

BAB V

PEMBUATAN FORM LOGIN DAN CRUD MODULE ADMIN

A. Deskripsi Pembelajaran

Pada bab ini kita akan membahas untuk pembuatan form login admin dan CRUD module admin pada project kita. Ada beberapa hal yang harus dipastikan sebelum memulai bab ini, yaitu :

1. Database sudah dibuat dan selesai.
2. Template sudah diatur sebagaimana disampaikan di bab sebelumnya.

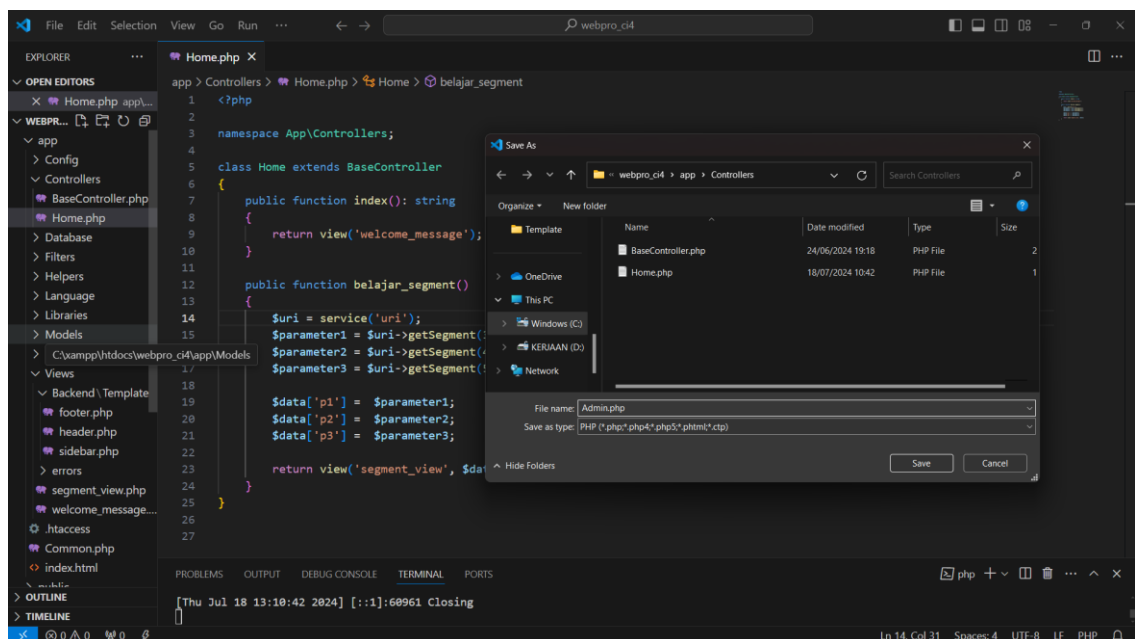
B. Tujuan Pembelajaran

Mahasiswa diharapkan mampu :

1. Membuat form login dan autentikasi login
2. Membuat dan menerapkan CRUD pada module admin
3. Menyelesaikan tugas mandiri

C. Pembuatan Controller Admin.php

Dalam pembuatan controller admin ini kita hanya perlu ATM (Amati, Tiru dan Modifikasi) dari controller yang sudah ada. Kita buka Controller **Home.php** di folder **app/Controller**. Kemudian kita **save as** dengan nama **Admin.php**.



Kemudian lakukan penyesuaian berikut ini, baris ke-5 **Home** replace to **Admin** :

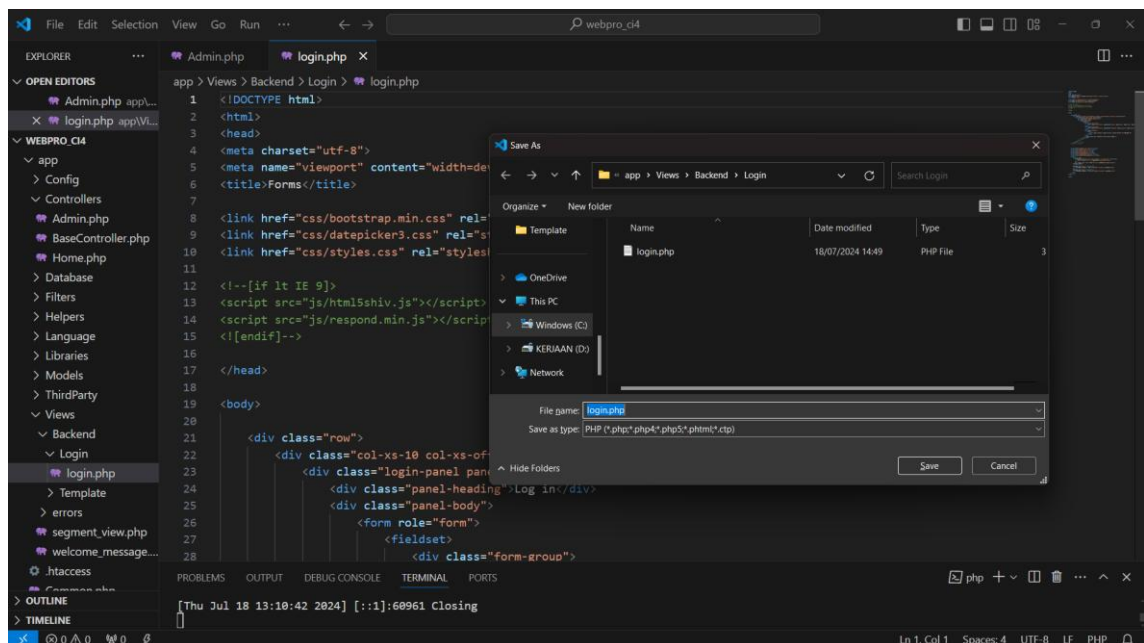
```

app > Controllers > Admin.php > ...
1  <?php
2
3  namespace App\Controllers;
4
5  class Admin extends BaseController
6  {
7      public function login()
8      {
9          return view('Backend/Login/login');
10     }
11 }

```

D. Tampilan Login Admin

Untuk membuat tampilan login admin kita buka kembali file master template Lumino kemudian cari **login.html**. Save as file **login.html** dengan nama **login.php** di dalam folder **app/Views/Backend/Login**. Cek gambar!!



Lakukan penyesuaian pada baris 6-10 seperti gambar di bawah ini :

```

6  <title>Login Aplikasi Perpustakaan</title>
7
8  <link href="/Assets/css/bootstrap.min.css" rel="stylesheet">
9  <link href="/Assets/css/datepicker3.css" rel="stylesheet">
10 <link href="/Assets/css/styles.css" rel="stylesheet">

```

Pada baris 29 seperti gambar di bawah ini :

```

<div class="form-group">
    <input class="form-control" placeholder="Username" name="username" type="text" autofocus="">
</div>

```

Dan pada baris 49-55 seperti gambar di bawah ini :

```

49 <script src="/Assets/js/jquery-1.11.1.min.js"></script>
50 <script src="/Assets/js/bootstrap.min.js"></script>
51 <script src="/Assets/js/chart.min.js"></script>
52 <script src="/Assets/js/chart-data.js"></script>
53 <script src="/Assets/js/easypiechart.js"></script>
54 <script src="/Assets/js/easypiechart-data.js"></script>
55 <script src="/Assets/js/bootstrap-datepicker.js"></script>

```

Jangan lupa lakukan simpan file secara berkala.

