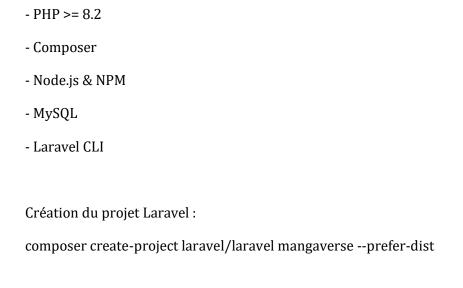
# Procédure Complète : Création, Documentation et Tests d'un Projet Laravel (Mangaverse)

# 1. Création du Projet Laravel

Pré-requis installés:



Se rendre dans le dossier du projet :

cd mangaverse

### 2. Initialisation de l'Environnement

cp .env.example .env

php artisan key:generate

Configurer la base de données dans le .env :

DB\_DATABASE=mangaverse

DB\_USERNAME=root

DB\_PASSWORD=

# 3. Installation des Dépendances

composer install

npm install

npm run dev

# 4. Création des Composants Laravel (MVC)

Créer un modèle et migration :

php artisan make:model Manga -m

Créer un contrôleur de ressources :

php artisan make:controller MangaController --resource

Créer une requête personnalisée :

php artisan make:request MangaRequest

Créer une vue Blade:

resources/views/mangas/index.blade.php

### 5. Routes Laravel

Fichier: routes/web.php

Utiliser : Route::resource('mangas', MangaController::class);

# 6. Migrations et Seeders

php artisan migrate

php artisan make:seeder UserSeeder

php artisan db:seed --class=UserSeeder

php artisan migrate:fresh --seed

### 7. Gestion des Assets avec Vite

npm run dev

npm run build (en production)

## 8. Documentation du Code avec Doxygen

Installer Doxygen: sudo apt install doxygen

Créer un fichier Doxyfile : doxygen -g

Modifier les options INPUT (src du projet) et OUTPUT\_DIRECTORY

Générer la documentation HTML : doxygen Doxyfile

## 9. Ajouter des DocBlocks Laravel

Exemple dans un contrôleur :

```
/**
```

- \* Affiche la liste des mangas.
- \* @return \Illuminate\Http\Response

\*/

public function index() {...}

# 10. Tests Fonctionnels & Unitaires (PHPUnit)

Créer un test:

php artisan make:test MangaTest

Lancer tous les tests:

php artisan test

Fichier de config : phpunit.xml

## 11. Lancement du Serveur de Développement

php artisan serve

Accessible via http://localhost:8000