

**LAPORAN  
PEMROGRAMAN MEMBUAT WEBSITE  
MENNGUNAKAAN LARAVEL\CI  
“PROGRAM WEBSITE PERPUSTAKAAN”**



**NAMA KELOMPOK:**

- 1. ANCEA SYAFIRYA(12221892)**
- 2. AYU MARYANI(12221893)**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER  
EL RAHMA  
YOGYAKARTA  
TAHUN 2023/2024**

## 1. Membuat Project baru pada laravel

### a. Membuat project baru dengan nama autodidak

```
Command Prompt - c:\xampp
Microsoft Windows [Version 10.0.22631.3737]
(c) Microsoft Corporation. All rights reserved.

C:\Users\acer>cd..

C:\Users>cd..

C:\>cd autodidak

C:\autodidak>c:\xampp\php\php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
'c:\xampp\php\php' is not recognized as an internal or external command,
operable program or batch file.
```

Hasil Project yang telah dibuat berada didalam laravel sudah berada dihtdocs

Name	Date modified	Type	Size
autodidak	29/06/2024 18:45	File folder	

## 2. Membuat Tampilan Login dan Register

### a. Menginstall composer UI pada laravel untuk tampilan Login dan Register

```
Command Prompt - c:\xampp
'c:\xampp\php\php' is not recognized as an internal or external command,
operable program or batch file.

C:\autodidak>c:\xampp\php\php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"

C:\autodidak>c:\xampp\php\php -r "if (hash_file('sha384', 'composer-setup.php') === 'dac665fdc30fdd8ec78b38b9800061b4150413ff2e3b6f88
543c636f7cd84f6db9189d43a81e5503cda447da73c7e5b6') { echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-s
etup.php'); } echo PHP_EOL;"
Installer verified

C:\autodidak>

C:\autodidak>

C:\autodidak>c:\xampp\php\php composer-setup.php
All settings correct for using Composer
Downloading...

Composer (version 2.7.7) successfully installed to: C:\autodidak\composer.phar
Use it: php composer.phar

C:\autodidak>c:\xampp\php\php -r "unlink('composer-setup.php');"

C:\autodidak>c:\xampp\php\php composer.phar create-project laravel/laravel perpus
Creating a "laravel/laravel" project at "./perpus"
Cannot use laravel/laravel's latest version v11.1.1 as it requires php ^8.2 which is not satisfied by your platform.
Installing laravel/laravel (v10.3.3)
- Downloading laravel/laravel (v10.3.3)
- Installing laravel/laravel (v10.3.3): Extracting archive
Created project in C:\autodidak\perpus
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
```

### b. Menjalankan perintah seperti dibawah pada cmd

```
Command Prompt - c:\xampp
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE
spatie/laravel-ignition ..... DONE

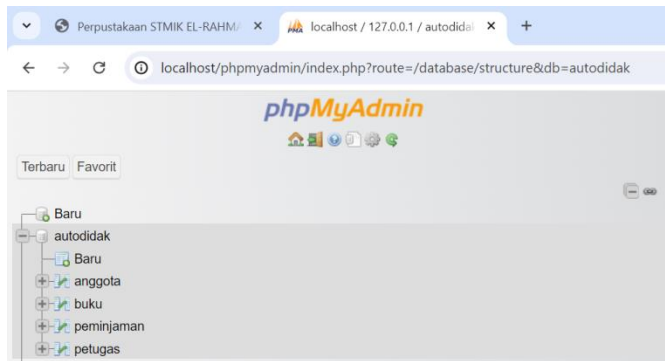
83 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

INFO No publishable resources for tag [laravel-assets].

No security vulnerability advisories found.
> @php artisan key:generate --ansi

INFO Application key set successfully.
```

### c. Membuat Database yang bernama autodidak yang berisi anggota,buku.peminjam dan petugas



### 3. Membuat BukuContorllor

#### a. Membuat BukuController

```
^C
C:\autodidak\perpus>c:\xampp\php\php artisan make:controller BukuController
INFO Controller [C:\autodidak\perpus\app\Http\Controllers\BukuController.php] created successfully.
```

#### b. Mengubah kode pada buku controller

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

//panggil model BukuModel
use App\Models\BukuModel;

class BukuController extends Controller
{
    //method untuk tampil data buku
    public function bukutampil()
    {
        $databuku = BukuModel::orderBy('kode_buku', 'ASC')
        ->paginate(5);

        return view('halaman/view_buku',['buku'=>$databuku]);
    }

    //method untuk tambah data buku
    public function bukutambah(Request $request)
    {
        $this->validate($request, [
            'kode_buku' => 'required',
            'judul' => 'required',
            'pengarang' => 'required',
            'kategori' => 'required'
        ]);

        BukuModel::create([
            'kode_buku' => $request->kode_buku,
            'judul' => $request->judul,
            'pengarang' => $request->pengarang,
```

```

        'kategori' => $request->kategori
    ]);

    return redirect('/buku');
}

//method untuk hapus data buku
public function bukuhapus($id_buku)
{
    $databuku=BukuModel::find($id_buku);
    $databuku->delete();

    return redirect()->back();
}

//method untuk edit data buku
public function bukuedit($id_buku, Request $request)
{
    $this->validate($request, [
        'kode_buku' => 'required',
        'judul' => 'required',
        'pengarang' => 'required',
        'kategori' => 'required'
    ]);

