

Laporan Presentasi Aplikasi Perpustakaan



Dosen Pengampu :

Herkules S.Kom.,M.Cs

Mahasiswa :

Misericordias Domini Natalia Boru Pandiangan (C2055201041)

Anjela Rastiana (C2055201042)

Angelina Rentika Karolina (C2055201048)

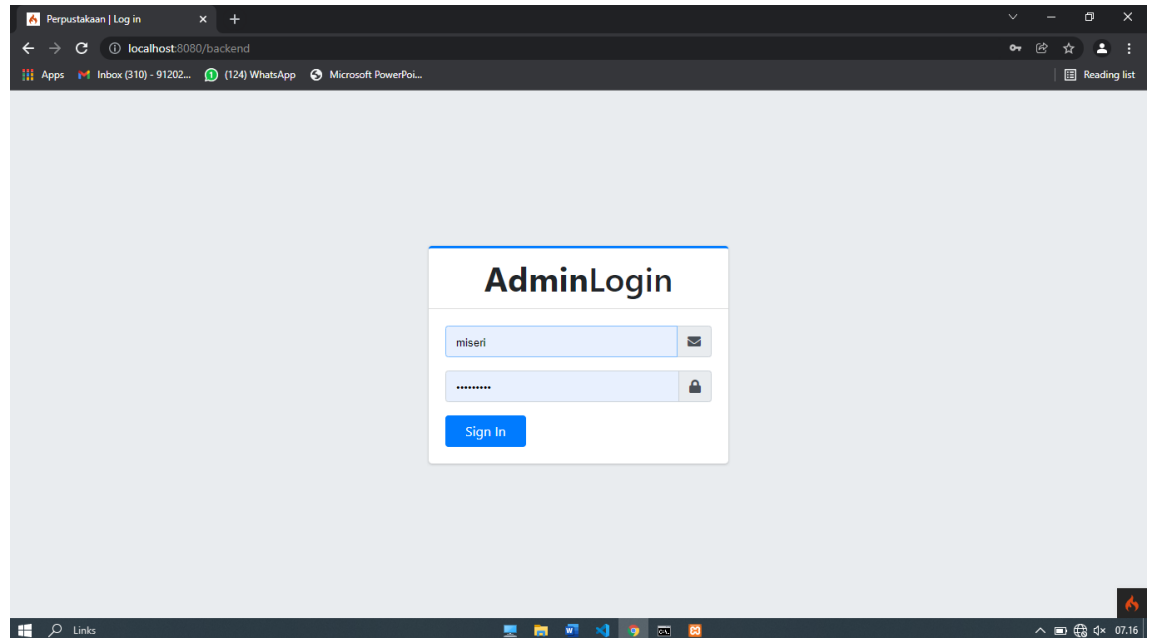
Sri Ayu (C2055201049)

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

A. Tampilan Aplikasi perpustakaan

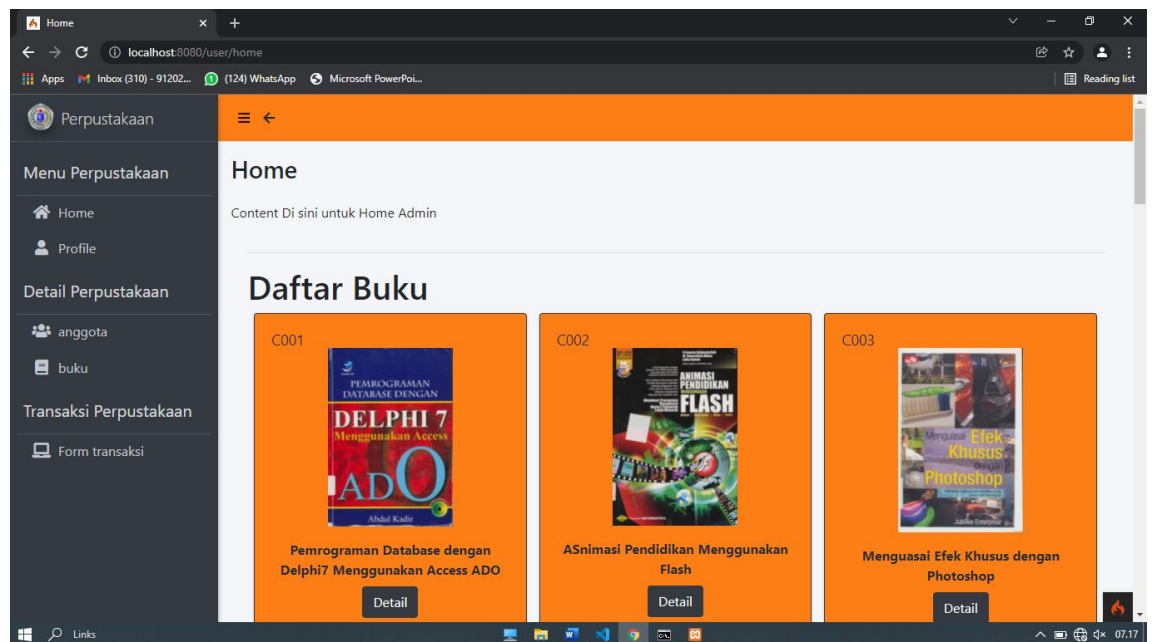
1. Login

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



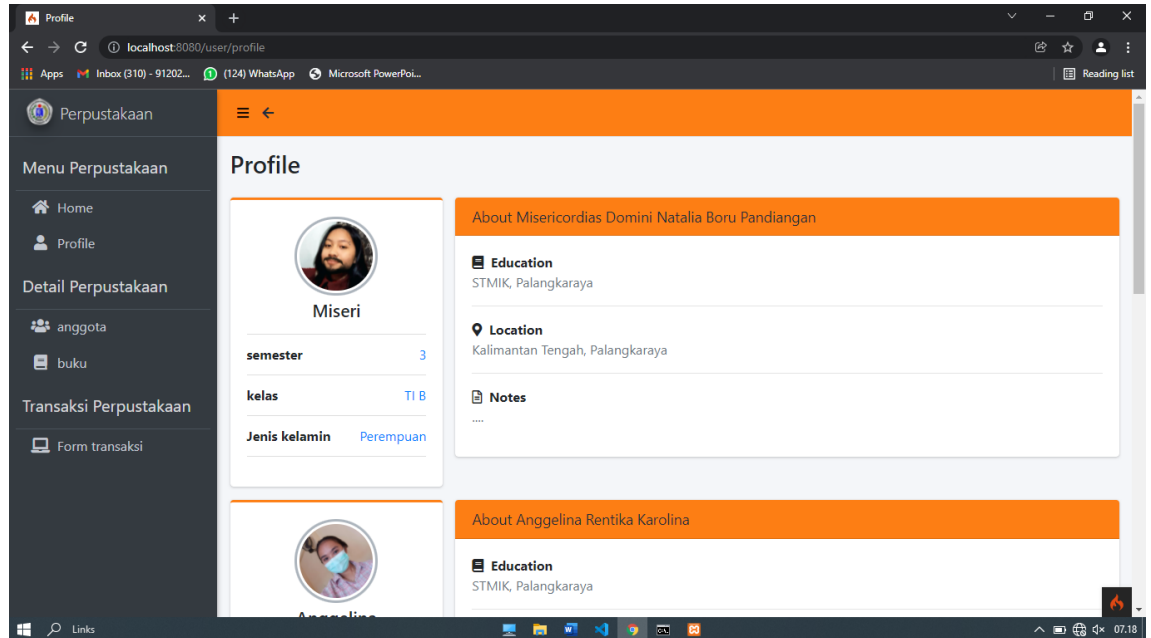
2. Home

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



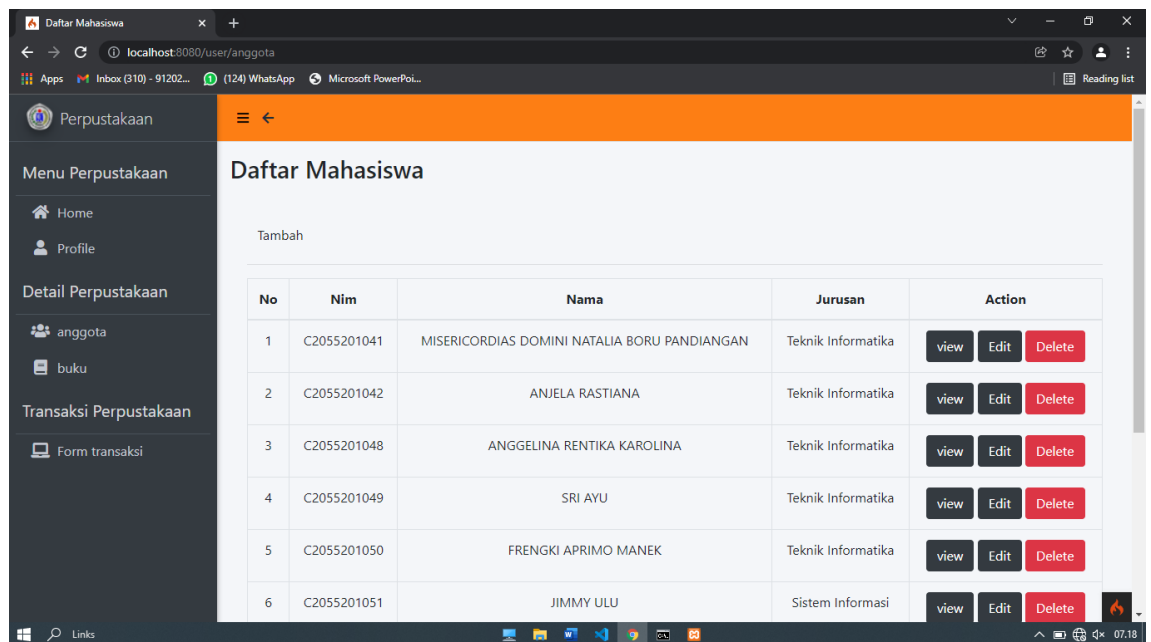
3. Profile

Fitur profile menampilkan data dari anggota kelompok yang mengedit aplikasi perpustakaan



