

SURAT TUGAS
2106/B.01/LPPM-UBSI/XII/2025

Tentang

**Pelindungan Ciptaan di Bidang Ilmu Pengetahuan, Seni dan Sastra Berdasarkan Undang-Undang
Nomor 28 Tahun 2014 tentang Hak Cipta**
Nomor dan Tanggal Permohonan : EC002025212382, 12 Desember 2025
Nomor Pencatatan : 001052642

**PADA SURAT PENCATATAN CIPTAAN KEMENTERIAN HUKUM DAN HAK ASASI
MANUSIA REPUBLIK INDONESIA**

Program Komputer

Judul Ciptaan :

**Rancang Bangun Program Pendataan Penduduk Berbasis Website Menggunakan Framework
Laravel (Studi Kasus : Kabupaten Bekasi)**

MEMUTUSKAN

Pertama : Menugaskan kepada saudara

1. 200904432 Melyani
2. 200609880 Indra Riyana Rahadjeng
3. 202203040 Roydawaty Bunga S
4. 202003024 Rizkiana Karmelia Shaura
5. 200903130 Daniel
6. 201108556 Fazhar Sumantri
7. 202009160 Syabrinildi

Sebagai Pencipta yang mempublikasikan karyanya.

Kedua : Mempunyai tugas sbb:
Melaksanakan Tugas yang diberikan dengan penuh rasa tanggung jawab.

Ketiga : Keputusan ini berlaku sejak tanggal ditetapkan, dengan ketentuan apabila
dikemudian hari terdapat kekeliruan akan diubah dan diperbaiki sebagaimana
mestinya.

Jakarta, 5 Desember 2025

LPPM Universitas Bina Sarana Informatika

Ketua



Agus Junaidi, M. Kom

Tembusan

- Ka. LPPM Universitas Bina Sarana Informatika
- Arsip
- Ybs

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC002025212382, 12 Desember 2025

Pencipta

Nama : **Melyani, Indra Riyana Rahadjeng dkk**
Alamat : Jl. Bunga Rampai VII gg.7 RT.010/Rw.006 No. 20 - Perumnas Klender,
Jakarta Timur 13470, Duren Sawit, Kota Adm. Jakarta Timur, DKI
Jakarta, 13470
Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : **Melyani, Indra Riyana Rahadjeng dkk**
Alamat : Jl. Bunga Rampai VII gg.7 RT.010/Rw.006 No. 20 - Perumnas Klender,
Jakarta Timur 13470, Duren Sawit, Kota Adm. Jakarta Timur, DKI
Jakarta, 13470
Kewarganegaraan : Indonesia
Jenis Ciptaan : **Program Komputer**
Judul Ciptaan : **Rancang Bangun Program Pendataan Penduduk Berbasis Website
Menggunakan Framework Laravel (Studi Kasus : Kabupaten
Bekasi)**

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : 12 Desember 2025, di Kota Adm. Jakarta Timur

Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.

Nomor Pencatatan : 001052642

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.

Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.



a.n. MENTERI HUKUM
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL
u.b
Direktur Hak Cipta dan Desain Industri

Agung Damarsasongko,SH.,MH.
NIP. 196912261994031001

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Melyani	Jl. Bunga Rampai VII gg.7 RT.010/Rw.006 No. 20 - Perumnas Klender, Jakarta Timur 13470 Duren Sawit, Kota Adm. Jakarta Timur
2	Indra Riyana Rahadjeng	Jl. Madrasah No. 31A, RT 004/03 Pondok Cina Depok, 16424 Beji, Kota Depok
3	Roydawaty Bunga S	Jl Cikunir Raya Gg H alam 3 rt 6 /15 no42 Jakasampurna Bekasi Barat 17145 Bekasi Barat, Kota Bekasi
4	Rizkiana Karmelia Shaura	Mutiara Depok Estate Blok FE/6. Jl. Tole Iskandar No.66 Sukmajaya, Depok. Sukmajaya, Kota Depok
5	Daniel	jl.Wedana Dalam No. 37 Rt. 006/001,Jakarta Timur, 13310 Matraman, Kota Adm. Jakarta Timur
6	Fazhar Sumantri	Apartment citylight light, jl.ir.h.juanda no.38, RT 001 RW 003, Ciputat timur, Tangerang selatan. Ciputat Timur, Kota Tangerang Selatan
7	Syabrinildi	Dukuh Zamrud, Blo S.16 No. 6, Padurenan, Mustika Jaya, Kota Bekasi Mustikajaya, Kota Bekasi

