

# Laporan Presentasi Aplikasi Perpustakaan



Dosen Pengampu :  
Herkules S.Kom.,M.Cs

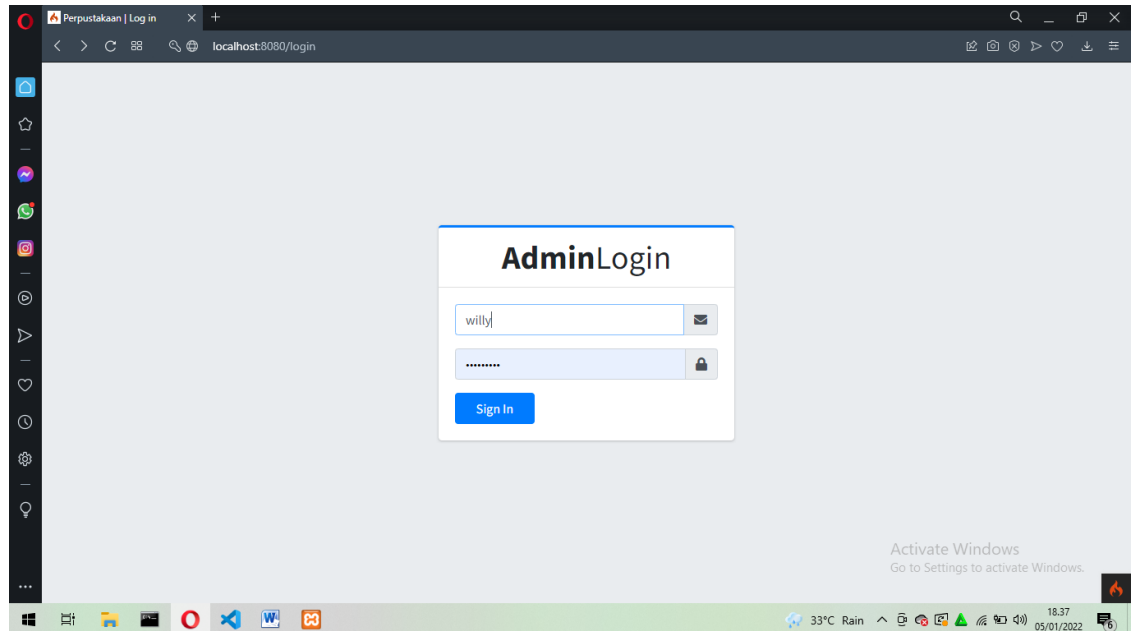
Mahasiswa :  
WILLYANTO NADELAKU (C2055201071)

PROGRAM STUDI TEKNIK INFORMATIKA  
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN  
KOMPUTER  
(STMIK)  
2022

## A. Tampilan Aplikasi perpustakaan

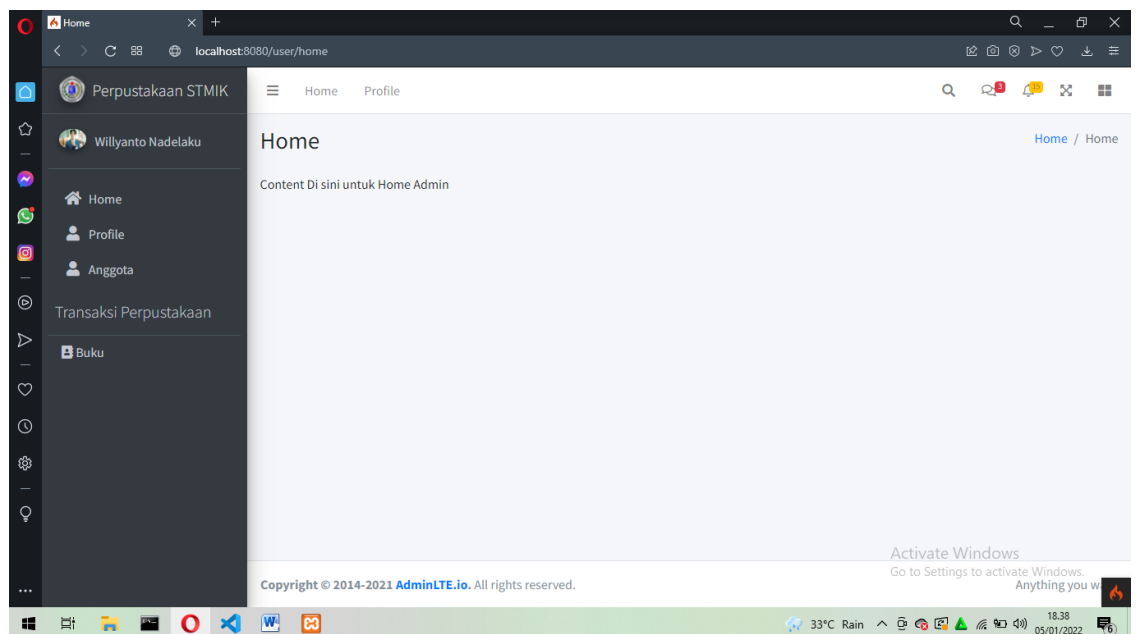
### 1. Login

Fitur login digunakan untuk memasukan data agar bisa masuk ke dalam aplikasi



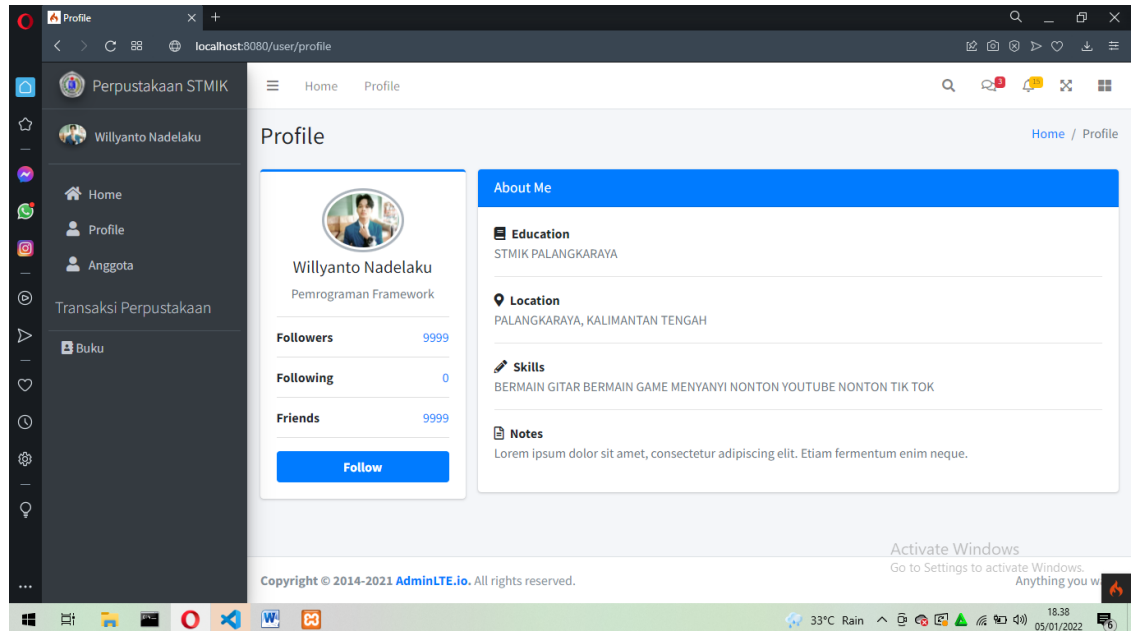
### 2. Home

Fitur home adalah halaman awal dari aplikasi Perpustakaan, home menampilkan daftar buku yang ada di dalam tabel buku di bukumodel.



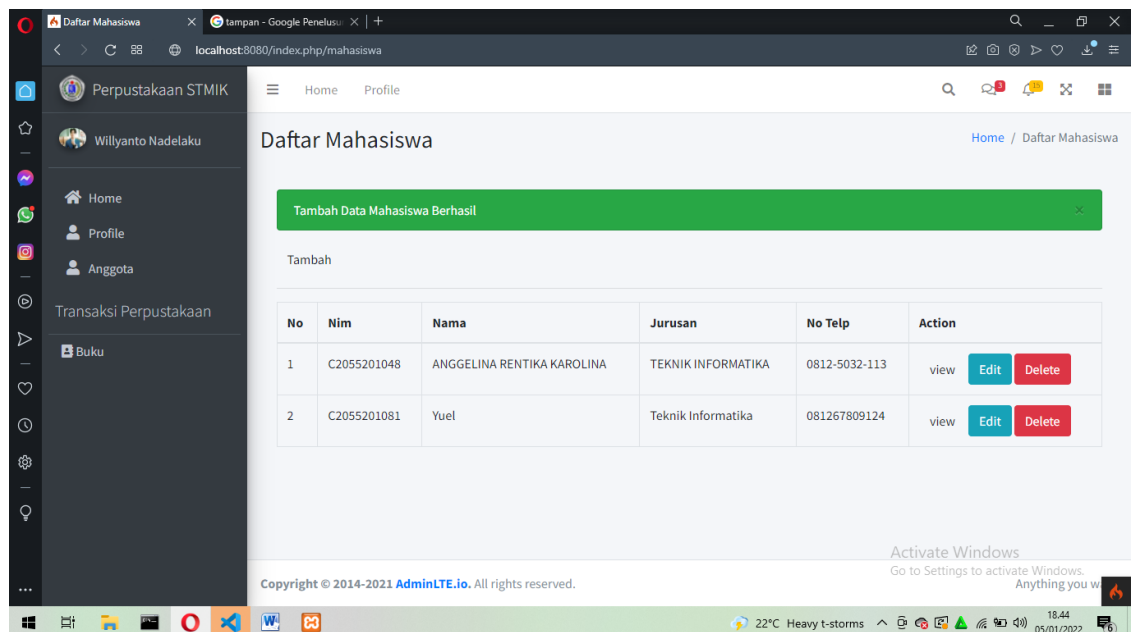
### 3. Profile

Fitur profile menampilkan data dari anggota kelompok yang mengedit aplikasi perpustakaan



