

# Guide d'installation - Application Laravel

## Prérequis

Avant de commencer l'installation, assurez-vous d'avoir les éléments suivants installés sur votre système Windows :

### Logiciels requis

- **PHP 8.2 ou supérieur** avec les extensions suivantes :
  - BCMath
  - Ctype
  - Fileinfo
  - JSON
  - Mbstring
  - OpenSSL
  - PDO
  - Tokenizer
  - XML
  - SQLite
- **Composer** (gestionnaire de dépendances PHP)
- **Node.js** (version 18 ou supérieure) avec npm
- **Git** (optionnel, pour cloner le projet)

### Vérification des prérequis

Ouvrez une invite de commande (cmd) et vérifiez les versions installées :

```
bash

php --version
composer --version
node --version
npm --version
```

## Installation

### 1. Récupération du projet

Si vous avez reçu le projet en archive, décompressez-le dans le dossier de votre choix.

Si le projet est sur un dépôt Git :

```
bash
```

```
git clone [URL_DU_DEPOT]  
cd [NOM_DU_PROJET]
```

## 2. Installation des dépendances PHP

Dans le dossier racine du projet, exécutez :

```
bash
```

```
composer install
```

## 3. Installation des dépendances JavaScript

Toujours dans le dossier racine :

```
bash
```

```
npm install
```

## 4. Configuration de l'environnement

Copiez le fichier d'exemple de configuration :

```
bash
```

```
copy .env.example .env
```

Générez la clé d'application Laravel :

```
bash
```

```
php artisan key:generate
```

## 5. Configuration de la base de données

Le projet utilise SQLite. La base de données sera automatiquement créée lors de la migration.

Exécutez les migrations pour créer les tables :

```
bash
```

```
php artisan migrate
```

## 6. Insertion des données de test

Pour charger les utilisateurs de test, exécutez :

```
bash
```

```
php artisan db:seed
```

## 7. Compilation des assets

Compilez les fichiers CSS et JavaScript :

```
bash
```

```
npm run build
```

Pour le développement avec rechargement automatique :

```
bash
```

```
npm run dev
```

## 8. Lancement de l'application

Démarrez le serveur de développement Laravel :

```
bash
```

```
php artisan serve
```

L'application sera accessible à l'adresse : **<http://localhost:8000>**

## Utilisation

### Connexion à l'application

Après avoir exécuté le seeder, vous pouvez vous connecter avec les comptes utilisateurs de test suivants :

#### Utilisateur standard :

- Email :
- Mot de passe :

#### Administrateur :

- Email :
- Mot de passe :

## Navigation

L'application utilise Laravel Breeze qui fournit :

- Page de connexion
- Page d'inscription
- Page de profil utilisateur
- Système de réinitialisation de mot de passe

## Résolution des problèmes courants

### Erreur "Class not found"

```
bash
composer dump-autoload
```

### Erreur de permissions (si applicable)

Sur Windows, généralement pas de problème de permissions, mais assurez-vous que le dossier est accessible en écriture.

### Erreur de base de données

Vérifiez que le fichier `database/database.sqlite` existe. S'il n'existe pas :

```
bash
touch database/database.sqlite
php artisan migrate
```

### Assets non compilés

Si les styles ne s'affichent pas correctement :

```
bash
npm run build
```

## Commandes utiles

### Réinitialiser la base de données

```
bash
php artisan migrate:fresh --seed
```

### Vider le cache

```
bash
```

```
php artisan cache:clear  
php artisan config:clear  
php artisan route:clear  
php artisan view:clear
```

## Redémarrer le serveur de développement

```
bash
```

```
php artisan serve
```

## Structure du projet

```
├─ app/                # Code de l'application  
├─ database/  
│   ├─ migrations/    # Migrations de base de données  
│   ├─ seeders/       # Seeders pour données de test  
│   └─ database.sqlite # Base de données SQLite  
├─ resources/  
│   ├─ views/         # Templates Blade  
│   ├─ css/           # Fichiers CSS  
│   └─ js/            # Fichiers JavaScript/Alpine.js  
├─ routes/            # Définition des routes  
├─ .env               # Configuration d'environnement  
└─ composer.json      # Dépendances PHP
```

## Support

En cas de problème lors de l'installation, vérifiez :

1. Que tous les prérequis sont correctement installés
2. Que vous êtes dans le bon dossier lors de l'exécution des commandes
3. Que les commandes sont exécutées dans l'ordre indiqué

Pour plus d'informations sur Laravel et Breeze, consultez la [documentation officielle Laravel](#).