Stephen Prasetya Chrismawan (H1D021025)
https://github.com/stephenprasetyachrismawan/karyawan
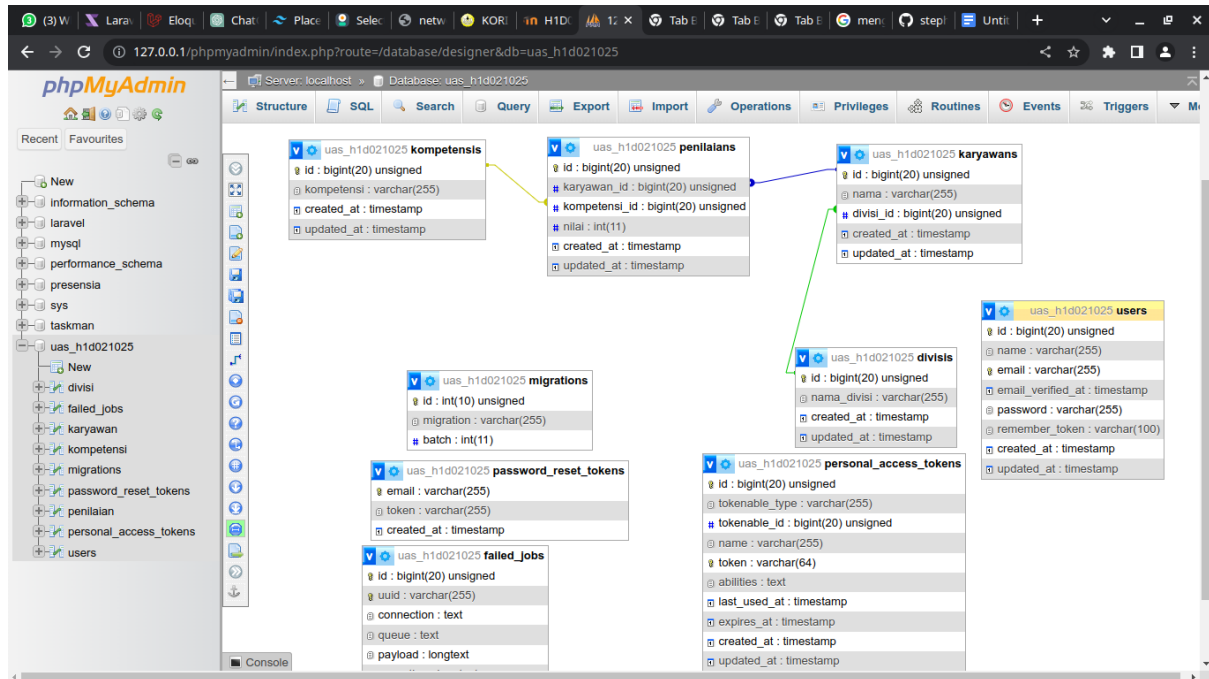
Dashboard

Dashboard

TAMBAH KARYAWAN

Show 10 entries

Search:

Nama	Divisi	Nilai Total	Aksi
Dika	IT	0	BERI NILAI EDIT PROFILE HAPUS KARYAWAN
Stephen P C	HRD	8	BERI NILAI EDIT PROFILE HAPUS KARYAWAN
Nama	Divisi	Nilai	Aksi

Showing 1 to 2 of 2 entries

Previous 1 Next

127.0.0.1:8000/karyawanedit/3

Dashboard

Admin

Form Penilaian Karyawan

Penilaian Karyawan

Dika

Rating Kehadiran
★★★★☆

Rating Tugas Selesai
★★★☆☆

Rating Kejujuran
★★☆☆☆

SIMPAN

Dashboard

Dashboard

TAMBAH KARYAWAN

Show 10 entries

Search:

Nama	Divisi	Nilai Total	Aksi
Dika	IT	9	BERI NILAI EDIT PROFILE HAPUS KARYAWAN
Stephen P C	HRD	8	BERI NILAI EDIT PROFILE HAPUS KARYAWAN
Nama	**Divisi**	**Nilai**	**Aksi**