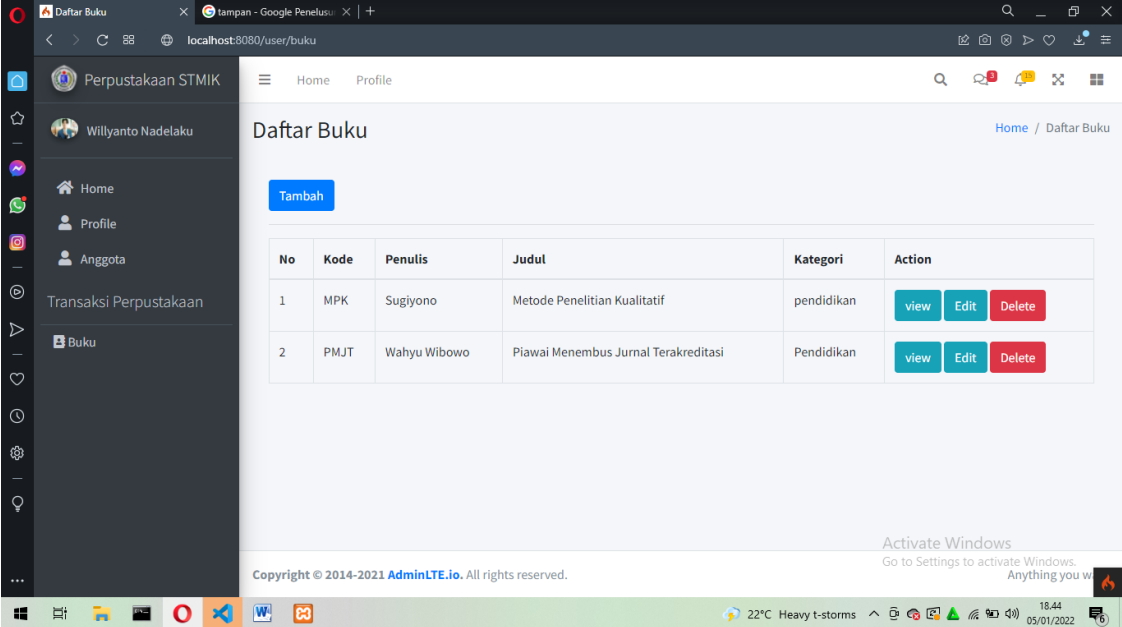
### 4. Anggota

Fitur anggota menampilkan daftar anggota yang ada didalam tabel mahasiswa, dimana dalam fitur ini terdapat tambah, edit, delete dan view



## 5. Buku

Fitur buku menampilkan daftar buku yang ada didalam tabel buku, dimana dalam fitur ini terdapat tambah, edit, delete dan view

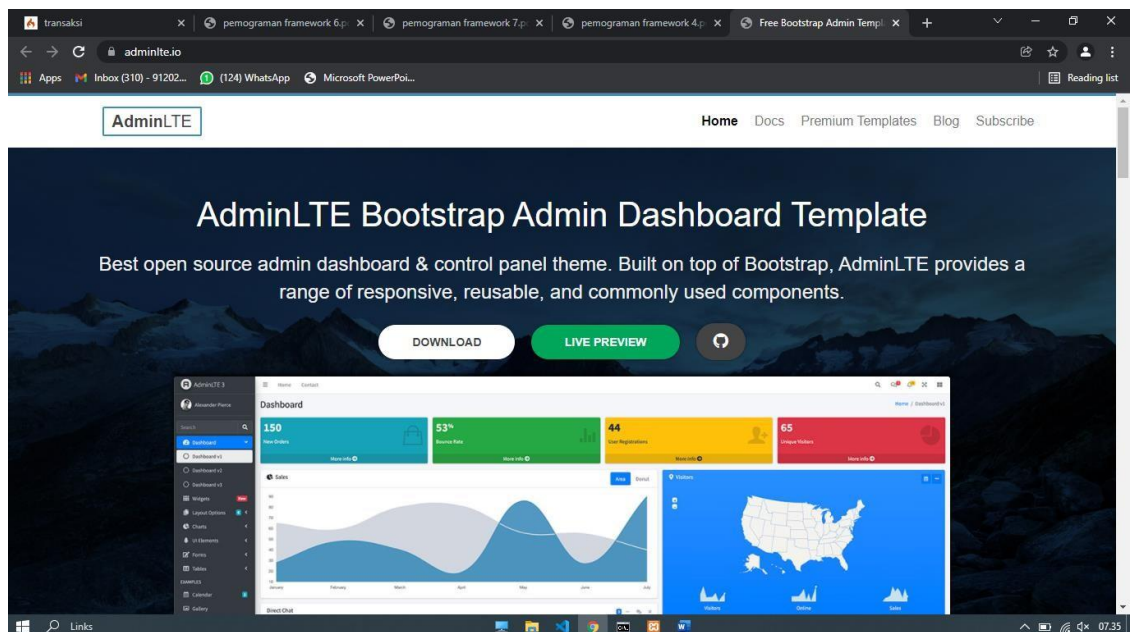


The screenshot displays a web application interface for a library system. The sidebar on the left includes navigation options: Home, Profile, Anggota, Transaksi Perpustakaan, and Buku. The main content area, titled 'Daftar Buku', features a 'Tambah' button and a table listing books. The table has columns for No, Kode, Penulis, Judul, Kategori, and Action. Two books are listed: one by Sugiyono and another by Wahyu Wibowo. Each book entry has 'view', 'Edit', and 'Delete' buttons. The footer shows copyright information for AdminLTE.io and a Windows activation notice.

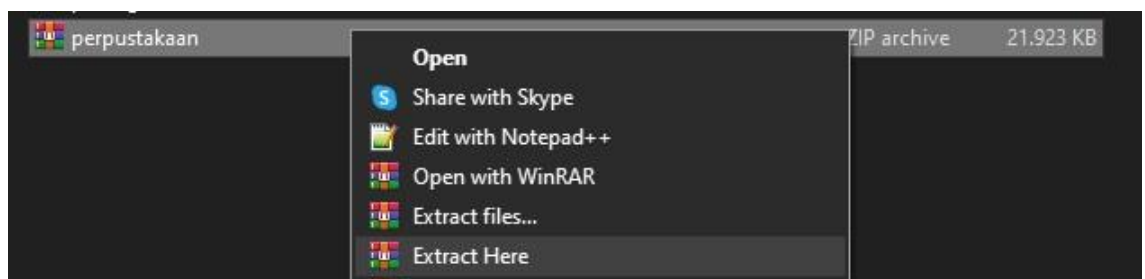
No	Kode	Penulis	Judul	Kategori	Action
1	MPK	Sugiyono	Metode Penelitian Kualitatif	pendidikan	<a href="#">view</a> <a href="#">Edit</a> <a href="#">Delete</a>
2	PMJT	Wahyu Wibowo	Piawai Menembus Jurnal Terakreditasi	Pendidikan	<a href="#">view</a> <a href="#">Edit</a> <a href="#">Delete</a>

## B. Integrasi Template AdminLTE

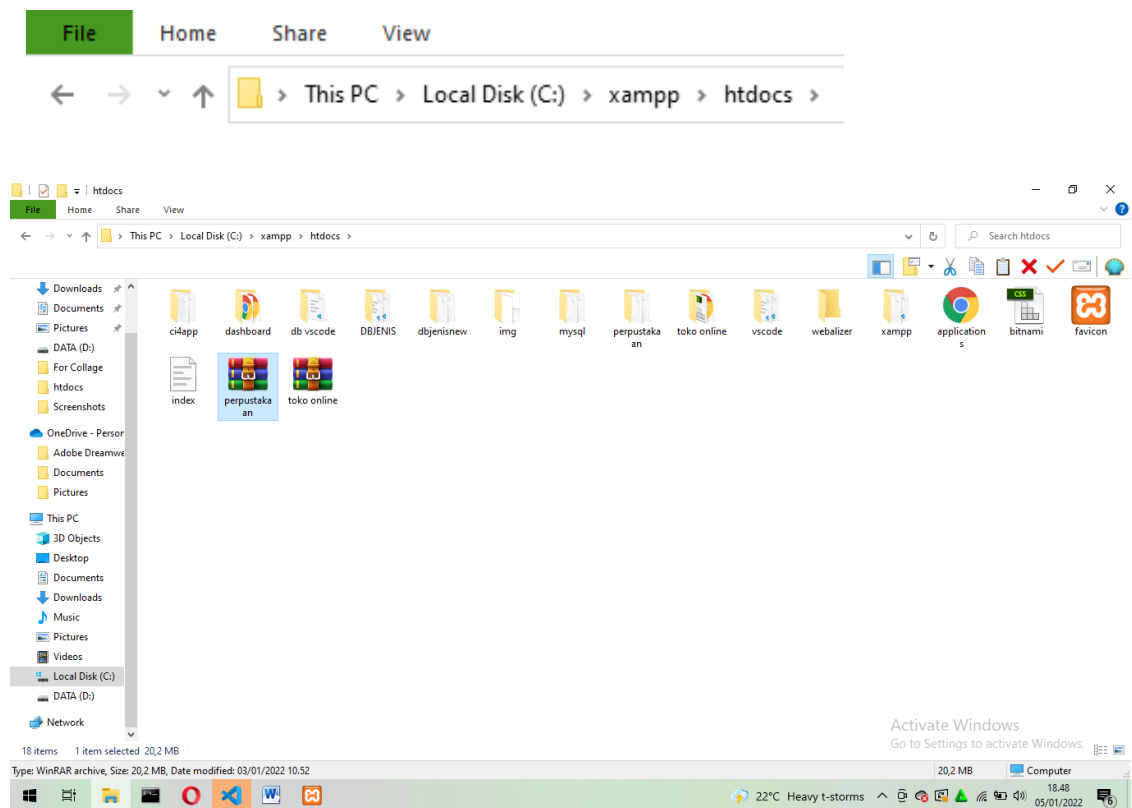
AdminLTE adalah template WebApp sumber terbuka populer untuk dasbord admin dan panel kontrol. Ini adalah template HTML responsif yang didasarkan pada kerangka CSS Bootstrap 4.6.0 pada versi terbarunya yaitu AdminLTE v3.1.0. Pada versi Ini menggunakan semua komponen Bootstrap dalam desainnya dan menata ulang banyak plugin yang biasa digunakan untuk membuat desain yang konsisten yang dapat digunakan sebagai antarmuka pengguna untuk aplikasi backend . AdminLTE didasarkan pada desain modular, yang memungkinkannya untuk disesuaikan dan dibangun dengan mudah. Untuk mendapatkannya kita dapat mendownload langsung di website resminya pada alamat <https://adminlte.io>



Dalam pengerjaan aplikasi perpustakaan, ekstrak file .zip perpustakaan



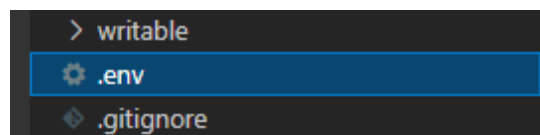
Hasil ekstrakan file .zip perpustakaan bisa di move ke local disk C -> xampp -> htdocs



Dalam pengerjaan Aplikasi Perpustakaan terdapat beberapa fitur, penjelasan pembuatan setiap fitur akan di jelaskan sebagai berikut :

### 1. Fitur login

- Langkah awal yang harus dilakukan adalah dengan melakukan konfigurasi untuk koneksi database pada file .env, hilangkan tanda “#” pada awal setiap baris kode



```

38 #-----
39
40 database.default.hostname = localhost
41 database.default.database = perpustakaan
42 database.default.username = root
43 database.default.password =
44 database.default.DBDriver = MySQLi
45 database.default.DBPrefix =
46

