?>>Diterima</option> 58. </select> 59. <?=form\_error('status')?> 60. </div> 62. <div class="form-group <?=form\_error('nilai') ? 'has-error' : null?>"> 63. <label>Nilai \*</label> 64. <input type="text" name="nilai" value="<?=$this->input->post('nilai') ? $this->input->post('nilai') : $row->nilai?>" class="form-control"> 65. <?=form\_error('nilai')?> 66. </div> 68. <div class="form-group"> 69. <label>Komentar </label> 70. <textarea name="komentar" class="form-control"><?=$this->input->post('komentar') ? $this->input->post('komentar') : $row->komentar?> 71. </div> 72. <div class="form-group"> 73. <button type="submit" class="btn btn-primary btn-flat"> 74. <i class="fa fa-paper-plane"></i> Simpan 75. </button> 76. <button type="Reset" class="btn btn-flat">Reset</button> 77. </div> 78. </form> 79. </div> 80. </div> 81. </div> 82. </div> 83. </section> |

* **data\_history.php**

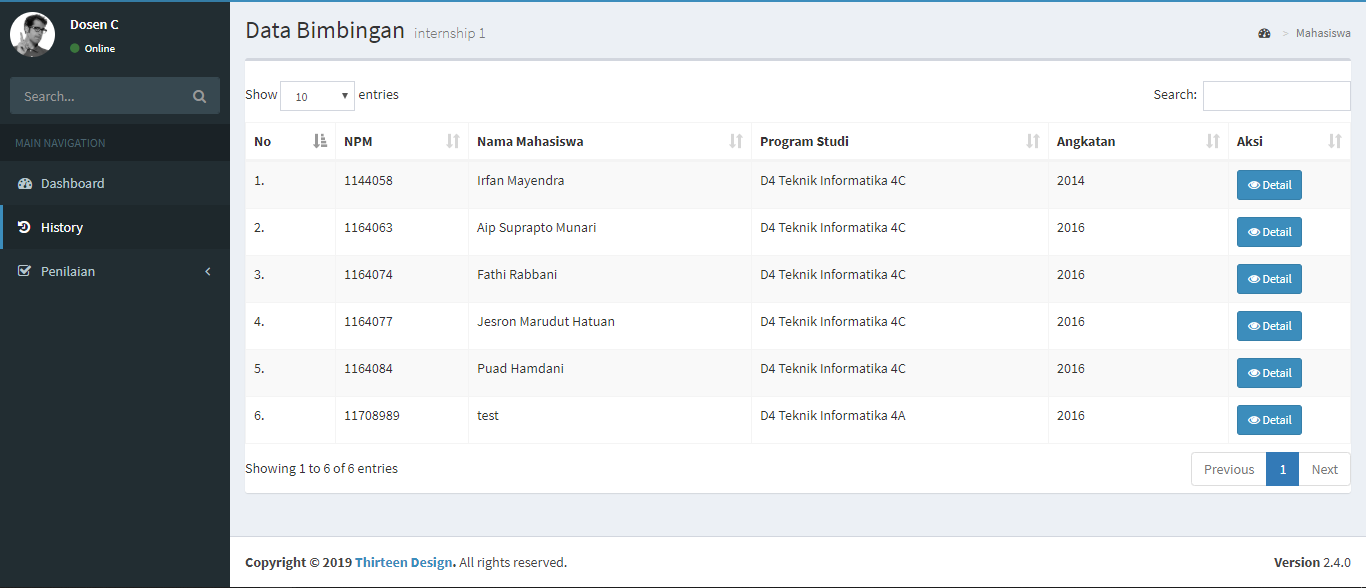
|  |
| --- |
| 1. <section class="content-header"> 2. <h1>Data Diterima 3. <small>Laporan </small> 4. </h1> 5. <ol class="breadcrumb"> 6. <li><a href="#"><i class="fa fa-dashboard"></i></a></li> 7. <li class="active">Mahasiswa</li> 8. </ol> 9. </section> 10. <section class="content"> 11. <div class="row"> 12. <div class="col-xs-12"> 13. <div class="box"> 14. <div class="box-header"> 16. </div> 17. <div class="table-responsive"> 18. <table id="example1" class="table table-bordered table-striped"> 19. <thead> 20. <tr> 21. <th style="width : 50px">No</th> 22. <th>NPM</th> 23. <th>Nama Mahasiswa</th> 24. <th>Program Studi</th> 25. <th>Angkatan</th> 26. <th style="width : 80px">Aksi</th> 27. </tr> 28. </thead> 29. <?php $no=1 ; foreach($penilaian as $penilaian) { ?> 30. <tr> 31. <td><?php echo $no++; ?>.</td> 32. <td><?php echo $penilaian->npm; ?></td> 33. <td><?php echo $penilaian->nama; ?></td> 34. <td><?php echo $penilaian->kelas; ?></i></td> 35. <td><?php echo $penilaian->angkatan; ?></td> 36. <td> 37. <a href="<?php echo base\_url('penilaian/index1/'.$penilaian->npm); ?>" class="btn 38. btn-primary btn-sm" type="button"><i class="fa fa-eye"></i> Detail</a> 39. </td> 40. </tr> 41. <?php } ?> 43. </table> 44. </div> 45. </div> 46. </div> 47. </div> 48. </section> |

