

Solution Laravel pour l'application des marchés du Bénin by Malthus AMETEPE

Table des matières

1. [Structure du projet](#)
 2. [Création du projet Laravel](#)
 3. [Configuration de la base de données](#)
 4. [Création des migrations](#)
 5. [Création des modèles](#)
 6. [Création des Seeders](#)
 7. [Configuration des Routes](#)
 8. [Création des Contrôleurs](#)
 9. [Création des Vues](#)
 10. [Instructions pour mettre en place et exécuter le projet](#)
-

Structure du projet

Le projet est organisé avec la structure suivante :

```
marches-benin/  
├── app/  
│   ├── Models/  
│   │   ├── Ville.php  
│   │   ├── Marche.php  
│   │   ├── Equipement.php  
│   │   └── TypeEquipement.php  
│   ├── Http/  
│   │   ├── Controllers/  
│   │   │   ├── MarcheController.php  
│   │   │   └── HomeController.php  
├── database/  
│   ├── migrations/  
│   │   ├── create_villes_table.php  
│   │   ├── create_marches_table.php  
│   │   ├── create_type_equipements_table.php  
│   │   └── create_equipements_table.php  
│   └── seeders/  
│       ├── VilleSeeder.php  
│       └── MarcheSeeder.php
```

```
|— resources/
|   |— views/
|   |   |— layouts/
|   |   |   |— app.blade.php
|   |   |   |— marches/
|   |   |       |— index.blade.php
|   |   |       |— create.blade.php
|   |   |       |— home.blade.php
|— routes/
|   |— web.php
```

1. Création du projet Laravel

```
composer create-project laravel/laravel marches-benin
cd marches-benin
```

2. Configuration de la base de données

Modifiez le fichier `.env` pour configurer la connexion à la base de données :

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=marchesBenin
DB_USERNAME=root
DB_PASSWORD=
```

3. Création des migrations

Migration pour la table Villes

```
<?php

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

return new class extends Migration
```

```
{  
    public function up(): void  
    {  
        Schema::create('villes', function (Blueprint $table) {  
            $table->id();  
            $table->string('nom');  
            $table->string('pays')->default('Bénin');  
            $table->timestamps();  
        });  
    }  
  
    public function down(): void  
    {  
        Schema::dropIfExists('villes');  
    }  
};
```

Migration pour la table Marches

```
<?php
```

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

```
return new class extends Migration
{
    public function up(): void
    {
        Schema::create('marches', function (Blueprint $table) {
            $table->id();
            $table->string('nom');
            $table->text('description')->nullable();
            $table->integer('capacite');
            $table->string('adresse')->nullable();
            $table->string('telephone')->nullable();
            $table->string('email')->nullable();
            $table->string('image')->nullable();
            $table->foreignId('ville_id')->constrained()->onDelete('cascade');
            $table->timestamps();
        });
    }

    public function down(): void
    {
        Schema::dropIfExists('marches');
    }
};
```

Migration pour la table TypeEquipements

```
<?php
```

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

```
return new class extends Migration
```

```
{
```

```
    public function up(): void
```

```
    {
```

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

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

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

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

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

```
        });
```

```
    }
```

```
    public function down(): void
```

```
    {
```

```
        Schema::dropIfExists('type_equipements');
```

```
}  
};
```

Migration pour la table Equipements

```
<?php
```

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

```
return new class extends Migration  
{  
    public function up(): void  
    {  
        Schema::create('equipements', function (Blueprint $table) {  
            $table->id();  
            $table->string('nom');  
            $table->text('description')->nullable();  
            $table->integer('quantite')->default(1);  
            $table->foreignId('marche_id')->constrained()->onDelete('cascade');  
            $table->foreignId('type_equipement_id')->constrained()->onDelete('cascade');  
            $table->timestamps();  
        });  
    }  
};
```



```
}

public function down(): void
{
    Schema::dropIfExists('equipements');
}
};
```

4. Création des modèles

Modèle Ville