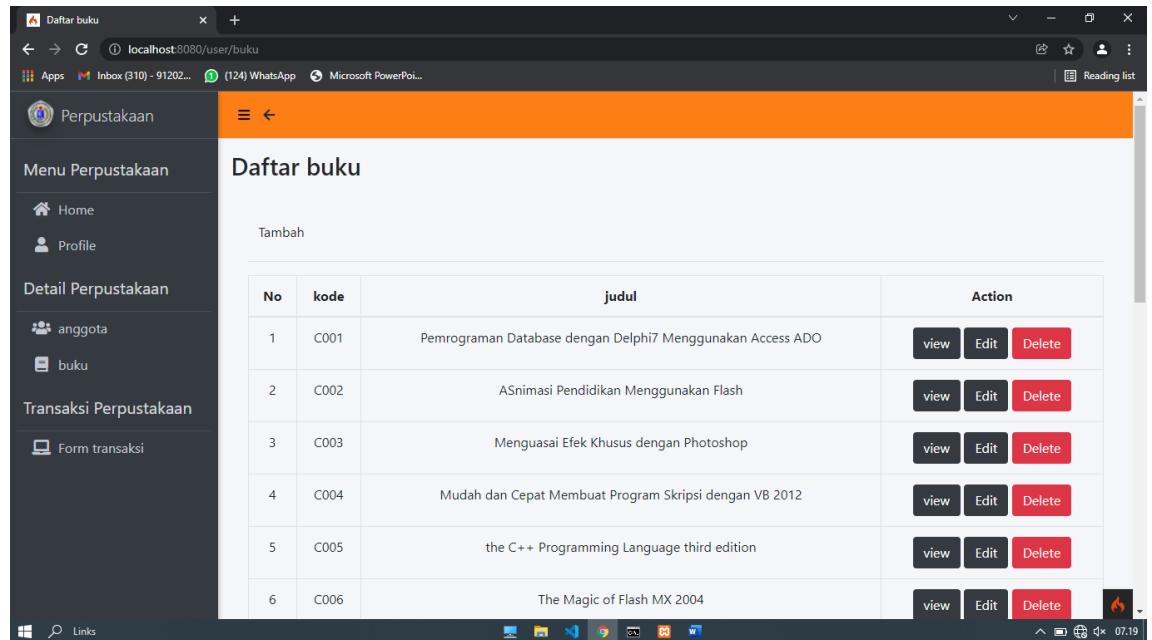
4. Anggota

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



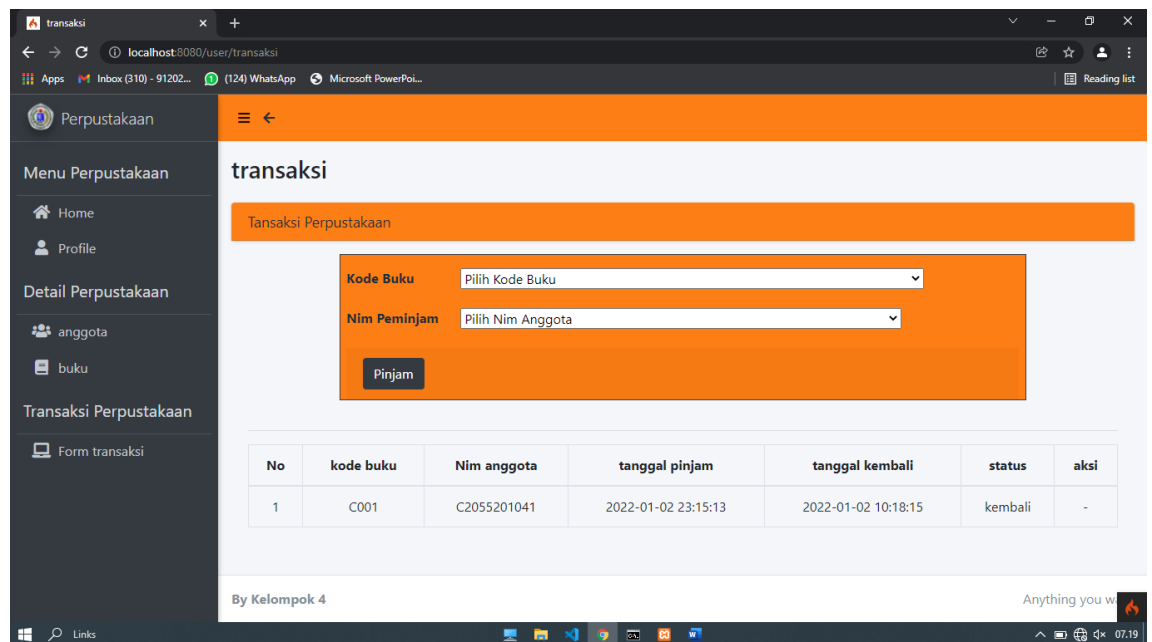
5. Buku

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



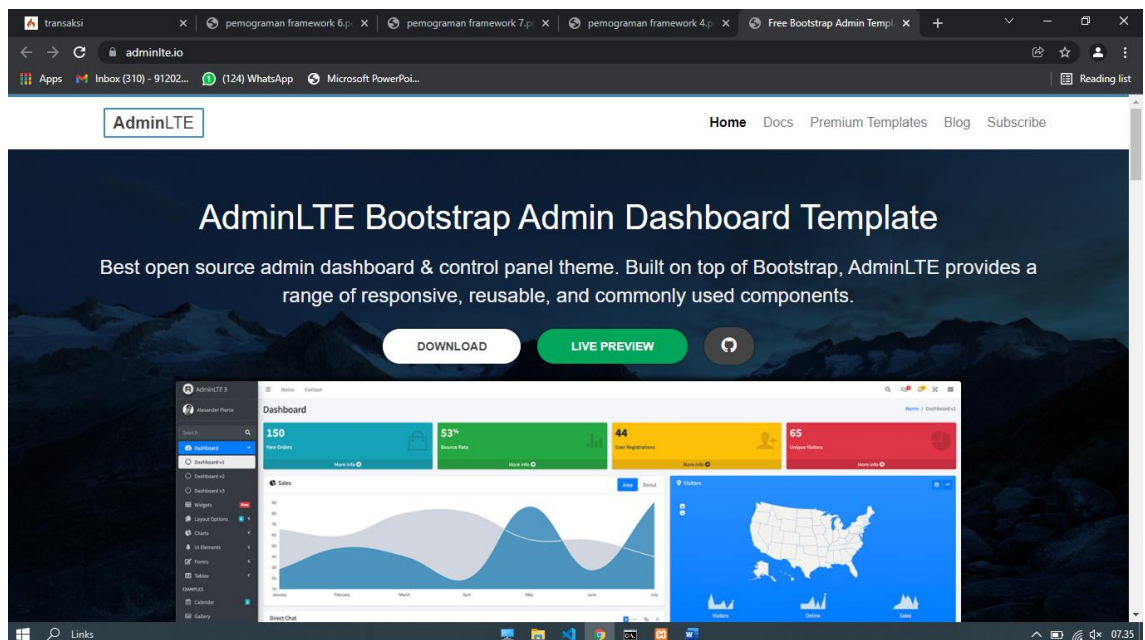
6. Transaksi

Fitur transaksi dilengkapi dengan peminjaman dan pengembalian buku, dimana kode buku di ambil dari tabel buku sedangkan nim peminjam di ambil dari tabel mahasiswa.