* **detail\_data.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 Laporan</div> 27. <div class="panel-body"> 29. <table class="table"> 30. <?php $no=1 ; foreach($detail as $detail) { ?> 32. <tr> 33. <th>Nama Mahasiswa</th> 34. <td>:</td> 35. <td><?php echo $detail->nama; ?></td> 36. </tr> 37. <tr> 38. <th>Nama Lokasi</th> 39. <td>:</td> 40. <td><?php echo $detail->lokasi; ?></td> 41. </tr> 42. <tr> 43. <th>Hasil Jarak</th> 44. <td>:</td> 45. <td><?php echo $detail->hasil; ?></td> 46. </tr> 47. <tr> 48. <th>Tanggal Waktu</th> 49. <td>:</td> 50. <td><?php echo $detail->date; ?></td> 51. </tr> 52. <tr> 53. <th>Berkas</th> 54. <td>:</td> 55. <td><?php echo $detail->upload; ?></td> 56. </tr> 57. <tr> 58. <tr> 59. <th>Nilai</th> 60. <td>:</td> 61. <td><?php echo $detail->nilai; ?></td> 62. <?php } ?> 63. </table> 64. <tr> 65. <button class="btn btn-success btn-flat"><i class="fa fa-plus"></i> Actived</button>  --> 66. <a href="<?=site\_url('penilaian/tampil')?>"> 67. <button class="btn btn-warning btn-flat"><i class="fa fa-undo"></i> Back</button> 68. </tr> 70. <div class="box box-primary"> 71. <div class="box-header with-border"> 72. <i class="fa fa-bar-chart-o"></i> 74. <h3 class="box-title">Donut Chart</h3> 76. <div class="box-tools pull-right"> 77. <button type="button" class="btn btn-box-tool" data-widget="collapse"><i class="fa fa-minus"></i> 78. </button> 79. <button type="button" class="btn btn-box-tool" data-widget="remove"><i class="fa fa-times"></i></button> 80. </div> 81. </div> 82. <div class="box-body"> 83. <div id="donut-chart" style="height: 300px;"></div> 84. </div> 85. </div> 86. </div> 87. </div> 88. </div> 89. </div> 90. </div> 91. </section> |

Keterangan :