```
<?php

namespace App\Models;

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

class Ville extends Model
```

```
{  
    use HasFactory;  
  
    protected $fillable = ['nom', 'pays'];  
  
    public function marches(): HasMany  
    {  
        return $this->hasMany(Marche::class);  
    }  
}
```

Modèle Marche

```
<?php  
  
namespace App\Models;  
  
use Illuminate\Database\Eloquent\Factories\HasFactory;  
use Illuminate\Database\Eloquent\Model;  
use Illuminate\Database\Eloquent\Relations\BelongsTo;  
use Illuminate\Database\Eloquent\Relations\HasMany;  
  
class Marche extends Model  
{
```

```
use HasFactory;
```

```
protected $fillable = [  
    'nom',  
    'description',  
    'capacite',  
    'adresse',  
    'telephone',  
    'email',  
    'image',  
    'ville_id'  
];
```

```
public function ville(): BelongsTo  
{  
    return $this->belongsTo(Ville::class);  
}
```

```
public function equipements(): HasMany  
{  
    return $this->hasMany(Equipement::class);  
}
```

```
}
```

Modèle TypeEquipement

```
<?php

namespace App\Models;

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

class TypeEquipement extends Model
{
    use HasFactory;

    protected $fillable = ['nom', 'description'];

    public function equipements(): HasMany
    {
        return $this->hasMany(Equipement::class);
    }
}
```

Modèle Equipement

```
<?php
```

```
namespace App\Models;
```

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

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

```
use Illuminate\Database\Eloquent\Relations\BelongsTo;
```

```
class Equipement extends Model
```

```
{
```

```
    use HasFactory;
```

```
    protected $fillable = [
```

```
        'nom',
```

```
        'description',
```

```
        'quantite',
```

```
        'marche_id',
```

```
        'type_equipement_id'
```

```
    ];
```

```
    public function marche(): BelongsTo
```

```
    {
```

```
        return $this->belongsTo(Marche::class);
```

```
    }
```

```
public function typeEquipement(): BelongsTo
{
    return $this->belongsTo(TypeEquipement::class);
}
}
```

5. Création des Seeders

VilleSeeder

```
<?php

namespace Database\Seeders;

use App\Models\Ville;
use Illuminate\Database\Seeder;

class VilleSeeder extends Seeder
{
    public function run(): void
    {
```

```
$villes = [  
  ['nom' => 'Cotonou'],  
  ['nom' => 'Porto-Novo'],  
  ['nom' => 'Abomey'],  
  ['nom' => 'Parakou'],  
  ['nom' => 'Natitingou'],  
  ['nom' => 'Lokossa'],  
  ['nom' => 'Ouidah'],  
  ['nom' => 'Bohicon'],  
  ['nom' => 'Djougou'],  
  ['nom' => 'Kandi'],  
  ['nom' => 'Abomey-Calavi'],  
  ['nom' => 'Dassa-Zoumé']  
];  
  
foreach ($villes as $ville) {  
  Ville::create($ville);  
}  
}  
}
```

MarcheSeeder

```
<?php
```

```
namespace Database\Seeders;
```

```
use App\Models\Marche;
```

```
use App\Models\Ville;
```

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

```
class MarcheSeeder extends Seeder
```

```
{
```

```
    public function run(): void
```

```
    {
```

```
        // Récupérer les ID des villes
```

```
        $cotonouId = Ville::where('nom', 'Cotonou')->first()->id;
```

```
        $portoNovoId = Ville::where('nom', 'Porto-Novo')->first()->id;
```

```
        // Marchés de Gbégamey, Ménontin, Wologuèdé à Cotonou
```

```
        $marches = [
```

```
            [
```

```
                'nom' => 'Gbégamey',
```

```
                'description' => 'Marché urbain de Gbégamey situé dans la ville de Cotonou',
```