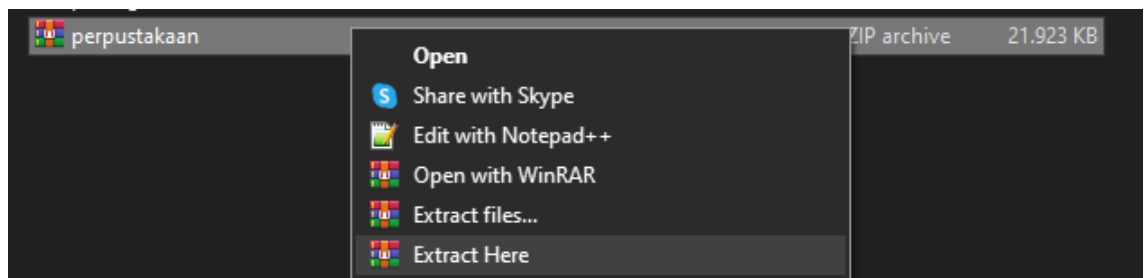


B. Integrasi Template AdminLTE

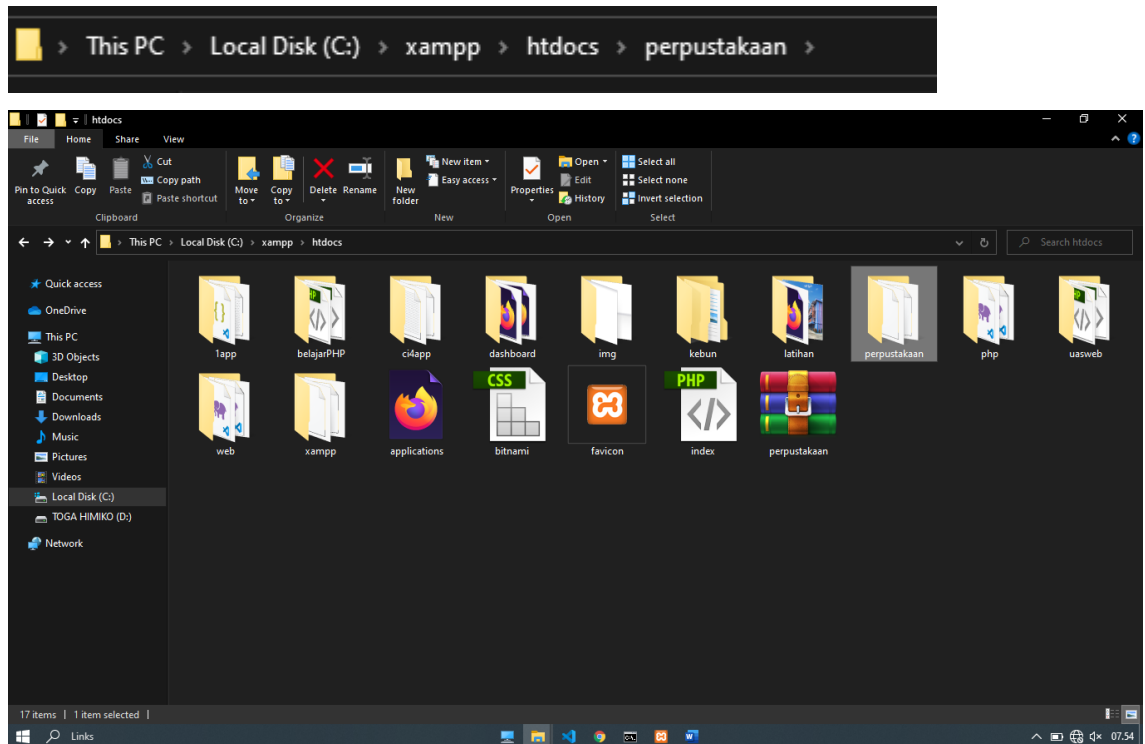
AdminLTE adalah template WebApp sumber terbuka populer untuk dasbord admin dan panel kontrol. Ini adalah template HTML responsif yang didasarkan pada kerangka CSS Bootstrap 4.6.0 pada versi terbarunya yaitu AdminLTE v3.1.0. Pada versi Ini menggunakan semua komponen Bootstrap dalam desainnya dan menata 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



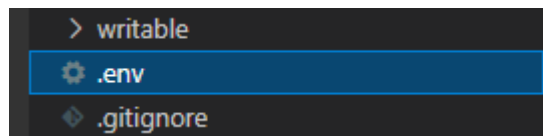
Hasil ekstrakan file .zip perpustakaan bisa di move ke local disk C -> xampp -> htdocs



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

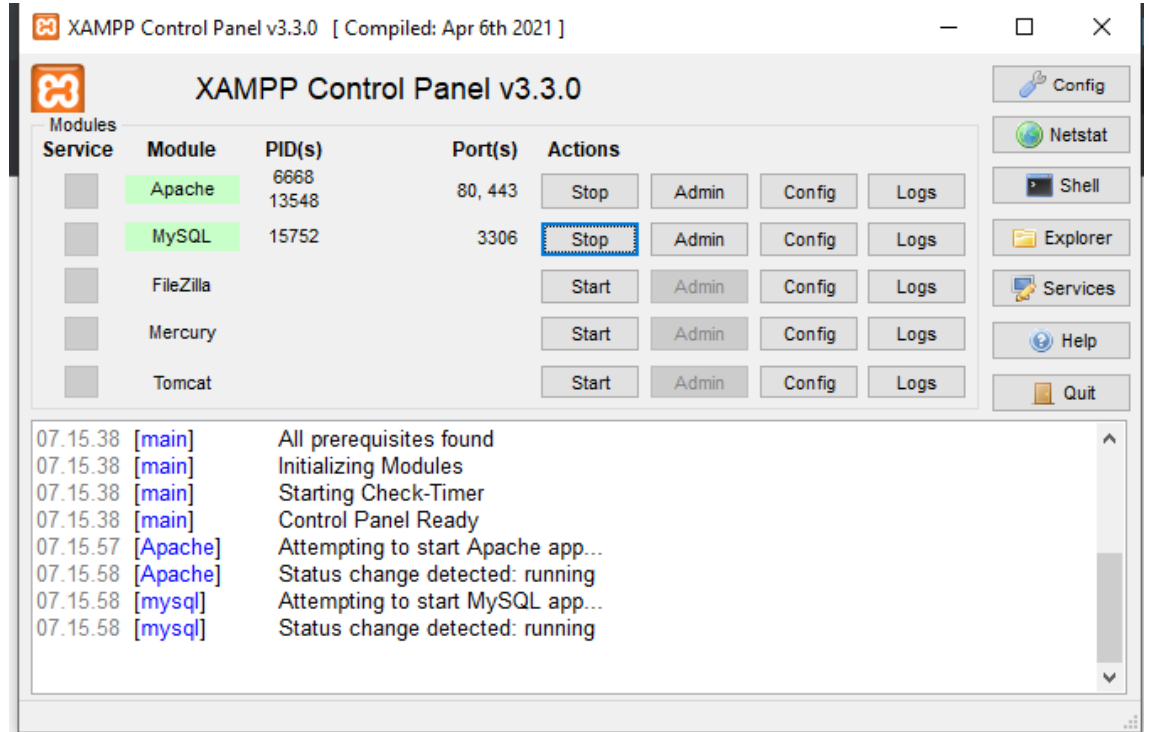
database.default.username = root

database.default.password =

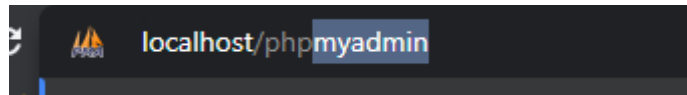
database.default.DBDriver = MySQLi

- 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

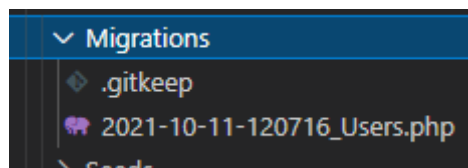
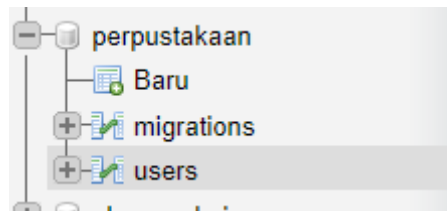
Create

- 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 }
```

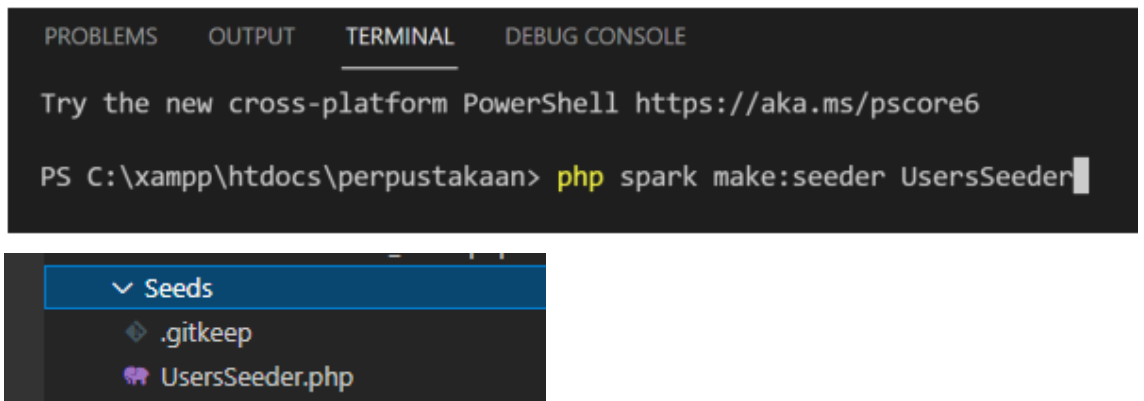
- Untuk membuat tabel users di database perpustakaan perlu menjalankan file _Users migration pada terminal root di rektori aplikasi projek

```
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



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

```

home.php  UsersSeeder.php X
app > Database > Seeds > UsersSeeder.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' => 'miseri',
13                 'password' => password_hash('offjumpol', PASSWORD_BCRYPT),
14                 'name' => 'miseri',
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

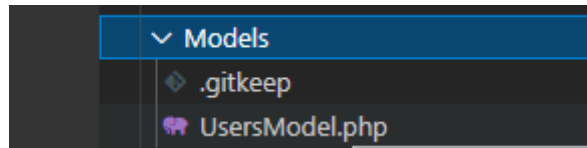
CodeIgniter v4.1.4 Command Line Tool - Server Time: 2021-10-14 10:07:01 UTC-05:00

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