    $id_buku = BukuModel::find($id_buku);
    $id_buku->kode_buku    = $request->kode_buku;
    $id_buku->judul        = $request->judul;
    $id_buku->pengarang     = $request->pengarang;
    $id_buku->kategori      = $request->kategori;

    $id_buku->save();

    return redirect()->back();
}
}

```

### c. Menjalankan migrate tabel

Hasilnya Memasukan data dari database, dan ini hasilnya

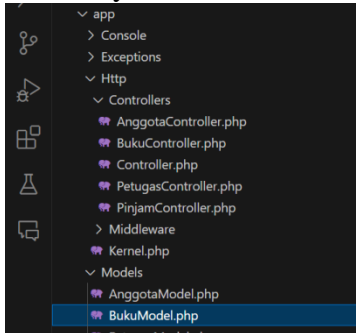
#	Nama	Jenis	Penyortiran	Atribut	Tak Terilai	Bawaan	Komentar	Ekstra	Tindakan
1	id_buku	int(255)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus
2	kode_buku	varchar(10)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus
3	judul	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus
4	pengarang	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus
5	kategori	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus
6	created_at	timestamp			Tidak	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()	Ubah Hapus
7	updated_at	datetime			Tidak	Tidak ada			Ubah Hapus

#### 4. Membuat Model

##### a. Membuat model pada di cmd

```
C:\autodidak\perpus>c:\xampp\php\php artisan make:model BukuModel  
  
INFO Model [C:\autodidak\perpus\app\Models\BukuModel.php] created successfully.
```

Hasilnya akan terbuat BukuModel

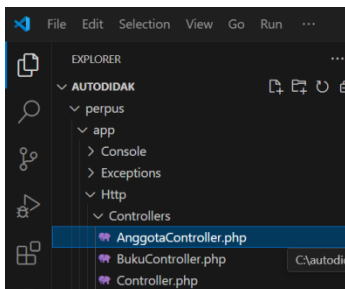


#### 5. Membuat Anggota

##### a. Membuat controller Anggota

```
C:\autodidak\perpus>c:\xampp\php\php artisan make:controller AnggotaController  
  
INFO Controller [C:\autodidak\perpus\app\Http\Controllers\AnggotaController.php] created successfully.
```

Hasilnya, controller akan terbuat dengan nama AnggotaController.php, controller ini berfungsi sebagai penghubung antara model dengan view



#### Program AnggotaController

```
<?php  
  
namespace App\Http\Controllers;  
  
use Illuminate\Http\Request;  
  
//panggil model AnggotaModel  
use App\Models\AnggotaModel;  
  
class AnggotaController extends Controller  
{  
    //method untuk tampil data anggota  
    public function anggotatampil()  
    {  
        $dataanggota = AnggotaModel::orderBy('id_anggota', 'ASC')  
        ->paginate(5);  
    }  
}
```

```

        return view('halaman/view_anggota',['anggota'=>$dataanggota]);
    }

    //method untuk tambah data anggota
    public function anggotatambah(Request $request)
    {
        $this->validate($request, [
            'nim' => 'required',
            'nama_anggota' => 'required',
            'prodi' => 'required',
            'hp' => 'required'
        ]);

        AnggotaModel::create([
            'nim' => $request->nim,
            'nama_anggota' => $request->nama_anggota,
            'prodi' => $request->prodi,
            'hp' => $request->hp
        ]);

        return redirect('/anggota');
    }

    //method untuk hapus data anggota
    public function anggotahapus($id_anggota)
    {
        $dataanggota=AnggotaModel::find($id_anggota);
        $dataanggota->delete();

        return redirect()->back();
    }

    //method untuk edit data anggota
    public function anggotaedit($id_anggota, Request $request)
    {
        $this->validate($request, [
            'nim' => 'required',
            'nama_anggota' => 'required',
            'prodi' => 'required',
            'hp' => 'required'
        ]);

        $id_anggota = AnggotaModel::find($id_anggota);
        $id_anggota->nim    = $request->nim;
        $id_anggota->nama_anggota    = $request->nama_anggota;
        $id_anggota->prodi    = $request->prodi;
        $id_anggota->hp    = $request->hp;

        $id_anggota->save();

        return redirect()->back();
    }
}

```

## 6. Membuat AnggotaModel

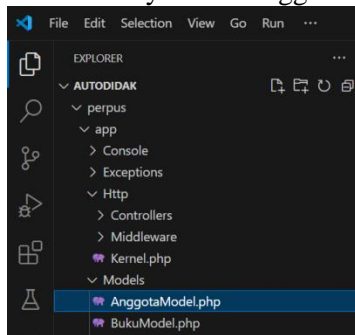
### a. Membuat AnggotaModel pada di cmd

```
^C
C:\autodidak\perpus>C:\xampp\php\php artisan make:model AnggotaModel

[INFO] Model [C:\autodidak\perpus\app\Models\AnggotaModel.php] created successfully.