LAMPIRAN PEMEGANG

No	Nama	Alamat
1	Melyani	Jl. Bunga Rampai VII gg.7 RT.010/Rw.006 No. 20 - Perumnas Klender, Jakarta Timur 13470 Duren Sawit, Kota Adm. Jakarta Timur
2	Indra Riyana Rahadjeng	Jl. Madrasah No. 31A, RT 004/03 Pondok Cina Depok, 16424 Beji, Kota Depok
3	Melyani	Jl. Bunga Rampai VII gg.7 RT.010/Rw.006 No. 20 - Perumnas Klender, Jakarta Timur 13470 Duren Sawit, Kota Adm. Jakarta Timur
4	Roydawaty Bunga S	Jl Cikunir Raya Gg H alam 3 rt 6 /15 no42 Jakasampurna Bekasi Barat 17145 Bekasi Barat, Kota Bekasi
5	Rizkiana Karmelia Shaura	Mutiara Depok Estate Blok FE/6. Jl. Tole Iskandar No.66 Sukmajaya, Depok. Sukmajaya, Kota Depok
6	Daniel	jl.Wedana Dalam No. 37 Rt. 006/001,Jakarta Timur, 13310 Matraman, Kota Adm. Jakarta Timur
7	Fazhar Sumantri	Apartment citylight light, jl.ir.h.juanda no.38, RT 001 RW 003, Ciputat timur, Tangerang selatan. Ciputat Timur, Kota Tangerang Selatan
8	Syabrinildi	Dukuh Zamrud, Blo S.16 No. 6, Padurenan, Mustika Jaya, Kota Bekasi Mustikajaya, Kota Bekasi



Menu dan Paduan Penggunaan Rancang Bangun Program Pendataan Penduduk Berbasis Website Menggunakan Framework Laravel (Studi Kasus : Kabupaten Bekasi)

I. PENDAHULUAN

Proses pendataan penduduk manual di Kabupaten Bekasi menghadapi kendala inefisiensi, redundansi data, dan keterlambatan pembaruan. Penelitian ini bertujuan merancang program pendataan berbasis website menggunakan framework Laravel untuk mengatasi masalah tersebut. Pengembangan sistem menggunakan model Waterfall yang meliputi analisis, perancangan dengan UML dan ERD, implementasi dengan arsitektur MVC, serta pengujian black-box. Hasilnya adalah sistem fungsional yang mampu mengelola data penduduk, mencatat mutasi, dan menghasilkan laporan secara efisien, akurat, dan aman, dengan tingkat keberhasilan pengujian fungsional mencapai 100%.

Pendataan penduduk merupakan fondasi krusial bagi perencanaan pembangunan dan pengambilan kebijakan publik yang efektif. Data yang akurat dan mutakhir memungkinkan pemerintah merumuskan program strategis di berbagai sektor. Namun, di Kabupaten Bekasi, proses pendataan yang masih berjalan secara manual menghadapi berbagai kendala signifikan. Sistem konvensional ini rentan terhadap redundansi data, keterlambatan pembaruan informasi mutasi seperti kelahiran dan kematian, serta inefisiensi dalam penyusunan laporan demografis. Ketergantungan pada proses manual tidak hanya memperlambat alur kerja, tetapi juga meningkatkan risiko kesalahan manusia yang dapat mengkompromikan validitas data kependudukan secara keseluruhan.