```

+ Options					username	password	name	created_at	updated_at
<input type="checkbox"/>	Edit	Copy	Delete		miseri	\$2y\$10\$rWqmESrib5PcASJA3v36JObIWtzJ22a2i9QWxuQcDgd...	miseri	2022-01-02 09:57:39	NULL
<input type="checkbox"/> Check all With selected: Edit Copy Delete Export									

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

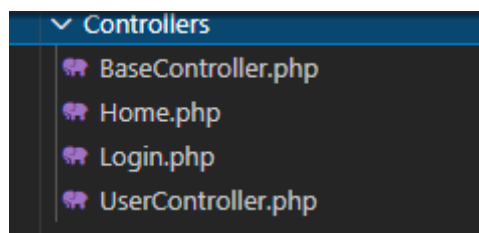


Model.php ini digunakan untuk merepresentasikan data yang terdapat pada tabel users di database perpustakaan

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



Controller 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.pgp terdapat 3 method yaitu

a. Idex()

Untuk mengembalikan halaman login yang ada pada foledr
view\dashboard

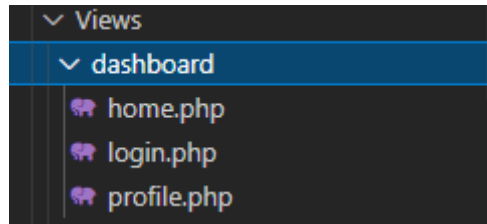
b. Process()

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
10     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">

```

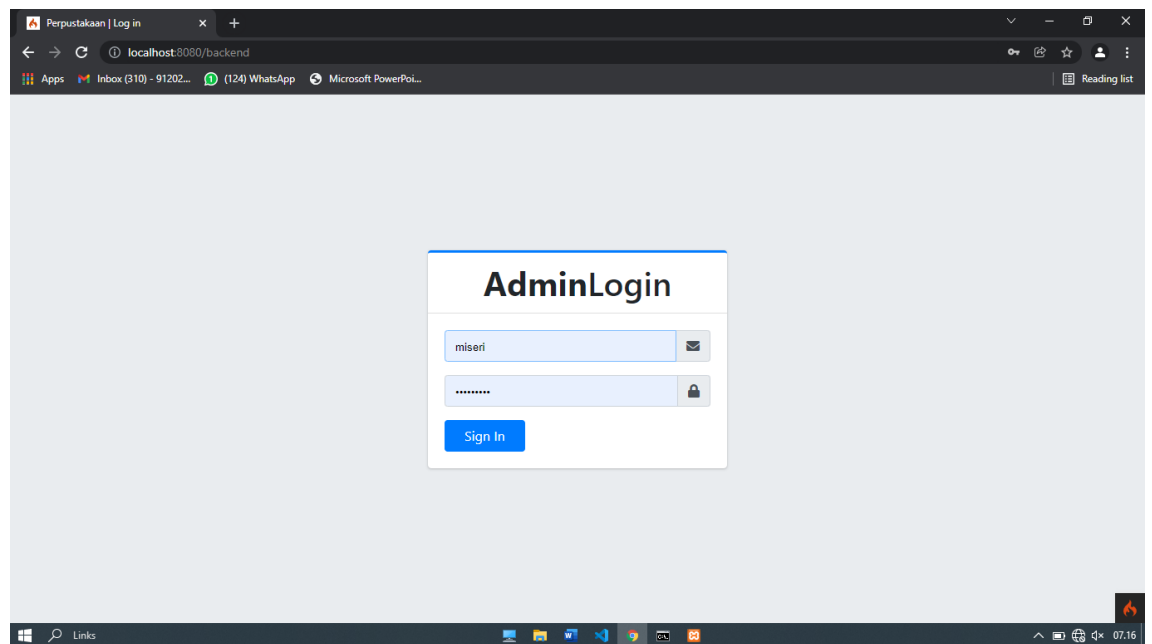
- menambahkan routing pada file routes.php

```

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

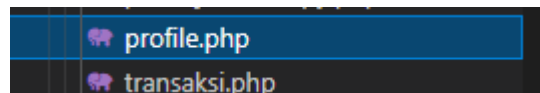
```

- Tampilan Login :



2. Fitur Profile

- Buatlah file view profile.php



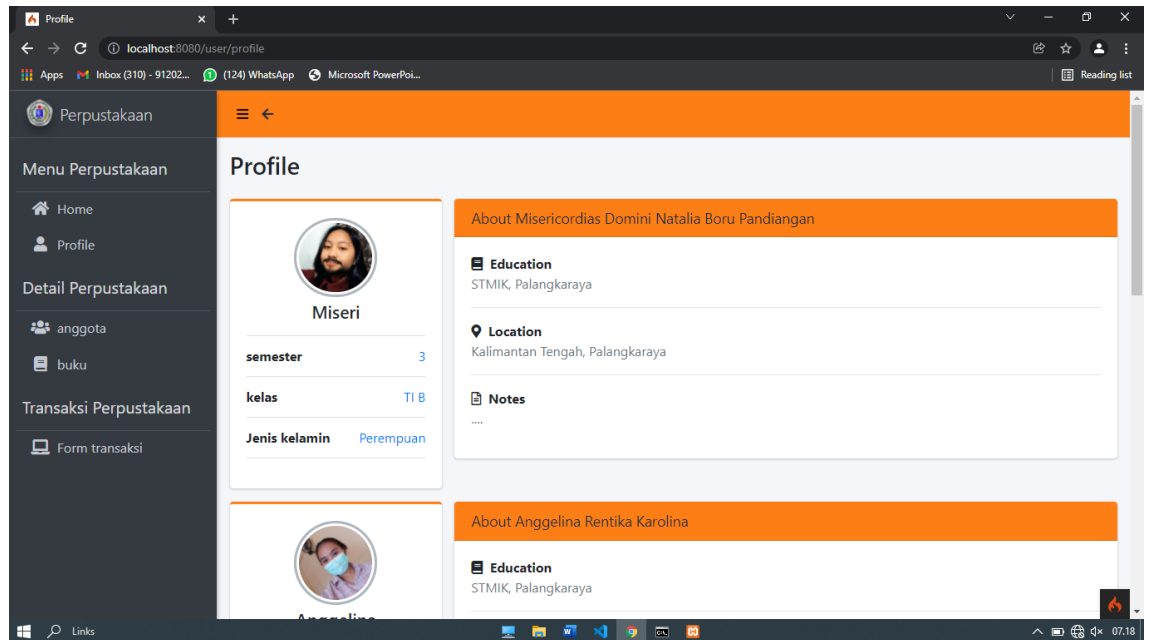
```
home.php  Mahasiswa.php  profile.php X
app > Views > dashboard > profile.php
1  <?= $this->extend('layout/dashboard-layout'); ?>
2  <?= $this->section('content'); ?>
3
4
5  <div class="row">
6      <div class="col-md-3">
7
8
9          <!-- Profile Image -->
10         <div class="card card-orange 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     </div>
```

```
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

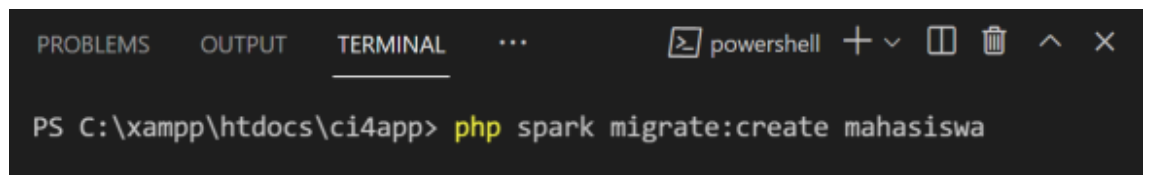
```
$routes->get('profile', 'UserController::profile', ['as' => 'user.profile']);
```

- Tampilan Profile

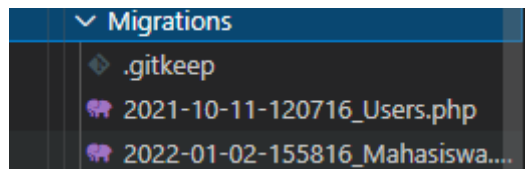


3. Fitur Anggota

- 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 .