```
                'capacite' => 1000,
```

```
                'adresse' => 'Quartier Gbégamey, Cotonou',
```

```
                'telephone' => '+229 97000001',
```

```
                'email' => 'gbegamey@marches.bj',
```



```
        'image' => 'gbegamey.jpg',
        'ville_id' => $cotonouId
    ],
    [
        'nom' => 'Ménontin',
        'description' => 'Marché urbain de Ménontin situé dans la ville de Cotonou',
        'capacite' => 800,
        'adresse' => 'Quartier Ménontin, Cotonou',
        'telephone' => '+229 97000002',
        'email' => 'menontin@marches.bj',
        'image' => 'menontin.jpg',
        'ville_id' => $cotonouId
    ],
    [
        'nom' => 'Wologuèdé',
        'description' => 'Marché urbain de Wologuèdé situé dans la ville de Cotonou',
        'capacite' => 500,
        'adresse' => 'Quartier Wologuèdé, Cotonou',
        'telephone' => '+229 97000003',
        'email' => 'wolguede@marches.bj',
        'image' => 'wolguede.jpg',
        'ville_id' => $cotonouId
    ],
    // Marchés mentionnés dans l'énoncé
    [
        'nom' => 'Ganhi',
```

```
'description' => 'Marché central de Ganhi',
'capacite' => 1200,
'adresse' => 'Centre-ville de Cotonou',
'telephone' => '+229 97000004',
'email' => 'ganhi@marches.bj',
'image' => 'market1.jpg',
'ville_id' => $cotonouId
],
[
  'nom' => 'Cadjèhoun',
  'description' => 'Marché de Cadjèhoun',
  'capacite' => 600,
  'adresse' => 'Quartier Cadjèhoun, Cotonou',
  'telephone' => '+229 97000005',
  'email' => 'cadjehoun@marches.bj',
  'image' => 'market2.jpg',
  'ville_id' => $cotonouId
],
[
  'nom' => 'Tokplégbé',
  'description' => 'Marché de Tokplégbé',
  'capacite' => 450,
  'adresse' => 'Quartier Tokplégbé, Cotonou',
  'telephone' => '+229 97000006',
  'email' => 'tokplegbe@marches.bj',
  'image' => 'market3.jpg',
```

```
    'ville_id' => $cotonouId
  ],
  [
    'nom' => 'Akpakpa',
    'description' => 'Marché d\'Akpakpa',
    'capacite' => 900,
    'adresse' => 'Quartier Akpakpa, Cotonou',
    'telephone' => '+229 97000007',
    'email' => 'akpakpa@marches.bj',
    'image' => 'market4.jpg',
    'ville_id' => $cotonouId
  ],
  [
    'nom' => 'Madombo',
    'description' => 'Marché de Madombo',
    'capacite' => 350,
    'adresse' => 'Quartier Madombo, Porto-Novo',
    'telephone' => '+229 97000008',
    'email' => 'madombo@marches.bj',
    'image' => 'market5.jpg',
    'ville_id' => $portoNovoId
  ],
  [
    'nom' => 'Sainte Trinité',
    'description' => 'Marché Sainte Trinité',
    'capacite' => 550,
```

```

        'adresse' => 'Quartier Sainte Trinité, Porto-Novo',
        'telephone' => '+229 97000009',
        'email' => 'saintetrinite@marches.bj',
        'image' => 'market6.jpg',
        'ville_id' => $portoNovoId
    ],
    [
        'nom' => 'PK3 route de Porto-Novo',
        'description' => 'Marché PK3 sur la route de Porto-Novo',
        'capacite' => 400,
        'adresse' => 'PK3 Route de Porto-Novo',
        'telephone' => '+229 97000010',
        'email' => 'pk3@marches.bj',
        'image' => 'market7.jpg',
        'ville_id' => $portoNovoId
    ]
];

foreach ($marches as $marche) {
    Marche::create($marche);
}
}
}

```

6. Mise à jour du fichier DatabaseSeeder

```
<?php

namespace Database\Seeders;

use Illuminate\Database\Seeder;

class DatabaseSeeder extends Seeder
{
    public function run(): void
    {
        $this->call([
            VilleSeeder::class,
            MarcheSeeder::class,
        ]);
    }
}
```

7. Configuration des Routes

