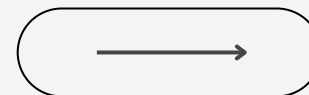


LARAVEL

Pour un back-end solide à l'épreuve des utilisateurs



IUT DE DIJON AUXERRE
dev back | 2024-2025

01

LE PROF'

Ancien MMI / Licence Pro à Dijon

Full stack PHP / JS | DevOps depuis 8 ans



LE DÉROULÉ

30 minutes de théorie max

Stand Up

Les mains sur le clavier pour développer un projet

C'EST QUOI AU JUSTE ?

“Laravel is a web application framework with expressive, elegant syntax.

A web framework provides a structure and starting point for creating your application, allowing you to focus on creating something amazing while we sweat the details. ”



04

POURQUOI ?

Authentification

Migrations

Eloquent

Blade

Caching

CLI

Asynchrone

...



L'HEURE EST VENUE

Création d'un repo **public** sur votre GitHub: [iut-weather](#)

Partage du lien

Création d'un compte sur OpenWeather



06

PARLONS ENVIRONNEMENT

Pas de MAMP / WAMP / XAMPP

Docker -> socle commun

QUI PREND QUOI ?

macOS -> Docker Desktop

Windows -> WSL 2 + Docker Desktop

PC IUT -> VM Ubuntu



Suivre les indications du README du repo GitHub

DONOVANBROQUIN/IUT-LARAVEL

COMMANDES

Composer - install

`composer install`

Composer - ajout dépendance

`composer require ...`

Laravel

`php artisan ...`

NPM - install

`npm i`

NPM - build dev

`npm run dev`



09

PROJET

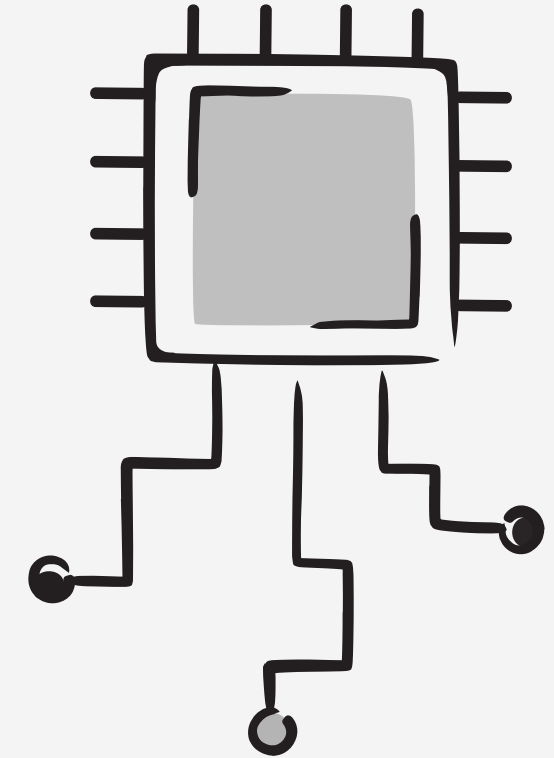
Un classique, une application météo

Avec une démo -> weather.dhonnabhain.me

API OpenWeather

Design sans importance

Laravel 



Énoncé sur GitHub [donovanbroquin/iut-laravel](#)

PROJECT.MD

MVC par nature

Model	Intéraktion avec la base de données (Eloquent)
View	Interface utilisateur (Blade)
Controller	Coordination (Controller)



ENVIRONNEMENT & CONFIGURATIONS

[Documentation](#)

01

Fichier .env

Définition des "secrets"

03

Dossier config/...

Agit sur le fonctionnement de Laravel

02

NE PAS METTRE DANS GIT

Les secrets doivent rester secret

04

METTRE DANS GIT

Sans passer de secret en dur

```
env('APP_URL');
```

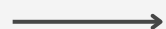


```
config('app.url');
```

CONFIGURATIONS

[Documentation](#)

```
DB_CONNECTION=pgsql
DB_HOST=postgres
DB_PORT=5432
DB_DATABASE=weather
DB_USERNAME=weather
DB_PASSWORD=weather
```



```
use Illuminate\Support\Str;

return [

    'default' => env('DB_CONNECTION', 'sqlite'),

    'connections' => [
        'pgsql' => [
            'driver' => 'pgsql',
            'url' => env('DB_URL'),
            'host' => env('DB_HOST', '127.0.0.1'),
            'port' => env('DB_PORT', '5432'),
            'database' => env('DB_DATABASE', 'laravel'),
            'username' => env('DB_USERNAME', 'root'),
            'password' => env('DB_PASSWORD', ''),
            'charset' => env('DB_CHARSET', 'utf8'),
            'prefix' => '',
            'prefix_indexes' => true,
            'search_path' => 'public',
            'sslmode' => 'prefer',
        ],
    ],
];
```

ROUTING

[Documentation](#)

```
Route::get('dashboard', DashboardController::class)→name('dashboard');
```

Verbe	GET / POST / PUT / PATCH / DELETE
Chemin	Tout ce qui est APRÈS l'extension de domaine
Contrôleur	Namespace du contrôleur
Nom	Facultatif mais permet de s'y retrouver

CONTRÔLEUR

[Documentation](#)

```
namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;

class DashboardController extends Controller
{
    public function __invoke(Request $request)
    {
        // ...

        return view('dashboard', [
            'place' => $place,
            'weather' => $weather,
        ]);
    }
}
```

Commande	php artisan make:controller <NOM>
Namespace	Namespace ~= chemin du fichier
Extends	Étends obligatoirement Controller
Return	Une vue, un tableau, un mot, ...
Request	Champs de formulaire, source de la requête, utilisateur courant, ...

VUE

[Documentation](#)

```
<form method="POST"
  action="{{ route($place->pivot ? 'places.dissociate' : 'places.associate', ['place' => $place]) }}"
  class="flex"
>
  @csrf

  @if($place->pivot)
    @method('DELETE')
  @endif

  <button type="submit" class="w-5 h-5">
    @if($place->pivot)
      Icon 1
    @else
      Icon 2
    @endif
  </button>
</form>
```

Moteur de rendu

Blade -> [documentation](#)

Extension

.blade.php