Untuk mengatasi permasalahan tersebut, pengembangan program pendataan penduduk berbasis website menjadi solusi strategis yang mendesak. Penelitian ini bertujuan untuk merancang dan membangun sebuah sistem informasi terpusat menggunakan framework Laravel. Pemilihan Laravel didasarkan pada arsitektur Model-View-Controller (MVC) yang memfasilitasi pengembangan terstruktur, serta menyediakan fitur-fitur modern untuk keamanan dan pengelolaan basis data. Melalui penerapan sistem ini, diharapkan proses pengelolaan data kependudukan di Kabupaten Bekasi dapat menjadi lebih efisien, akurat, dan aman. Sistem ini dirancang untuk mengotomatisasi pencatatan, pembaruan data mutasi, serta penyajian laporan demografis secara real-time.

II. Landasan Teori

Kerangka kerja teoretis yang digunakan dalam penelitian ini adalah model pengembangan perangkat lunak Waterfall. Model ini dipilih karena pendekatannya yang sistematis dan sekuensial, di mana setiap fase pengembangan harus diselesaikan sepenuhnya sebelum melanjutkan ke fase berikutnya. Tahapan dalam model ini meliputi analisis kebutuhan[3], perancangan sistem, implementasi, pengujian, serta penerapan dan pemeliharaan. Struktur yang jelas ini sangat sesuai untuk proyek perancangan program pendataan penduduk yang memiliki

ruang lingkup dan tujuan yang terdefinisi dengan baik, memastikan proses pengembangan yang terorganisir.

Tahap pertama, analisis kebutuhan, melibatkan pengumpulan data secara mendalam mengenai proses pendataan penduduk yang sedang berjalan di Kabupaten Bekasi. Kebutuhan fungsional dan non-fungsional diidentifikasi melalui wawancara dan observasi untuk memastikan sistem dapat mengatasi permasalahan yang ada. Selanjutnya, pada tahap perancangan, seluruh kebutuhan tersebut diterjemahkan ke dalam cetak biru arsitektur sistem. Ini mencakup perancangan basis data, desain antarmuka pengguna, serta pemodelan alur proses menggunakan diagram UML untuk memvisualisasikan struktur dan interaksi antar komponen sistem.

Fase implementasi merupakan tahap di mana rancangan sistem diwujudkan menjadi kode program yang fungsional. Pada tahap ini, pengembang menulis kode menggunakan bahasa pemrograman PHP dengan framework Laravel sesuai dengan arsitektur dan desain yang telah ditetapkan sebelumnya. Setelah proses pengkodean selesai, sistem memasuki fase pengujian. Pengujian dilakukan secara komprehensif, menggunakan metode seperti black-box testing, untuk memverifikasi bahwa semua fungsi berjalan sesuai spesifikasi kebutuhan, bebas dari kesalahan, dan mampu menangani input data kependudukan secara akurat dan aman.

Setelah berhasil melewati fase pengujian, sistem siap untuk diterapkan pada lingkungan operasional sesungguhnya di instansi terkait di Kabupaten Bekasi. Tahap penerapan ini mencakup instalasi perangkat lunak, migrasi data awal jika diperlukan, serta pelatihan bagi para pengguna akhir. Fase terakhir adalah pemeliharaan, yang merupakan proses berkelanjutan untuk memastikan sistem tetap berjalan optimal. Kegiatan dalam fase ini meliputi perbaikan bug yang ditemukan pasca-implementasi, penyesuaian minor, serta pembaruan sistem untuk mengakomodasi perubahan kebutuhan di masa mendatang.