```
<?php
```

```
use App\Http\Controllers\HomeController;  
use App\Http\Controllers\MarcheController;  
use Illuminate\Support\Facades\Route;
```

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

```
Route::get('/marches', [MarcheController::class, 'index'])->name('marches.index');  
Route::get('/marches/create', [MarcheController::class, 'create'])->name('marches.create');  
Route::post('/marches', [MarcheController::class, 'store'])->name('marches.store');  
Route::get('/marches/{marche}', [MarcheController::class, 'show'])->name('marches.show');
```

8. Création des Contrôleurs

HomeController

```
<?php
```

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

```
use App\Models\Marche;  
use Illuminate\Http\Request;  
  
class HomeController extends Controller  
{  
    public function index()  
    {  
        // Rediriger vers la liste des marchés  
        return redirect()->route('marches.index');  
    }  
}
```

MarcheController

```
<?php  
  
namespace App\Http\Controllers;  
  
use App\Models\Marche;  
use App\Models\Ville;  
use Illuminate\Http\Request;  
use Illuminate\Support\Facades\Storage;
```

```
class MarcheController extends Controller
{
    public function index()
    {
        $marches = Marche::with('ville')->get();
        return view('marches.index', compact('marches'));
    }

    public function create()
    {
        $villes = Ville::all();
        return view('marches.create', compact('villes'));
    }

    public function store(Request $request)
    {
        $validated = $request->validate([
            'nom' => 'required|string|max:255',
            'description' => 'nullable|string',
            'capacite' => 'required|integer|min:1',
            'adresse' => 'nullable|string|max:255',
            'telephone' => 'nullable|string|max:20',
            'email' => 'nullable|email|max:255',
            'ville_id' => 'required|exists:villes,id',
            'image' => 'nullable|image|max:2048', // max 2MB
        ]);
    }
}
```



```
if ($request->hasFile('image')) {  
    $path = $request->file('image')->store('marches', 'public');  
    $validated['image'] = basename($path);  
}  
  
$marche = Marche::create($validated);  
  
return redirect()->route('marches.index')  
    ->with('success', 'Marché créé avec succès!');  
}  
  
public function show(Marche $marche)  
{  
    return view('marches.show', compact('marche'));  
}  
}
```

9. Création des Vues

Layout principal (app.blade.php)

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>@yield('title', 'Marchés du Bénin')</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="styl
  <style>
    body {
      padding-top: 20px;
    }
    .card-img-top {
      height: 200px;
      object-fit: cover;
    }
  </style>
</head>
<body>
  <div class="container">
    <header class="mb-4">
      <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
        <div class="container-fluid">
          <a class="navbar-brand" href="{{ route('home') }}">Marchés du Bénin</a>
          <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs
            <span class="navbar-toggler-icon"></span>

```

```

        </button>
        <div class="collapse navbar-collapse" id="navbarNav">
            <ul class="navbar-nav">
                <li class="nav-item">
                    <a class="nav-link" href="{{ route('marches.index') }}">Liste des
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{ route('marches.create') }}">Ajouter u
                </li>
            </ul>
        </div>
    </div>
</nav>
</header>

<main>
    @if(session('success'))
        <div class="alert alert-success">
            {{ session('success') }}
        </div>
    @endif

    @yield('content')
</main>

<footer class="mt-5 pt-3 border-top text-center text-muted">

```

```
        <p>&copy; {{ date('Y') }} - Application pour les marchés du Bénin</p>
    </footer>
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></s
</body>
</html>
```

Index des marchés (index.blade.php)

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

@section('title', 'Liste des marchés du Bénin')

@section('content')
    <div class="row mb-4">
        <div class="col">
            <h1>Liste des marchés du Bénin</h1>
            <p class="lead">Découvrez les différents marchés construits dans le cadre du programme
        </div>
    </div>

