

# Dossier 3 : Développement Back-end (Laravel)

Solution Documentée

Juin 2025

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Commandes de Création des Migrations</b>	<b>2</b>
<b>3</b>	<b>Migration de la Table Séminaires</b>	<b>2</b>
<b>4</b>	<b>Commandes de Création des Modèles</b>	<b>2</b>
<b>5</b>	<b>Modèle Séminaire avec Relations</b>	<b>2</b>
<b>6</b>	<b>Contrôleur de Ressource SéminaireController</b>	<b>3</b>
<b>7</b>	<b>Vue index.blade.php</b>	<b>4</b>
<b>8</b>	<b>Vue show.blade.php</b>	<b>5</b>
<b>9</b>	<b>Routes pour SéminaireController</b>	<b>6</b>
<b>10</b>	<b>Conclusion</b>	<b>6</b>

# 1 Introduction

Ce document présente une solution complète pour le Dossier 3, axée sur le développement back-end avec Laravel. Il couvre les migrations, modèles, relations, contrôleur, vues et routes nécessaires pour gérer les séminaires.

## 2 Commandes de Création des Migrations

Les commandes suivantes créent les fichiers de migration pour les tables nécessaires :

```
1 php artisan make:migration create_animateurs_table
2 php artisan make:migration create_seminaires_table
3 php artisan make:migration create_activites_table
```

## 3 Migration de la Table Séminaires

La méthode `up()` de la migration pour la table `seminaires` est définie comme suit :

```
1 public function up()
2 {
3     Schema::create('seminaires', function (Blueprint $table) {
4         $table->id();
5         $table->string('theme');
6         $table->date('date_debut');
7         $table->date('date_fin');
8         $table->text('description');
9         $table->decimal('cout_journalier', 10, 2);
10        $table->unsignedBigInteger('animateur_id');
11        $table->timestamps();
12
13        $table->foreign('animateur_id')
14            ->references('id')
15            ->on('animateurs')
16            ->onDelete('cascade');
17    });
18 }
```

## 4 Commandes de Création des Modèles

Les modèles sont créés avec les commandes suivantes :

```
1 php artisan make:model Animateur
2 php artisan make:model Seminaire
3 php artisan make:model Activite
```

## 5 Modèle Séminaire avec Relations

Le modèle `Seminaire` inclut les relations avec `Animateur` et `Activite` :

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Seminaire extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'theme', 'date_debut', 'date_fin', 'description',
14         'cout_journalier', 'animateur_id'
15     ];
16
17     // Relation One To Many (inverse) avec Animateur
18     public function animateur()
19     {
20         return $this->belongsTo(Animateur::class);
21     }
22
23     // Relation One To Many avec Activité
24     public function activites()
25     {
26         return $this->hasMany(Activite::class);
27     }
28 }

```

## 6 Contrôleur de Ressource SéminaireController

Le contrôleur `SeminaireController` gère les opérations CRUD pour les séminaires :

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Seminaire;
6 use Illuminate\Http\Request;
7
8 class SeminaireController extends Controller
9 {
10     // a) Méthode index()
11     public function index()
12     {
13         $seminaires = Seminaire::with('animateur')->get();
14         return view('viewed.seminaires.index', compact('seminaires'));
15     }
16
17     // b) Méthode show()
18     public function show($id)
19     {
20         $seminaire = Seminaire::with(['animateur', 'activites'])->
21         findOrFail($id);
22         return view('viewed.seminaires.show', compact('seminaire'));
23     }
24 }

```

```

24 // c) Méthode destroy()
25 public function destroy($id)
26 {
27     $seminaire = Seminaire::findOrFail($id);
28     $seminaire->delete();
29
30     return redirect()->route('seminaires.index')
31         ->with('success', 'Séminaire supprimé avec
succès');
32 }
33 }

```

## 7 Vue index.blade.php

La vue index.blade.php affiche la liste des séminaires :

```

1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container">
5     <h1>Liste des Séminaires</h1>
6
7     <table class="table table-bordered">
8         <thead>
9             <tr>
10                 <th>Tâcher</th>
11                 <th>Date début</th>
12                 <th>Date fin</th>
13                 <th>Description</th>
14                 <th>Côté par jour</th>
15                 <th>Admission</th>
16                 <th>Actions</th>
17             </tr>
18         </thead>
19         <tbody>
20             @foreach($seminaires as $seminaire)
21                 <tr>
22                     <td>{{ $seminaire->theme }}</td>
23                     <td>{{ $seminaire->date_debut }}</td>
24                     <td>{{ $seminaire->date_fin }}</td>
25                     <td>{{ $seminaire->description }}</td>
26                     <td>{{ number_format($seminaire->cout_journalier, 0,
',', ' ') }}</td>
27                     <td>{{ $seminaire->animateur->nomAnim }}</td>
28                     <td>
29                         <a href="{{ route('seminaires.show', $seminaire->id
) }}" class="btn btn-info">Consulter</a>
30                         <a href="#" class="btn btn-warning">Modifier</a>
31                         <form action="{{ route('seminaires.destroy',
$seminaire->id) }}" method="POST" style="display:inline;">
32                             @csrf
33                             @method('DELETE')
34                             <button type="submit" class="btn btn-danger">
Supprimer</button>
35                         </form>
36                     </td>
37                 </tr>

```

```

38         @endforeach
39     </tbody>
40 </table>
41 </div>
42 @endsection

```

## 8 Vue show.blade.php

La vue show.blade.php affiche les détails d'un séminaire :

```

1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container">
5     <h1>Détails du séminaire : {{ $seminaire->id }}</h1>
6
7     <table class="table table-bordered">
8         <tr>
9             <th>Tâcher</th>
10            <td>{{ $seminaire->theme }}</td>
11        </tr>
12        <tr>
13            <th>Date début</th>
14            <td>{{ $seminaire->date_debut }}</td>
15        </tr>
16        <tr>
17            <th>Date fin</th>
18            <td>{{ $seminaire->date_fin }}</td>
19        </tr>
20        <tr>
21            <th>Description</th>
22            <td>{{ $seminaire->description }}</td>
23        </tr>
24        <tr>
25            <th>Côté par jour</th>
26            <td>{{ number_format($seminaire->cout_journalier, 0, ',', '
27        </td>
28        </tr>
29        <tr>
30            <th>Animateur</th>
31            <td>{{ $seminaire->animateur->nomAnim }} {{ $seminaire->
32            animateur->prenomAnim }}</td>
33        </tr>
34    </table>
35
36    <h2>Liste des activités assurées :</h2>
37
38    <table class="table table-bordered">
39        <thead>
40            <tr>
41                <th>Nom de l'activité</th>
42                <th>Description de l'activité</th>
43            </tr>
44        </thead>
45        <tbody>
46            @foreach($seminaire->activites as $activite)
47                <tr>

```

```

46         <td>{{ $activite->nomActivite }}</td>
47         <td>{{ $activite->descriptionActivite }}</td>
48     </tr>
49     @endforeach
50 </tbody>
51 </table>
52
53 <a href="{{ route('seminaires.index') }}" class="btn btn-primary">
Retour à la liste</a>
54 </div>
55 @endsection

```

## 9 Routes pour SéminaireController

Les routes pour le contrôleur `SeminaireController` sont définies comme suit :

```

1 use App\Http\Controllers\SeminaireController;
2
3 Route::resource('seminaires', SeminaireController::class);

```

## 10 Conclusion

Cette solution fournit une implémentation complète pour la gestion des séminaires dans une application Laravel, respectant les bonnes pratiques de développement et les conventions du framework.