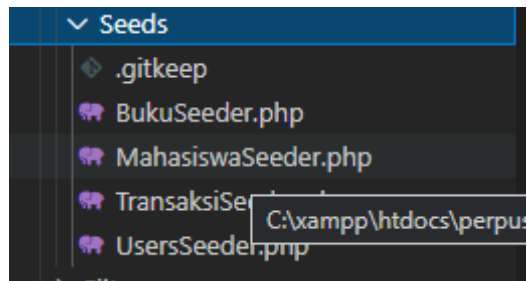
Menjalankan file migration untuk membuat tabel di database

```
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 melalui terminal
php spark make:seeder MahasiswaSeeder



Lalu edit file mahasiswaSeeder.php tersebut dengan kebutuhan masing-masing

```
home.php  MahasiswaSeeder.php X
app > Database > Seeds > MahasiswaSeeder.php
1  <?php
2  namespace App\Database\Seeds;
3  use CodeIgniter\Database\Seeder;
4  use CodeIgniter\I18n\Time;
5  class MahasiswaSeeder extends Seeder
6  {
7      public function run()
8      {
9          $data = [
10             [
11                 'nim' => 'C2055201048',
12                 'nama' => 'ANGGELINA RENTIKA KAROLINA',
13                 'jurusan' => 'Teknik Informatika',
14                 'jenis_kelamin' => 'wanita',
15                 'no_telp' => '085750841933',
16                 'email' => 'anggelinarentikakarolina@gmail.com',
17                 'alamat' => 'Jalan Pramuka 1 Kos Milenial NO.10 Palangkaraya',
18                 'created_at' => Time::now()
19             ],

```

```
home.php  MahasiswaSeeder.php X
app > Database > Seeds > MahasiswaSeeder.php
30  [
31      'nim' => 'C2055201042',
32      'nama' => 'ANJELA RASTIANA',
33      'jurusan' => 'Teknik Informatika',
34      'jenis_kelamin' => 'wanita',
35      'no_telp' => '087838547716',
36      'email' => 'anjelarastiana@gmail.com',
37      'alamat' => 'Jalan Tjilik Riwut Palangkaraya',
38      'created_at' => Time::now()
39  ],
40  [
41      'nim' => 'C2055201049',
42      'nama' => 'SRI AYU',
43      'jurusan' => 'Teknik Informatika',
44      'jenis_kelamin' => 'wanita',
45      'no_telp' => '081350201762',
46      'email' => 'sriayu@gmail.com',
47      'alamat' => 'Jalan Jati Palangkaraya',
48      'created_at' => Time::now()
49  ],
50  [
51      'nim' => 'C2055201050',
52      'nama' => 'FRENGKI APRIMO MANEK',
53      'jurusan' => 'Teknik Informatika',
54      'jenis_kelamin' => 'pria',
55      'no_telp' => '081350200171',
56      'email' => 'frengkiaprimomanek@gmail.com',
57      'alamat' => 'Jalan G.Obos 7 Palangkaraya',
58      'created_at' => Time::now()
59  ],
60  [
61      'nim' => 'C2055201051',
62      'nama' => 'JIMMY ULU',
```

home.php

MahasiswaSeeder.php X

app > Database > Seeds > MahasiswaSeeder.php

```
59     ],
60     [
61         'nim' => 'C2055201051',
62         'nama' => 'JIMMY ULU',
63         'jurusan' => 'Sistem Informasi',
64         'jenis_kelamin' => 'pria',
65         'no_telp' => '081339179796',
66         'email' => 'jimmyulu@gmail.com',
67         'alamat' => 'Jalan Rajawali Palangkaraya',
68         'created_at' => Time::now()
69     ],
70     [
71         'nim' => 'C2055201052',
72         'nama' => 'JANUARIUS OLLIN',
73         'jurusan' => 'Sistem Informasi',
74         'jenis_kelamin' => 'pria',
75         'no_telp' => '081288121169',
76         'email' => 'januariusollin@gmail.com',
77         'alamat' => 'Jalan Garuda Palangkaraya',
78         'created_at' => Time::now()
79     ],
80     [
81         'nim' => 'C2055201060',
82         'nama' => 'SANTOS',
83         'jurusan' => 'Sistem Informasi',
84         'jenis_kelamin' => 'pria',
85         'no_telp' => '081255674682',
86         'email' => 'santos@gmail.com',
87         'alamat' => 'Jalan Temanggung Tilung Palangkaraya',
88         'created_at' => Time::now()
89     ],
90     [
91         'nim' => 'C2055201065',
```

```
home.php  MahasiswaSeeder.php X
app > Database > Seeds > MahasiswaSeeder.php
83         'jurusan' => 'Sistem Informasi',
84         'jenis_kelamin' => 'pria',
85         'no_telp' => '081255674682',
86         'email' => 'santos@gmail.com',
87         'alamat' => 'Jalan Temanggung Tilung Palangkaraya',
88         'created_at' => Time::now()
89     ],
90     [
91         'nim' => 'C2055201065',
92         'nama' => 'YANTO MANTANO',
93         'jurusan' => 'Sistem Informasi',
94         'jenis_kelamin' => 'pria',
95         'no_telp' => '085705150121',
96         'email' => 'yantomantano@gmail.com',
97         'alamat' => 'Jalan Menteng Palangkaraya',
98         'created_at' => Time::now()
99     ],
100    [
101        'nim' => 'C2157201021',
102        'nama' => 'RISMA UTAMI HANDAYANI',
103        'jurusan' => 'Sistem Informasi',
104        'jenis_kelamin' => 'wanita',
105        'no_telp' => '081351680382',
106        'email' => 'rismautamihandayani@gmail.com',
107        'alamat' => 'Jalan B.Koetin Kos Putri Berlian NO.4 Palangkaraya',
108        'created_at' => Time::now()
109    ]
110 ];
111 $this->db->table('mahasiswa')->insertBatch($data);
112 }
113 }
114
```

Keterangan :

Perintah diatas, insert 10 data mahasiswa.

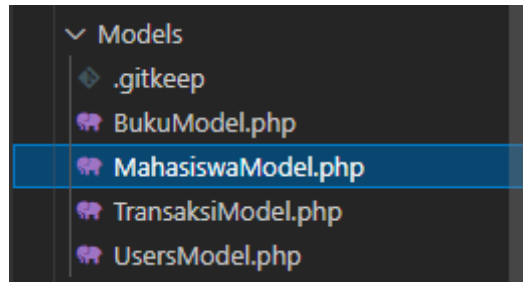
Berikut jalankan perintah diatas untuk insert data mahasiswa ke dalam tabell mahasiswa, denngan menuliskan perintah berikut ke dalam terminal Vscode.

```
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 :

```

home.php  MahasiswaModel.php X
app > Models > MahasiswaModel.php
1  <?php
2  namespace App\Models;
3  use CodeIgniter\Model;
4  class MahasiswaModel extends Model
5  {
6      protected $table = "mahasiswa";
7      protected $primaryKey = "nim";
8      protected $returnType = "object";
9      protected $useTimestamps = true;
10     protected $allowedFields = ['nim', 'nama', 'jurusan', 'jenis_kelamin', 'no_telp', 'email', 'alamat', 'foto'];
11
12     function getAll()
13     {
14         $builder = $this->db->table('mahasiswa');
15         $builder->join('transaksi', 'transaksi.nim_anggota = mahasiswa.nim', 'left');
16         $query = $builder->get();
17         return $query->getResult();
18     }
19
20 }

```

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

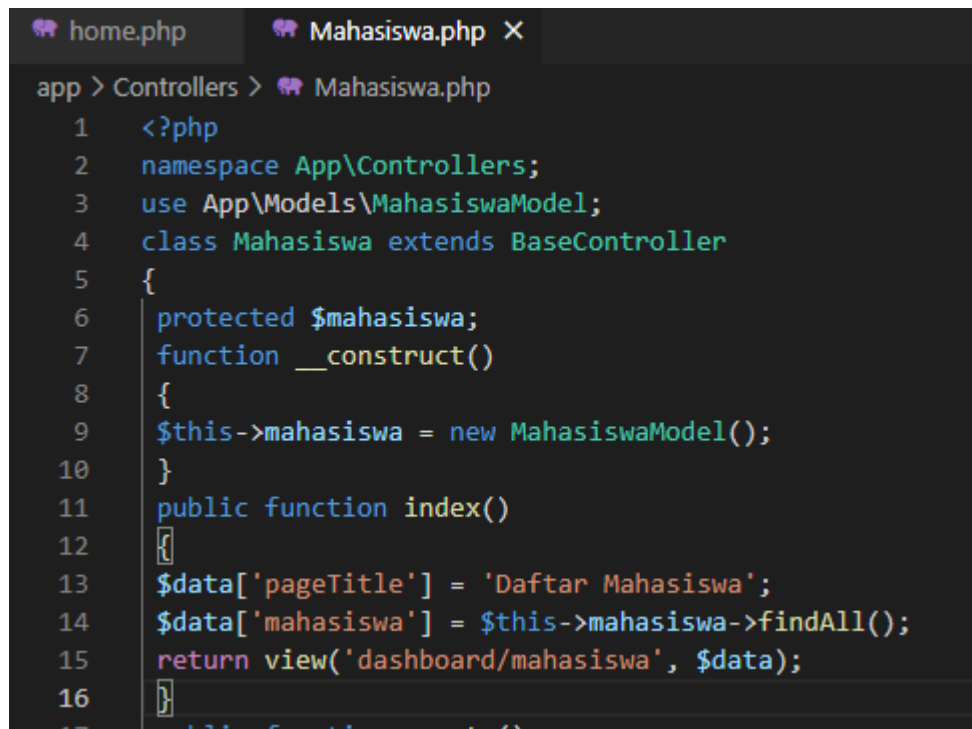
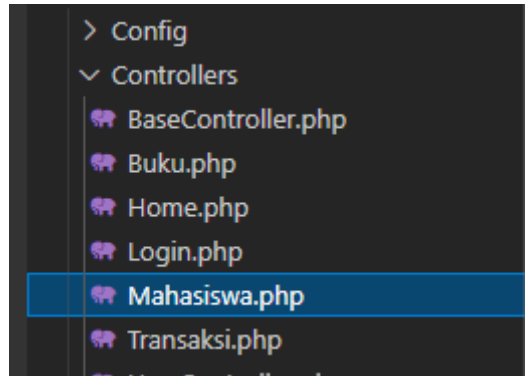
\$primaryKey dengan nilai nim karena kolom nim merupakan primary key dari tabel mahasiswa

\$returnType dengan nilai object karena kita ingin untuk return type 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