Kemudian kita setting default controller, sebelumnya default controller telah dijelaskan pada Bab II. Kita setting default Controller kita ke Admin dan default method ke login.

```

public string $defaultController = 'Admin';
public string $defaultMethod = 'login';

```

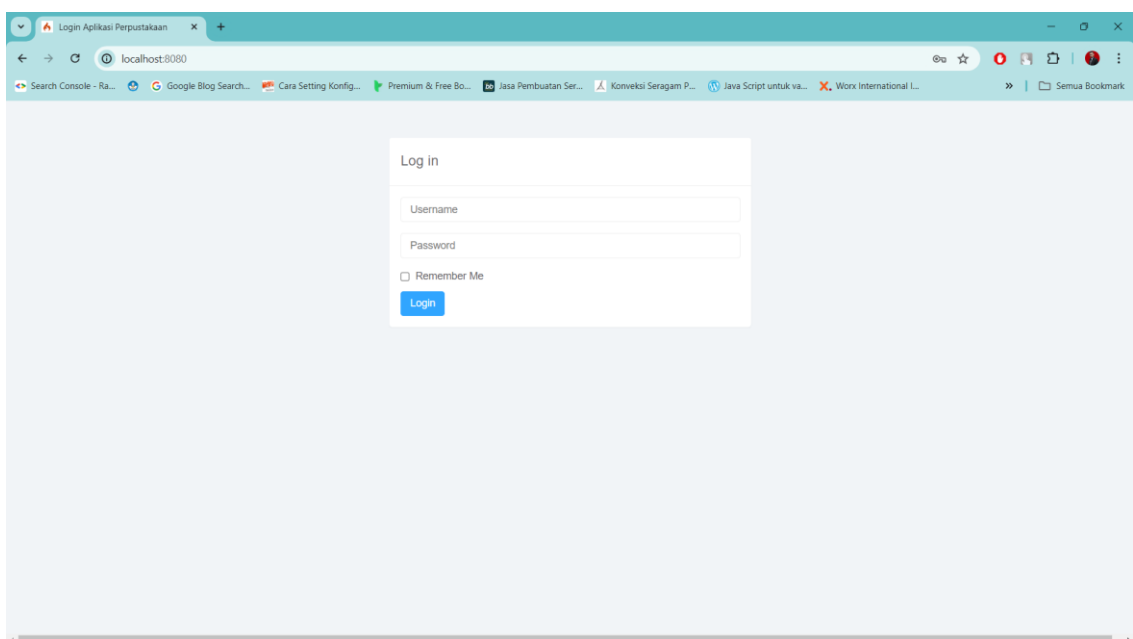
Selanjutnya kita setting Routes.php untuk halaman login admin ini.

```

app > Config > Routes.php
1 <?php
2
3 use CodeIgniter\Router\RouteCollection;
4
5 /**
6  * @var RouteCollection $routes
7  */
8 $routes->get('/', 'Admin::login');
9 $routes->get('/home/coba-parameter/(:alpha)/(:num)/(:alphanum)', 'Home::belajar_segment/$1/$2/$3');
10
11 // Routes untuk login admin
12 $routes->get('/admin/login-admin', 'Admin::login');

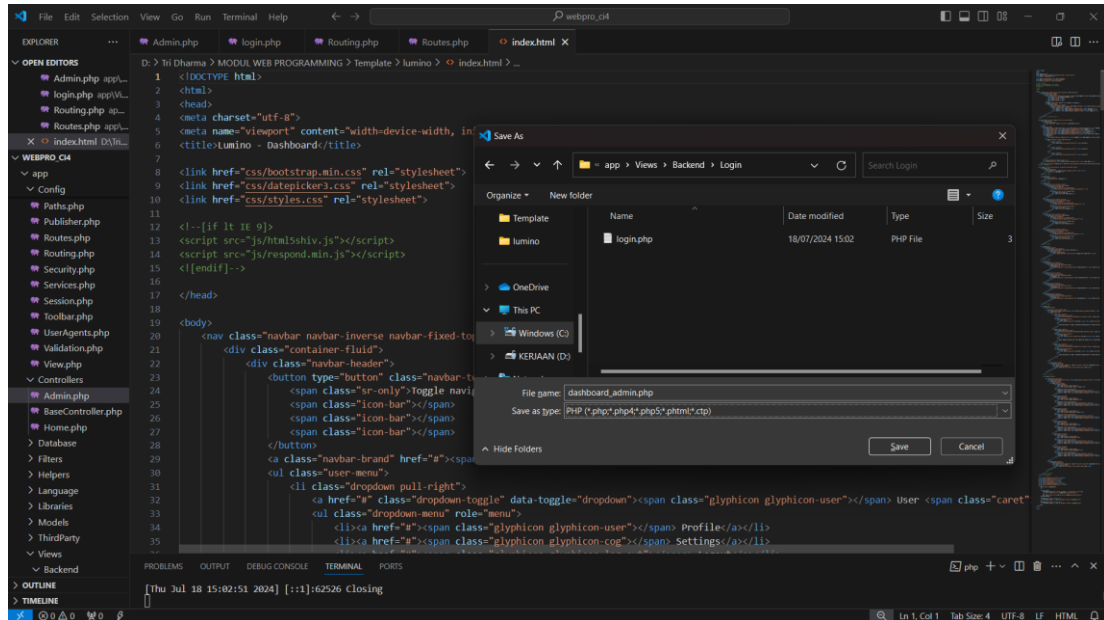
```

Selanjutnya coba buka laman <http://localhost:8080/> Maka tampilan halaman login admin sudah bisa diakses seperti gambar di bawah ini :



E. Siapkan tampilan dashboard admin

Untuk membuat tampilan dashboard admin kita ambil dari template Lumino. Cari file **index.html** kemudian **save as** dengan nama **dashboard_admin.php** ke dalam folder **app/Views/Backend/Login**. Lihat gambar!!



Lakukan beberapa penyesuaian berikut ini :