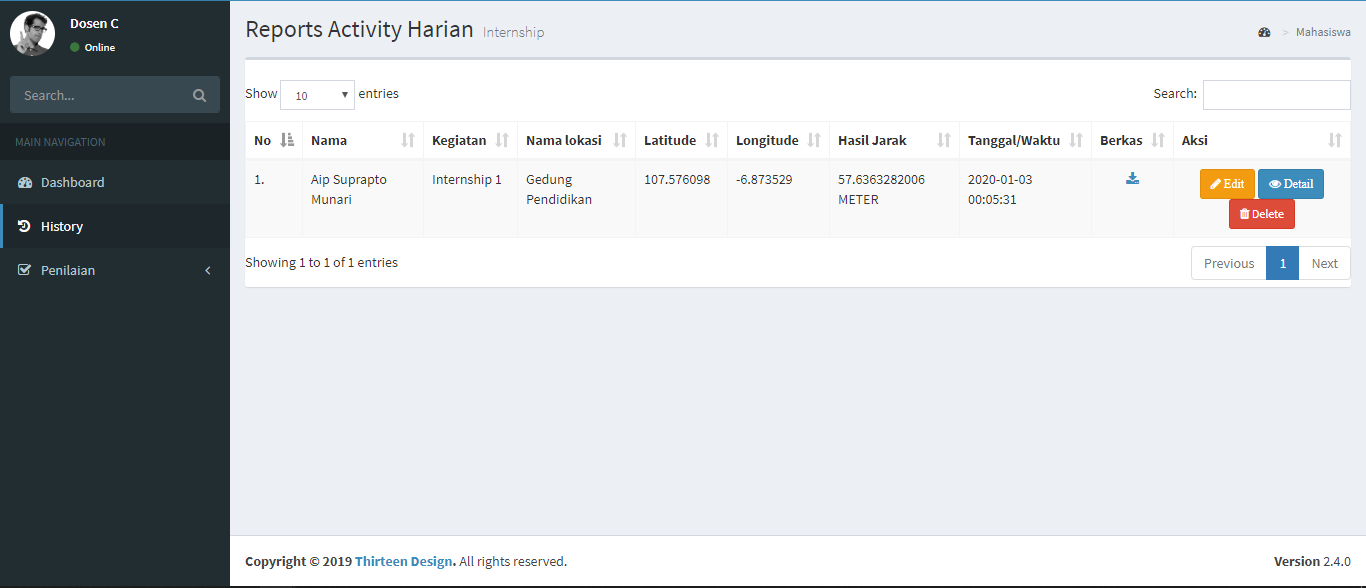
Folder yang berisi-kan *file*-*file* untuk desain antarmuka *website* yang dibangun dan contoh untuk antarmuka pada konten penilaian adalah sebagai berikut :