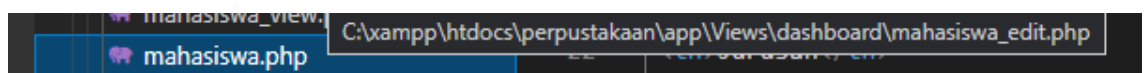
- Functions getAll() berfungsi sebagai data berelasi dengan tabel transaksi
- Membuat controller mahasiswa



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>

```

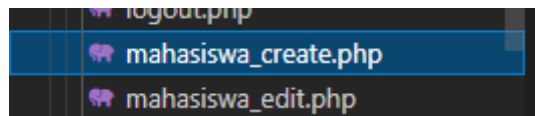
```

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

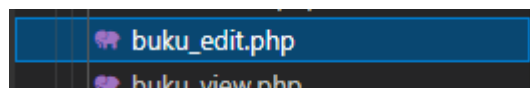
Buatlah method edit di controller Mahasiswa

```
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        $dataMahasiswa->update($this->request->getPost());
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->validator->errors());
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  <?= $this->section('content'); ?>
3  <div class="col-md-10">
4      <!-- general form elements -->
5      <div class="card card-orange">
6          <div class="card-header">
7              <h3 class="card-title">Edit Data Buku</h3>
8          </div>
9          <!-- /.card-header -->
10         <!-- form start -->
11         <?php if (!empty(session()->getFlashdata('error'))): ?>
12             <div class="alert alert-light alert-dismissible fade show" role="alert">
13                 <h4 class="alert-heading">Periksa Entrian Form</h4>
14                 <hr />
15                 <?php echo session()->getFlashdata('error'); ?>
16                 <button type="button" class="close" data-dismiss="alert" arialabel="Close">
17                     <span aria-hidden="true">&times;</span>
18                 </button>
19             </div>
20         <?php endif; ?>
21         <form class="form-horizontal" method="post" action="<?=base_url('buku/update/'.$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);

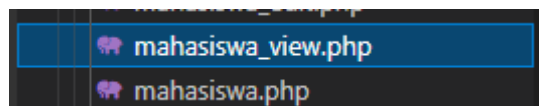
session()->setFlashdata('message', 'Delete Data Mahasiswa
Berhasil');
return redirect()->to('/mahasiswa');
}
```

- 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);
}
}
```

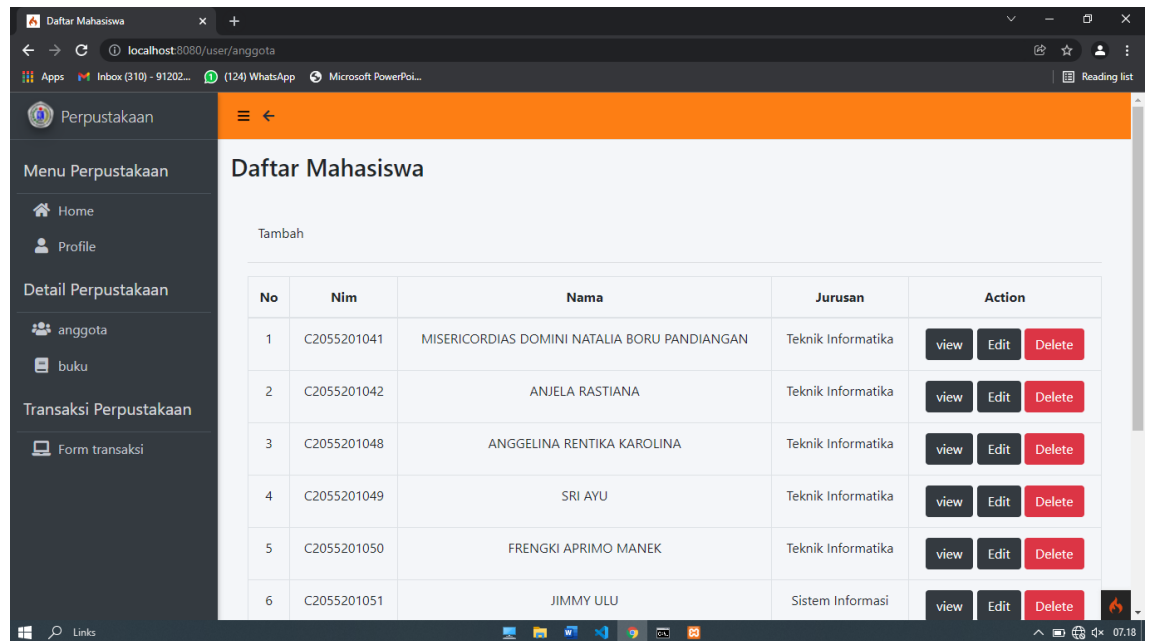
Buat file mahasiswa_view.php di folder app/views



```
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 Telpn / 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'); ?>
```

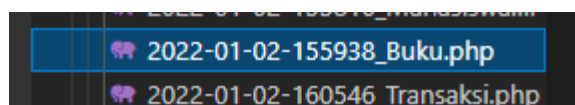
- Tampilan Anggota :



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 .

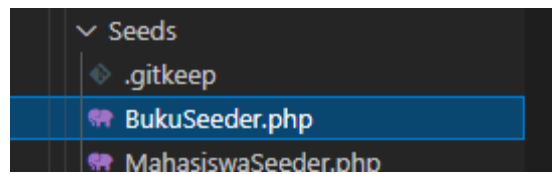
Menjalankan file migration untuk membuat tabel di database

```
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
```

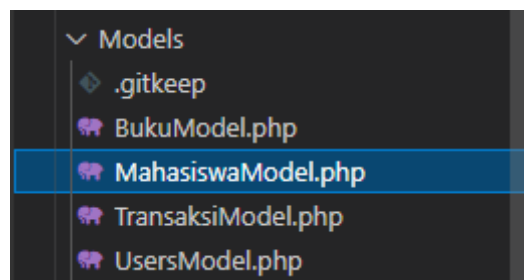
- Membuat BukuSeeder melalun terminal
php spark make:seeder BukuSeeder



```
app > Database > Seeds > BukuSeeder.php
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' => 'C001',
12             'judul' => 'Pemrograman Database dengan Delphi7 Menggunakan Access ADO',
13             'slug' => 'pemrograman-database-dengan-delphi7-menggunakan-access-ado',
14             'penulis' => 'Abdul Kadir',
15             'penerbit' => 'Penerbit Andi',
16             'created_at' => Time::now()
17         ],
18     ]
```

```
$this->db->table('buku')->insertBatch($data);
}
}
```

- Membuat file mahasiswamodel.php did alam folder app/models



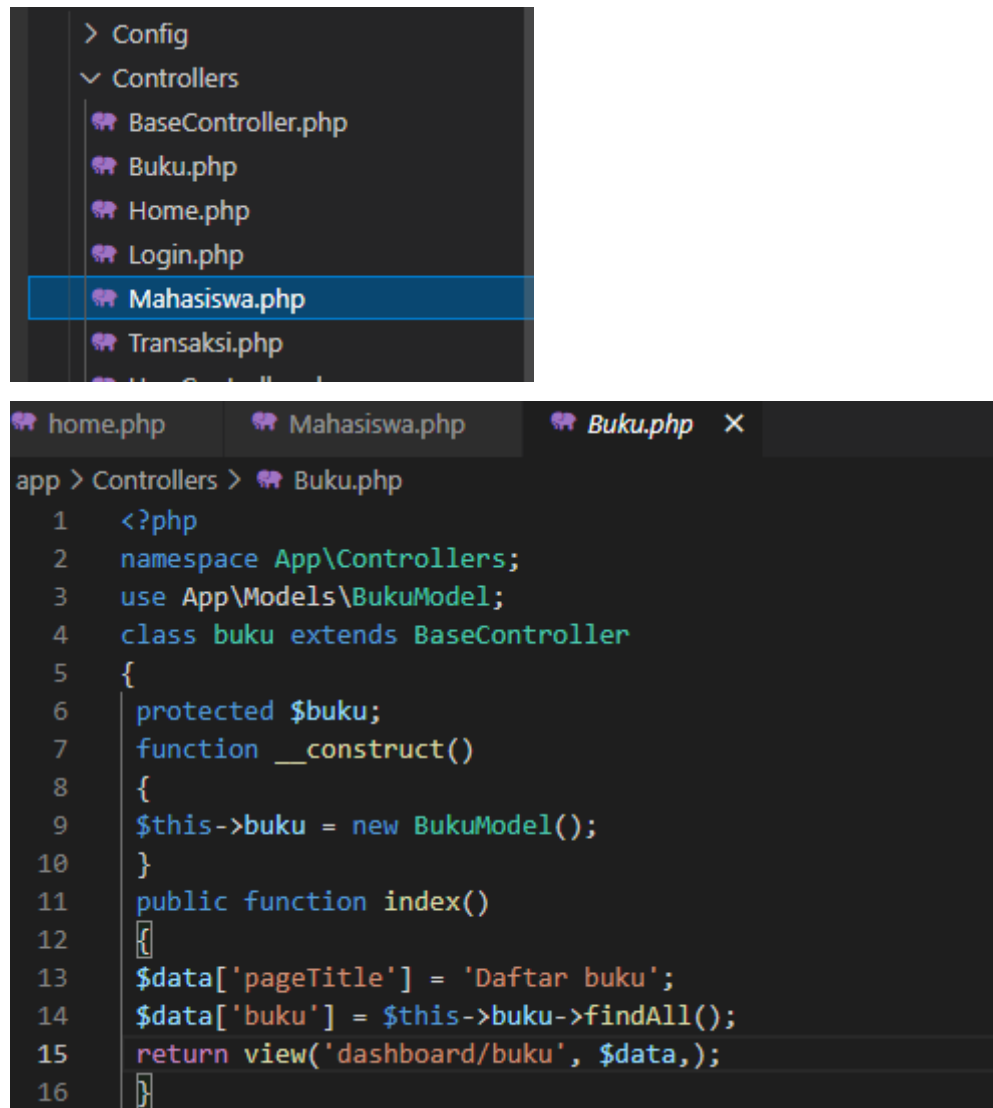
Program dari file bukumodel.php :