C:\autodidak\perpus>
```

### b. Hasilnya akan AnggotaModel



## Program AnggotaModel

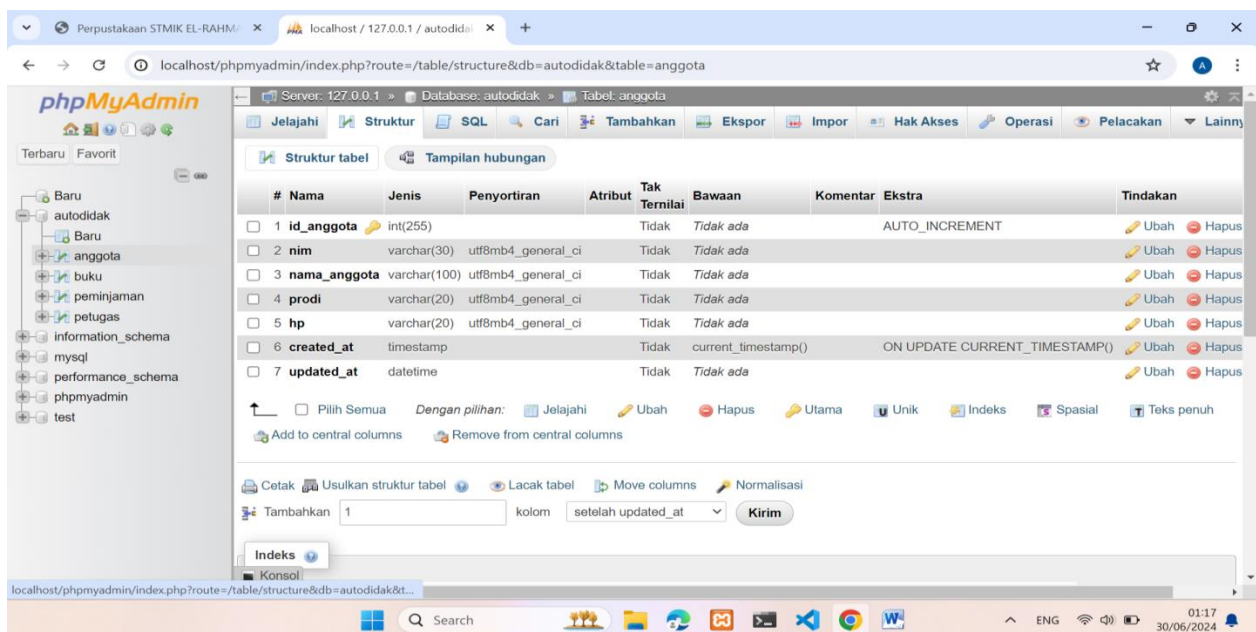
```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class AnggotaModel extends Model
{
    use HasFactory;
    protected $table = "anggota";
    protected $primaryKey = "id_anggota";
    protected $fillable = ['id_anggota', 'nim', 'nama_anggota', 'prodi', 'hp'];
}
```

Hasilnya akan tampil data yang ada didalam database



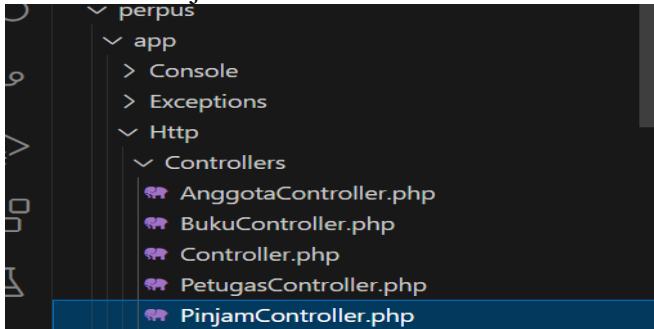


## 7. Membuat PinjamController

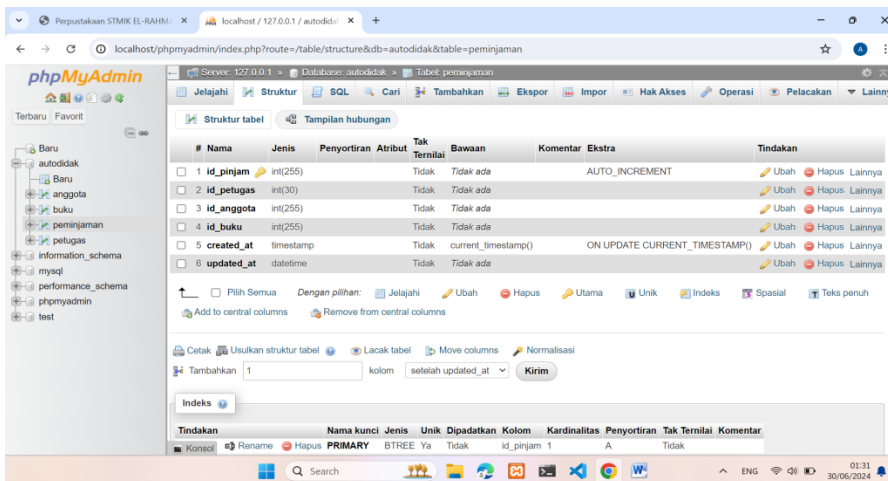
### a. Membuat PinjamController pada cmd