Gambar 5.20 Tampilan Data Bimbingan

Sumber : *Template* AdminLTE versi 2.4.0

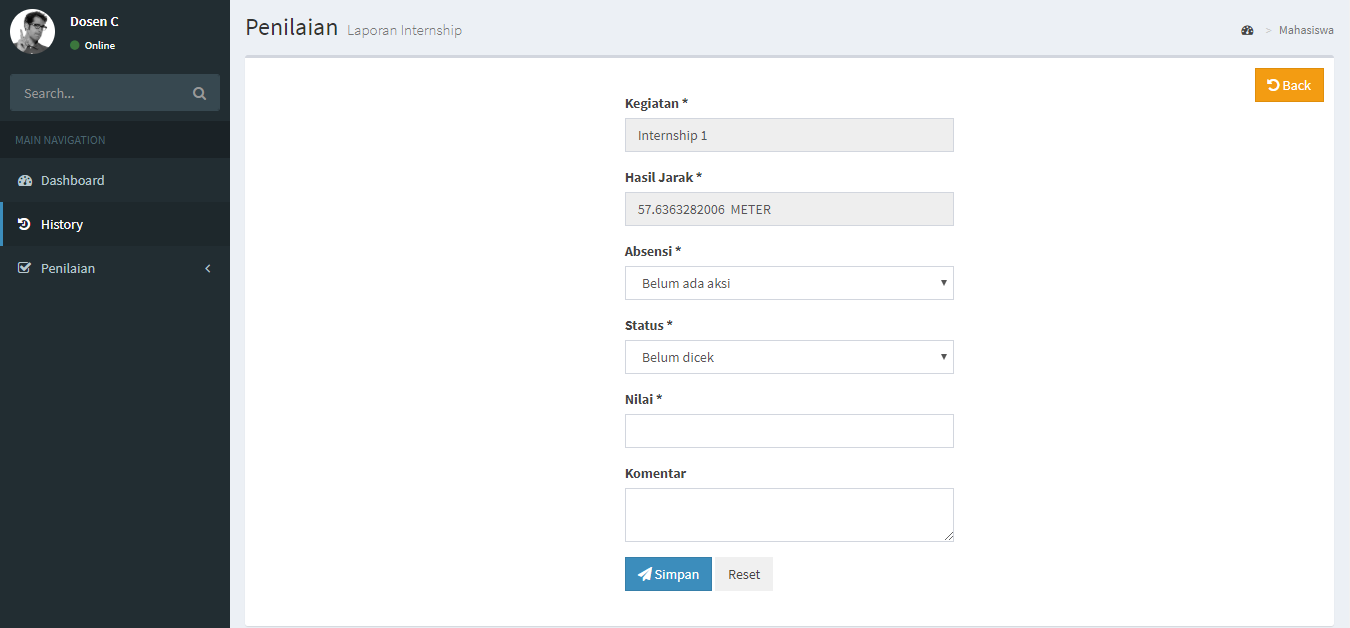
Untuk menampilkan halaman tersebut yang bisa mengakses adalah aktor pembimbing. Dimana pada proses ini data yang muncul adalah data bimbingan mahasiswa/i saat mengisi pendaftaran akun. Otomatis sistem akan menampilkan-nya kepada level yang dituju seperti pada gambar 5.20. Cara penggunaan untuk proses ini, pembimbing memilih menu penilaian dan sistem memunculkan opsi pilihan terdiri dari pilihan *internship* I dan *internship* II. Setelah memilih salah satu opsi yang ada pada menu penilaian, seperti pada gambar diatas (*internship* I) maka sistem akan memuncul data-data mahasiswa/i bimbingan.