1. Hapus baris 1-84, kemudian simpan (ctrl+s).
2. Hapus baris 270-299, kemudian simpan (ctrl+s).

Lakukan penambahan function di dalam controller admin. Buat function untuk menampilkan dashboard admin seperti gambar di bawah ini :

```

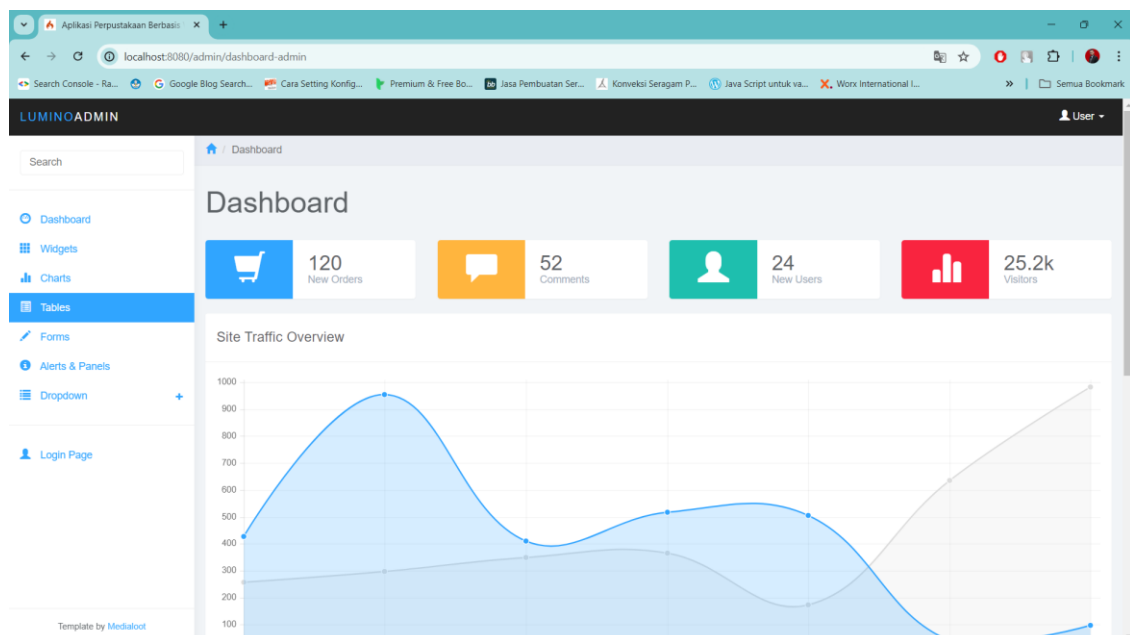
app > Controllers > Admin.php > Admin > dashboard
1  <?php
2
3  namespace App\Controllers;
4
5  class Admin extends BaseController
6  {
7      public function login()
8      {
9          return view('Backend/Login/login');
10     }
11
12     public function dashboard()
13     {
14         echo view('Backend/Template/header');
15         echo view('Backend/Template/sidebar');
16         echo view('Backend/Login/dashboard_admin');
17         echo view('Backend/Template/footer');
18     }
19 }

```

Tambahkan baris berikut pada file **Routes.php**

```
$routes->get('/admin/dashboard-admin', 'Admin::dashboard');
```

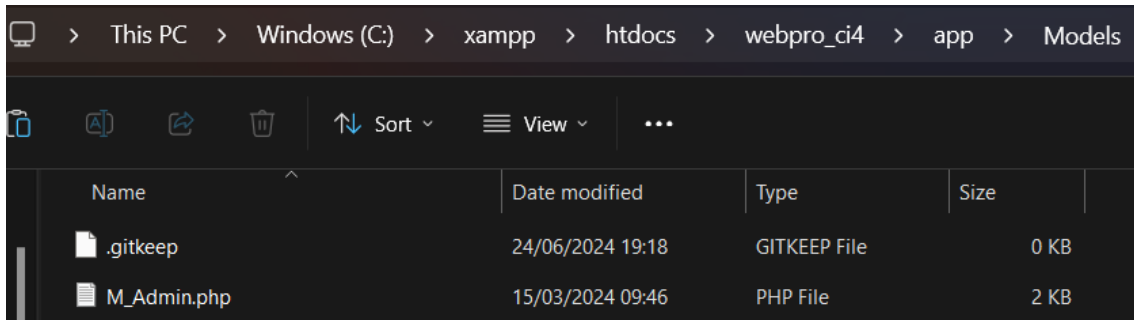
Buka laman <http://localhost:8080/admin/dashboard-admin> Pastikan tampil halaman seperti gambar di bawah ini :



F. Model Admin dan Autentikasi Login

Selanjutnya kita buat model admin. Model disini akan melakukan manipulasi database, peran model adalah menghubungkan antara controller dengan database. Idealnya 1 tabel di dalam database mempunyai 1 model atau 1 tabel di dalam database mempunyai 1 prosedur dalam pengelolaan databasenya.

Buatlah file baru kemudian simpan dengan nama **M_Admin.php** di dalam folder **app/Models**.



Name	Date modified	Type	Size
.gitkeep	24/06/2024 19:18	GITKEEP File	0 KB
M_Admin.php	15/03/2024 09:46	PHP File	2 KB

Kemudian tuliskan baris kode di bawah ini :

```
app > Models > M_Admin.php > ...
1  <?php
2  namespace App\Models;
3
4  use CodeIgniter\Model;
5
6  class M_Admin extends Model
7  {
8      protected $table = 'tbl_admin';
9
10     public function getDataAdmin($where = false)
11     {
12         if ($where === false) {
13             $builder = $this->db->table($this->table);
14             $builder->select('*');
15             $builder->orderBy('nama_admin', 'ASC');
16             return $query = $builder->get();
17         } else {
18             $builder = $this->db->table($this->table);
19             $builder->select('*');
20             $builder->where($where);
21             $builder->orderBy('nama_admin', 'ASC');
22             return $query = $builder->get();
23         }
24     }
25
26     public function saveDataAdmin($data)
27     {
28         $builder = $this->db->table($this->table);
29         return $builder->insert($data);
30     }
}
```

```

31
32     public function updateDataAdmin($data, $where)
33     {
34         $builder = $this->db->table($this->table);
35         $builder->where($where);
36         return $builder->update($data);
37     }
38
39     public function autoNumber() {
40         $builder = $this->db->table($this->table);
41         $builder->select("id_admin");
42         $builder->orderBy("id_admin", "DESC");
43         $builder->limit(1);
44         return $query = $builder->get();
45     }
46 }
47 ?>

```

Setelah itu jangan lupa disimpan. Tekan CTRL+S!

Hampir selesai untuk autentikasi login.

Untuk membuat fungsi autentikasi login maka kita perlu kembali ke controller **Admin.php**. Tambahkan baris di bawah ini dibagian atas file controller setelah namespace App\Controllers;

```

3     namespace App\Controllers;
4     // Load models
5     use App\Models\M_Admin;

```

Tambahkan sebuah fungsi dengan nama **autentikasi()**. Kemudian tuliskan baris kode yang ada di bawah ini :

```

22 public function autentikasi(){
23     $modelAdmin = new M_Admin; // proses inisiasi model
24     $username = $this->request->getPost('username');
25     $password = $this->request->getPost('password');
26
27     $cekUsername = $modelAdmin->getDataAdmin(['username_admin' => $username, 'is_delete_admin' => '0'])->getNumRows();
28     if($cekUsername == 0){
29         session()->setFlashdata('error','Username Tidak Ditemukan!');
30         ?>
31         <script>
32             history.go(-1);
33         </script>
34         <?php
35     }
36     else{
37         $dataUser = $modelAdmin->getDataAdmin(['username_admin' => $username, 'is_delete_admin' => '0'])->getRowArray();
38         $passwordUser = $dataUser['password_admin'];
39
40         $verifikasiPassword = password_verify($password, $passwordUser);
41         if(!$verifikasiPassword){
42             session()->setFlashdata('error','Password Tidak Sesuai!');
43             ?>
44             <script>
45                 history.go(-1);
46             </script>
47             <?php
48         }

```

```

49         else{
50             $dataSession = [
51                 'ses_id' => $dataUser['id_admin'],
52                 'ses_user' => $dataUser['nama_admin'],
53                 'ses_level' => $dataUser['akses_level']
54             ];
55             session()->set($dataSession);
56             session()->setFlashdata('success','Login Berhasil!');
57             ?>
58             <script>
59                 document.location = "<?>= base_url('admin/dashboard-admin');?>";
60             </script>
61             <?php
62         }
63     }
64 }
65 }