    <div class="row row-cols-1 row-cols-md-3 g-4">
        @foreach($marches as $marche)
```

```

<div class="col">
    <div class="card h-100">
        @if($marche->image)
            
            <h5 class="card-title">{{ $marche->nom }}</h5>
            <h6 class="card-subtitle mb-2 text-muted">{{ $marche->ville->nom }}</h6>
            <p class="card-text">{{ \Illuminate\Support\Str::limit($marche->descriptio
            <p class="card-text"><small class="text-muted">Capacité: {{ $marche->capac
        </div>
        <div class="card-footer">
            <a href="{{ route('marches.show', $marche) }}" class="btn btn-sm btn-prima
        </div>
    </div>
</div>
@endforeach
</div>
@endsection

```

Formulaire de création (create.blade.php)

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

@section('title', 'Ajouter un marché')

@section('content')
    <div class="row mb-4">
        <div class="col">
            <h1>Ajouter un nouveau marché</h1>
            <p class="lead">Remplissez le formulaire ci-dessous pour ajouter un nouveau marché à l
        </div>
    </div>

    <div class="row">
        <div class="col-md-8">
            <div class="card">
                <div class="card-body">
                    <form action="{{ route('marches.store') }}" method="POST" enctype="multipart/f
                        @csrf

                    <div class="mb-3">
                        <label for="nom" class="form-label">Nom du marché *</label>
                        <input type="text" class="form-control @error('nom') is-invalid @enderror
                            @error('nom')
                            <div class="invalid-feedback">{{ $message }}</div>
                        @enderror
```

```
</div>
```

```
<div class="mb-3">
```

```
  <label for="ville_id" class="form-label">Ville *</label>
```

```
  <select class="form-select @error('ville_id') is-invalid @enderror" id
```

```
    <option value="">Sélectionnez une ville</option>
```

```
    @foreach($villes as $ville)
```

```
      <option value="{{ $ville->id }}" {{ old('ville_id') == $ville-  
        {{ $ville->nom }}
```

```
      </option>
```

```
    @endforeach
```

```
  </select>
```

```
  @error('ville_id')
```

```
    <div class="invalid-feedback">{{ $message }}</div>
```

```
  @enderror
```

```
</div>
```

```
<div class="mb-3">
```

```
  <label for="description" class="form-label">Description *</label>
```

```
  <textarea class="form-control @error('description') is-invalid @enderr
```

```
  @error('description')
```

```
    <div class="invalid-feedback">{{ $message }}</div>
```

```
  @enderror
```

```
</div>
```

```
<div class="mb-3">
```

```
<label for="capacite" class="form-label">Capacité du marché *</label>
<input type="number" class="form-control @error('capacite') is-invalid
@error('capacite')
    <div class="invalid-feedback">{{ $message }}</div>
@enderror
</div>

<div class="mb-3">
    <label for="adresse" class="form-label">Adresse</label>
    <input type="text" class="form-control @error('adresse') is-invalid @e
@error('adresse')
        <div class="invalid-feedback">{{ $message }}</div>
    @enderror
</div>

<div class="mb-3">
    <label for="telephone" class="form-label">Téléphone</label>
    <input type="text" class="form-control @error('telephone') is-invalid
@error('telephone')
        <div class="invalid-feedback">{{ $message }}</div>
    @enderror
</div>

<div class="mb-3">
    <label for="email" class="form-label">Email</label>
    <input type="email" class="form-control @error('email') is-invalid @en
```



```

        @error('email')
        <div class="invalid-feedback">{{ $message }}</div>
    @enderror
</div>

<div class="mb-3">
    <label for="image" class="form-label">Image</label>
    <input type="file" class="form-control @error('image') is-invalid @enderror" @error('image')
        <div class="invalid-feedback">{{ $message }}</div>
    @enderror
</div>

<div class="d-flex justify-content-between">
    <button type="submit" class="btn btn-primary">Enregistrer</button>
    <a href="{{ route('marches.index') }}" class="btn btn-secondary">Annuler</a>
</div>
</form>
</div>
</div>

<div class="col-md-4">
    <div class="card">
        <div class="card-header bg-info text-white">
            Guide
        </div>
    </div>
</div>

