Laravel Livewire CRUD Exemple01

1. Objectifs

- Comment effectuer l'opération CRUD **livewire** dans une application **laravel**.

2. Présentation

- **Livewire** est un framework **full-stack** pour le framework Laravel qui simplifie la création d'interfaces dynamiques, sans quitter le confort de Laravel.
- Si vous utilisez **livewire** avec laravel, alors vous ne vous souciez pas d'écrire du code jquery ajax, livewire vous aidera à écrire de manière très simple du code jquery ajax en utilisant php.
- Sans actualisation de la page, la validation laravel fonctionnera, le formulaire sera soumis, etc.
- Dans cet exemple, nous allons créer un module de **suppression** de **mise à jour** après **insertion**.
- Alors, suivons l'étape ci-dessous:
  - Téléchargez l'application Laravel
  - Créer des itinéraires
  - Créer un modèle et une migration
  - Installez Livewire
  - Créer un composant et une vue Livewire
  - Créer un fichier Blade
  - Exécuter le projet
- Vous obtiendrez la mise en page ci-dessous

3. Aperçu de notre application

- Nous allons créer une application CRUD en Laravel et Livewire. Il s'agit donc essentiellement d'une application CRUD sans que la page ne se recharge.
- Livewire gérera toutes les requêtes AJAX requises pour mettre à jour l'interface utilisateur.

4. Réalisation

1. **Créer un nouveau projet Laravel8**

apcpedagogie.com

- Dans la première étape, nous devons obtenir une nouvelle application de version laravel à l'aide de la commande ci-dessous. Alors ouvrons le terminal et exécutons la commande ci-dessous.

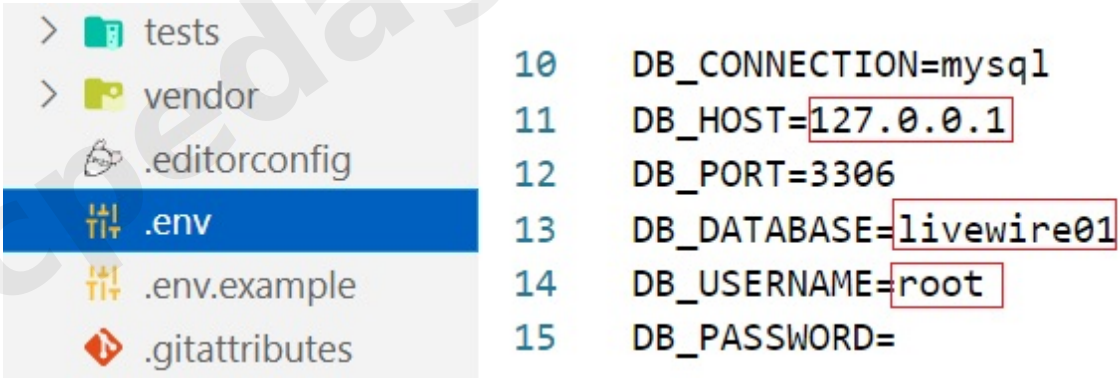
```
composer create-project --prefer-dist laravel/laravel livewire-projet01
```

2. Quand on installe Laravel, si on veut déjà un système d'**authentification complet**, on installe **Jetstream** qui va lui-même installer Livewire comme dépendance (à moins qu'on choisisse Inertia) :

```
composer require laravel/jetstream
php artisan jetstream:install livewire
npm install
npm run dev
php artisan migrate
```

3. **Configurer la base de données**

- Pour configurer la base de données. Ouvrir le fichier "`.env`"
- changer le nom de la base de données (livewire01), le nom d'utilisateur et le mot de passe.



4. **Créer une migration et un modèle**

- Ici, créer une migration de base de données pour la table "**users**" et créer un modèle pour la table "**User**".