Struktur file (yang disertakan di sini)

- app/Models/Penduduk.php
- app/Http/Controllers/PendudukController.php
- app/Http/Controllers/DashboardController.php
- routes/web.php
- database/migrations/2025_01_01_000000_create_penduduk_table.php
- database/seeder/DatabaseSeeder.php
- database/seeder/PendudukSeeder.php
- resources/views/layouts/app.blade.php
- resources/views/dashboard.blade.php

- resources/views/penduduk/index.blade.php
- resources/views/penduduk/create.blade.php
- resources/views/penduduk/edit.blade.php
- resources/views/penduduk/show.blade.php
- public/css/app.css (basic)
- README (instruksi)

File: app/Models/Penduduk.php

```php

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

class Penduduk extends Model

{

use HasFactory;

protected \$table = 'penduduk';

protected \$fillable = [

'nik',

'nama',

'jenis\_kelamin',

'tempat\_lahir',

```
'tanggal_lahir',
'alamat',
'rt',
'rw',
'desa',
'kecamatan',
'agama',
'status_kawin',
'pekerjaan',
'pendidikan'
];
```

```
protected $dates = ['tanggal_lahir'];
}
...
```

----

### File: app/Http/Controllers/PendudukController.php

```php

<?php

```
namespace App\Http\Controllers;
```

```
use App\Models\Penduduk;
```

```
use Illuminate\Http\Request;
```

```
use Illuminate\Support\Facades\Validator;
```

```
class PendudukController extends Controller
```



```

{
    public function __construct()
    {
        $this->middleware('auth');
    }

    public function index(Request $request)
    {
        $q = $request->query('q');
        $penduduk = Penduduk::when($q, function($query) use ($q){
            $query->where('nama','like','%'.$q.'%')
                ->orWhere('nik','like','%'.$q.'%')
                ->orWhere('alamat','like','%'.$q.'%');
        })->orderBy('nama')->paginate(15);

        return view('penduduk.index', compact('penduduk','q'));
    }

    public function create()
    {
        return view('penduduk.create');
    }

    public function store(Request $request)
    {
        $rules = [
            'nik' => 'required|digits:16|unique:penduduk,nik',
            'nama' => 'required|string|max:255',
            'jenis_kelamin' => 'required|in:L,P',

```

```

        'tanggal_lahir' => 'required|date',
    ];

    $validator = Validator::make($request->all(), $rules);
    if ($validator->fails()) return redirect()->back()->withErrors($validator)->withInput();

    Penduduk::create($request->all());
    return redirect()->route('penduduk.index')->with('success','Data penduduk berhasil ditambahkan');
}

public function show(Penduduk $penduduk)
{
    return view('penduduk.show', compact('penduduk'));
}

public function edit(Penduduk $penduduk)
{
    return view('penduduk.edit', compact('penduduk'));
}

public function update(Request $request, Penduduk $penduduk)
{
    $rules = [
        'nik' => 'required|digits:16|unique:penduduk,nik,',$penduduk->id,
        'nama' => 'required|string|max:255',
        'jenis_kelamin' => 'required|in:L,P',
        'tanggal_lahir' => 'required|date',
    ];

```

```
$validator = Validator::make($request->all(), $rules);
```

```
if ($validator->fails()) return redirect()->back()->withErrors($validator)->withInput();
```

```
$penduduk->update($request->all());
```

```
return redirect()->route('penduduk.index')->with('success','Data penduduk berhasil diupdate');
```

```
}
```

```
public function destroy(Penduduk $penduduk)
```

```
{
```

```
$penduduk->delete();
```

```
return redirect()->route('penduduk.index')->with('success','Data penduduk berhasil dihapus');
```

```
}
```

```
}
```

```
...
```

```
----
```

```
### File: app/Http/Controllers/DashboardController.php
```

```
```php
```

```
<?php
```

```
namespace App\Http\Controllers;
```

```
use App\Models\Penduduk;
```

```
class DashboardController extends Controller
```

```
{
```

```
 public function __construct()
```

```
 {
```

```

 $this->middleware('auth');
 }

 public function index()
 {
 $total = Penduduk::count();

 $male = Penduduk::where('jenis_kelamin','L')->count();

 $female = Penduduk::where('jenis_kelamin','P')->count();

 return view('dashboard', compact('total','male','female'));
 }
}
...

```

----

### File: routes/web.php