Gambar 5.21 *Detail* Data Bimbingan

Sumber : *Template* AdminLTE versi 2.4.0

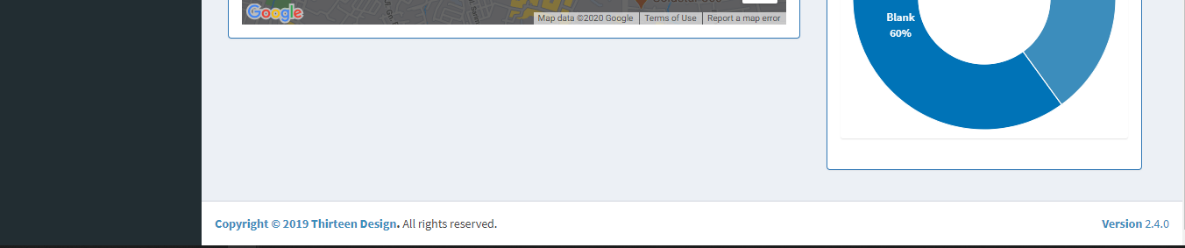
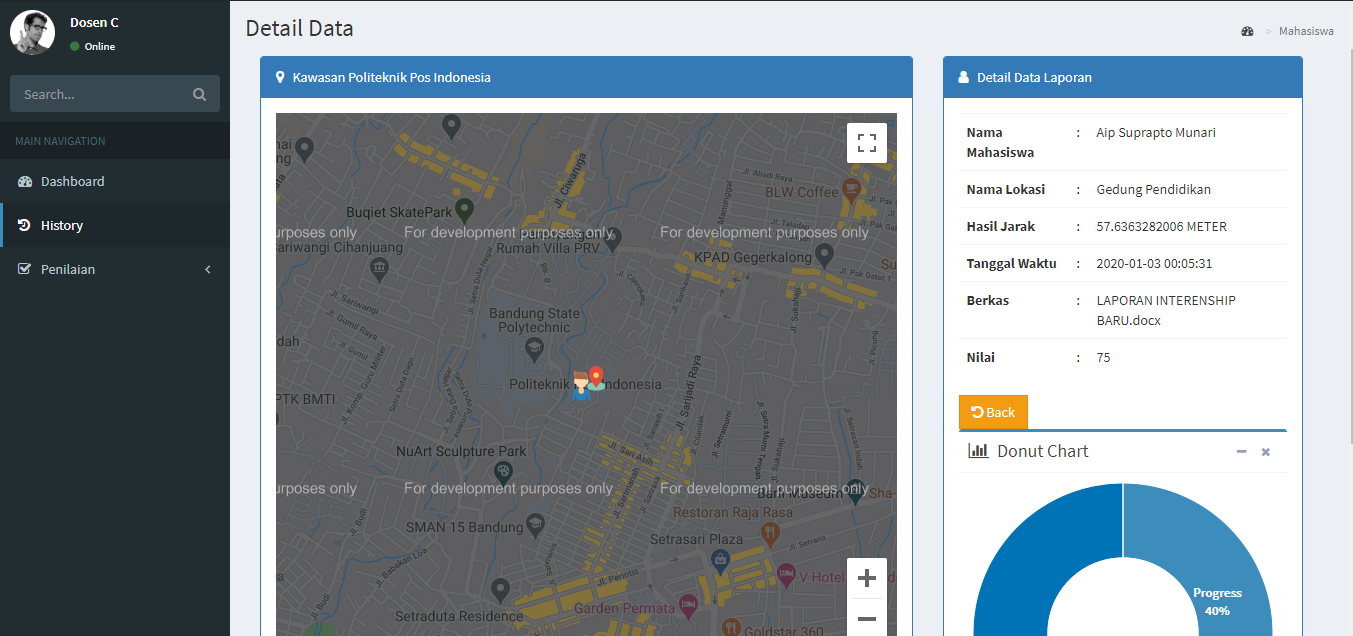
Proses ini adalah lanjutan dari gambar 5.20, apabila pembimbing menekan tombol detail kepada salah satu mahasiswa/i maka sistem akan memuncul seperti pada gambar 5.21. Dimana terdapat 3 buah tombol yang diantaranya tombol untuk *edit*. Tombol *edit* disini berfungsi untuk meng-*update* absensi, status dan nilai mahasiswa/i yang telah mengirimkan berkas kepada pembimbing. Untuk itu detail berfungsi untuk memperincikan data dari setiap kegiatan mahasisw/i untuk dinilai. Cara penggunaan akan dijelaskan pada gambar selanjutnya yaitu gambar 5.22.