```

Kemudian lakukan penyesuaian untuk fungsi **dashboard** di controller **Admin.php**!

```

14 public function dashboard()
15 {
16     if(session()->get('ses_id')=="" or session()->get('ses_user')=="" or session()->get('ses_level')=="){
17         session()->setFlashdata('error','Silakan login terlebih dahulu!');
18         ?>
19         <script>
20             document.location = "<?>= base_url('admin/login-admin');?>";
21         </script>
22         <?php
23     }
24     else{
25         echo view('Backend/Template/header');
26         echo view('Backend/Template/sidebar');
27         echo view('Backend/Login/dashboard_admin');
28         echo view('Backend/Template/footer');
29     }
30 }

```

Lakukan penyesuaian pada file **login.php** yang berada pada folder **app/Views/Login** sebagai berikut :

1. Baris ke-26

```
<form role="form" action="<?>= base_url('admin/autentikasi-login');?>" method="post">
```

2. Baris ke-39

```
<button type="submit" class="btn btn-primary btn-block">Login</button>
```

3. Tambahkan style untuk Sweetalert. Download style sweetalert di sini :

<https://drive.google.com/drive/folders/1XV9FzIOO-WNTLqF6Fn8QlqA12sHIRE7i?usp=sharing>

Langkah menambahkan sweetalert2 :

- ✓ Download file css (**sweetalert2.min.css**) kemudian pindahkan ke folder **public/Assets/css**.
- ✓ Download file js (**sweetalert2.min.js**) kemudian pindahkan ke folder **public/Assets/js**.
- ✓ Tambahkan baris berikut di file **login.php** setelah baris ke-10.

```
11 <link href="/Assets/css/sweetalert2.min.css" rel="stylesheet">
```

- ✓ Tambahkan baris berikut di file **login.php** setelah baris ke-56.

```
57 <script src="/Assets/js/sweetalert2.min.js"></script>
```


- ✓ Tambahkan baris berikut di file login.php setelah baris ke-72.

```
73     <?php if (session()->getFlashdata('success')) : ?>
74         <script type="text/javascript">
75             $(document).ready(function() {
76                 swal("Success!", "<?php echo $_SESSION['success'] ?>", "success");
77             });
78         </script>
79     <?php endif; ?>
80     <?php if (session()->getFlashdata('error')) : ?>
81         <script type="text/javascript">
82             $(document).ready(function() {
83                 swal("Sorry!", "<?php echo $_SESSION['error'] ?>", "error");
84             });
85         </script>
86     <?php endif; ?>
87     <?php if (session()->getFlashdata('warning')) : ?>
88         <script type="text/javascript">
89             $(document).ready(function() {
90                 swal("Warning!", "<?php echo $_SESSION['warning'] ?>", "warning");
91             });
92         </script>
93     <?php endif; ?>
94     <?php if (session()->getFlashdata('info')) : ?>
95         <script type="text/javascript">
96             $(document).ready(function() {
97                 swal("Info!", "<?php echo $_SESSION['info'] ?>", "info");
98             });
99         </script>
100     <?php endif; ?>
```

- ✓ Tambahkan baris berikut di file **header.php** setelah baris ke-11.

```
12 <link href="/Assets/css/sweetalert2.min.css" rel="stylesheet">
```

- ✓ Tambahkan baris berikut di file **footer.php** setelah baris ke-8.

```
9 <script src="/Assets/js/sweetalert2.min.js"></script>
```

- ✓ Tambahkan baris berikut di file footer.php setelah baris ke-24.

```

25     <?php if (session()->getFlashdata('success')) : ?>
26         <script type="text/javascript">
27             $(document).ready(function() {
28                 swal("Success!", "<?php echo $_SESSION['success'] ?>", "success");
29             });
30         </script>
31     <?php endif; ?>
32     <?php if (session()->getFlashdata('error')) : ?>
33         <script type="text/javascript">
34             $(document).ready(function() {
35                 swal("Sorry!", "<?php echo $_SESSION['error'] ?>", "error");
36             });
37         </script>
38     <?php endif; ?>
39     <?php if (session()->getFlashdata('warning')) : ?>
40         <script type="text/javascript">
41             $(document).ready(function() {
42                 swal("Warning!", "<?php echo $_SESSION['warning'] ?>", "warning");
43             });
44         </script>
45     <?php endif; ?>
46     <?php if (session()->getFlashdata('info')) : ?>
47         <script type="text/javascript">
48             $(document).ready(function() {
49                 swal("Info!", "<?php echo $_SESSION['info'] ?>", "info");
50             });
51         </script>
52     <?php endif; ?>

```

4. Tambahkan baris berikut pada file **Routes.php**

```
14 $routes->post('/admin/autentikasi-login', 'Admin::autentikasi');
```

5. Silakan tambahkan 1 baris record di dalam tabel admin di database dengan isian seperti dibawah ini :

Kolom	Jenis	Fungsi	Tak Terilai	Nilai
id_admin	varchar(6)			ADM000
nama_admin	varchar(50)			Developer
username_admin	varchar(20)			developer
password_admin	varchar(255)			\$2y\$10\$BtHHWFXmLuhnP79ievN58O8EivCDmojcmNDivaVhmIIBQNSiq9Ku
akses_level	enum	--		<input checked="" type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3
is_delete_admin	enum	--		<input checked="" type="radio"/> 0 <input type="radio"/> 1
created_at	datetime			2024-07-18 16:11:01
updated_at	datetime			2024-07-18 16:11:01

Kirim

Password admin isikan hasil enkripsi dari **developer** yaitu:

\$2y\$10\$BtHHWFXmLuhnP79ievN58O8EivCDmojcmNDivaVhmIIBQNSiq9Ku

6. Terakhir silakan lakukan penyesuaian pada file environment (.env) pada baris ke 62-63 seperti terlihat pada gambar.

```
58 #-----
59 # SESSION
60 #-----
61
62 session.driver = 'CodeIgniter\Session\Handlers\FileHandler'
63 session.savePath = ci_session
```

7. **Coba jalankan form login dengan username dan password => developer**
8. Perintah untuk logout sebagai berikut :

```
public function logout(){
    session()->remove('ses_id');
    session()->remove('ses_user');
    session()->remove('ses_level');
    session()->setFlashdata('info','Anda telah keluar dari sistem!');
    ?>
    <script>
        document.location = "<?= base_url('admin/login-admin');?>";
    </script>
    <?php
}
```

Dan untuk setting di Routes.php maka tambahkan baris kode berikut ini :

```
$routes->get('/admin/logout', 'Admin::logout');
```

G. Create, Read, Update, Delete Module Admin

1. Create

Sebelum membuat module create kita siapkan terlebih dahulu design dari form input admin nya. Silakan dituliskan kode program seperti di bawah ini :

```
app > Views > Backend > MasterAdmin > input-admin.php
1 <div class="col-sm-9 col-sm-offset-3 col-lg-10 col-lg-offset-2 main">
2 <div class="row">
3 <ol class="breadcrumb">
4 <li><a href="#"><span class="glyphicon glyphicon-home"></span></a></li>
5 <li>Master Data Admin</li>
6 <li class="active">Input Data Admin</li>
7 </ol>
8 </div><!--/.row-->
9
10 <div class="row">
11 <div class="col-md-12">
12 <div class="panel panel-default">
13 <div class="panel-body">
14 <h3>Input Admin</h3>
15 <hr />
16 <form action="<?php echo base_url('admin/simpan-admin');?>" method="post">
17 <div class="form-group col-md-6">
18 <label>Nama Admin</label>
19 <input type="text" class="form-control" name="nama" placeholder="Masukkan Nama Admin" required="required">
20 </div>
21 <div style="clear:both;"></div>
22
23 <div class="form-group col-md-6">
24 <label>Username Admin</label>
25 <input type="text" class="form-control" onKeyPress="return goodchars(event,
'abcdefghijklmnopqrstuvwxyz_ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890',this)" name="username" placeholder="Masukkan Username
Admin" required="required">
26 </div>
27 <div style="clear:both;"></div>
```

```

28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