```
C:\autodidak\perpus>c:\xampp\php\php artisan make:controller PinjamController  
INFO Controller [C:\autodidak\perpus\app\Http\Controllers\PinjamController.php] created successfully.
```

### b. Hasil akan PinjamContoroller



### c. Hasilnya akan tampil data yang ada didalam database

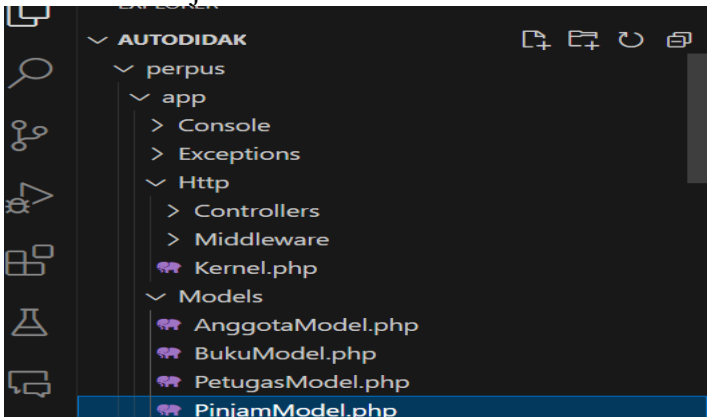


## 7. Membuat PinjamModel

### a. Membuat PinjamModel Pada mcd

```
C:\autodidak\perpus>c:\xampp\php\php artisan make:model PinjamModel  
INFO Model [C:\autodidak\perpus\app\Models\PinjamModel.php] created successfully.
```

### b. Hasil akan PinjamModel





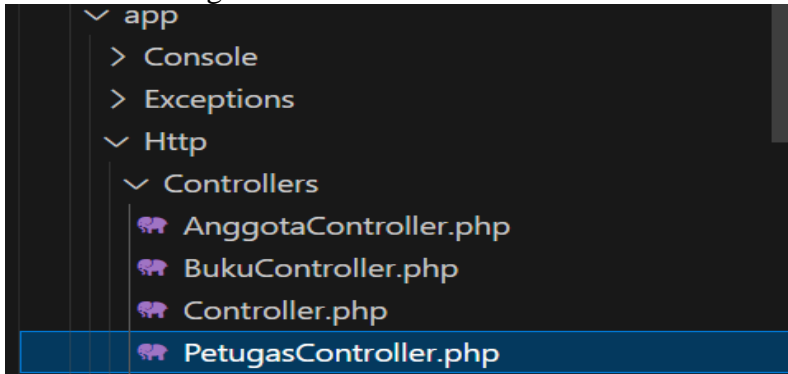
## 8. Membuat PetugasController

### a. Membuat PetugasController pada mcd

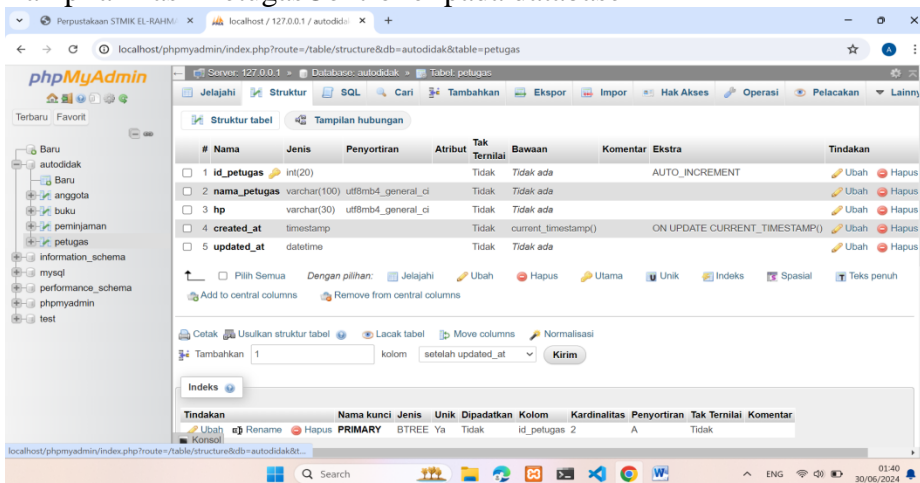
```
Command Prompt - c:\xampp
2024-06-29 21:45:32
C:\autodidak\perpus>c:\xampp\php\php artisan make:controller PetugasController