```
```php
```

```
<?php
```

```

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\PendudukController;
use App\Http\Controllers\DashboardController;

```

```
Route::get('/', function(){ return redirect()->route('dashboard'); });
```

```
Auth::routes(); // jika menggunakan laravel/ui atau breeze
```

```
Route::get('/dashboard', [DashboardController::class,'index']->name('dashboard');
```

```
Route::resource('penduduk', PendudukController::class);
```

```
...
```

```
----
```

```
### File: database/migrations/2025_01_01_000000_create_penduduk_table.php
```

```
```php
```

```
<?php
```

```
use Illuminate\Database\Migrations\Migration;
```

```
use Illuminate\Database\Schema\Blueprint;
```

```
use Illuminate\Support\Facades\Schema;
```

```
return new class extends Migration
```

```
{
```

```
 public function up()
```

```
 {
```

```
 Schema::create('penduduk', function (Blueprint $table) {
```

```
 $table->id();
```

```
 $table->string('nik',16)->unique();
```

```
 $table->string('nama');
```

```
 $table->enum('jenis_kelamin',['L','P']);
```

```
 $table->string('tempat_lahir')->nullable();
```

```
 $table->date('tanggal_lahir')->nullable();
```

```
 $table->text('alamat')->nullable();
```

```
 $table->string('rt',5)->nullable();
```

```
 $table->string('rw',5)->nullable();
```

```
 $table->string('desa')->nullable();
```



```

 $table->string('kecamatan')->nullable();
 $table->string('agama')->nullable();
 $table->string('status_kawin')->nullable();
 $table->string('pekerjaan')->nullable();
 $table->string('pendidikan')->nullable();
 $table->timestamps();
 });
}

```

```

public function down()
{
 Schema::dropIfExists('penduduk');
}
};
...

```

----

### File: database/seeder/PendudukSeeder.php

```
```php
```

```
<?php
```

```
namespace Database\Seeders;
```

```
use Illuminate\Database\Seeder;
```

```
use App\Models\Penduduk;
```

```
class PendudukSeeder extends Seeder
```

```
{
```

```

    public function run()
    {
        Penduduk::factory()->count(50)->create();
    }
}
'''

```

File: database/seeder/DatabaseSeeder.php

```

```php

```

```

<?php

```

```

namespace Database\Seeders;

```

```

use Illuminate\Database\Seeder;

```

```

class DatabaseSeeder extends Seeder

```

```

{
 public function run()
 {
 // \App\Models\User::factory(10)->create();
 $this->call([PendudukSeeder::class]);
 }
}
'''

```

----

### File: database/factories/PendudukFactory.php

```php

<?php

namespace Database\Factories;

use Illuminate\Database\Eloquent\Factories\Factory;

use Illuminate\Support\Str;

class PendudukFactory extends Factory

{

public function definition()

{

return [

'nik' => \$this->faker->unique()->numerify('#####'),

'nama' => \$this->faker->name(),

'jenis_kelamin' => \$this->faker->randomElement(['L', 'P']),

'tempat_lahir' => \$this->faker->city(),

'tanggal_lahir' => \$this->faker->date('Y-m-d', '2005-01-01'),

'alamat' => \$this->faker->address(),

'rt' => \$this->faker->numberBetween(1,9),

'rw' => \$this->faker->numberBetween(1,9),

'desa' => 'Desa Contoh',

'kecamatan' => 'Kecamatan Contoh',

'agama' => \$this->faker->randomElement(['Islam', 'Kristen', 'Katolik', 'Hindu', 'Buddha']),

'status_kawin' => \$this->faker->randomElement(['Belum Kawin', 'Kawin', 'Cerai']),

'pekerjaan' => \$this->faker->jobTitle(),

'pendidikan' => \$this->faker->randomElement(['SD', 'SMP', 'SMA', 'S1', 'S2'])

];

```
}  
}  
...
```

```
----
```

```
### File: resources/views/layouts/app.blade.php
```

```
```blade
```