Showing 1 to 2 of 2 entries

Previous 1 Next
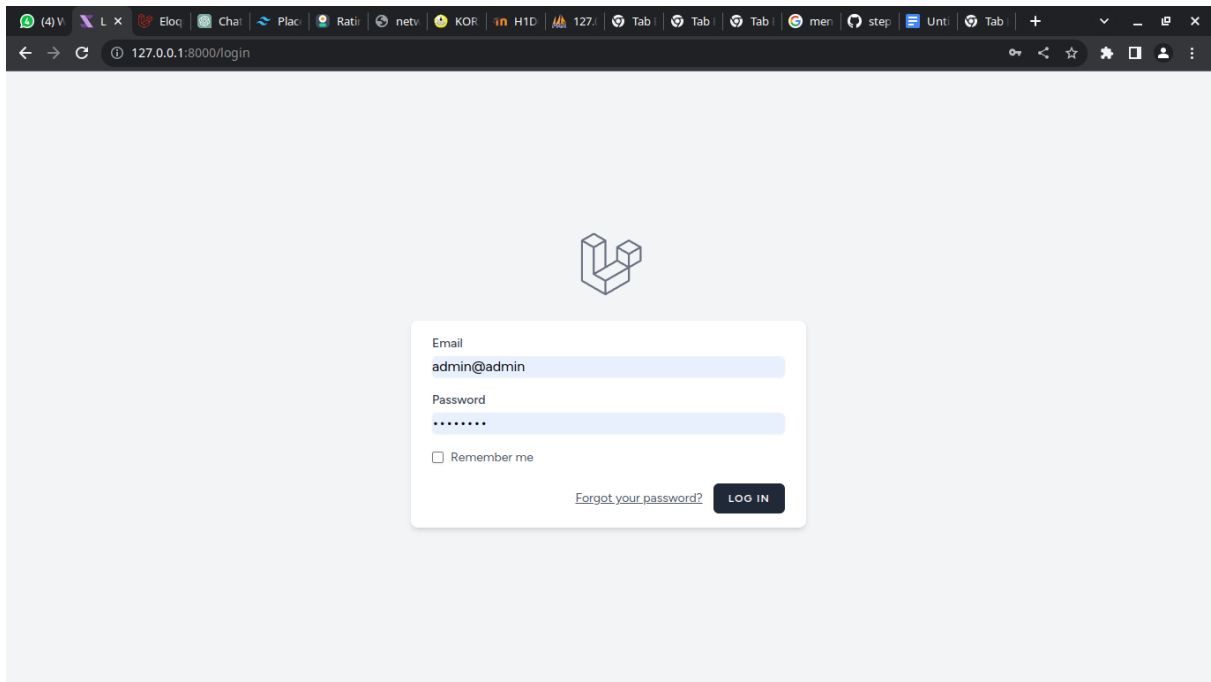
Dashboard

Admin

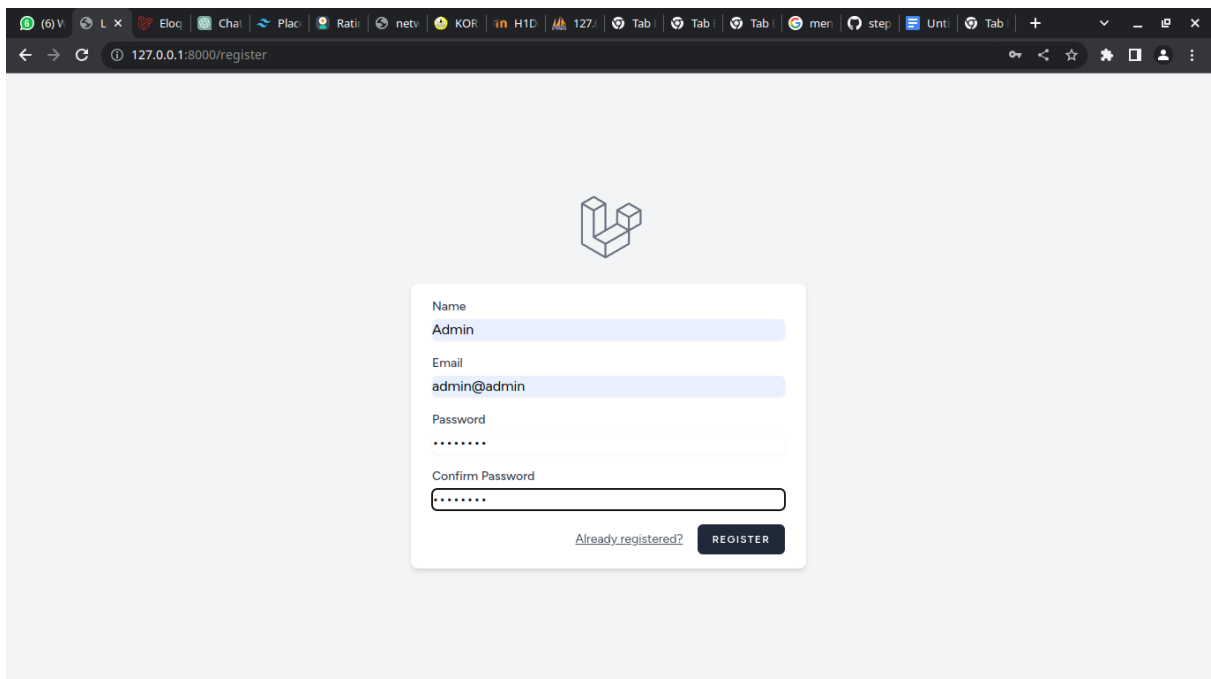**Form Edit Karyawan**

Edit Data Karyawan

Dika

HRD

SIMPAN

Login : admin@admin pass: 12345678

File migrations

```php
<?php

use Illuminate\Database\Migrations\Migration;

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

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

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('divisis', function (Blueprint $table) {
            $table->id();
            $table->string('nama_divisi');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('divisi');
    }
};
```

```php
<?php

use Illuminate\Database\Migrations\Migration;

use Illuminate\Database\Schema\Blueprint;

use Illuminate\Support\Facades\Schema;


return new class extends Migration

{

/**

* Run the migrations.

*/

public function up(): void

{

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

$table->id();

$table->string('nama');

$table->unsignedBigInteger('divisi_id');

$table->foreign('divisi_id')->references('id')->on('divisis');

$table->timestamps();

});

}


/**

* Reverse the migrations.

*/

public function down(): void
```

```php
{
Schema::dropIfExists('karyawan');
}
};
```

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
/**
* Run the migrations.
*/
public function up(): void
{
Schema::create('kompetensis', function (Blueprint $table) {
$table->id();
$table->string('kompetensi');
$table->timestamps();
});
}
```

```php
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('kompetensi');
    }
};
```

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('penilaians', function (Blueprint $table) {
            $table->id();
            $table->unsignedBigInteger('karyawan_id');
```

```php
$table->foreign('karyawan_id')->references('id')->on('karyawan
s')->onDelete('cascade');

$table->unsignedBigInteger('kompetensi_id');

$table->foreign('kompetensi_id')->references('id')->on('kompet
ensis');

$table->integer('nilai');

$table->timestamps();

});

}


/**
* Reverse the migrations.
*/
public function down(): void
{
Schema::dropIfExists('penilaian');
}
};
```

File model

```php
<?php


namespace App\Models;


use App\Models\Karyawan;