```

database.default.hostname = localhost

database.default.database = perpustakaan

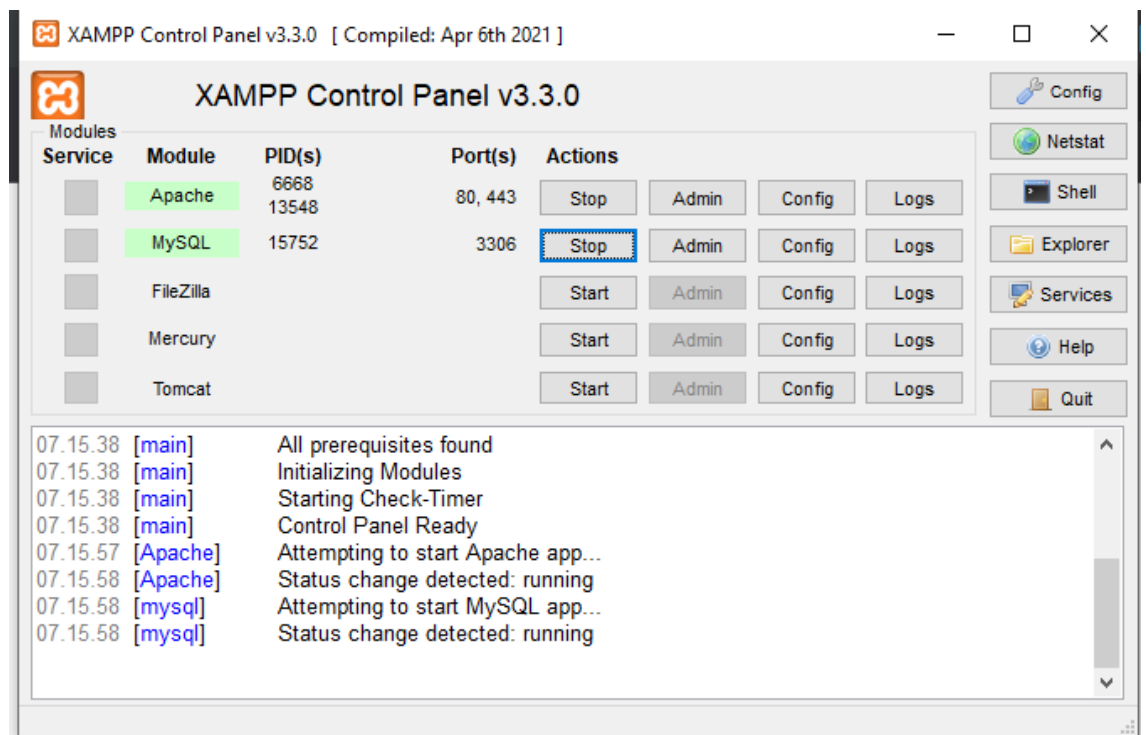
database.default.username = root

database.default.password =

database.default.DBDriver = MySQLi

- Membuat database perpustakaan melalui phpmyadmin

Jangan lupa mengaktifkan apache dan mysql pada aplikasi xampp terlebih dahulu



Akses localhost/phpmyadmin



Buat database yang bernama perpustakaan

## Databases

Create database ?

perpustakaan

utf8mb4\_general\_ci

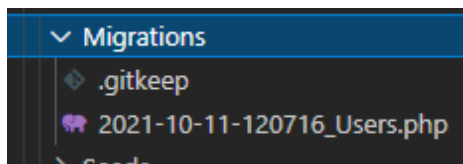
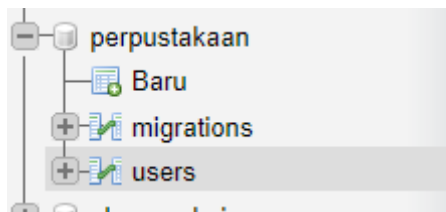
Create

- Membuat migration untuk table users didatabas

```
Try the new cross-platform PowerShell https://aka.ms/pscore6
```

```
PS C:\xampp\htdocs\perpustakaan> php spark migrate:create Users
```

```
File created: APPPATH\Database\Migrations\2021-10-14-150020_Users.php
```





- Edit users.php untuk membuat skema table users pada database perpustakaan

```
app > Database > Migrations > 2021-10-11-120716_Users.php
1  <?php
2  namespace App\Database\Migrations;
3  use CodeIgniter\Database\Migration;
4  class Users extends Migration
5  {
6  public function up()
7  {
8  $this->forge->addField([
9  'username' => [
10 'type' => 'VARCHAR',
11 'constraint' => '100',
12 ],
13 'password' => [
14 'type' => 'VARCHAR',
15 'constraint' => '100',
16 ],
17 'name' => [
18 'type' => 'VARCHAR',
19 'constraint' => '100',
20 ],
21 'created_at' => [
22 'type' => 'DATETIME',
23 'null' => true,
24 ],
25 'updated_at' => [
26 'type' => 'DATETIME',
27 'null' => true,
28 ]
29 ]);
30 $this->forge->addPrimaryKey('username', true);
31 $this->forge->createTable('users');
32 }
33 //-----
```

```
app > Database > Migrations > 2021-10-11-120716_Users.php
33 //-----
34 public function down()
35 {
36 $this->forge->dropTable('users');
37 }
38 }
```

- Untuk membuat tabel users di database perpustakaan perlu menjalankan fike \_Users migration pada terminal root di rektori aplikasi proyek

```
Running all new migrations...
Done migrations.
PS C:\xampp\htdocs\perpustakaan> php spark migrate

Running all new migrations...
Running: (App) 2021-10-11-120716_App\Database\Migrations\Users
Done migrations.

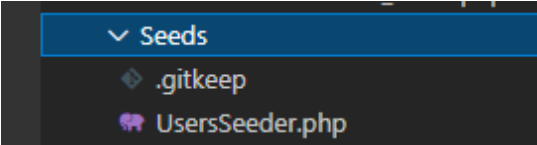
CodeIgniter v4.1.4 Command Line Tool - Server Time: 2021-10-14 10:05:37 UTC-05:00
File created: APPPATH\Database\Seeds\UsersSeeder.php
```

- Membuat seeding data users

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\xampp\htdocs\perpustakaan> php spark make:seeder UsersSeeder
```



- Buka file UsersSeeder.php lalu tuliskan code di dalam file UsersSeder.php

```

1  <?php
2  namespace App\Database\Seeds;
3  use CodeIgniter\Database\Seeder;
4  use CodeIgniter\I18n\Time;
5  class UsersSeeder extends Seeder
6  {
7      public function run()
8      {
9          //
10         $data = [
11             [
12                 'username' => 'willy',
13                 'password' => password_hash('willy2512', PASSWORD_BCRYPT),
14                 'name' => 'Williyanto Nadelaku',
15                 'created_at' => Time::now()
16             ]
17         ];
18         $this->db->table('users')->insertBatch($data);
19     }
20 }

```

- Keterangan :

Dalam perintah di atas, kita telah menginsert 1 data users dengan username miseri password offjumpol, dan namanya miseri ke dalam tabel users database.

- Menjalankan usersseeder untuk insert data user

```

PS C:\xampp\htdocs\perpustakaan> php spark db:seed UsersSeeder

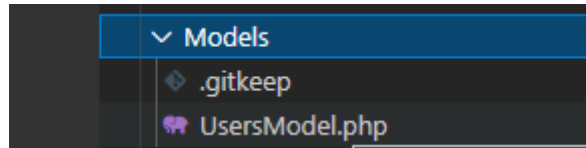
CodeIgniter v4.1.4 Command Line Tool - Server Time: 2021-10-14 10:07:01 UTC-05:00

Seeded: App\Database\Seeds\UsersSeeder
PS C:\xampp\htdocs\perpustakaan>

```

+ Opsi					
	username	password	name	created_at	updated_at
<input type="checkbox"/>	Ubah	Salin	Hapus	willy	\$2y\$10\$7uXyAiSpaHoC/BWb4nKjM.ryyIDi24dpn5rYytVjxnx... Williyanto Nadelaku 2021-12-02 01:56:57 NULL

- Membuat file UsersModel.php pada app\model\usermodel.php

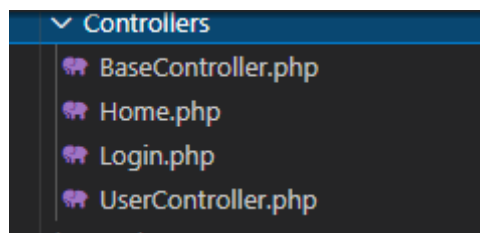


Model.php ini digunakan untuk merepresentasikan data yang terdapat pada tabel users di database perpustakaan

Programnya :

```
app > Models > UsersModel.php
1  <?php
2  namespace App\Models;
3  use CodeIgniter\Model;
4  class UsersModel extends Model
5  {
6      protected $table = "users";
7      protected $primaryKey = "username";
8      protected $returnType = "object";
9      protected $useTimestamps = true;
10     protected $allowedFields = ['username', 'password', 'name'];
11 }
```

- Membuat controller Login.php



Controler login.Php ini berfungsi untuk menangani proses login pada system perpustakaan

Programnya

:

```
app > Controllers > Login.php
21     'username' => $dataUser->username,
22     'name' => $dataUser->name,
23     'logged_in' => TRUE
24 ];
25 return redirect()->to(base_url('user/home'));
26 } else {
27     session()->setFlashdata('error', 'Username & Password Salah');
28     return redirect()->to(base_url('/backend'));
29 }
30 } else {
31     session()->setFlashdata('error', 'Username & Password Salah');
32     return redirect()->to(base_url('/backend'));
33 }
34 }
35 function logout()
36 {
37     session()->destroy();
38     return redirect()->to(base_url('/backend'));
39 }
40 }
```

Keterangan :

Pada controler login.php terdapat 3 method yaitu

a. Index()

Untuk mengembalikan halaman login yang ada pada folder view\dashboard

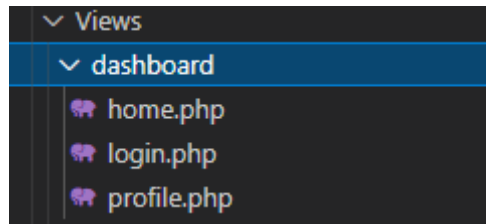
b. Process()

Untuk proses verifikasi data username dan password yang di inputkan melalui form login dengan data yang ada pada table users di database perpustakaan

c. Logout()

Untuk keluar dari system aplikasi backend

- Membuat view login.php di app\views\dashboard\login.php



View login.php berfungsi agar users login pada system project perpustakaan.

Halaman ini dapat digunakan dari template adminLTE sebelumnya

Programnya :