Gambar 5.22 *Edit* Data Penilaian

Sumber : *Template* AdminLTE versi 2.4.0

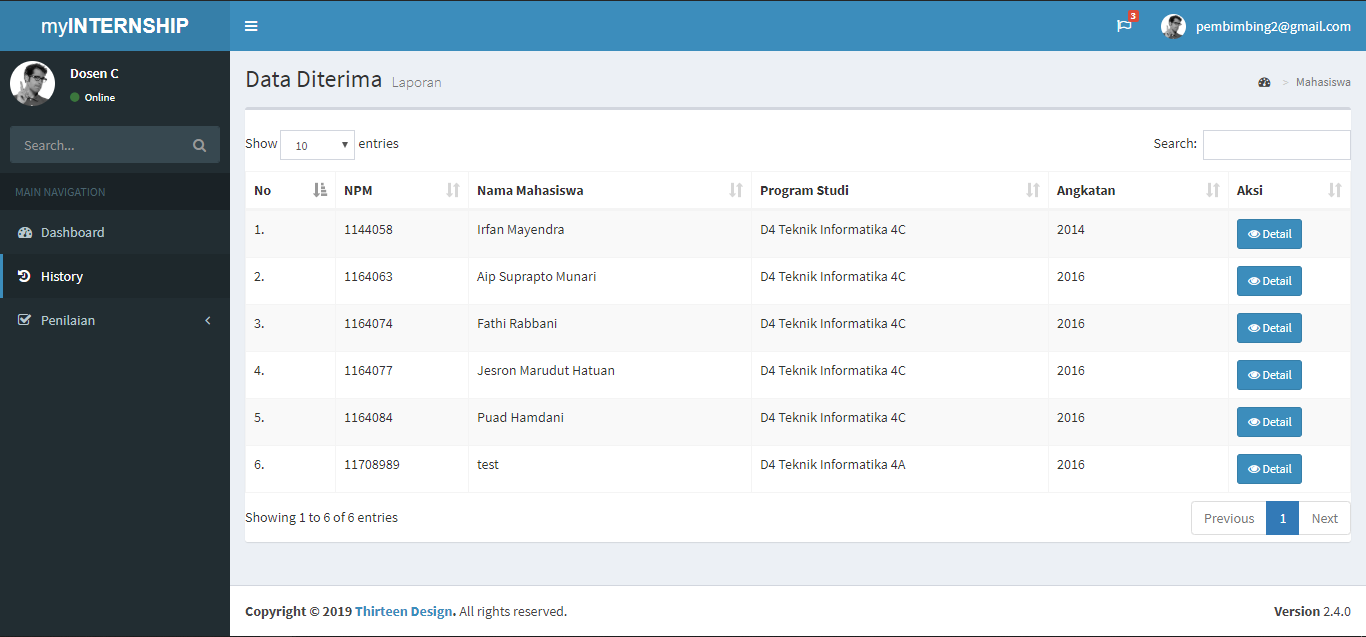
Proses selanjutnya adalah memperbarui data yang masuk untuk dinilai oleh pembimbing dan Cara penggunaan untuk halaman ini, pembimbing memilih tombol *edit* dan sistem akan memvalidasi dan menampilkan data yang akan diperbarui. Lalu apabila data absensi, status dan nilai telah diperbarui otomatis data yang tadinya ada didalam menu penilaian akan pindah ke menu *history*. Dengan aturan data absensi telah berubah menjadi hadir dan status data laporan untuk penilaian berubah menjadi diterima atau direvisi.



Gambar 5.23 *Detail* Data Penilaian

Sumber : *Template* AdminLTE versi 2.4.0

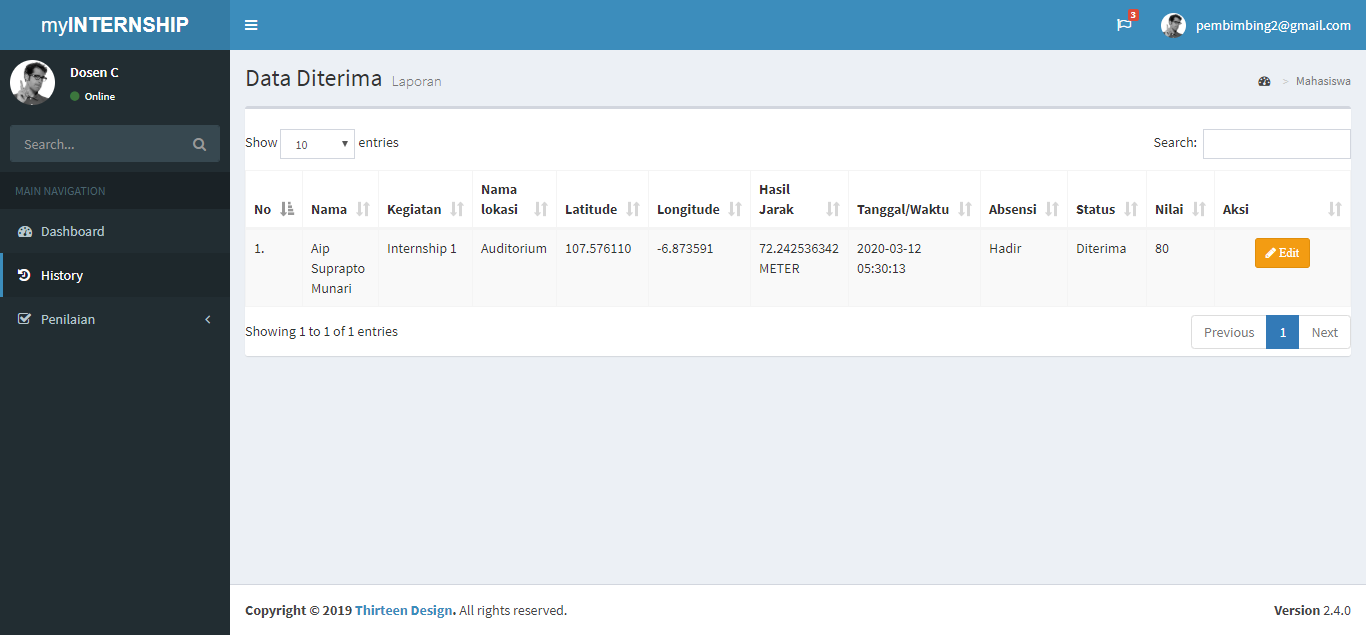
Tahap selanjutnya adalah proses detail data penilaian apabila pembimbing ingin melihat secara terperinci maka sistem akan menampilkan halaman *detail* data penilaian seperti pada gambar 5.23. Cara penggunaan untuk masuk halaman tersebut, aktor menekan tombol detail yang ada pada tampilan gambar 5.21.