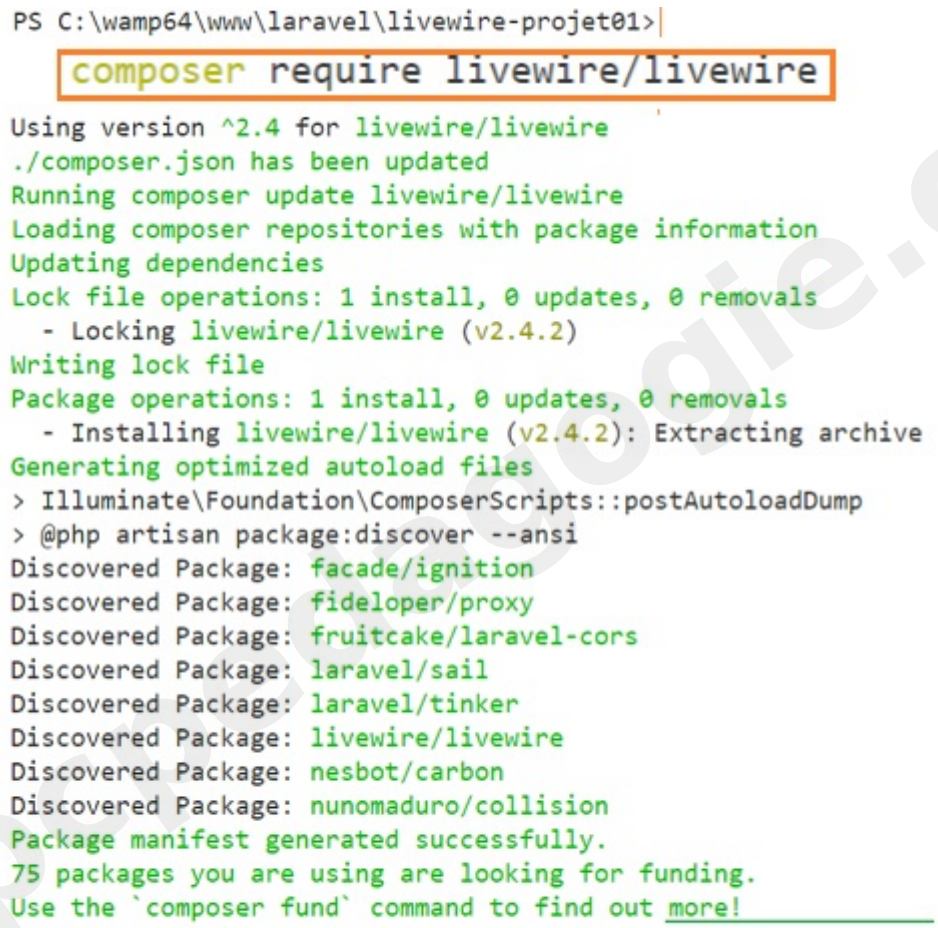
```
php artisan make:migration create_users_table --create=users
```