```

```

        </div>
        <div class="card-body">
            <p>Remplissez tous les champs obligatoires marqués par un astérisque (*).</p>
            <p>Pour l'image, vous pouvez télécharger une photo du marché au format JPG, PNG</p>
            <p>Assurez-vous que les informations saisies sont correctes avant de soumettre</p>
        </div>
    </div>
</div>
</div>
@endsection

```

Vue détaillée d'un marché (show.blade.php)

```

@extends('layouts.app')

@section('title', $marche->nom . ' - Détails du marché')

@section('content')
    <div class="row mb-4">
        <div class="col">
            <h1>{{ $marche->nom }}</h1>
            <p class="lead">Marché situé à {{ $marche->ville->nom }}</p>
        </div>
    </div>
</div>

```

```

<div class="row">
  <div class="col-md-8">
    <div class="card mb-4">
      @if($marche->image)
        
        <h5 class="card-title">Description</h5>
        <p class="card-text">{{ $marche->description }}</p>

        <h5 class="card-title mt-4">Informations</h5>
        <ul class="list-group list-group-flush">
          <li class="list-group-item"><strong>Capacité:</strong> {{ $marche->capacit
          <li class="list-group-item"><strong>Adresse:</strong> {{ $marche->adresse
          <li class="list-group-item"><strong>Téléphone:</strong> {{ $marche->teleph
          <li class="list-group-item"><strong>Email:</strong> {{ $marche->email ?? '
        </ul>
      </div>
    </div>
  </div>
</div>

<div class="col-md-4">
  <div class="card">

```

```

<div class="card-header bg-primary text-white">
    Localisation
</div>
<div class="card-body">
    <p><strong>Ville:</strong> {{ $marche->ville->nom }}</p>
    <p><strong>Pays:</strong> {{ $marche->ville->pays }}</p>

    <div class="d-grid gap-2 mt-3">
        <a href="{{ route('marches.index') }}" class="btn btn-outline-secondary">R
    </div>
</div>
</div>

<!-- Équipements si disponibles -->
@if($marche->equipements->count() > 0)
<div class="card mt-4">
    <div class="card-header bg-success text-white">
        Équipements
    </div>
    <div class="card-body">
        <ul class="list-group">
            @foreach($marche->equipements as $equipement)
                <li class="list-group-item d-flex justify-content-between align-items-cent
                    {{ $equipement->nom }}
                    <span class="badge bg-primary rounded-pill">{{ $equipement->quantite }}
                </li>
            @endforeach
        </ul>
    </div>
</div>

```

```
        @endforeach
      </ul>
    </div>
  </div>
@endif
</div>
</div>
@endsection
```

10. Instructions pour mettre en place et exécuter le projet

Configuration initiale du projet

1. Cloner le projet

```
git clone https://votre-repo/marches-benin.git
cd marches-benin
```

2. Installer les dépendances

```
composer install
```

3. Configurer le fichier d'environnement

```
cp .env.example .env  
php artisan key:generate
```

4. Modifier le fichier `.env` pour configurer la base de données

```
DB_CONNECTION=mysql  
DB_HOST=127.0.0.1  
DB_PORT=3306  
DB_DATABASE=marchesBenin  
DB_USERNAME=root  
DB_PASSWORD=
```

5. Créer la base de données

```
mysql -u root -p
```

Dans MySQL :

```
CREATE DATABASE marchesBenin;  
EXIT;
```

6. Exécuter les migrations et les seeders

```
php artisan migrate --seed
```

7. Créer le lien symbolique pour le stockage

```
php artisan storage:link
```

8. Démarrer le serveur de développement

```
php artisan serve
```

9. Accéder à l'application Ouvrez votre navigateur et allez à l'adresse : <http://localhost:8000>