Gambar 5.24 *Detail* Data Diterima Laporan

Sumber : *Template* AdminLTE versi 2.4.0

Mengenai gambar diatas adalah tampilan untuk setiap data mahasiswa apabila dilakukan aksi pada salah satu mahasiswa/i dengan menekan tombol detail maka sistem akan menampilkan data *history* laporan yang telah diterima oleh pembimbing dengan meng-*update* status laporan tersebut. Cara penggunaan akan dijelaskan pada gambar 5.25.



Gambar 5.25 Data Laporan Diterima

Sumber : *Template* AdminLTE versi 2.4.0

Tahap selanjutnya apabila pembimbing melakukan aksi dengan menekan tombol *detail* maka sistem akan menampilkan halaman data laporan yang diterima. Pada halaman ini terdapat tombol *edit*, apabila ada suatu kendala data dapat bisa diubah. Jadi dengan adanya menu *history* sangat membantu untuk mengatasi masalah yang terjadi.

# DAFTAR PUSTAKA

1. Ningrum, J. A. *SISTEM INFORMASI AKADEMIK SMK PENERBANGAN AAG ADI SUTJIPTO BERBASIS WEB*. Diss. STMIK AKAKOM Yogyakarta, 2008.
2. Indrajit, Richardus Eko. "Manajemen sistem informasi dan teknologi informasi." *Jakarta: PT Elex Media Komputindo* (2000).
3. Warsito, Ary Budi, Muhamad Yusup, and Yulianto Yulianto. "Kajian Yii Framework dalam Pengembangan Website Perguruan Tinggi." *Creative Communication and Innovative Technology Journal* 7.3 (2014): 437-451.
4. Subagia A. Kolaborasi Codeigniter dan Ajax dalam Perancangan CMS. Elex Media Komputindo; 2018 Apr 23.
5. Adolf, Mediocto Sahat, Denny Darlis, and Ari Machmuddi Kanosri. "Implementasi Sistem Tampilan Lokasi Berbasis Gps Di Kereta Api Sebagai Pemandu Otomatis." *eProceedings of Applied Science* 1.3 (2015).
6. Kaleb Yaroseray, Metode Pemetaan.
7. Putra, Faisal Setia. *Layanan Berbasis Lokasi Untuk Menemukan Koki Rumahan Yang Terdekat Dan Tersedia*. Diss. Institut Teknologi Sepuluh Nopember, 2017.