apcpedagogie.com

5. **Installer Livewire**

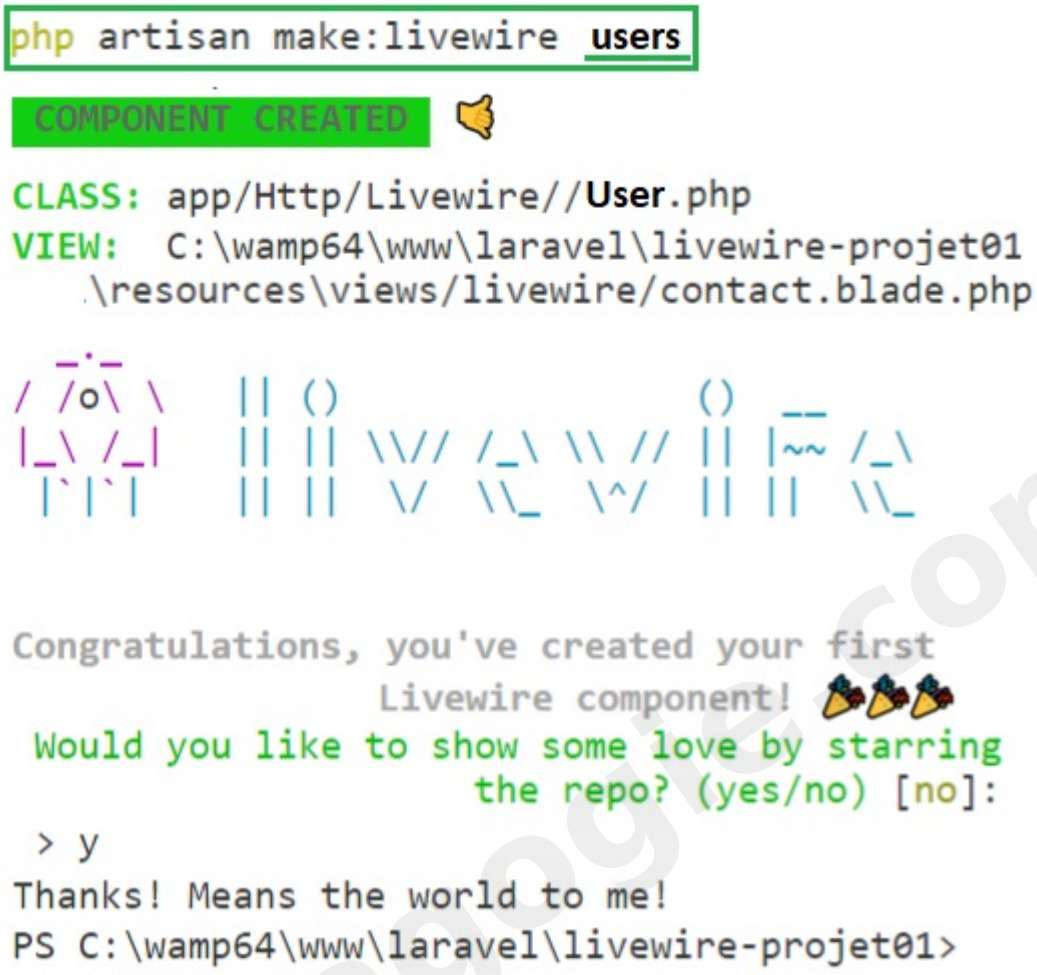
   - **Livewire** est un package pour Laravel qui s'installe classiquement avec composer :
   - Dans cette étape, installer **livewire** sur l'application laravel en utilisant la commande ci-dessous:

```
composer require livewire/livewire
```



6. **Créer Component**

apcpedagogie.com



- Deux fichiers ont été créé dans les deux chemins suivants:

```
app/Http/Livewire/Users.php
resources/views/livewire/users.blade.php
```

7. Contenu du ficher: **`app/Http/Livewire/Users.php`**

8. **Mettre à jour le fichier de component**

   - Contenu du ficher: **`app/Http/Livewire/Users.php`**

app > Http > Livewire > Users.php

```php
<?php
namespace App\Http\Livewire;

use Livewire\Component;
use App\Models\User;

class Users extends Component
{
    public $users, $name, $email, $user_id;
    public $updateMode = false;

    public function render()
    {
        $this->users = User::all();
        return view('livewire.users');
    }

    private function resetInputFields(){
        $this->name = '';
        $this->email = '';
    }

    public function store()
    {
        $validatedDate = $this->validate([
            'name' => 'required',
            'email' => 'required|email',
        ]);
        User::create($validatedDate);
        session()->flash('message',
        'Utilisateurs créés avec succès');
        $this->resetInputFields();
    }

    public function edit($id)
    {
        $this->updateMode = true;
        $user = User::where('id',$id)->first();
        $this->user_id = $id;
        $this->name = $user->name;
        $this->email = $user->email;
    }

    public function cancel()
    {
        $this->updateMode = false;
        $this->resetInputFields();
    }

    public function update()
    {
        $validatedDate = $this->validate([
            'name' => 'required',
            'email' => 'required|email',
        ]);
        if ($this->user_id) {
            $user = User::find($this->user_id);
            $user->update([
                'name' => $this->name,
                'email' => $this->email,
            ]);
            $this->updateMode = false;
            session()->flash('message',
            'Utilisateurs mis à jour avec succès.');
            $this->resetInputFields();
        }
    }

    public function delete($id)
    {
        if($id){
            User::where('id',$id)->delete();
            session()->flash('message',
            'Utilisateurs supprimés avec succès');
        }
    }
}
```

9. **Ajouter une route**

   - Pour ajouter une route dans l'application laravel. ouvrir le fichier "**routes/web.php** " et ajouter la route suivante.

```
Route::view('users','livewire.home');
```