INFO Controller [C:\autodidak\perpus\app\Http\Controllers\PetugasController.php] created successfully.
```

### b. Hasil akan PetugasController



### c. Tampilan hasil PetugasController pada database



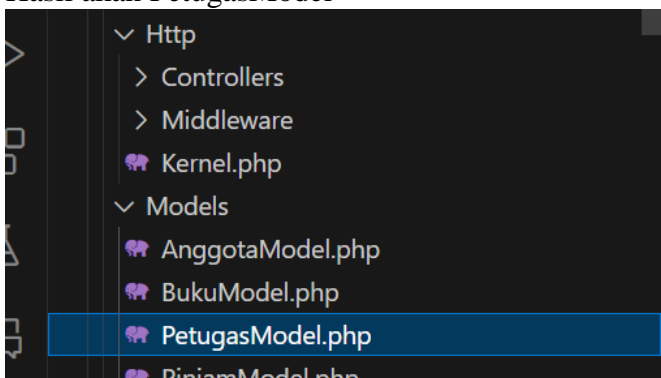
## 9. Membuat PetugasModel

### a. Membuat PetugasModel di cmd

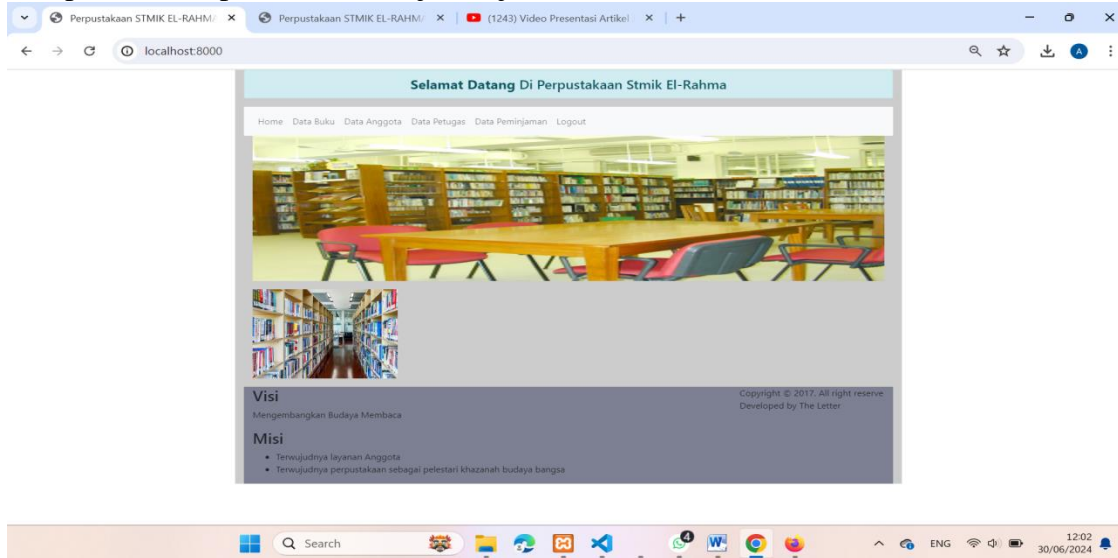
```
C:\autodidak\perpus>c:\xampp\php\php artisan make:model PetugasModel