```

.env 2021-10-11-120716_Users.php UsersSeeder.php UsersModel.php Logi
app > Views > dashboard > login.php
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1">
6 <title>Perpustakaan | Log in</title>
7 <!-- Google Font: Source Sans Pro -->
8 <link rel="stylesheet"
9 href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,70
0&display=fallback">
11 <!-- Font Awesome -->
12 <link rel="stylesheet" href="plugins/fontawesome-free/css/all.min.css">
13 <!-- icheck bootstrap -->
14 <link rel="stylesheet" href="plugins/icheck-bootstrap/icheck-bootstrap.min.css">
15 <!-- Theme style -->
16 <link rel="stylesheet" href="dist/css/adminlte.min.css">
17 </head>
18 <body class="hold-transition login-page">
19 <div class="login-box">
20 <!-- /.login-logo -->
21 <div class="card card-outline card-primary">

```

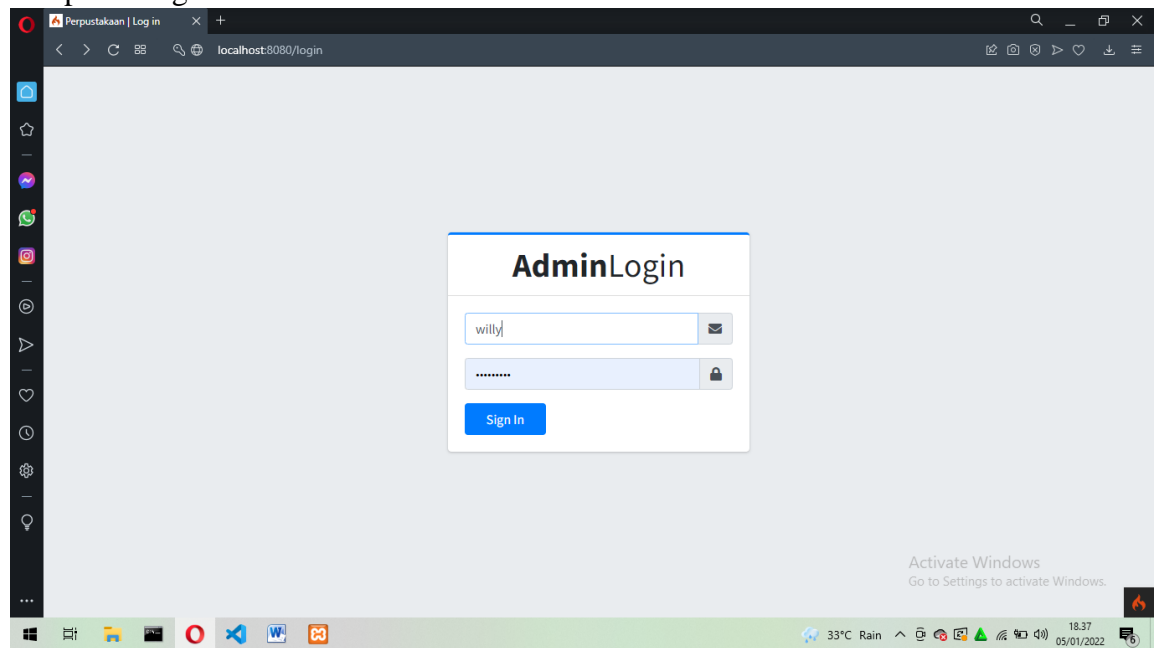
- menambahkan routing pada file routes.php

```

* -----
*/
$routes->get('/backend', 'Login::index');
$routes->post('/login/process', 'Login::process');
$routes->get('/logout', 'Login::logout');
/*

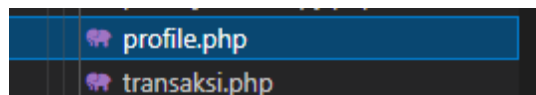
```

- Tampilan Login :



2. Fitur Profile

- Buatlah file view profile.php



```

home.php  Mahasiswa.php  profile.php X
app > Views > dashboard > profile.php
1  <?= $this->extend('layout/dashboard-layout'); ?>
2  <?= $this->section('content'); ?>
3
4
5  <div class="row">
6      <div class="col-md-3">
7
8
9          <!-- Profile Image -->
10         <div class="card card-orange card-outline">
11             <div class="card-body box-profile">
12                 <div class="text-center">
13                     Miseri</h3>
17
18                 <p class="text-muted text-center"></p>
19
20                 <ul class="list-group list-group-unbordered mb-3">
21                     <li class="list-group-item">
22                         <b>semester</b> <a class="float-right">3</a>
23                     </li>
24                     <li class="list-group-item">
25                         <b>kelas</b> <a class="float-right">TI B</a>
26                     </li>
27                     <li class="list-group-item">
28                         <b>Jenis kelamin</b> <a class="float-right">Perempuan</a>
29                     </li>
30                 </ul>
31             </div>
32         </div>
33

```

```

home.php  Mahasiswa.php  profile.php X
app > Views > dashboard > profile.php
35
36
37     </div>
38     <!-- /.col -->
39     <div class="col-md-9">
40         <!-- About Me Box -->
41         <div class="card card-orange">
42             <div class="card-header">
43                 <h3 class="card-title">About Misericordias Domini Natalia Boru Pandiangan</h3>
44             </div>
45             <!-- /.card-header -->
46             <div class="card-body">
47                 <strong><i class="fas fa-book mr-1"></i> Education</strong>
48
49                 <p class="text-muted">
50                     STMIK, Palangkaraya
51                 </p>
52
53                 <hr>
54
55                 <strong><i class="fas fa-map-marker-alt mr-1"></i> Location</strong>
56
57                 <p class="text-muted">Kalimantan Tengah, Palangkaraya</p>
58                 <hr>
59
60                 <strong><i class="far fa-file-alt mr-1"></i> Notes</strong>
61
62                 <p class="text-muted">...</p>
63             </div>
64             <!-- /.card-body -->
65         </div>
66         <!-- /.card -->
67

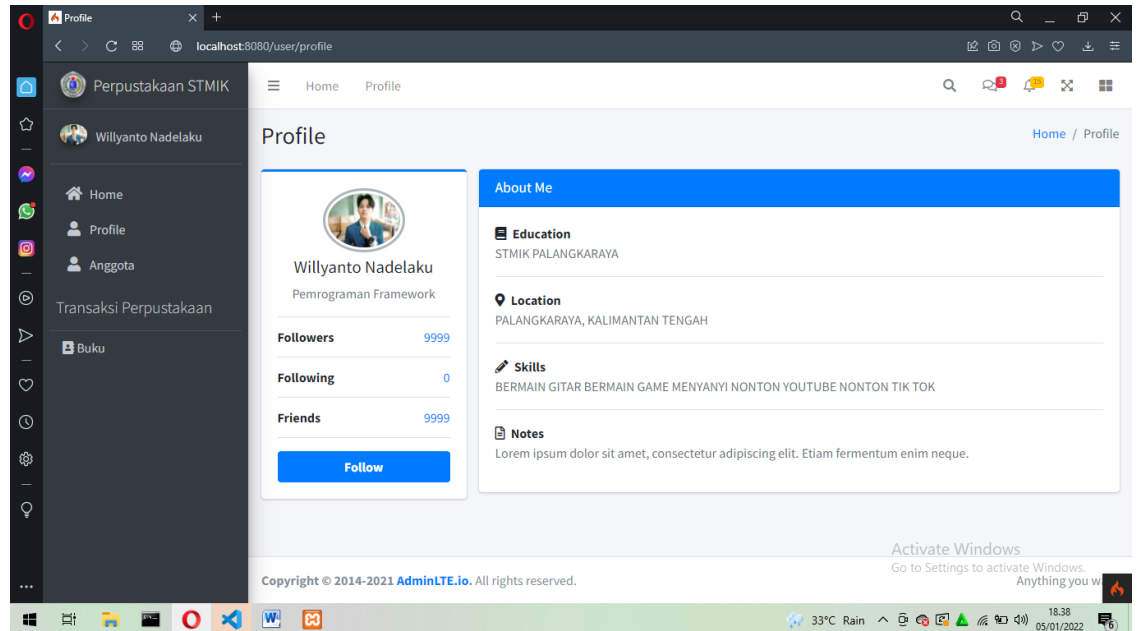
```



- Edit routes.php untuk menampilkan view profile.php

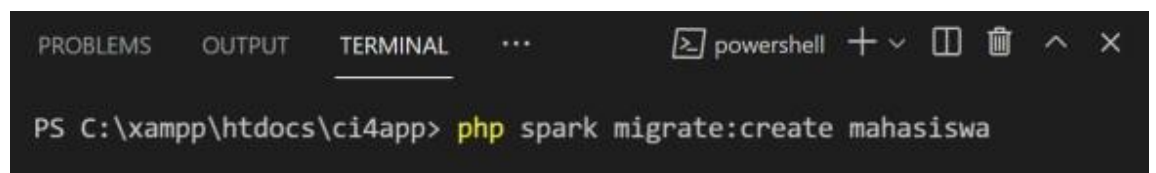
```
$routes->get('profile', 'UserController::profile', ['as' => 'user.profile']);
```

- Tampilan Profile

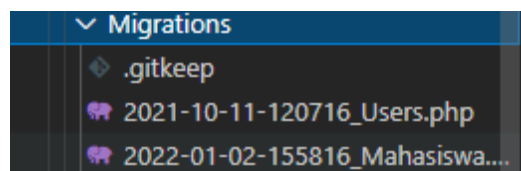


### 3. Fitur Anggota

- Membuat tabel mahasiswa dengan cara ketikkan command berikut di terminal VScode



Akan muncul file baru di folder app/Database/Migrations



Edit file mahasiswa.php tersebut untuk function menjadi :

```
home.php 2022-01-02-155816_Mahasiswa.php X
app > Database > Migrations > 2022-01-02-155816_Mahasiswa.php
1  <?php
2  namespace App\Database\Migrations;
3  use CodeIgniter\Database\Migration;
4  class Mahasiswa extends Migration
5  {
6  public function up()
7  {
8  $this->forge->addField([
9  'nim' => [
10 'type' => 'VARCHAR',
11 'constraint' => '11',
12 ],
13 'nama' => [
14 'type' => 'VARCHAR',
15 'constraint' => '255',
16 ],
17 'jurusan' => [
18     'type' => 'VARCHAR',
19     'constraint' => '255',
20 ],
21 'jenis_kelamin' => [
22     'type' => 'ENUM',
23     'constraint' => "'pria','wanita'",
24 ],
25 'no_telp' => [
26     'type' => 'VARCHAR',
27     'constraint' => '100',
28 ],
29 'email' => [
30     'type' => 'VARCHAR',
31     'constraint' => '100',
32 ],
33 'alamat' => [
```

```
home.php 2022-01-02-155816_Mahasiswa.php X
app > Database > Migrations > 2022-01-02-155816_Mahasiswa.php
33     'alamat' => [
34         'type' => 'VARCHAR',
35         'constraint' => '255',
36     ],
37     'foto' => [
38         'type' => 'VARCHAR',
39         'constraint' => '500',
40     ],
41     'created_at' => [
42         'type' => 'DATETIME',
43         'null' => true,
44     ],
45     'updated_at' => [
46         'type' => 'DATETIME',
47         'null' => true,
48     ],
49 ];
50 $this->forge->addPrimaryKey('nim');
51 $this->forge->createTable('mahasiswa');
52 }
53 //-----
54 public function down()
55 {
56     $this->forge->dropTable('mahasiswa');
57 }
58 }
```

Keterangan :

Perintah diatas digunakan untuk membuat sebuah tabel dengan nama mahasiswa .

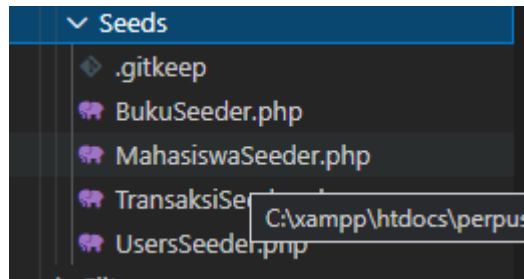
Menjalankan file migration untuk membuat tabel di database

```
Running all new migrations...
Done migrations.
PS C:\xampp\htdocs\perpustakaan> php spark migrate

Running all new migrations...
Running: (App) 2021-10-11-120716_App\Database\Migrations\Users
Done migrations.