10. **Créer les views**

    - **resources/views/livewire/users.blade.php**

apcpedagogie.com

```blade
<div class="container">
    <div class="row justify-content-center">
        <div class="col-md-8">
            <div class="card">
                <div class="card-header">
                    <div class="row justify-content-evenly">
                        <div class="col-sm-8">
                            <h4>Laravel Livewire Crud - apcpedagogie.com</h4>
                        </div>
                        <div class="col-sm-4">
                            <button type="button" class="btn btn-primary"
                            data-bs-toggle="modal" data-bs-target="#exampleModal" >
                                Ajouter un utilisateur
                            </button>
                        </div>
                    </div>
                </div>
                <div class="card-body">
                    <div>
                        @if($updateMode)
                            @include('livewire.update')
                        @else
                            @include('livewire.create')
                        @endif
                        <table class="table table-bordered mt-5">
                            <thead>
                                <tr><th>No.</th><th>Name</th><th>Email</th><th>Action</th></tr>
                            </thead>
                            <tbody>
                                @foreach($users as $value)
                                <tr>
                                    <td>{{ $value->id }}</td><td>{{ $value->name }}</td><td>{{ $value->email }}</td>
                                    <td>
                                    <button wire:click="edit({{ $value->id }})" class="btn btn-primary btn-sm"
                                        data-bs-toggle="modal" data-bs-target="#updateModal">Modifier</button>
                                    <button wire:click="delete({{ $value->id }})" class="btn btn-danger btn-sm">
                                    </td>                                                    Supprimer</button>
                                </tr>
                                @endforeach
                            </tbody>
                        </table>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

- **resources/views/livewire/home.blade.php**

```blade
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <!--<script src="{{ asset('js/app.js') }}" defer></script>
    <link href="{{ asset('css/app.css') }}" rel="stylesheet">-->
    @livewireStyles
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css"
    rel="stylesheet" integrity="sha384-eOJMYsd53ii+scO/bJGFsiCZc+KMDaMqyr8-8MDcr8gJ8+qr48bdcxlpbtzKgavrs"
     crossorigin="anonymous">
</head>
<body>

                    @if (session()->has('message'))
                        <div class="alert alert-success">
                            {{ session('message') }}
                        </div>
                    @endif
                    @livewire('users')

    @livewireScripts
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-JEW9xMcG8R+pH31jmWH6WWP0WintQrMb4s7ZOdauHnUtxwoG2vI5DkLtS3qm9Ekf" crossorigin="anonymous"></script>
</body>
</html>
```

apcpedagogie.com

- **resources/views/livewire/create.blade.php**

```
<form>
    <div class="form-group">
        <label for="exampleFormControlInput1">Name</label>
        <input type="text" class="form-control" id="exampleFormControlInput1"
        placeholder="Enter Name" wire:model="name">
        @error('name') <span class="text-danger">{{ $message }}</span>@enderror
    </div>
    <div class="form-group">
        <label for="exampleFormControlInput2">Email address</label>
        <input type="email" class="form-control" id="exampleFormControlInput2"
        wire:model="email" placeholder="Enter Email">
        @error('email') <span class="text-danger">{{ $message }}</span>@enderror
    </div>
    <div class="form-group mt-4">
    <button wire:click.prevent="store()" class="btn btn-success">Save</button>
    </div>
</form>
```

- **resources/views/livewire/update.blade.php**

```
<form>
    <div class="form-group">
        <input type="hidden" wire:model="user_id">
        <label for="exampleFormControlInput1">Name</label>
        <input type="text" class="form-control" wire:model="name"
        id="exampleFormControlInput1" placeholder="Enter Name">
        @error('name') <span class="text-danger">{{ $message }}</span>@enderror
    </div>
    <div class="form-group">
        <label for="exampleFormControlInput2">Email address</label>
        <input type="email" class="form-control" wire:model="email"
        id="exampleFormControlInput2" placeholder="Enter Email">
        @error('email') <span class="text-danger">{{ $message }}</span>@enderror
    </div>
    <button wire:click.prevent="update()" class="btn btn-dark">Update</button>
    <button wire:click.prevent="cancel()" class="btn btn-danger">Cancel</button
</form>
```

Sommaire du cours Laravel

Source : https://apcpedagogie.com/laravel-livewire-crud-exemple01 · téléchargé le 07/11/2026 · © APC Pédagogie · apcpedagogie.com