INFO Model [C:\autodidak\perpus\app\Models\PetugasModel.php] created successfully.
```

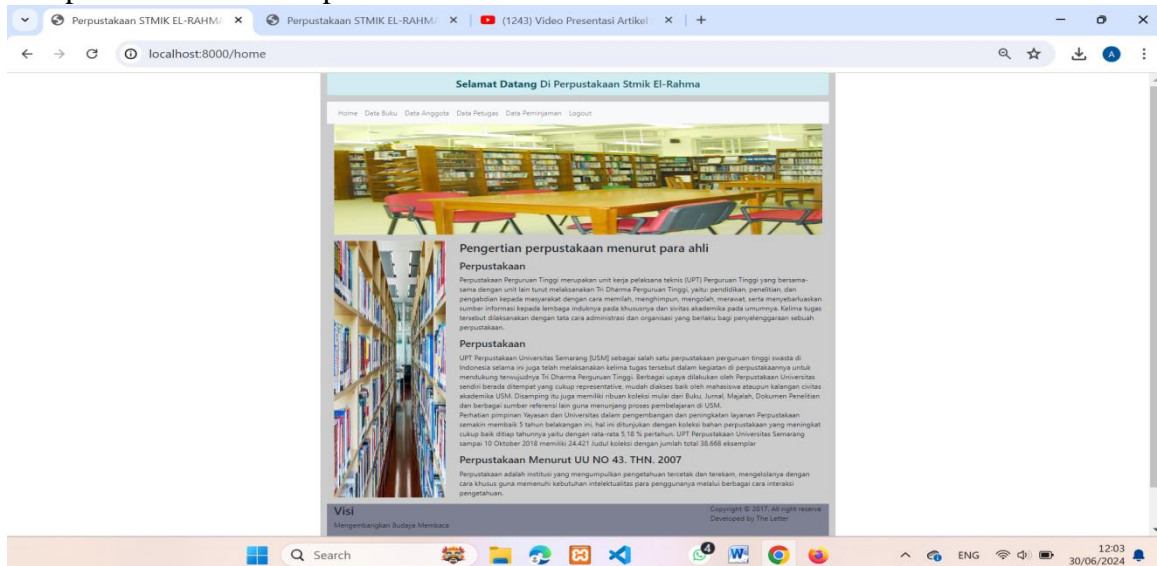
### b. Hasil akan PetugasModel



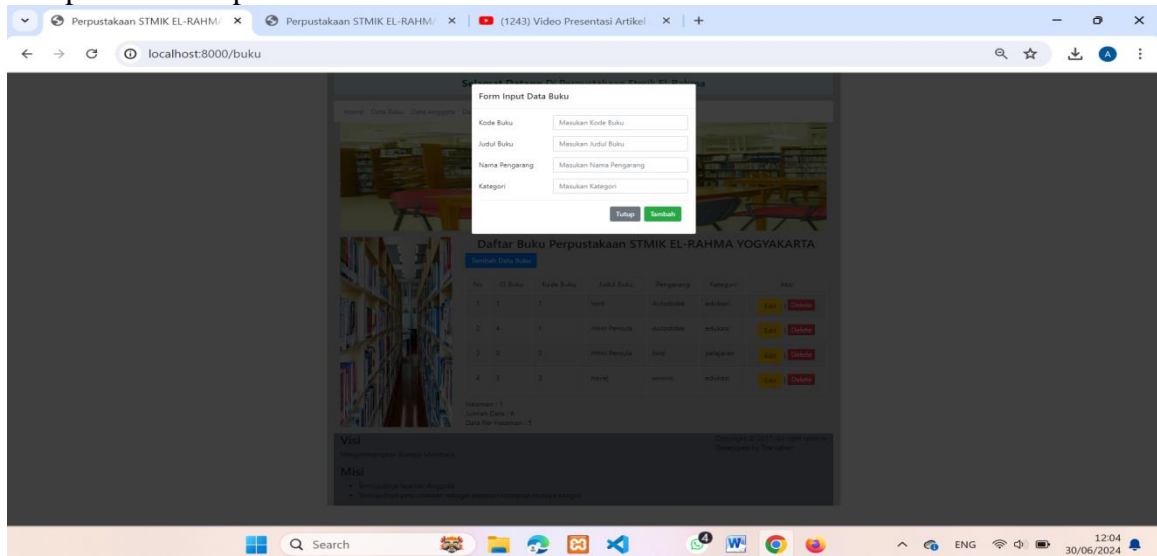
## Tampilan awal aplikasi website jika dijalankan ke browser



## Tampilan home website pada browser



## Tampilan form input data buku



## Tampilan tambah data buku yang sudah diinput

Daftar Buku Perpustakaan STMIK EL-RAHMA YOGYAKARTA

Tambah Data Buku

No	ID Buku	Kode Buku	Judul Buku	Pengarang	Kategori	Aksi
1	1	1	lord	Autodidak	edukasi	Edit Delete
2	4	1	Html Pemula	Autodidak	edukasi	Edit Delete
3	2	2	Html Pemula	lord	pelajaran	Edit Delete
4	3	3	novel	www	edukasi	Edit Delete

Halaman : 1  
Jumlah Data : 4  
Data Per Halaman : 5

Copyright © 2017. All right reserve

## Tampilan form edit data buku

Form Edit Data Buku

Kode Buku:

Judul Buku:

Nama Pengarang:

Kategori:

Tutup Edit

## Tampilan form data anggota

Form Input Data Anggota

NIM:

Nama Anggota:

Prodi:

HP:

Tutup Tambah

Daftar Anggota Perpustakaan STMIK EL-RAHMA YOGYAKARTA

No	ID Anggota	NIM	Nama Anggota	Prodi	HP	Aksi
1	2	12345	neta	s	08567	Edit Delete

Halaman : 1  
Jumlah Data : 1  
Data Per Halaman : 5