CodeIgniter v4.1.4 Command Line Tool - Server Time: 2021-10-14 10:05:37 UTC-05:00
File created: APPPATH\Database\Seeds\UsersSeeder.php
```

- Membuat MahasiswaSeeder melalun terminal  
php spark make:seeder MahasiswaSeeder



Lalu edit dile mahasiswaSeeder.php tersebut dengan kebutuhan masing masing.

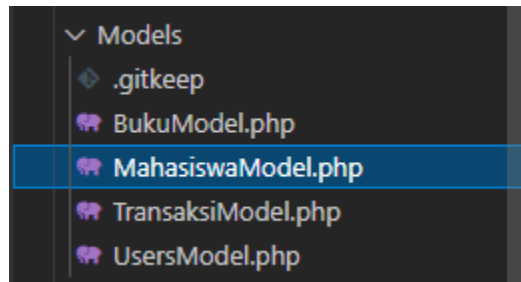
Keterangan :

Perintah diatas, insert 1 data mahasiswa.

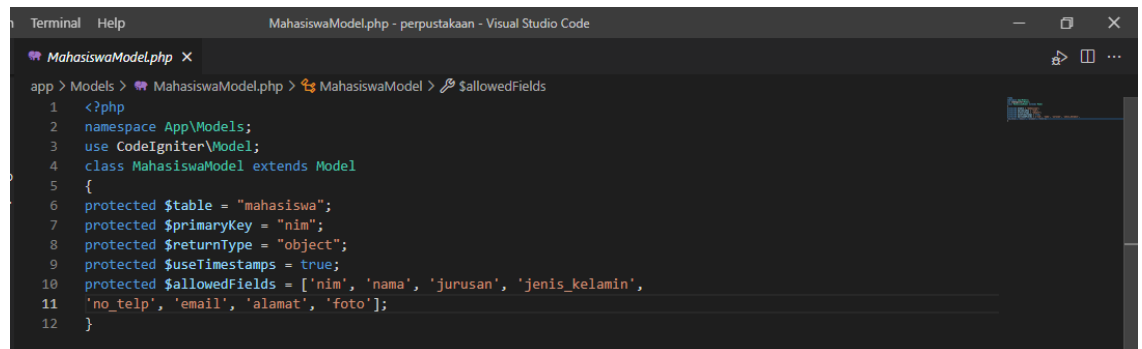
Berikut jalankan perintah diatas untuk insert data mahasiswa ke dalam tabell mahasiswa, denngan menuliskan perintah berikut ke dalam terminal VScode.

```
Done migrations.  
PS C:\xampp\htdocs\perpustakaan> php spark db:seed MahasiswaSeeder  
  
CodeIgniter v4.1.4 Command Line Tool - Server Time: 2021-12-01 23:15:13 UTC-06:00  
  
Seeded: App\Database\Seeds\MahasiswaSeeder  
PS C:\xampp\htdocs\perpustakaan> 
```

- Membuat file mahasiswamodel.php di dalam folder app/models



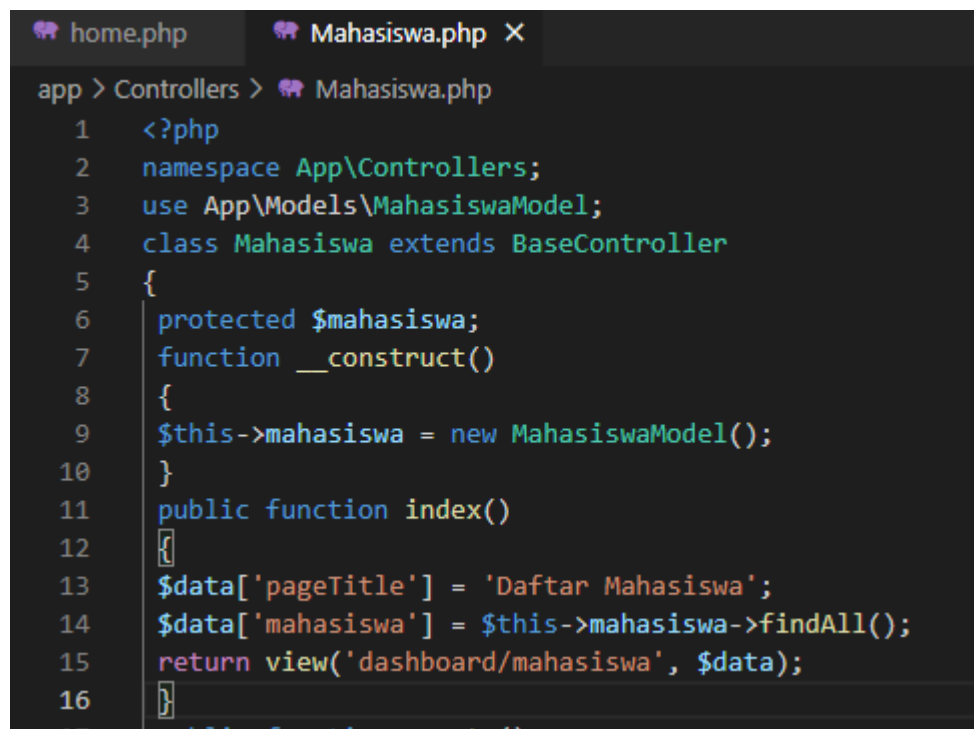
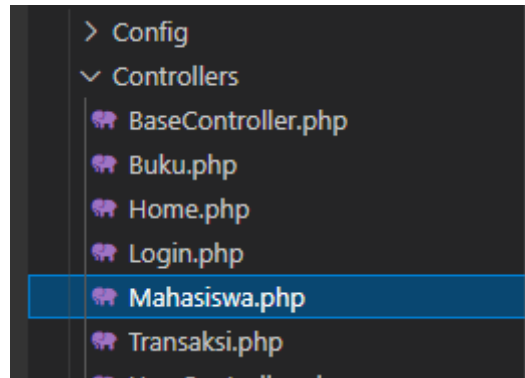
Program dari file mahasiswamodel.php :



Keterangan :

- Untuk classnya diberikan nama yang sama dengan filenya yaitu MahasiswaModel
- Beberapa settingan tulisannya :  
\$table dengan nilai mahasiswa karena kita akan komunikasi dengan tabel mahasiswa
- \$primaryKey dengan nilai nim karena kolom nim merupakan primary key dari tabel mahasiswa
- \$returnType dengan nilai object karena kita ingin untuk return type nantinya berupa object
- \$useTimestamps dengan nilai true karena kita akan mengisi kolom created\_at (saat insert data), dan kolom updated\_at (saat update data)
- \$allowedFields kita isikan nama kolom di tabel mahasiswa yang boleh diinsert data

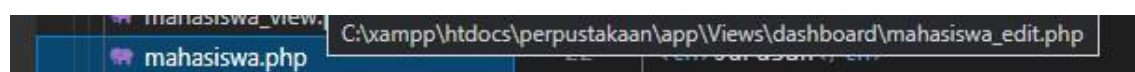
- Functions getAll() berfungsi sebagai data berelasi dengan tabel transaksi
- Membuat controller mahasiswa



Di file routes.php tambahkan :

```
$routes->get('/mahasiswa', 'Mahasiswa::index');
```

- Membuat halaman view mahasiswa pada folder app/views/dashboard/



```

home.php mahasiswa.php X
app > Views > dashboard > mahasiswa.php
1 <?= $this->extend('layout/dashboard-layout'); ?>
2 <?= $this->section('content'); ?>
3 <!-- /.card-header -->
4 <div class="card-body">
5 <?php if (!empty(session()->getFlashdata('message'))) : ?>
6 <div class="alert alert-success alert-dismissible fade show"
7 role="alert">
8 <?php echo session()->getFlashdata('message'); ?>
9 <button type="button" class="close" data-dismiss="alert" arialabel="Close">
10 <span aria-hidden="true">&times;</span>
11 </button>
12 </div>
13 <?php endif; ?>
14 <a href="<?= base_url('/mahasiswa/create'); ?>" class="btn btnprimary">Tambah</a>
15 <hr />
16 <table class="table table-bordered text-center">
17 <thead>
18 <tr>
19 <th>No</th>
20 <th>Nim</th>
21 <th>Nama</th>
22 <th>Jurusan</th>
23 <th>Action</th>
24 </tr>
25 </thead>
26 <tbody>
27 <?php
28 $no = 1;
29 foreach ($mahasiswa as $row) {
30 ?>
31 <tr>
32 <td><?= $no++; ?></td>
33 <td><?= $row->nim; ?></td>

```

```

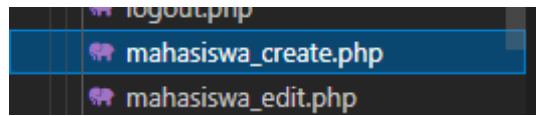
home.php mahasiswa.php X
app > Views > dashboard > mahasiswa.php
27 <?php
28 $no = 1;
29 foreach ($mahasiswa as $row) {
30 ?>
31 <tr>
32 <td><?= $no++; ?></td>
33 <td><?= $row->nim; ?></td>
34 <td><?= $row->nama; ?></td>
35 <td><?= $row->jurusan; ?></td>
36 <td>
37 <a title="View" href="<?=base_url("mahasiswa/view/$row->nim"); ?>" class="btn btn-dark">view</a>
38 <a title="Edit" href="<?= base_url("mahasiswa/edit/$row->nim"); ?>" class="btn btn-dark">Edit</a>
39 <a title="Delete" href="<?=base_url("mahasiswa/delete/$row->nim") ?>" class="btn btn-danger"
40 onclick="return confirm('Apakah Anda yakin ingin menghapus data?')">Delete</a>
41 </td>
42 </tr>
43 <?php
44 }
45 ?>
46 </tbody>
47 </table>
48 </div>
49 <!-- /.card-body -->
50 <?= $this->endSection('content'); ?>

```



- Membuat fitur create

Buatlah file mahasiswa\_create.php pada folder app/views/dashboard/