```

```

<div class="form-group col-md-6">
  <label>Akses Level</label>
  <select class="form-control" name="level" required="required">
    <option value="">-- Pilih Level --</option>
    <option value="2">Kepala Perpustakaan</option>
    <option value="3">Admin Perpustakaan</option>
  </select>
</div>
<div style="clear:both;"></div>

<div class="form-group col-md-6">
  <button type="submit" class="btn btn-primary">Simpan</button>
  <button type="reset" class="btn btn-danger">Batal</button>
</div>
<div style="clear:both;"></div>
</form>
</div>
</div>
</div><!-- /.row -->
</div>

```

Dalam baris kode diatas menggunakan fitur validasi dari **HTML5** yaitu **required**. Selain itu terdapat juga baris kode untuk verifikasi inputan user pada **textfield username** dengan menggunakan **event keypress**. Yang tertera dalam baris kode ke-25 pada fungsi **goodchars**. Semua karakter yang didefinisikan pada fungsi **goodchars** inilah yang bisa diinputkan oleh user. Simpan file tersebut dengan nama **input-admin.php** di dalam folder **app/Views/Backend/MasterAdmin**. Selanjutnya kita lakukan penyesuaian di file **sidebar.php** dan **header.php** yang berada di folder **Template** sebagai berikut :

- ✓ Hapus baris 2-6 pada **sidebar.php**. Lihat gambar!

```

app > Views > Backend > Template > sidebar.php
1
2
3
4
5
6

```

```

<div id="sidebar-collapse" class="col-sm-3 col-lg-2 sidebar">
  <form role="search">
    <div class="form-group">
      <input type="text" class="form-control" placeholder="Search">
    </div>
  </form>

```

- ✓ Sesuaikan baris ke-3 pada **sidebar.php**. Lihat gambar!

```

3

```

```

<li><a href="{?= base_url('admin/dashboard-admin')}?>"><span class="glyphicon glyphicon-dashboard"></span> Dashboard</a></li>

```

- ✓ Hapus baris 4-8 pada **sidebar.php**. Lihat gambar!

```

4
5
6
7
8

```

```

<li><a href="widgets.html"><span class="glyphicon glyphicon-th"></span> Widgets</a></li>
<li><a href="charts.html"><span class="glyphicon glyphicon-stats"></span> Charts</a></li>
<li class="active"><a href="tables.html"><span class="glyphicon glyphicon-list-alt"></span> Tables</a></li>
<li><a href="forms.html"><span class="glyphicon glyphicon-pencil"></span> Forms</a></li>
<li><a href="panels.html"><span class="glyphicon glyphicon-info-sign"></span> Alerts & Panels</a></li>

```

- ✓ Sesuaikan baris 4-36 pada **sidebar.php**. Lihat gambar!

```
app > Views > Backend > Template > sidebar.php
4      <li class="parent ">
5          <a href="#">
6              <span class="glyphicon glyphicon-list"></span> Master Data <span data-toggle="collapse" href="#sub-item-master" class="icon
              pull-right"><em class="glyphicon glyphicon-s glyphicon-plus"></em></span>
7          </a>
8          <ul class="children collapse" id="sub-item-master">
9              <li>
10                 <a href="= base_url('admin/master-data-admin');?&gt;"&gt;
11                     &lt;span class="glyphicon glyphicon-share-alt"&gt;&lt;/span&gt; Data Admin
12                 &lt;/a&gt;
13             &lt;/li&gt;
14             &lt;li&gt;
15                 &lt;a href="<?= base_url('admin/master-data-anggota');?&gt;"&gt;
16                     &lt;span class="glyphicon glyphicon-share-alt"&gt;&lt;/span&gt; Data Anggota
17                 &lt;/a&gt;
18             &lt;/li&gt;
19             &lt;li&gt;
20                 &lt;a href="<?= base_url('admin/master-data-kategori');?&gt;"&gt;
21                     &lt;span class="glyphicon glyphicon-share-alt"&gt;&lt;/span&gt; Data Kategori
22                 &lt;/a&gt;
23             &lt;/li&gt;
24             &lt;li&gt;
25                 &lt;a href="<?= base_url('admin/master-data-rak');?&gt;"&gt;
26                     &lt;span class="glyphicon glyphicon-share-alt"&gt;&lt;/span&gt; Data Rak
27                 &lt;/a&gt;
28             &lt;/li&gt;
29             &lt;li&gt;
30                 &lt;a href="<?= base_url('admin/master-data-buku');?&gt;"&gt;
31                     &lt;span class="glyphicon glyphicon-share-alt"&gt;&lt;/span&gt; Data Buku
32                 &lt;/a&gt;
33             &lt;/li&gt;
34         &lt;/ul&gt;
35     &lt;/li&gt;
36     &lt;li role="presentation" class="divider"&gt;&lt;/li&gt;</pre
```

- ✓ Sesuaikan baris ke-37 pada **sidebar.php**. Lihat gambar!

```
37     <li><a href="= base_url('admin/logout');?&gt;"&gt;&lt;span class="glyphicon glyphicon-log-out"&gt;&lt;/span&gt; Logout&lt;/a&gt;&lt;/li&gt;
38 &lt;/ul&gt;
39 &lt;div class="attribution"&gt;Template by &lt;a href="http://www.medialoot.com/item/lumino-admin-bootstrap-template/"&gt;Medialoot&lt;/a&gt;&lt;/div&gt;
40 &lt;/div&gt;&lt;!--.sidebar--&gt;</pre
```

- ✓ Sisipkan baris kode berikut pada **header.php** sebelum tag </head>.

```
<script type="text/javascript">
    function getkey(e){
        if (window.event)
            return window.event.keyCode;
        else if (e)
            return e.which;
        else
            return null;

    }

    function goodchars(e, goods, field){
        var key, keychar;
        key = getkey(e);
        if (key == null) return true;

        keychar = String.fromCharCode(key);
        //keychar = keychar.toLowerCase();
        //goods = goods.toLowerCase();

        // check goodkeys
        if (goods.indexOf(keychar) != -1)
            return true;
        // control keys
        if ( key==null || key==0 || key==8 || key==9 || key==27 )
            return true;

        if (key == 13) {
            var i;
            for (i = 0; i < field.form.elements.length; i++)
                if (field == field.form.elements[i])
                    break;
            i = (i + 1) % field.form.elements.length;
            field.form.elements[i].focus();
            return false;
        };
        // else return false
        return false;
    }
</script>
```

Fungsi diatas digunakan untuk membatasi inputan yang dilakukan oleh user pada formulir yang kita buat. Fungsi ini akan berguna di beberapa kondisi formulir yang nantinya kita buat.