```
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



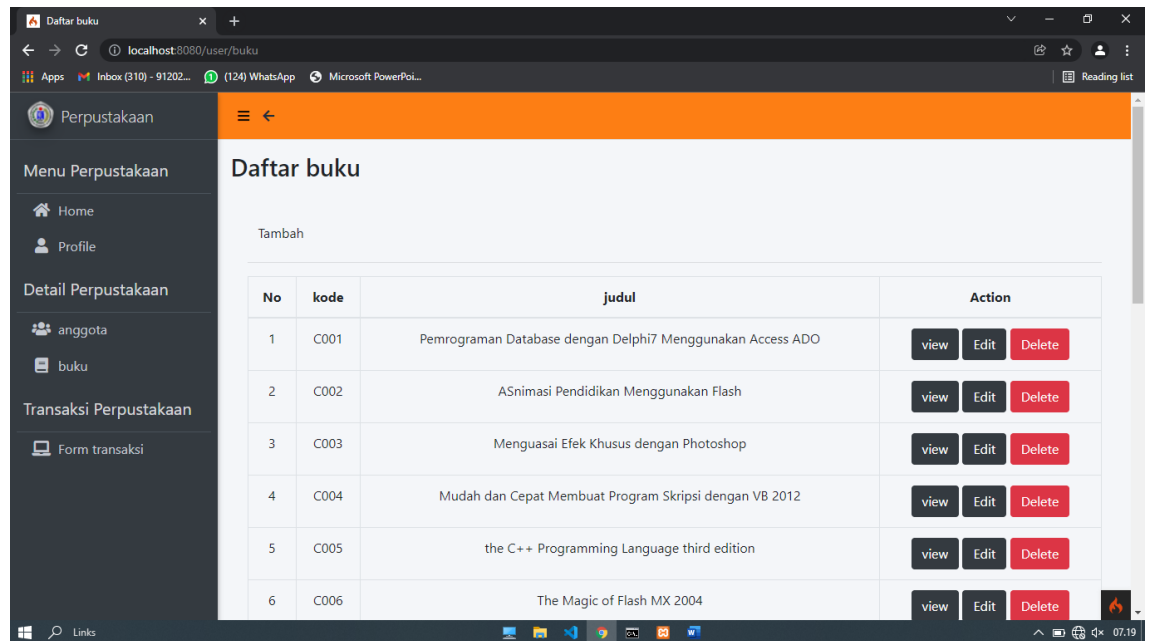
Di file routes.php tambahkan :

```
$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.

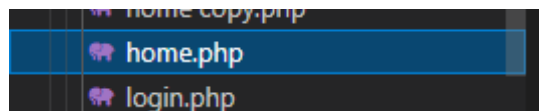
- Tampilan Buku



5. Fitur Home

Pada fitur Home, data yang ditampilkan adalah data dari tabel buku, jadi ada hal yang harus dilakukan untuk menampilkan data buku menggunakan controller home.

- Membuat file home.php di folder views/dashboard



```
home.php X Mahasiswa.php Buku.php
app > Views > dashboard > home.php
1 <?= $this->extend('layout/dashboard-layout'); ?>
2 <?= $this->section('content'); ?>
3 Content Di sini untuk Home Admin
4
5 <!-- /.card-header -->
6 <div class="card-body">
7     <?php if (empty(session()->getFlashdata('message'))): ?>
8     <div class="alert alert-success alert-dismissible fade show" role="alert">
9     <?php echo session()->getFlashdata('message'); ?>
10    <button type="button" class="close" data-dismiss="alert" arialabel="Close">
11    <span aria-hidden="true">&times;</span>
12    </button>
13    </div>
14    <?php endif; ?>
15    <hr />
16    <h1>Daftar Buku</h1>
17    <div class="content">
18    <div class="container">
19    <div class="row">
20    <?php
21    $no = 1;
22    foreach ($buku as $row) {
23    ?>
24    <div class="col-lg-4">
25        <div class="card bg-orange border border-dark ">
26            <div class="card-body text-center ">
27
28                <h5 class="card-title"><?= $row->kode; ?></h5>
29                <p class="card-text">
30                    
31                    </p>
32
33                    <label>
```

```
home.php X Mahasiswa.php Buku.php
app > Views > dashboard > home.php
30 
31 </p>
32
33 <label>
34 <label><?= $row->judul; ?></label>
35
36 <p>
37 <td><a title="View" href="<?=base_url("buku/view/$row->kode"); ?>"
38 class="btn btn-dark">Detail</a> </td></p>
39 </div>
40 </div>
41 </div>
42 <?php
43 }
44 ?>
45 </div>
46 </div>
47 </div>
48
49 </div>
50 <!-- /.card-body -->
51 <?= $this->endSection('content'); ?>
```

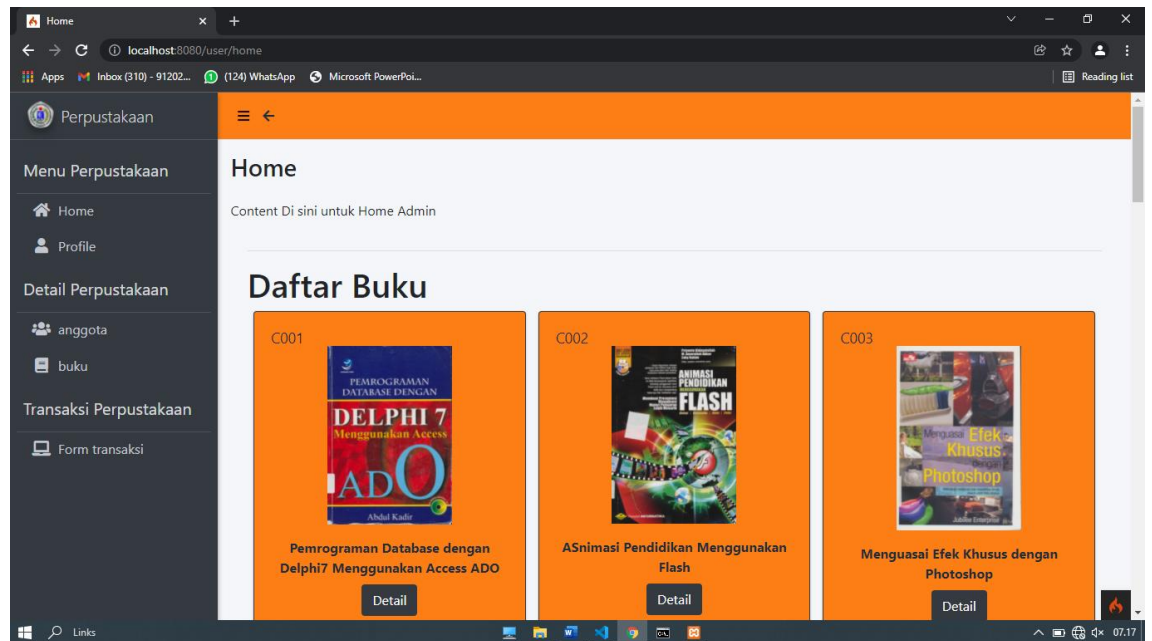
- Membuat controller home

Karena home mengambil data dari tabel buku, jadi bukumodel merupakan patokan controller home

```
home.php  Home.php x  Mahasiswa.php

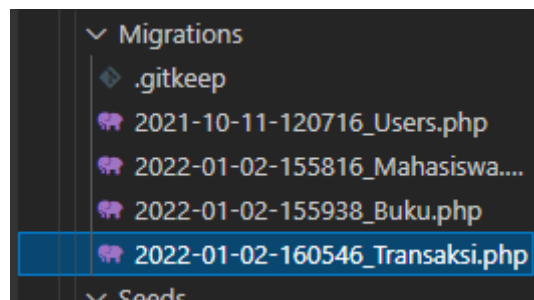
app > Controllers > Home.php
1  <?php
2  namespace App\Controllers;
3  use App\Models\BukuModel;
4  class Home extends BaseController
5  {
6
7      protected $buku;
8      function __construct()
9      {
10         $this->buku = new BukuModel();
11     }
12     public function index()
13     {
14         $data['pageTitle'] = 'Home';
15         $data['buku'] = $this->buku->findAll();
16         return view('dashboard/home', $data);
17     }
18 }
19
```


- Tampilan Home



6. Fitur Transaksi

- Membuat migration transaksi dengan nama transaksi



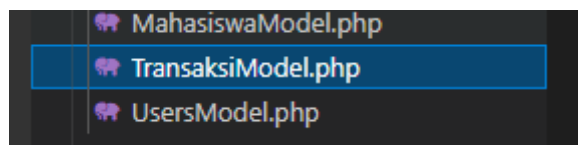
home.php 2022-01-02-160546_Transaksi.php X Mahasiswa.php

app > Database > Migrations > 2022-01-02-160546_Transaksi.php