```

home.php  mahasiswa_create.php X
app > Views > dashboard > mahasiswa_create.php
1  <?= $this->extend('layout/dashboard-layout'); ?>
2  <?= $this->section('content'); ?>
3  <div class="col-md-10">
4      <!-- general form elements -->
5      <div class="card card-orange">
6          <div class="card-header">
7              <h3 class="card-title">Tambah Data Mahasiswa</h3>
8          </div>
9          <!-- /.card-header -->
10         <!-- form start -->
11         <?php if (!empty(session()->getFlashdata('error'))): ?>
12             <div class="alert alert-light alert-dismissible fade show"
13                 role="alert">
14                 <h4 class="alert-heading">Periksa Entrian Form</h4>
15                 </hr />
16                 <?php echo session()->getFlashdata('error'); ?>
17                 <button type="button" class="close" data-dismiss="alert" arialabel="Close">
18                     <span aria-hidden="true">&times;</span>
19                 </button>
20             </div>
21         <?php endif; ?>
22         <form class="form-horizontal" method="post" action="<?= base_url('mahasiswa/store') ?>"
23             enctype="multipart/form-data">
24             <?= csrf_field(); ?>
25             <div class="card-body">
26                 <div class="form-group row">
27                     <label for="nim" class="col-sm-2 col-formlabel">Nim</label>
28                     <div class="col-sm-6">
29                         <input type="text" class="form-control" id="nim"
30                             name="nim" placeholder="Nim" value="<?= old('nim'); ?>">
31                     </div>
32                 </div>
33                 <div class="form-group row">

```

```
home.php mahasiswa_create.php X
app > Views > dashboard > mahasiswa_create.php
38 </div>
39 </div>
40 <div class="form-group row">
41 <label for="jurusan" class="col-sm-2 col-formlabel">Jurusan</label>
42 <div class="col-sm-6">
43 <input type="text" class="form-control" id="jurusan"
44 name="jurusan" value="<?= old('jurusan'); ?>">
45 </div>
46 </div>
47 <div class="form-group row">
48 <label for="jenis_kelamin" class="col-sm-2 col-formlabel">Jenis Kelamin</label>
49 <div class="col-sm-6">
50 <select name="jenis_kelamin" class="form-control"
51 id="jenis_kelamin">
52 <option value="pria">Pria</option>
53 <option value="wanita">Wanita</option>
54 </select>
55 </div>
56 </div>
57 <div class="form-group row">
58 <label for="no_telp" class="col-sm-2 col-form-label">No
59 Telp</label>
60 <div class="col-sm-6">
61 <input type="text" class="form-control" id="no_telp"
62 name="no_telp" value="<?= old('no_telp') ?>" />
63 </div>
64 </div>
65 <div class="form-group row">
66 <label for="email" class="col-sm-2 col-formlabel">Email</label>
67 <div class="col-sm-6">
68 <input type="text" class="form-control" id="email" name="email" value="<?= old('email') ?>" />
69 </div>
70 </div>
```

```
home.php mahasiswa_create.php X
app > Views > dashboard > mahasiswa_create.php
61 <input type="text" class="form-control" id="no_telp"
62 name="no_telp" value="<?= old('no_telp') ?>" />
63 </div>
64 </div>
65 <div class="form-group row">
66 <label for="email" class="col-sm-2 col-formlabel">Email</label>
67 <div class="col-sm-6">
68 <input type="text" class="form-control" id="email" name="email" value="<?= old('email') ?>" />
69 </div>
70 </div>
71 <div class="form-group row">
72 <label for="alamat" class="col-sm-2 col-formlabel">Alamat</label>
73 <div class="col-sm-10">
74 <textarea class="form-control" name="alamat"
75 id="alamat"><?= old('alamat') ?></textarea>
76 </div>
77 </div>
78 <div class="form-group row" >
79 <label for="foto" class="col-sm-2 col-form-label">Foto Diri</label>
80 <div class="custom-file col-sm-6">
81 <input type="file" class="custom-file-input" id="foto" name="foto">
82 <label class="custom-file-label" for="foto">Pilih Foto Diri</label>
83 </div>
84 </div>
85
86 </div>
87 <!-- /.card-body -->
88 <div class="card-footer">
89 <button type="submit" class="btn btn-dark">Simpan</button>
90 </div>
91 <!-- /.card-footer -->
92 </form>
93 </div>
94 <!-- /.card -->
```

Tambahkan method baru create() dan store() di bawah ini pada file controller Mahasiswa.php

home.php

Mahasiswa.php X

app > Controllers > Mahasiswa.php

```
17 public function create()
18 {
19     $data['pageTitle'] = 'Input Data Mahasiswa';
20     return view('dashboard/mahasiswa_create', $data);
21 }
22 public function store()
23 {
24     if (!$this->validate([
25         'nim' => [
26             'rules' => 'required',
27             'errors' => [
28                 'required' => '{field} Harus diisi'
29             ]
30         ],
31         'nama' => [
32             'rules' => 'required',
33             'errors' => [
34                 'required' => '{field} Harus diisi'
35             ]
36         ],
37         'jurusan' => [
38             'rules' => 'required',
39             'errors' => [
40                 'required' => '{field} Harus diisi'
41             ]
42         ],
43         'jenis_kelamin' => [
44             'rules' => 'required',
45             'errors' => [
46                 'required' => '{field} Harus diisi'
47             ]
48         ],
49         'no_telp' => [
```

home.php

Mahasiswa.php X

app > Controllers > Mahasiswa.php

```
17 public function create()
18 {
19     $data['pageTitle'] = 'Input Data Mahasiswa';
20     return view('dashboard/mahasiswa_create', $data);
21 }
22 public function store()
23 {
24     if (!$this->validate([
25         'nim' => [
26             'rules' => 'required',
27             'errors' => [
28                 'required' => '{field} Harus diisi'
29             ]
30         ],
31         'nama' => [
32             'rules' => 'required',
33             'errors' => [
34                 'required' => '{field} Harus diisi'
35             ]
36         ],
37         'jurusan' => [
38             'rules' => 'required',
39             'errors' => [
40                 'required' => '{field} Harus diisi'
41             ]
42         ],
43         'jenis_kelamin' => [
44             'rules' => 'required',
45             'errors' => [
46                 'required' => '{field} Harus diisi'
47             ]
48         ],
49         'no_telp' => [
```

```
home.php  Mahasiswa.php X
app > Controllers > Mahasiswa.php
48     ],
49     'no_telp' => [
50         'rules' => 'required',
51         'errors' => [
52             'required' => '{field} Harus diisi'
53         ]
54     ],
55     'email' => [
56         'rules' => 'required|valid_email',
57         'errors' => [
58             'required' => '{field} Harus diisi',
59             'valid_email' => 'Email Harus Valid'
60         ]
61     ],
62     'alamat' => [
63         'rules' => 'required',
64         'errors' => [
65             'required' => '{field} Harus diisi'
66         ]
67     ],
68     'foto' => [
69         'rules' =>
70         'uploaded[foto]|mime_in[foto,image/jpg,image/jpeg,image/gif,image/png]|max_size[foto,1024]',
71         'errors' => [
72             'uploaded' => 'Harus Ada File yang diupload',
73             'mime_in' => 'File Extension Harus Berupa jpg,jpeg,gif,png',
74             'max_size' => 'Ukuran File Maksimal 1 MB'
75         ]
76     ],
77     ))) {
78         session()->setFlashdata('error', $this->validator->listErrors());
79         return redirect()->back()->withInput();
80     }
81 }
```

```
home.php  Mahasiswa.php X
app > Controllers > Mahasiswa.php
80
81     }
82     $filfoto = $this->request->getFile('foto');
83     $namaFilefoto = $filefoto->getRandomName();
84     $filefoto->move('img', $namaFilefoto);
85
86     $this->mahasiswa->insert([
87         'nim' => $this->request->getVar('nim'),
88         'nama' => $this->request->getVar('nama'),
89         'jurusan' => $this->request->getVar('jurusan'),
90         'jenis_kelamin' => $this->request->getVar('jenis_kelamin'),
91         'no_telp' => $this->request->getVar('no_telp'),
92         'email' => $this->request->getVar('email'),
93         'alamat' => $this->request->getVar('alamat'),
94         'foto' => $namaFilefoto
95     ]);
96     session()->setFlashdata('message', 'Tambah Data Mahasiswa
97     Berhasil');
98     return redirect()->to('/mahasiswa');
99 }
```



- Membuat fitur edit

Buatlah method edit di controller Mahasiswa

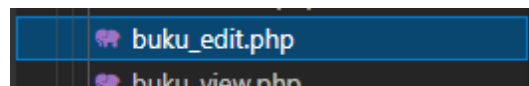
```
home.php  Mahasiswa.php X
app > Controllers > Mahasiswa.php
98  return redirect()->to( /mahasiswa );
99  }
100 function edit($nim)
101 {
102     $dataMahasiswa = $this->mahasiswa->find($nim);
103     if (empty($dataMahasiswa)) {
104         throw new
105         \CodeIgniter\Exceptions\PageNotFoundException('Data Mahasiswa
106         Tidak ditemukan !');
107     }
108     $data['pageTitle'] = 'Edit Data Mahasiswa';
109     $data['mahasiswa'] = $dataMahasiswa;
110     return view('dashboard/mahasiswa_edit', $data);
111 }
112 public function update($nim)
113 {
114     if (!$this->validate([
115         'nama' => [
116             'rules' => 'required',
117             'errors' => [
118                 'required' => '{field} Harus diisi'
119             ]
120         ],
121         'jurusan' => [
122             'rules' => 'required',
123             'errors' => [
124                 'required' => '{field} Harus diisi'
125             ]
126         ],
127         'jenis_kelamin' => [
128             'rules' => 'required',
129             'errors' => [
130                 'required' => '{field} Harus diisi'
131             ]
132         ]
133     ])) {
134         return redirect()->to( /mahasiswa );
135     }
136     $dataMahasiswa = $this->mahasiswa->find($nim);
137     $dataMahasiswa->update($nim);
138     return redirect()->to( /mahasiswa );
139 }
```

```
home.php  Mahasiswa.php X
app > Controllers > Mahasiswa.php
130     'required' => '{field} Harus diisi'
131     ],
132 ],
133     'no_telp' => [
134         'rules' => 'required',
135         'errors' => [
136             'required' => '{field} Harus diisi'
137         ]
138     ],
139     'email' => [
140         'rules' => 'required|valid_email',
141         'errors' => [
142             'required' => '{field} Harus diisi',
143             'valid_email' => 'Email Harus Valid'
144         ]
145     ],
146     'alamat' => [
147         'rules' => 'required',
148         'errors' => [
149             'required' => '{field} Harus diisi'
150         ]
151     ],
152     'foto' => [
153         'rules' =>
154             'mime_in[foto,image/jpg,image/jpeg,image/gif,image/png]|max_size[foto,1024]',
155         'errors' => [
156             'mime_in' => 'File Extention Harus Berupa jpg,jpeg,gif,png',
157             'max_size' => 'Ukuran File Maksimal 1 MB'
158         ]
159     ],
160 )) {
161     session()->setFlashdata('error', $this->validator->listErrors());
162     return redirect()->back();
```