Selanjutnya kita setting **Routes.php** untuk keseluruhan CRUD Module Admin sebagai berikut :

```
16 // Routes untuk module admin
17 $routes->get('/admin/master-data-admin', 'Admin::master_data_admin');
18 $routes->get('/admin/input-data-admin', 'Admin::input_data_admin');
19 $routes->post('/admin/simpan-admin', 'Admin::simpan_data_admin');
20 $routes->get('/admin/edit-data-admin/{:alphanum}', 'Admin::edit_data_admin/{:alphanum}');
21 $routes->post('/admin/update-admin', 'Admin::update_data_admin');
22 $routes->get('/admin/hapus-data-admin/{:alphanum}', 'Admin::hapus_data_admin/{:alphanum}');
```

Setelah kita selesai menyiapkan design tampilan untuk form input admin, mengatur link pada sidebar dan menyesuaikan Routes.php maka selanjutnya kita buat perintah untuk menampilkan Form Input Admin dan proses **CREATE** nya di controller **Admin.php**. Tuliskan baris kode berikut ini :

```
public function input_data_admin(){
    if(session()->get('ses_id')=="" or session()->get('ses_user')=="" or session()->get('ses_level')== ""){
        session()->setFlashdata('error','Silakan login terlebih dahulu!');
        ?>
        <script>
        | document.location = "<? = base_url('admin/login-admin');?>";
        </script>
        <?php
    }
    else{
        echo view('Backend/Template/header');
        echo view('Backend/Template/sidebar');
        echo view('Backend/MasterAdmin/input-admin');
        echo view('Backend/Template/footer');
    }
}
```

```
public function simpan_data_admin(){
    if(session()->get('ses_id')=="" or session()->get('ses_user')=="" or session()->get('ses_level')== ""){
        session()->setFlashdata('error','Silakan login terlebih dahulu!');
        ?>
        <script>
        | document.location = "<? = base_url('admin/login-admin');?>";
        </script>
        <?php
    }
    else{
        $modelAdmin = new M_Admin; // inisiasi

        $nama = $this->request->getPost('nama');
        $username = $this->request->getPost('username');
        $level = $this->request->getPost('level');

        $cekUsername = $modelAdmin->getDataAdmin(['username_admin' => $username])->getNumRows();
        if($cekUsername > 0){
            session()->setFlashdata('error','Username sudah digunakan!!');
            ?>
            <script>
            | history.go(-1);
            </script>
            <?php
        }
    }
}
```

```

else{
    $hasil = $modelAdmin->autoNumber()->getRowArray();
    if(!$hasil){
        $id = "ADM001";
    }
    else{
        $kode = $hasil['id_admin'];
        $noUrut = (int) substr($kode, -3);
        $noUrut++;
        $id = "ADM".sprintf("%03s", $noUrut);
    }

    $dataSimpan = [
        'id_admin' => $id,
        'nama_admin' => $nama,
        'username_admin' => $username,
        'password_admin' => password_hash('pass_admin', PASSWORD_DEFAULT),
        'akses_level' => $level,
        'is_delete_admin' => '0',
        'created_at' => date('Y-m-d H:i:s'),
        'updated_at' => date('Y-m-d H:i:s')
    ];
    $modelAdmin->saveDataAdmin($dataSimpan);
    session()->setFlashdata('success', 'Data Admin Berhasil Ditambahkan!!');
    ?>
    <script>
        document.location = "<?>= base_url('admin/master-data-admin');?>";
    </script>
    <?php
}
}
}

```

2. Read

Pada bagian ini kita akan mengimplementasikan percabangan **IF...ELSE** dan perulangan dengan **FOREACH** seperti yang tertera pada Bab sebelumnya. Silakan buat file baru dengan nama **master-data-admin.php** yang disimpan di folder **app/Views/Backend/MasterAdmin** kemudian ketikkan baris kode di bawah ini :

```

app > Views > Backend > MasterAdmin > master-data-admin.php
1 <div class="col-sm-9 col-sm-offset-3 col-lg-10 col-lg-offset-2 main">
2 <div class="row">
3 <ol class="breadcrumb">
4 <li><a href="#"><span class="glyphicon glyphicon-home"></span></a></li>
5 <li class="active">Master Data Admin</li>
6 </ol>
7 </div><!--/.row-->
8
9 <div class="row">
10 <div class="col-lg-12">
11 <div class="panel panel-default">
12 <div class="panel-body">
13 <h3>Master Data Admin
14 <a href="<?>= base_url('admin/input-data-admin');?>"><button type="button" class="btn btn-sm btn-primary pull-right">Input Data
Admin</button></a>
15 </h3>
16 <hr />
17 <table data-toggle="table" data-show-refresh="true" data-show-toggle="true" data-show-columns="true" data-search="true"
data-select-item-name="toolbar1" data-pagination="true" data-sort-name="name" data-sort-order="desc">
18 <thead>
19 <tr>
20 <th data-sortable="true">#</th>
21 <th data-sortable="true">Nama Admin</th>
22 <th data-sortable="true">Username Admin</th>
23 <th data-sortable="true">Akses Level</th>
24 <th data-sortable="true">Ops</th>
25 </tr>
26 </thead>

```

```

27 <tbody>
28 <?php
29 $no = 0;
30 foreach($data_user as $data){
31     ?>
32     <tr>
33         <td data-sortable="true"><?=$no-$no+1;></td>
34         <td data-sortable="true"><?=$data['nama_admin'];></td>
35         <td data-sortable="true"><?=$data['username_admin'];></td>
36         <td data-sortable="true"><?=$data['akses_level'];></td>
37         <td data-sortable="true">
38             <?php
39             if(session()->get('ses_level')=="1"){
40                 ?>
41                 <a href="<?=$base_url('admin/edit-data-admin/'.$data['id_admin']);>"><button type="button" class="btn btn-sm
42                 btn-success">Edit</button></a>
43                 <a href="#" onclick="doDelete('<?=$data['id_admin']);>'"><button type="button" class="btn btn-sm
44                 btn-danger">Hapus</button></a>
45             <?php }
46             else echo "#"; ?>
47         </td>
48     </tr>
49 </tbody>
50 </table>
51 </div>
52 </div>
53 </div><!--/.row-->
54 </div><!--/.main-->

```

Pada baris kode 30 adalah implementasi perulangan dengan **FOREACH**. Dan pada baris kode 39 adalah implementasi percabangan dengan **IF...ELSE**.

```

55
56 <script type="text/javascript">
57     function doDelete(idDelete){
58         swal({
59             title : "Hapus Data Admin?",
60             text : "Data ini akan terhapus secara permanen!!",
61             icon : "warning",
62             buttons : true,
63             dangerMode : false,
64         })
65         .then(ok => {
66             if(ok){
67                 window.location.href = '<?=$base_url();>/admin/hapus-data-admin/' + idDelete;
68             }
69             else{
70                 $(this).removeAttr('disabled')
71             }
72         })
73     }
74 </script>