use Illuminate\Database\Eloquent\Model;
```

```php
use Illuminate\Database\Eloquent\Factories\HasFactory;

class Divisi extends Model
{
use HasFactory;
public function karyawan()
{
return $this->hasMany(Karyawan::class);
}
}
```

```php
<?php

namespace App\Models;

use App\Models\Divisi;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Factories\HasFactory;

class Karyawan extends Model
{
use HasFactory;
public function divisi()
{
return $this->belongsTo(Divisi::class);
```

```php
}

public function penilaian()

{

return $this->hasMany(Penilaian::class);

}

}
```

```php
<?php


namespace App\Models;


use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;


class Kompetensi extends Model

{

use HasFactory;

public function penilaian()

{

return $this->hasMany(Penilaian::class);

}

}
```

```php
<?php
```

```php
namespace App\Models;


use App\Models\Karyawan;

use App\Models\Kompetensi;

use Illuminate\Database\Eloquent\Model;

use Illuminate\Database\Eloquent\Factories\HasFactory;


class Penilaian extends Model

{

use HasFactory;

public function karyawan()

{

return $this->belongsTo(Karyawan::class);

}

public function kompetensi()

{

return $this->belongsTo(Kompetensi::class);

}

}
```

File DatabaseSeeder

```php
<?php


namespace Database\Seeders;


// use Illuminate\Database\Console\Seeds\WithoutModelEvents;
```

```php
use Illuminate\Database\Seeder;

use Illuminate\Support\Facades\DB;


class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     */
    public function run(): void
    {
        // \App\Models\User::factory(10)->create();


        // \App\Models\User::factory()->create([
        //     'name' => 'Test User',
        //     'email' => 'test@example.com',
        // ]);
        DB::table('divisis')->insert([
            'nama_divisi' => "HRD"
        ]);
        DB::table('divisis')->insert([
            'nama_divisi' => "IT"
        ]);
        DB::table('divisis')->insert([
            'nama_divisi' => "Pemasaran"
        ]);
```

```php
DB::table('kompetensis')->insert([

'kompetensi' => "Kehadiran"

]);

DB::table('kompetensis')->insert([

'kompetensi' => "Tugas Selesai"

]);

DB::table('kompetensis')->insert([

'kompetensi' => "Kejujuran"

]);

}

}
```