```
home.php  Mahasiswa.php X
app > Controllers > Mahasiswa.php
161 session()->setFlashdata('error', $this->validation->hasErrors());
162 return redirect()->back();
163 }
164 //ambil nama fotolama dari variabel tipe hidden
165 $filefotolama = $this->request->getVar('fotolama');
166 //cek data file foto
167 $filefoto = $this->request->getFile('foto');
168 //jika file kosong jalankan perintah mengambil data nama file lama
169 if ($filefoto->getError() == 4) {
170 $namaFilefoto = $this->request->getVar('fotolama');
171 } else {
172 // jika tidak hapus file foto lama dan ambil data file foto baru
173 $path = '../public/img/';
174 @unlink($path . $filefotolama);
175 //generate nama file random
176 $namaFilefoto = $filefoto->getRandomName();
177 //pindahkan gambar
178 $filefoto->move('img', $namaFilefoto);
179 }
180 $this->mahasiswa->update($nim, [
181 'nama' => $this->request->getVar('nama'),
182 'jurusan' => $this->request->getVar('jurusan'),
183 'jenis_kelamin' => $this->request->getVar('jenis_kelamin'),
184 'no_telp' => $this->request->getVar('no_telp'),
185 'email' => $this->request->getVar('email'),
186 'alamat' => $this->request->getVar('alamat'),
187 'foto' => $namaFilefoto
188 ]);
189
190 session()->setFlashdata('message', 'Update Data Mahasiswa
191 Berhasil');
192 return redirect()->to('/mahasiswa');
193 }
194 function delete($nim)
```

- Kemudian buatlah form edit data pada folder  
app/views/dassboard/mahasiswa\_edit.php



home.php Mahasiswa.php buku\_edit.php X

app > Views > dashboard > buku\_edit.php

```
1 <?= $this->extend('layout/dashboard-layout'); ?>
2 <?= $this->section('content'); ?>
3 <div class="col-md-10">
4     <!-- general form elements -->
5     <div class="card card-orange">
6         <div class="card-header">
7             <h3 class="card-title">Edit Data Buku</h3>
8         </div>
9     <!-- /.card-header -->
10    <!-- form start -->
11    <?php if (!empty(session()->getFlashdata('error'))): ?>
12        <div class="alert alert-light alert-dismissible fade show" role="alert">
13            <h4 class="alert-heading">Periksa Entrian Form</h4>
14            </div>
15        <?php echo session()->getFlashdata('error'); ?>
16        <button type="button" class="close" data-dismiss="alert" arialabel="Close">
17            <span aria-hidden="true">&times;</span>
18        </button>
19    </div>
20    <?php endif; ?>
21    <form class="form-horizontal" method="post" action="<?=base_url('buku/update/'.$buku->kode) ?>"
22        enctype="multipart/form-data">
23        <?= csrf_field(); ?>
24        <div class="card-body">
25
26            <div class="form-group row">
27                <label for="kode" class="col-sm-2 col-formlabel">Kode Buku</label>
28                <div class="col-sm-6">
29                    <input type="text" class="form-control" id="kode"
30                        name="kode" value="<?= $buku->kode; ?>">
31                </div>
32            </div>
33
```

```
home.php  Mahasiswa.php  buku_edit.php X
app > Views > dashboard > buku_edit.php
33
34 <div class="form-group row">
35 <label for="judul" class="col-sm-2 col-formlabel">Judul Buku</label>
36 <div class="col-sm-6">
37 <input type="text" class="form-control" id="judul"
38 name="judul" value="<?=$buku->judul; ?>">
39 </div>
40 </div>
41 <div class="form-group row">
42 <label for="penulis" class="col-sm-2 col-formlabel">Nama Penulis</label>
43 <div class="col-sm-6">
44 <input type="text" class="form-control" id="penulis"
45 name="penulis" value="<?=$buku->penulis; ?>">
46 </div>
47 </div>
48
49 <div class="form-group row">
50 <label for="penerbit" class="col-sm-2 col-formlabel">Penerbit Buku</label>
51 <div class="col-sm-6">
52 <input type="text" class="form-control" id="penerbit"
53 name="penerbit" value="<?=$buku->penerbit; ?>">
54 </div>
55 </div>
56
57 <div class="form-group row">
58 <input type="hidden" id="fotolama" name="fotolama" value="<?=$buku->foto; ?>">
59 <label for="foto" class="col-sm-2 col-form-label">Sampul Buku</label>
60 <div class="custom-file col-sm-6">
61 <input type="file" class="custom-file-input" id="foto" name="foto">
62 <label class="custom-file-label" for="foto"><?=$buku->foto; ?></label>
63 </div>
64 <?php if (!empty($buku->foto)) : ?>
65 <div class="card-body box-profile col-sm-6">
```

```
home.php  Mahasiswa.php  buku_edit.php X
app > Views > dashboard > buku_edit.php
61 <input type="file" class="custom-file-input" id="foto" name="foto">
62 <label class="custom-file-label" for="foto"><?=$buku->foto; ?></label>
63 </div>
64 <?php if (!empty($buku->foto)) : ?>
65 <div class="card-body box-profile col-sm-6">
66 <div class="text-center">
67 
68 </div>
69 </div>
70 <?php endif; ?>
71 </div>
72
73 </div>
74 <!-- /.card-body -->
75 <div class="card-footer">
76 <button type="submit" class="btn btn-dark">Simpan</button>
77 </div>
78 <!-- /.card-footer -->
79 </form>
80 </div>
81 <!-- /.card -->
82 </div>
83 <?=$this->endSection('content'); ?>
84
```

- Membuat fitur delete

Buat method delete di controller mahasiswa.php

```
$this->mahasiswa->delete($nim);

//hapus file image dari folder public
$filefoto = $dataMahasiswa->foto;
$path = '../public/img/';
@unlink($path . $filefoto);

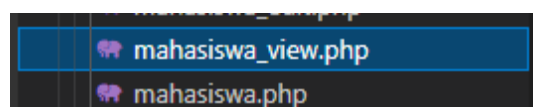
session()->setFlashdata('message', 'Delete Data Mahasiswa
Berhasil');
return redirect()->to('/mahasiswa');
}
```

- Membuat fitur view

Buat method view di controller mahasiswa.php

```
public function view($nim)
{
    $dataMahasiswa = $this->mahasiswa->find($nim);
    if (empty($dataMahasiswa)) {
        throw new
        \CodeIgniter\Exceptions\PageNotFoundException('Data Mahasiswa
        Tidak ditemukan !');
    }
    $data['pageTitle'] = 'Data Mahasiswa';
    $data['mahasiswa'] = $dataMahasiswa;
    return view('dashboard/mahasiswa_view', $data);
}
}
```

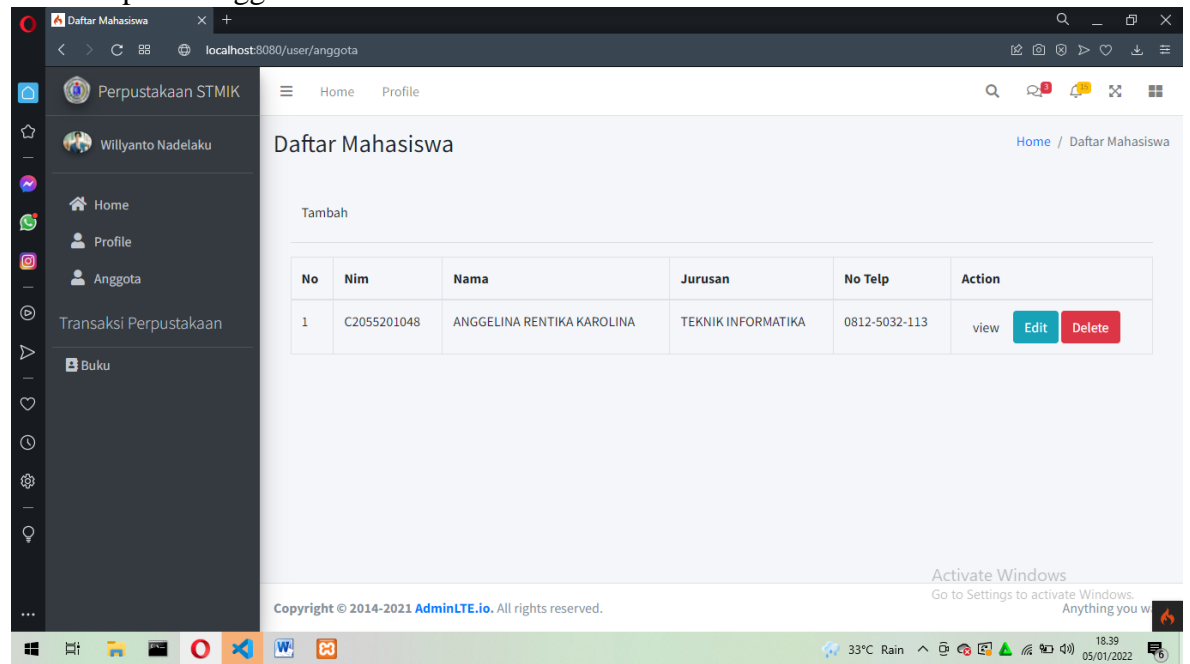
Buat file mahasiswa\_view.php di folder app/views



```
home.php Mahasiswa.php mahasiswa_view.php X
app > Views > dashboard > mahasiswa_view.php
1 <?= $this->extend('layout/dashboard-layout'); ?>
2 <?= $this->section('content'); ?>
3 <div class="col-md-12">
4 <!-- general form elements -->
5 <section class="content">
6 <div class="container-fluid">
7 <div class="row">
8 <!-- /.card-header -->
9 <!-- form start -->
10 <div class="col-md-3">
11 <!-- Profile Image -->
12 <div class="card card-orange card-outline">
13 <div class="card-body box-profile">
14 <div class="text-center">
15 
16 </div>
17 </div>
18 <!-- /.card-body -->
19 </div>
20 <!-- /.card -->
21 </div>
22 <div class="col-md-9">
23 <div class="card card-orange">
24 <div class="card-header">
25 <h3 class="card-title">Data Anggota</h3>
26 </div>
27 <!-- /.card-header -->
28 <div class="card-body">
29 <strong><i></i>Nama Mahasiswa</strong>
30 <p class="text-muted"><?= $mahasiswa->nama; ?> ( <?= $mahasiswa->nim; ?> )</p>
31 <hr>
32 <strong><i></i>Jurusan</strong>
33 <p class="text-muted"><?= $mahasiswa->jurusan;
```