```

Setelah selesai dengan design module Admin **READ**. Silakan tambahkan baris kode dibawah ini pada controller **Admin.php**.

```

public function master_data_admin(){
    if(session()->get('ses_id')=="" or session()->get('ses_user')=="" or session()->get('ses_level')=="){
        session()->setFlashdata('error','Silakan login terlebih dahulu!');
        ?>
        <script>
        | document.location = "<?=$base_url('admin/login-admin');>";
        </script>
        <?php
    }
    else{
        $modelAdmin = new M_Admin; // inisiasi

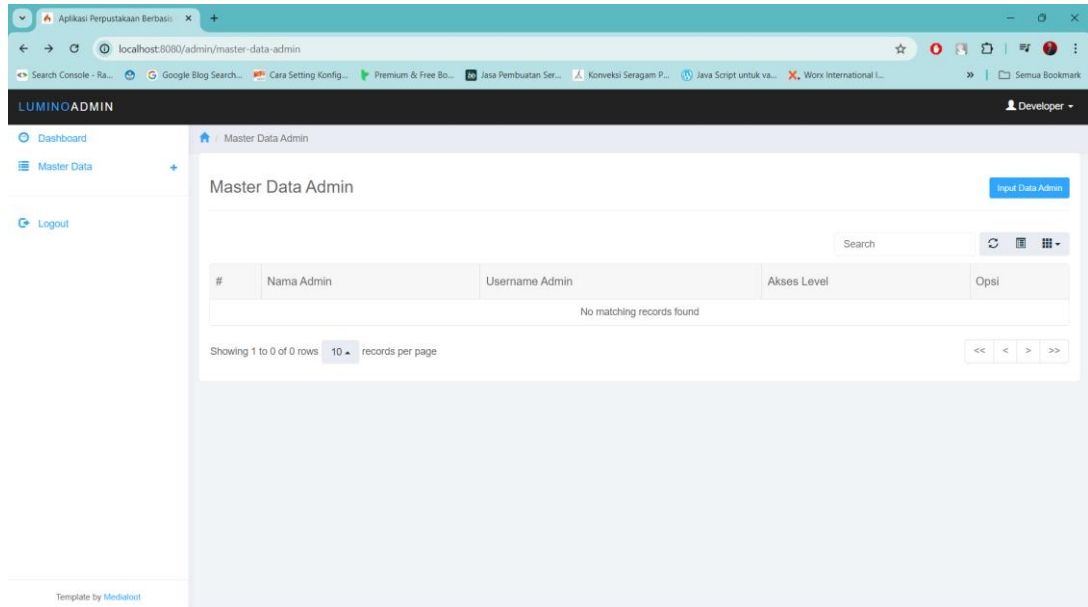
        $uri = service('uri');
        $pages = $uri->getSegment(2);
        $dataUser = $modelAdmin->getDataAdmin(['is_delete_admin' => '0', 'akses_level !=' => '1'])->getResultArray();

        $data['pages'] = $pages;
        $data['data_user'] = $dataUser;

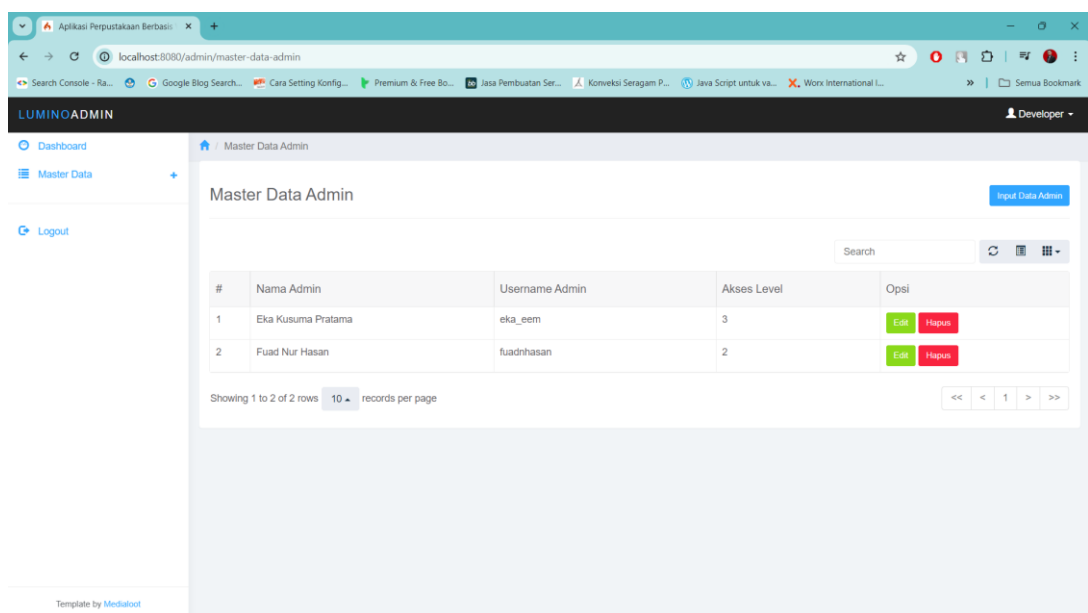
        echo view('Backend/Template/header', $data);
        echo view('Backend/Template/sidebar', $data);
        echo view('Backend/MasterAdmin/master-data-admin', $data);
        echo view('Backend/Template/footer', $data);
    }
}

```


Pada baris kode diatas maka variabel `data_user` yang dikirimkan hanya akan mengambil data admin dengan status delete nya = 0 dan akses level nya != 1. Maka jika kita akses halaman berikut <http://localhost:8080/admin/master-data-admin> hasilnya seperti gambar di bawah ini :



Tampilan jika data admin sudah diinput seperti di bawah ini :



Soal!!

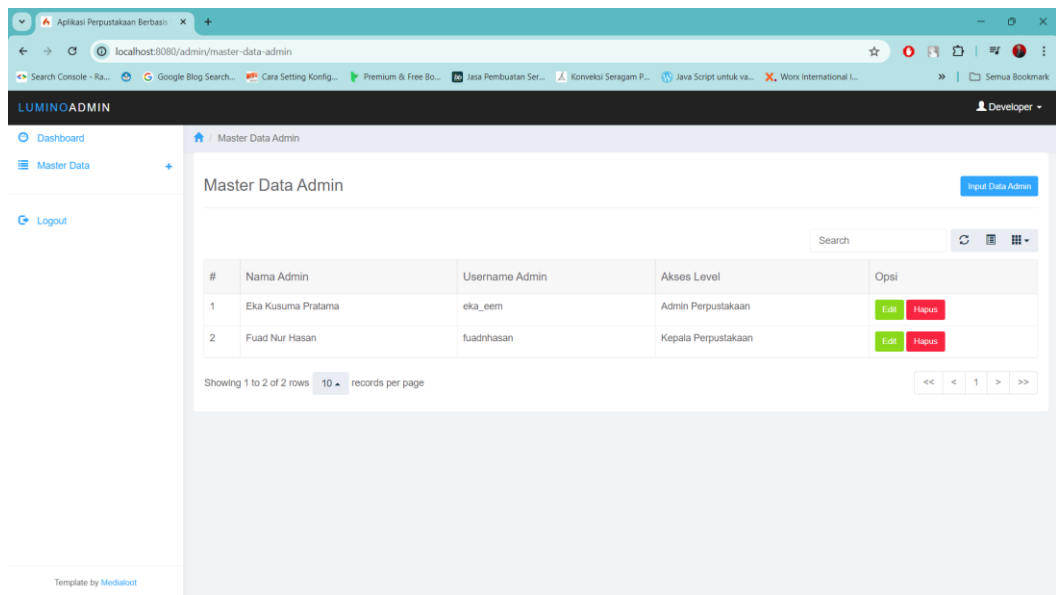
Pada gambar di atas terlihat bahwa Akses Level masih tampil angka. Ubahlah tampilan angka 2 dan 3 tersebut dengan percabangan **IF...ELSE**. Ketentuan :