```
1  <?php
2
3  namespace App\Database\Migrations;
4
5  use CodeIgniter\Database\Migration;
6
7  class Transaksi extends Migration
8  {
9      public function up()
10     {
11         $this->forge->addField([
12             'id' => [
13                 'type' => 'int',
14                 'constraint' => '10' ,
15                 'unsigned' => true,
16                 'auto_increment' => true,
17             ],
18             'kode_buku' => [
19                 'type' => 'varchar',
20                 'constraint' => '255',
21             ],
22             'nim_anggota' => [
23                 'type' => 'varchar',
24                 'constraint' => '11',
25             ],
26             'tanggal_pinjam' => [
27                 'type' => 'datetime',
28                 'default' => 'CURRENT_TIMESTAMP()',
29             ],
30             'tanggal_kembali' => [
31                 'type' => 'datetime',
32             ],
33             'status' => [
```

```
home.php 2022-01-02-160546_Transaksi.php X Mahasiswa.php
app > Database > Migrations > 2022-01-02-160546_Transaksi.php
38     'denda' => [
39         'type' => 'float',
40         'null' => true,
41     ],
42     'created_at' => [
43         'type' => 'DATETIME',
44         'null' => true,
45     ],
46     'updated_at' => [
47         'type' => 'DATETIME',
48         'null' => true,
49     ],
50 ];
51
52 $this->forge->addPrimaryKey('id');
53 $this->forge->addForeignKey('kode_buku', 'buku', 'kode');
54 $this->forge->addForeignKey('nim_anggota', 'mahasiswa', 'nim');
55 $this->forge->createTable('transaksi');
56 }
57 //-----
58 public function down()
59 {
60     $this->forge->addForeignKey('transaksi', 'transaksi_kode_buku_foreign');
61     $this->forge->addForeignKey('transaksi', 'transaksi_nim_anggota_foreign');
62     $this->forge->dropTable('transaksi');
63 }
64 }
```

- Membuat model transaksi dengan nama transaksimodel



```
home.php TransaksiModel.php Mahasiswa.php
app > Models > TransaksiModel.php
1  <?php
2  namespace App\Models;
3  use CodeIgniter\Model;
4  class TransaksiModel extends Model
5  {
6      protected $table = "transaksi";
7      protected $primaryKey = "id";
8      protected $returnType = "object";
9      protected $useTimestamps = true;
10     protected $allowedFields = ['id', 'kode_buku', 'nim_anggota',
11     'tanggal_pinjam', 'tanggal_kembali', 'status', 'denda'];
12 }
```

- Membuat view transaksi

```
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 <!-- from tambah transaksi -->
15 <?=$this->extend('layout/dashboard-layout'); ?>
16 <?=$this->section('content'); ?>
17
18 <div class="card card-orange">
19 <div class="card-header">
20 <h3 class="card-title">Tansaksi Perpustakaan</h3>
21 </div>
22 </div>
23 <div class="row">
24 <div class="col-md-9 ms-md-auto position-relative m-auto border border-dark  bg-orange ">
25 <p> </p>
26
27 <div class="form-group row">
28 <label for="kode" class="col-sm-2 col-formlabel">Kode Buku</label>
29 <form class="form-horizontal" method="post" action="<?= base_url('transaksi/create') ?>">
30 <?csrf_field() ?>
31 <div class="col-sm-5">
32 <select name="kode_buku" class="form-control" REQUIRED="">
```

```
33 <option value=""> Pilih Kode Buku </option>
34 <?php if ($buku): ?>
35 <?php foreach ($buku as $1b): ?>
36 <option value="<?php echo $1b->kode?>"> <?php echo $1b->kode?> - <?php echo $1b->judul?> </option>
37 <?php endforeach; ?>
38 <?php endforeach; ?>
39 </select>
40 </div>
41 </div>
42 </div>
43
44 <div class="form-group row">
45 <label for="nim" class="col-sm-2 col-formlabel">Nim Peminjam</label>
46 <div class="col-sm-5">
47 <select name="nim_anggota" class="form-control" REQUIRED="">
48 <option value=""> Pilih Nim Anggota </option>
49 <?php if ($mahasiswa): ?>
50 <?php foreach ($mahasiswa as $1a): ?>
51 <option value="<?php echo $1a->nim?>"> <?php echo $1a->nim?> - <?php echo $1a->nama?> </option>
52 <?php endforeach; ?>
53 <?php endforeach; ?>
54 </select>
55 </div>
56 </div>
57 </div>
58
59 <div class="card-footer">
60 <button type="submit" class="btn btn-dark " value="create">Pinjam</button>
61 </div>
62 </form>
```

```
64 </div>
65
66 <?= $this->endSection('content'); ?>
67 <!-- end from transaksi -->
68 <hr />
69
70 <table class="table table-bordered text-center">
71 <thead>
72 <tr>
73 <th>No</th>
74 <th>kode buku</th>
75 <th>nim anggota</th>
76 <th>tanggal pinjam</th>
77 <th>tanggal kembali</th>
78 <th>status</th>
79 <th>aksi</th>
80 </tr>
81 </thead>
82 <tbody>
83 <?php
84 $no = 1;
85 foreach ($transaksi as $row) {
86 >
87 <tr>
88 <td><?= $no++; ?></td>
89 <td><?= $row->kode_buku; ?></td>
90 <td><?= $row->nim_anggota; ?></td>
91 <td><?= $row->tanggal_pinjam; ?> </td>
92 <td><?= $row->tanggal_kembali; ?></td>
93 <td><?= $row->status; ?> </td>
94 <td>
95 <?php if($row->status == "Dipinjam") : ?>
96 <a href="<?php echo site_url() ?>/transaksi/kembali/<?php echo $row->id ?>">kembali </a>
```

```
97 <?php else: ?>
98 <a href="<?php echo site_url() ?>/transaksi/kembali/<?php echo $row->id ?>">kembali </a>
99 <?php endif ?>
100 </td>
101 </tr>
102 <?php
103 }
104 >
105 </tbody>
106 </table>
107 </div>
108 <!-- /.card-body -->
109 <?= $this->endSection('content'); ?>
```

- Membuat controler transaksi

```
home.php TransaksiModel.php Transaksi.php X Mahasiswa.php
app > Controllers > Transaksi.php
1  <?php
2  namespace App\Controllers;
3  use App\Models\TransaksiModel;
4  use App\Models\MahasiswaModel;
5  use App\Models\BukuModel;
6
7  class transaksi extends BaseController
8  {
9      function __construct() {
10         $this->transaksi = new TransaksiModel();
11         $this->mahasiswa = new MahasiswaModel();
12         $this->buku      = new BukuModel();
13     }
14
15     public function index()
16     {
17         $data['buku'] = $this->buku->getAll();
18         $data['mahasiswa'] = $this->mahasiswa->getAll();
19         $data['pageTitle'] = 'transaksi';
20         $data['transaksi'] = $this->transaksi->findAll();
21         return view('dashboard/transaksi', $data);
22     }
23
24     function pinjam()
25     {
26
27         $data['pageTitle'] = 'Data Peminjaman';
28         $data['buku'] = $this->buku->getAll();
29         $data['mahasiswa'] = $this->mahasiswa->getAll();
30         $data['transaksi'] = $this->transaksi->findAll();
31         return view('dashboard/peminjaman', $data);
32     }
33 }
```

```

home.php TransaksiModel.php Transaksi.php X Mahasiswa.php
app > Controllers > Transaksi.php

33
34
35     public function create()
36     {
37         $data = $this->request->getpost();
38         $lama_pinjam = 7; //7 hari
39         $sekarang = date('Y-m-d H:i:s');
40         $tanggal_kembali = date('Y-m-d H:i:s', strtotime('+7 days'));
41         $this->transaksi->insert([
42             'kode_buku' => $this->request->getVar('kode_buku'),
43             'nim_anggota' => $this->request->getVar('nim_anggota'),
44             'tanggal_kembali' => $tanggal_kembali,
45             'denda' => '0',
46         ]);
47         session()->setFlashdata('message', 'Pinjam buku Berhasil');
48         return redirect()->to('/transaksi');
49     }
50
51     public function kembali($id)
52     {
53         if ($id) {
54             $this->transaksi->update($id, [
55                 'tanggal_kembali' => date('Y-m-d H:i:s'),
56                 'status' => 'kembali',
57             ]);
58             session()->setFlashdata('message', 'Kembali Buku Berhasil');
59             return redirect()->to('/transaksi');
60         }
61     }
62 }
63
64 }

```

- Tampilan Transaksi

The screenshot displays a web application interface for a library system. The sidebar menu includes 'Perpustakaan', 'Menu Perpustakaan' (Home, Profile), 'Detail Perpustakaan' (anggota, buku), and 'Transaksi Perpustakaan' (Form transaksi). The main content area is titled 'transaksi' and contains a 'Transaksi Perpustakaan' section with a form for borrowing a book. The form has two dropdown menus: 'Kode Buku' (Pilih Kode Buku) and 'Nim Peminjam' (Pilih Nim Anggota), and a 'Pinjam' button. Below the form is a table showing transaction details.

No	kode buku	Nim anggota	tanggal pinjam	tanggal kembali	status	aksi
1	C001	C2055201041	2022-01-02 23:15:13	2022-01-02 10:18:15	kembali	-

By Kelompok 4