## Tampilan form data anggota yang sudah ditambahkan

The screenshot shows a web application for a library. The header includes a navigation menu with links: Home, Data Buku, Data Anggota, Data Petugas, Data Peminjaman, and Logout. The main content area features a large image of a library interior. Below the image, there is a section titled "Daftar Anggota Perpustakaan STMIK EL-RAHMA YOGYAKARTA" with a button "Tambah Data anggota". A table displays the list of members:

No	ID Anggota	NIM	Nama Anggota	Prodi	HP	Aksi
1	2	124567	ayu	ti	085647	<a href="#">Edit</a> <a href="#">Delete</a>
2	3	12221892	ancea syafriya	informatika	088985040898	<a href="#">Edit</a> <a href="#">Delete</a>

Below the table, it shows "Halaman : 1", "Jumlah Data : 2", and "Data Per Halaman : 5". The footer includes the library's vision statement, a copyright notice for 2017, and the developer's name, "The Letter".

## Tampilan form edit data anggota

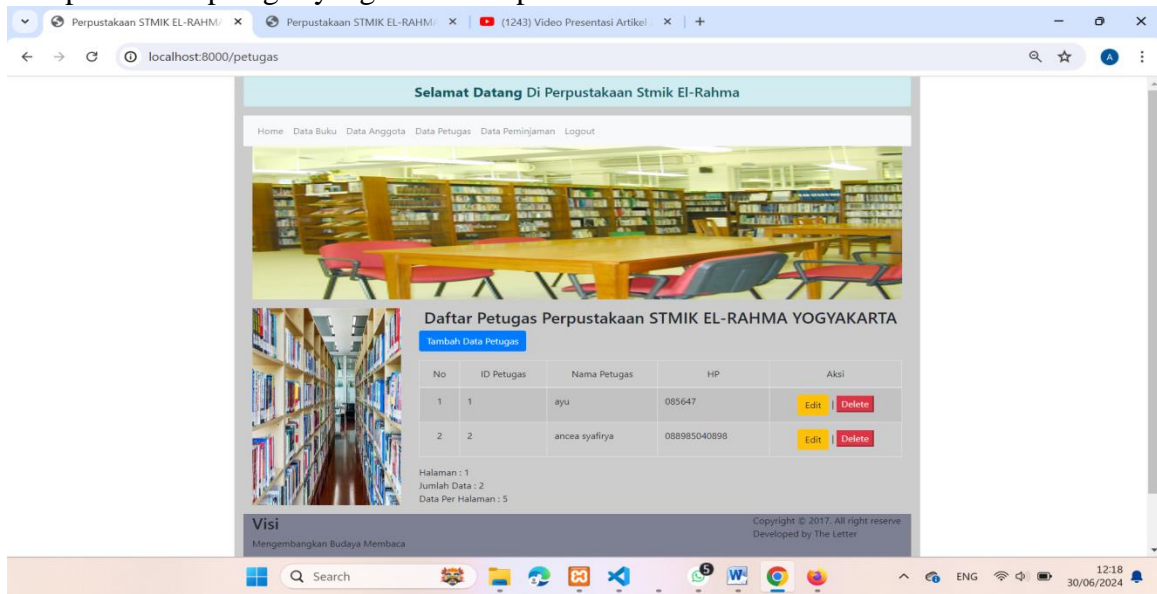
The screenshot shows the "Form Edit Data Anggota" modal. It contains input fields for "nim", "Nama Anggota", "Prodi", and "Hp". The "nim" field is pre-filled with "Masukan NIM", "Nama Anggota" with "ayu", "Prodi" with "ti", and "Hp" with "085647". There are "Tutup" and "Edit" buttons at the bottom of the form. The background shows the same library interface as the previous screenshot, but with a dark overlay.

## Tampilan form data petugas

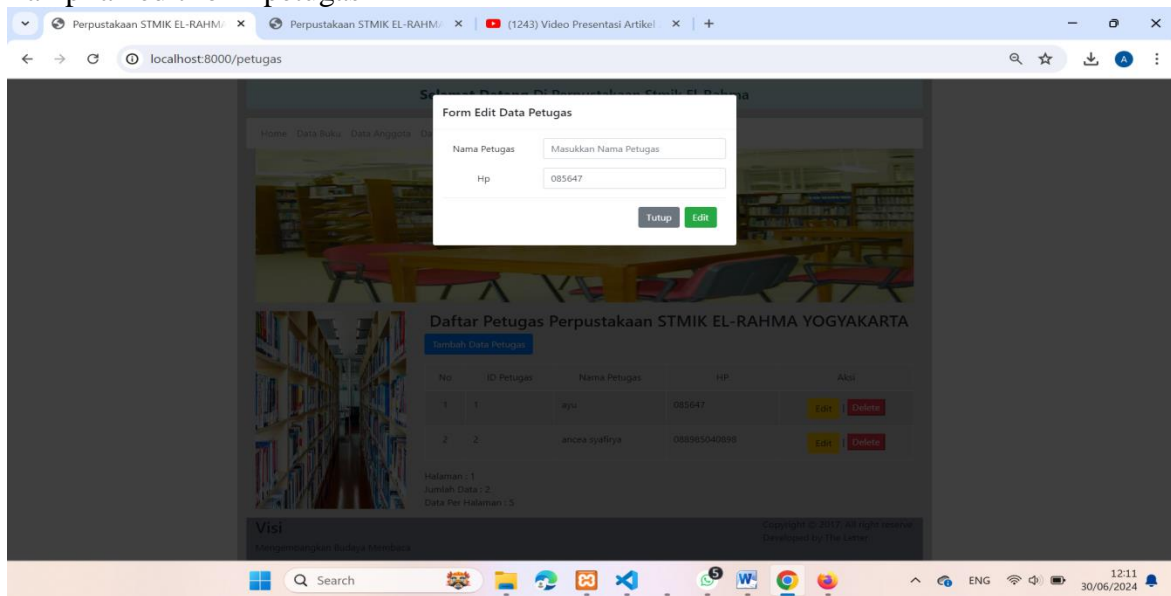
The screenshot shows the "Form Input Data Petugas" modal. It contains input fields for "Nama Petugas" and "HP". Both fields are pre-filled with "Masukan Nama Petugas" and "Masukan HP" respectively. There are "Tutup" and "Tambah" buttons at the bottom of the form. The background shows the same library interface as the previous screenshots, but with a dark overlay.



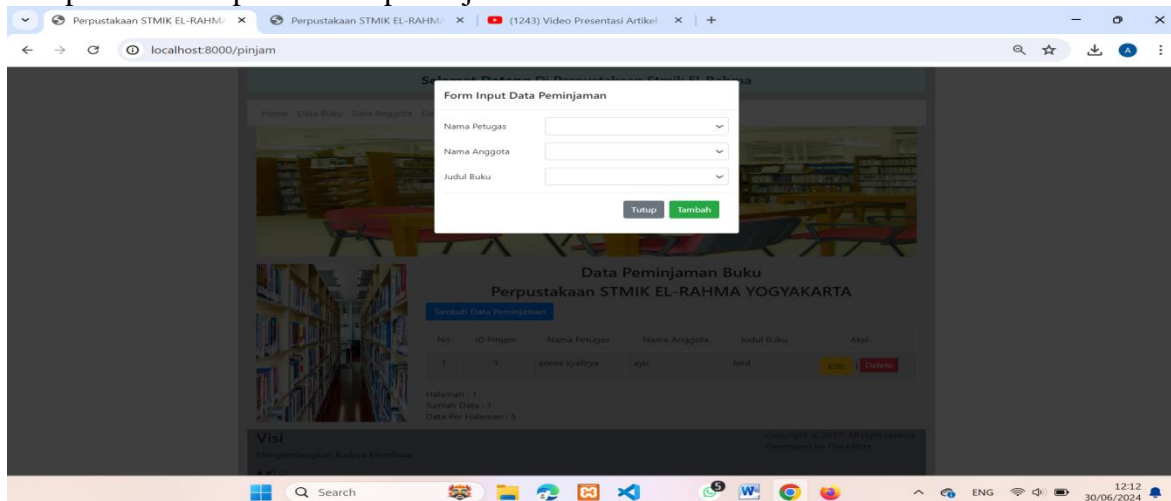
## Tampilan form petugas yang sudah diinputkan



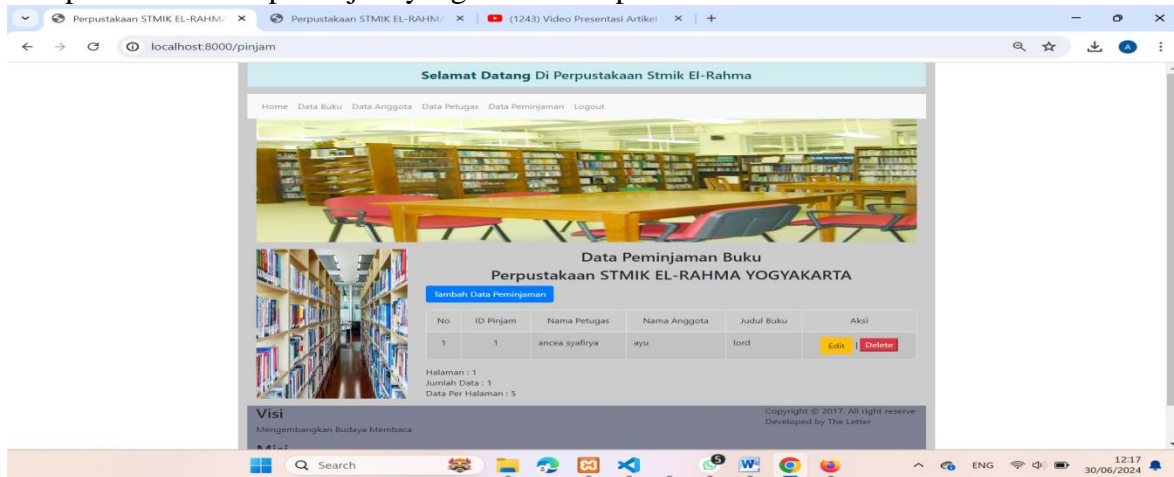
## Tampilan edit form petugas



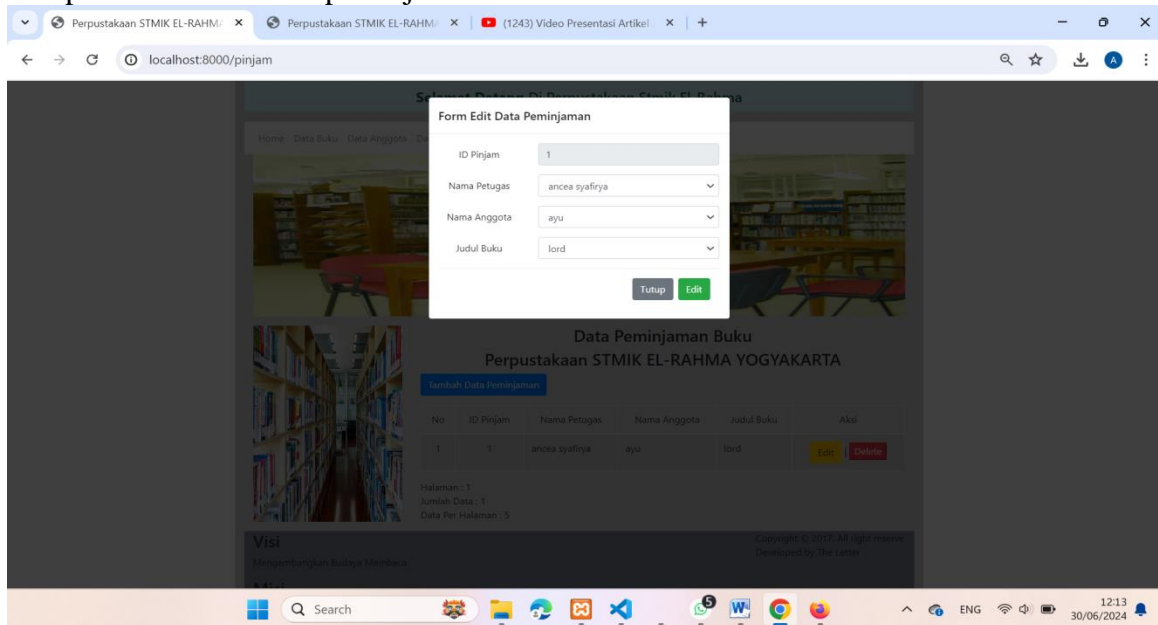
## Tampilan form input tambah peminjam



## Tampilan form data peminjam yang sudah di input



## Tampilan edit form data peminjam



## LINK VIDEO YOUTUBE:

Hasil presentasi: [https://youtu.be/4iD50BpYTjg?si=exFtpuaW\\_ixyeF1d](https://youtu.be/4iD50BpYTjg?si=exFtpuaW_ixyeF1d)

Tutor cara bikin: <https://youtu.be/U0675B2Pzto>