2 = Kepala Perpustakaan

3 = Admin Perpustakaan

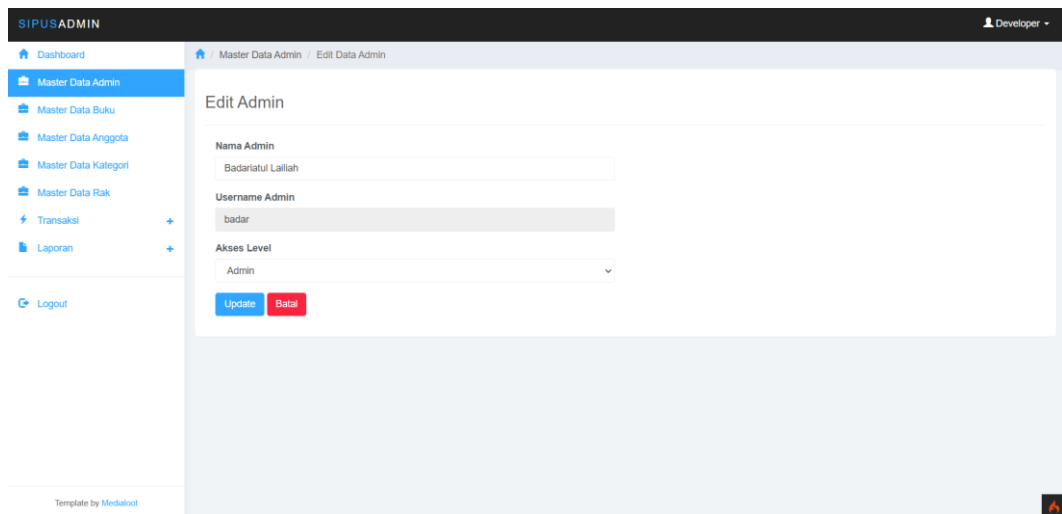
Kerjakan dan beritahu dosenmu jika kamu sudah selesai!!

Hasil yang benar adalah seperti gambar di bawah ini :



3. Edit dan Update

Pada bagian ini kita akan membuat tampilan edit data dan proses update. Tentunya kita akan mengadopsi tampilan yang sudah kita buat sebelumnya dalam proses input admin. Dari tampilan tersebut kita modifikasi menjadi seperti dibawah ini :



Kode program lihat dibawah ini :

```
.....
<h3>Edit Admin</h3>
<hr />
<form action="php echo base_url('admin/update-admin'); ?" method="post">
    <div class="form-group col-md-6">
        <label>Nama Admin</label>
        <input type="text" class="form-control" name="nama" placeholder="Masukkan Nama Admin" value="php echo $data_admin['nama_admin']; ?"
            required="required">
    </div>
    <div style="clear:both;"></div>

    <div class="form-group col-md-6">
        <label>Username Admin</label>
        <input type="text" class="form-control" value="php echo $data_admin['username_admin']; ?" readonly="readonly" onKeyPress="return goodchars(event,
            'abcdefghijklmnopqrstuvwxyz_ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890',this)" name="username" placeholder="Masukkan Username Pengguna"
            required="required">
    </div>
    <div style="clear:both;"></div>

    <div class="form-group col-md-6">
        <label>Akses Level</label>
        <select class="form-control" name="level" required="required">
            <option value="">-- Pilih Level --</option>
            <option value="2" <?php if($data_admin['akses_level']=="2"){ echo "selected"; }else echo "";>>Superadmin</option>
            <option value="3" <?php if($data_admin['akses_level']=="3"){ echo "selected"; }else echo "";>>Admin</option>
        </select>
    </div>
    <div style="clear:both;"></div>

    <div class="form-group col-md-6">
        <button type="submit" class="btn btn-primary">Update</button>
        <a href="php echo base_url('admin/master-admin'); ?"><button type="button" class="btn btn-danger">Batal</button></a>
    </div>
    <div style="clear:both;"></div>
</form>
.....
```

Save as dari file **input-admin.php** dan beri nama **edit-admin.php** di folder yang sama dengan penyimpanan file sebelumnya. Kemudian sesuaikan form menjadi seperti gambar di atas.

Setelah itu kita tambahkan baris kode di bawah ini dalam controller **Admin.php**.

```
public function edit_data_admin()
{
    $uri = service('uri');
    $idEdit = $uri->getSegment(3);
    $modelAdmin = new M_Admin;
    // Mengambil data admin dari table admin di database berdasarkan parameter yang dikirimkan
    $dataAdmin = $modelAdmin->getDataAdmin(['sha1(id_admin)' => $idEdit])>getRowArray();
    session()->set(['idUpdate' => $dataAdmin['id_admin']]);

    $page = $uri->getSegment(2);

    $data['page'] = $page;
    $data['web_title'] = "Edit Data Admin";
    $data['data_admin'] = $dataAdmin; // mengirim array data admin ke view

    echo view('Backend/Template/header', $data);
    echo view('Backend/Template/sidebar', $data);
    echo view('Backend/MasterAdmin/edit-admin', $data);
    echo view('Backend/Template/footer', $data);
}
```

Kemudian untuk proses simpan updatanya silakan tambahkan kode di controller **Admin.php** seperti di bawah ini :

```

public function update_admin()
{
    $modelAdmin = new M_Admin;

    $idUpdate = session()->get('idUpdate');
    $nama = $this->request->getPost('nama');
    $level = $this->request->getPost('level');

    if($nama=="" or $level==""){
        session()->setFlashdata('error', 'Isian tidak boleh kosong!!');
        ?>
        <script>
            history.go(-1);
        </script>
        <?php
    }
    else{
        $dataUpdate = [
            'nama_admin' => $nama,
            'akses_level' => $level,
            'updated_at' => date("Y-m-d H:i:s")
        ];
        $whereUpdate = ['id_admin' => $idUpdate];

        $modelAdmin->updateDataAdmin($dataUpdate, $whereUpdate);
        session()->remove('idUpdate');
        session()->setFlashdata('success', 'Data Admin Berhasil Diperbaharui!');
        ?>
        <script>
            document.location = "<?= base_url('admin/master-data-admin');?>";
        </script>
        <?php
    }
}
}

```

4. Delete

Pada bagian delete kita tidak membutuhkan view hanya satu langkah saja yang perlu kita lakukan yaitu menambahkan baris kode berikut pada Controller **Admin.php**.

```

public function hapus_data_admin()
{
    $modelAdmin = new M_Admin;

    $uri = service('uri');
    $idHapus = $uri->getSegment(3);

    $dataUpdate = [
        'is_delete_admin' => '1',
        'updated_at' => date("Y-m-d H:i:s")
    ];
    $whereUpdate = ['sha1(id_admin)' => $idHapus];
    $modelAdmin->updateDataAdmin($dataUpdate, $whereUpdate);
    session()->setFlashdata('success', 'Data Admin Berhasil Dihapus!');
    ?>
    <script>
        document.location = "<?= base_url('admin/master-data-admin');?>";
    </script>
    <?php
}
// Akhir modul admin

```

H. Tugas!!

Buatlah CRUD untuk module Anggota, Rak, Kategori dengan mengikuti contoh yang sudah ada di Module Admin!!