```
<!doctype html>
```

```
<html lang="id">
```

```
<head>
```

```
 <meta charset="UTF-8">
```

```
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
 <title>Pendataan Penduduk - Kabupaten Bekasi</title>
```

```
 <link href="{{ asset('css/app.css') }}" rel="stylesheet">
```

```
</head>
```

```
<body>
```

```
<nav>
```

```
 Dashboard |
```

```
 Penduduk
```

```
 @auth
```

```
 | Hai, {{ auth()->user()->name }}
```

```
 <form style="display:inline" method="POST" action="{{ route('logout') }}">@csrf<button
type="submit">Logout</button></form>
```

```
 @endauth
```

```
</nav>
```

```
<div class="container">
```

```
 @if(session('success'))
```

```
 <div class="alert success">{{ session('success') }}</div>
```

```
@endif
```

```
@yield('content')
```

```
</div>
```

```
</body>
```

```
</html>
```

```
...
```

```

```

```
File: resources/views/dashboard.blade.php
```

```
```blade
```

```
@extends('layouts.app')
```

```
@section('content')
```

```
<h1>Dashboard</h1>
```

```
<p>Total penduduk: {{ $total }}</p>
```

```
<p>Laki-laki: {{ $male }}</p>
```

```
<p>Perempuan: {{ $female }}</p>
```

```
@endsection
```

```
...
```

```
----
```

```
### File: resources/views/penduduk/index.blade.php
```

```
```blade
```

```
@extends('layouts.app')
```

```
@section('content')
```

```
<h1>Daftar Penduduk</h1>
```

```
Tambah Penduduk
```

```
<form method="get">
```

```
 <input type="text" name="q" value="{{ $q ?? '' }}" placeholder="cari nama / NIK">
```

```
 <button type="submit">Cari</button>
```

```
</form>
```

```
<table border="1" cellpadding="6" cellspacing="0">
```

```
 <thead>
```

```
 <tr>
```

```
 <th>#</th>
```

```
 <th>NIK</th>
```

```
 <th>Nama</th>
```

```
 <th>JK</th>
```

```
 <th>TTL</th>
```

```
 <th>Aksi</th>
```

```
 </tr>
```

```
 </thead>
```

```
 <tbody>
```

```
 @foreach($penduduk as $p)
```

```
 <tr>
```

```
 <td>{{ $loop->iteration + ($penduduk->currentPage()-1)*$penduduk->perPage() }}</td>
```

```
 <td>{{ $p->nik }}</td>
```

```
 <td>{{ $p->nama }}</td>
```

```
 <td>{{ $p->jenis_kelamin }}</td>
```

```
 <td>{{ $p->tempat_lahir }}, {{ $p->tanggal_lahir->format('d-m-Y') }}</td>
```

```
 <td>
```

```
 Lihat |
```

```

 Edit |

 <form action="{{ route('penduduk.destroy',$p) }}" method="POST"
style="display:inline">@csrf @method('DELETE')<button onclick="return
confirm('Hapus?')">Hapus</button></form>

 </td>

</tr>

@endforeach

</tbody>

</table>

```

```

 {{ $penduduk->withQueryString()->links() }}

@endsection
...

```

----

### File: resources/views/penduduk/create.blade.php

```
``blade
```

```
@extends('layouts.app')
```

```
@section('content')
```

```
<h1>Tambah Penduduk</h1>
```

```
@if($errors->any())
```

```
<div class="alert error">
```

```

```

```
@foreach($errors->all() as $err)
```

```
{{ $err }}
```

```
@endforeach
```

```

```

</div>

@endif

<form method="POST" action="{{ route('penduduk.store') }}">

@csrf

<label>NIK</label><br>

<input name="nik" value="{{ old('nik') }}"><br>

<label>Nama</label><br>

<input name="nama" value="{{ old('nama') }}"><br>

<label>Jenis Kelamin</label><br>

<select name="jenis\_kelamin"><option value="L">L</option><option value="P">P</option></select><br>

<label>Tempat Lahir</label><br>

<input name="tempat\_lahir" value="{{ old('tempat\_lahir') }}"><br>

<label>Tanggal Lahir</label><br>

<input type="date" name="tanggal\_lahir" value="{{ old('tanggal\_lahir') }}"><br>

<label>Alamat</label><br>

<textarea name="alamat">{{ old('alamat') }}</textarea><br>

<button type="submit">Simpan</button>

</form>

@endsection

...

----

### File: resources/views/penduduk/edit.blade.php