```
home.php Mahasiswa.php mahasiswa_view.php X
app > Views > dashboard > mahasiswa_view.php
35 <hr>
36 <strong><i></i>Jenis Kelamin</strong>
37 <p class="text-muted"><?= $mahasiswa->jenis_kelamin;
38 ?></p>
39 <hr>
40 <strong><i></i>No Telpon / Email</strong>
41 <p class="text-muted"><?= $mahasiswa->no_telp;?> / <?= $mahasiswa->email;?></p>
42 <hr>
43 <strong><i></i>Alamat Mahasiswa</strong>
44 <p class="text-muted"><?= $mahasiswa->alamat;
45 ?></p>
46 </hr>
47 </div>
48 <!-- /.card-body -->
49 </div>
50 <!-- /.card -->
51 </div>
52 </div>
53 <!-- /.card -->
54 </div>
55 </div>
56 <?= $this->endSection('content'); ?>
```

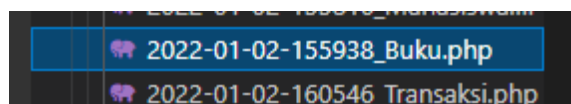
- Tampilan Anggota :



4. Fitur Buku

Pada fitur buku, sebenarnya pembuatanya cukup sama dengan fitur anggota. Jadi pada penjelasan fitur buku hanya di terapkan beberapa hal yang tidak umum yang terdapat pada fitur anggota.

- Membuat migration buku





home.php

Mahasiswa.php

2022-01-02-155938\_Buku.php X

app > Database > Migrations > 2022-01-02-155938\_Buku.php

```
1  <?php
2  namespace App\Database\Migrations;
3  use CodeIgniter\Database\Migration;
4  class Buku extends Migration
5  {
6  public function up()
7  {
8  $this->forge->addField([
9  'kode' => [
10  'type' => 'VARCHAR',
11  'constraint' => '255',
12  ],
13  'judul' => [
14  'type' => 'VARCHAR',
15  'constraint' => '255',
16  ],
17  'slug' => [
18  'type' => 'VARCHAR',
19  'constraint' => '255',
20  ],
21  'penulis' => [
22  'type' => 'VARCHAR',
23  'constraint' => '255',
24  ],
25  'penerbit' => [
26  'type' => 'VARCHAR',
27  'constraint' => '255',
28  ],
29  'foto' => [
30  'type' => 'VARCHAR',
31  'constraint' => '500'
32  ],
33  'created_at' => [
```

```
home.php  Mahasiswa.php  2022-01-02-155938_Buku.php X
app > Database > Migrations > 2022-01-02-155938_Buku.php
27  'constraint' => '255',
28  ],
29  'foto' => [
30  'type' => 'VARCHAR',
31  'constraint' => '500'
32  ],
33  'created_at' => [
34  'type' => 'DATETIME',
35  'null' => true,
36  ],
37  'updated_at' => [
38  'type' => 'DATETIME',
39  'null' => true,
40  ]
41  ]);
42  $this->forge->addPrimaryKey('kode');
43  $this->forge->createTable('buku');
44  }
45  //-----
46  public function down()
47  {
48  $this->forge->dropTable('buku');
49  }
50  }
```

Keterangan :

Perintah diatas digunakan untuk membuat sebuah tabel dengan nama buku .

Menjalankan file migration untuk membuat tabel di database

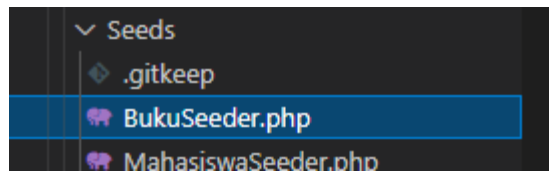
```
Running all new migrations...
Done migrations.
PS C:\xampp\htdocs\perpustakaan> php spark migrate

Running all new migrations...
Running: (App) 2021-10-11-120716_App\Database\Migrations\Users
Done migrations.

CodeIgniter v4.1.4 Command Line Tool - Server Time: 2021-10-14 10:05:37 UTC-05:00
File created: APPPATH\Database\Seeds\UsersSeeder.php
```

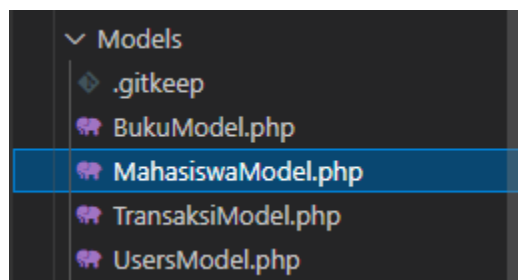


- Membuat BukuSeeder melalun terminal  
php spark make:seeder BukuSeeder



```
Terminal Help BukuSeeder.php - perpustakaan - Visual Studio Code
BukuSeeder.php X
app > Database > Seeds > BukuSeeder.php > BukuSeeder
1 <?php
2 namespace App\Database\Seeds;
3 use CodeIgniter\Database\Seeder;
4 use CodeIgniter\I18n\Time;
5 class BukuSeeder extends Seeder
6 {
7     public function run()
8     {
9         $data = [
10         [
11             'kode' => 'MPK',
12             'penulis' => 'Sugiyono',
13             'judul' => 'Metode Penelitian Kualitatif',
14             'kategori' => 'pendidikan',
15             'created_at' => Time::now()
16         ],
17         [
18             'kode' => 'MLKS',
19             'penulis' => 'Wildan Zulkarnain',
20             'judul' => 'Manajemen Layanan Khusus di Sekolah',
21             'kategori' => 'pendidikan',
22             'created_at' => Time::now()
23         ]
24     ];
25     $this->db->table('buku')->insertBatch($data);
26 }
27 }
```

- Membuat file mahasiswamodel.php di dalam folder app/models



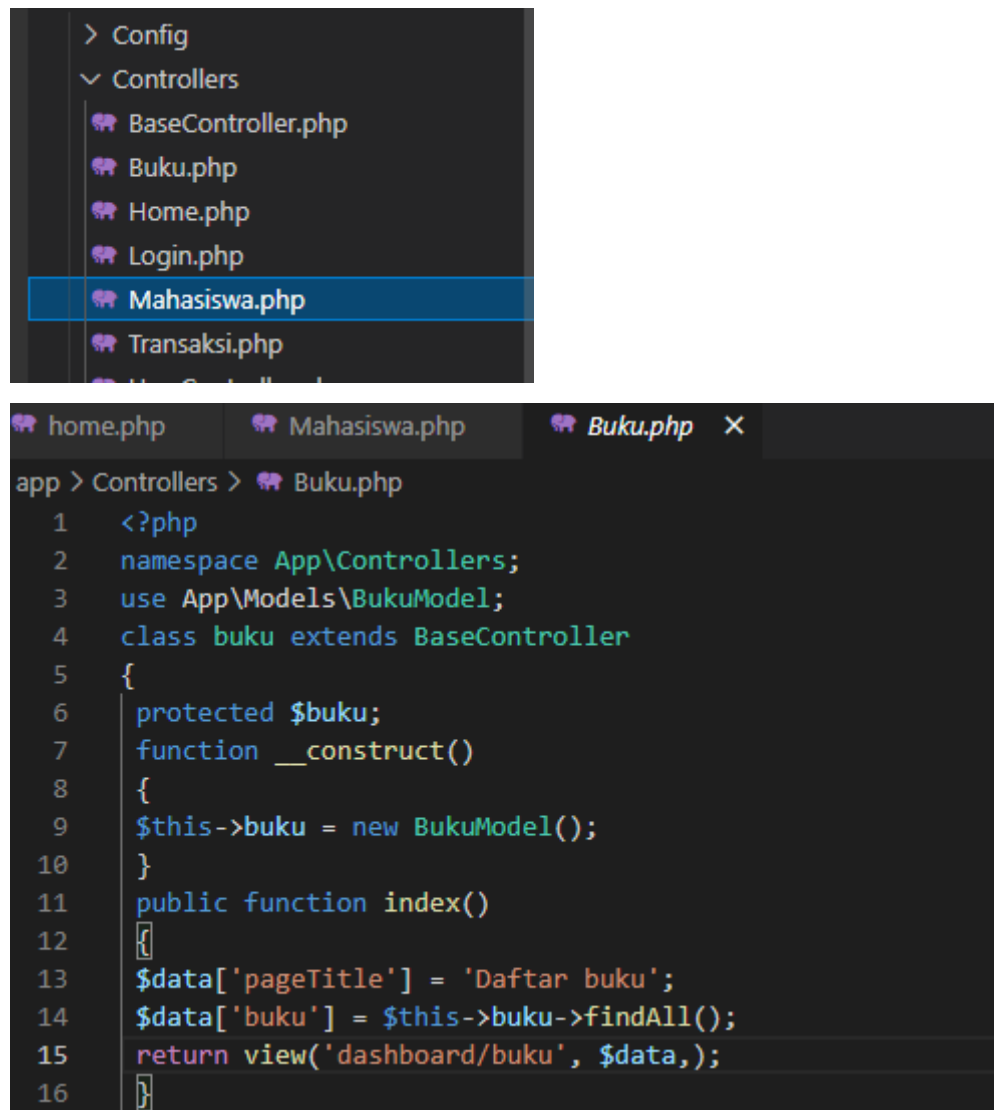
Program dari file bukumodel.php :

```
app > Models > BukuModel.php
1  <?php
2  namespace App\Models;
3  use CodeIgniter\Model;
4  class BukuModel extends Model
5  {
6      protected $table = "buku";
7      protected $primaryKey = "kode";
8      protected $returnType = "object";
9      protected $useTimestamps = true;
10     protected $allowedFields = ['kode', 'judul', 'penulis', 'penerbit', 'foto'];
11
12
13     function getAll()
14     {
15         $builder = $this->db->table('buku');
16         $builder->select('*');
17         $builder->join('transaksi', 'transaksi.kode_buku=buku.kode', 'left');
18         $query = $builder->get();
19         return $query->getResult();
20     }
21 }
22 ?>
```

Keterangan :

- Untuk classnya diberikan nama yang sama dengan filenya yaitu bukumodel
- Beberapa settingan tulisannya :
  - \$table dengan nilai mahasiswa karena kita akan komunikasi dengan tabel buku
  - \$primaryKey dengan nilai nim karena kolom kode merupakan primary key dari tabel buku
  - \$returnType dengan nilai object karena kita ingin untuk return type nanti berupa object
  - \$useTimestamps dengan nilai true karena kita akan mengisi kolom created\_at (saat insert data), dan kolom updated\_at (saat update data)
  - \$allowedFields kita isikan nama kolom di tabel mahasiswa yang boleh diinsert data
- Functions getAll() berfungsi sebagai data berelasi dengan tabel transaksi

- Membuat controler buku



Di file routes.php tambahkan :

```
$routes->get('/buku, buku::index');
```

- Catatan :

Untuk komponen lainnya seperti buku.php pada folder views/dashboard, form tambah, edit, view dan delete dapat dilihat pada bagian fitur anggota karena pembuatannya hampir sama saja.

- Tampilan Buku

Home

Profile

Daftar Buku

Home / Daftar Buku

Tambah

No	Kode	Penulis	Judul	Kategori	Action
1	MPK	Sugiyono	Metode Penelitian Kualitatif	pendidikan	<div>viewEditDelete</div>
2	PMJT	Wahyu Wibowo	Piawai Menembus Jurnal Terakreditasi	Pendidikan	<div>viewEditDelete</div>