Controle 2 laravel

Hasnaa lemalmi

1. Modèle

a-) php artisan make:model Stagiaire

b-) class Stagiaire extends Model

{

    protected $fillable=[

        'prenom','nom','email','domaine','date\_debut' , 'date\_fin'

    ];

}

2. Migration

a-)php artisan make:migration create\_stagiaires\_table --create=stagiaires

b-)  public function up(): void

    {

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

            $table->increments('id');

            $table->string('prenom', 255);

            $table->string('nom', 255);

            $table->string('email', 255)->unique();

            $table->string('domaine', 255);

            $table->date('date\_debut');

            $table->date('date\_fin');

            $table->timestamps();

        });

c-) php artisan migrate || php artisan migrate:rollback

3. Seeders

a-) php artisan make:seeder StagiairesTableSeeder

b-) use Illuminate\Support\Facades\DB;

 public function run(): void

    {

   DB::table('stagiaires')->insert([

'prenom' => 'hasnaa',

 'nom' => 'lemalmi',

 'email' => 'hasnaa.doe@example.com',

 'domaine' => 'Informatique',

 'date\_debut' => '2023-01-01',

 'date\_fin' => '2023-06-30',

        ]);

    }

c-) php artisan db:seed --class=StagiairesTableSeeder

si non ==> php artisan db :seed

4. Factory

a-) php artisan make:factory StagiaireFactory --model=Stagiaire

b-) use App\Models\Stagiaire;

class StagiaireFactory extends Factory

{

    protected $model = Stagiaire::class;

    public function definition(): array

    {

        return [

        'nom' => $this->faker->lastName,

        'prenom' => $this->faker->firstName,

        'email' => $this->faker->unique()->safeEmail,

        'domaine' => $this->faker->randomElement(['Informatique', 'Finance','Marketing','formateur','doctor']),

        'date\_debut' => $this->faker->date(),

        'date\_fin' => $this->faker->date(),

        ];

    }

}

use Dtabase\Seeders\DtabaseSeeder;

use Database\Factories\StagiaireFactory;

 public function run(): void

    {

        for ($i = 0; $i < 30; $i++) {

            $stagiaire = StagiaireFactory::new()->create();

        }

    }

5. Contrôleur

a=> php artisan make:controller StagiaireController --resource --model=Stagiaire

b=> use StagiaireController

    public function index()

    {

        $stagiaires = Stagiaire::all();

 return view('stagiaires.index', compact('stagiaires'));

    }

6. Routes

A=) wep.php

B=) use App\Http\Controllers\StagiaireController;

Route::get('/stagiaires', [StagiaireController::class, 'index'])->name('stagiaires.index');

7. Vue

A=) resource/view/stagiaires/index.blade.php

B=) <table class='table table-striped'>

    <thead>

    <tr>

        <th>ID</th>

       <th>nom</th>

        <th>prenom</th>

        <th>domaine</th>

        <th>email</th>

        <th>Date début </th>

        <th>date de fin </th>

        <th>action</th>

    </tr>

</thead>

<tbody>

@foreach($stagiaires as $stagiaire)

<tr>

    <td>{{$stagiaire->id}}</td>

    <td>{{$stagiaire->nom}}</td>

    <td>{{$stagiaire->prenom}}</td>

    <td>{{$stagiaire->domaine}}</td>

    <td>{{$stagiaire->email}}</td>

    <td>{{$stagiaire->date\_debut}}</td>

    <td>{{$stagiaire->date\_fin}}</td>

     <td>

     <a href="{{route('stagiaires.edit',$stagiaire->id)}}">edit</a>

                   <form method="post" action="{{route('stagiaires.destroy', $stagiaire->id)}}"  enctype="multipart/form-data">

                    @csrf

                    @method('DELETE')

                    <button type="submit" class="btn btn-danger">Delete</button>

                    </form>

                  </td>

</tr>

@endforeach

</tbody>

</table>

8. Configuration du fichier .env

A=) configuration de l’enronnement

= securite

= flexbilite

=le fichier .env joue un role cle dans la configuration et la gestion des variables d’environnement specifiques a votre project laravel . il offre flexibilite securite et facilite le deploiment sur differentsenvironnements.

B=) DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE= entreprise\_stagiaires

DB\_USERNAME=root

DB\_PASSWORD=

9. Table et modèle des missions

A=) php artisan make:model Mission –migration

B=)  Schema::create('missions', function (Blueprint $table) {

            $table->increments('id');

            $table->unsignedInteger('employe\_id');

            $table->string('nom',255);

            $table->text('description');

            $table->date('date\_debut');

            $table->date('date\_fin');

            $table->timestamps();

            $table->foreign('employe\_id')->references('id')->on('stagiaires')->onDelete('cascade');

        });

C=)    protected $fillable = [

    'employe\_id',

    'nom',

    'description',

    'date\_debut',

    'date\_fin'

   ];

   public function stagiaire(){

    return $this->belongsTo(Stagiaire::class, 'employe\_id');

}

D=) use model Stagiaire

public function missions(){

    return $this->hasMany(Mission::class, 'employe\_id');

}

E=) use controller missionsController

   public function showMissionsByStagiaire($employe\_id)

    {

        $stagiaire = Stagiaire::findOrFail($employe\_id);

        $missions = $stagiaire->missions;

        return view('missions.list' , compact('missions' , 'stagiaire'));

    }

Use page view\mission\list

<h1>liste des misions</h1>

<h1> Stagiaire : {{ $stagiaire ->nom}}</h1>

<table class='table table-striped'>

    <thead>

    <tr>

       <th>nom</th>

        <th>description</th>

        <th>Date début </th>

        <th>date de fin </th>

    </tr>

</thead>

<tbody>

@foreach($missions as $mission)

<tr>

    <td>{{$mission->nom}}</td>

    <td>{{$mission->description}}</td>

    <td>{{$mission->date\_debut}}</td>

    <td>{{$mission->date\_fin}}</td>

</tr>

@endforeach

</tbody>

</table>

Use web.php

Route::get('/missions/{employe\_id}',[MissionController::class , 'showMissionsByStagiaire'])->name('missions.showByStagiaire');

10. Authentification

A=) composer require Laravel\breeze –dev

B=) login register logout forgot-password reset-password

C=) php artisan breeze :install