```blade

@extends('layouts.app')


```

@section('content')

<h1>Edit Penduduk</h1>

@if($errors->any())

    <div class="alert error">

        <ul>

            @foreach($errors->all() as $err)

                <li>{{ $err }}</li>

            @endforeach

        </ul>

    </div>

@endif

<form method="POST" action="{{ route('penduduk.update',$penduduk) }}">

    @csrf @method('PUT')

    <label>NIK</label><br>

    <input name="nik" value="{{ old('nik',$penduduk->nik) }}"><br>

    <label>Nama</label><br>

    <input name="nama" value="{{ old('nama',$penduduk->nama) }}"><br>

    <label>Jenis Kelamin</label><br>

    <select name="jenis_kelamin"><option value="L" {{ $penduduk->jenis_kelamin=='L'? 'selected': '' }}>L</option><option value="P" {{ $penduduk->jenis_kelamin=='P'? 'selected': '' }}>P</option></select><br>

    <label>Tempat Lahir</label><br>

    <input name="tempat_lahir" value="{{ old('tempat_lahir',$penduduk->tempat_lahir) }}"><br>

    <label>Tanggal Lahir</label><br>

    <input type="date" name="tanggal_lahir" value="{{ old('tanggal_lahir',$penduduk->tanggal_lahir->format('Y-m-d')) }}"><br>

    <label>Alamat</label><br>

    <textarea name="alamat">{{ old('alamat',$penduduk->alamat) }}</textarea><br>

    <button type="submit">Update</button>

```

```

    </form>

@endsection
...

----

### File: resources/views/penduduk/show.blade.php
```blade
@extends('layouts.app')

@section('content')
 <h1>Detail Penduduk</h1>
 <p>NIK: {{ $penduduk->nik }}</p>
 <p>Nama: {{ $penduduk->nama }}</p>
 <p>JK: {{ $penduduk->jenis_kelamin }}</p>
 <p>TTL: {{ $penduduk->tempat_lahir }}, {{ $penduduk->tanggal_lahir->format('d-m-Y') }}</p>
 <p>Alamat: {{ $penduduk->alamat }}</p>
 Kembali
@endsection
...

File: public/css/app.css
```css
body{ font-family: Arial, Helvetica, sans-serif; margin:20px }
.container{ max-width:1000px; margin:20px auto }
nav{ margin-bottom:10px }

```

```
.alert{ padding:10px; margin-bottom:10px }  
.alert.success{ background:#d4edda }  
.alert.error{ background:#f8d7da }  
table{ width:100%; border-collapse:collapse }  
...
```

README: Cara deploy singkat

...

1. composer create-project laravel/laravel pendataan
2. cd pendataan
3. copy file-file dari dokumen ini ke folder proyek (timpa jika perlu)
4. composer require laravel/breeze --dev
5. php artisan breeze:install
6. npm install && npm run dev
7. cp .env.example .env // set DB_DATABASE, DB_USERNAME, DB_PASSWORD
8. php artisan key:generate
9. php artisan migrate --seed
10. php artisan serve

...

Catatan: Dokumen ini berisi file-file inti aplikasi. Untuk mendapatkan fitur autentikasi dan UI yang lengkap disarankan menginstal Laravel Breeze atau Jetstream sesuai preferensi. Jika ingin, saya juga bisa:

- Menyediakan file zip berisi semua file (siap ekstrak ke project).

- Membuat versi lengkap yang langsung berupa repository Git.

Sampaikan pilihan Anda.