# **Documentation du Web Service pour l’intégration**

# **Concept**

Le concept est de créer un API RESTFUL pour une plate-forme de commerce électronique qui permet aux clients de parcourir et d’acheter des produits.

Technologies utilisées

La réalisation de ce projet a nécessité l’utilisation de plusieurs technologies notamment Laravel, Vue JS, et le logiciel XAMP avec le système de gestion de base de données relationnelles phpMyAdmin.

# Laravel

Laravel est un Framework web open-source écrit en PHP respectant le principe modèle-vue-controller et entièrement développé en programmation orientée objet. Laravel est distribué sous licence MIT, avec ses sources hébergées sur GitHub. (Wikipédia, 2023).

# L’API

Laravel nous a permis dans ce projet de créer l’API. Laravel a été conçu dès le départ pour prendre en charger la création d’API RESTFUL et fournit de nombreuses fonctionnalités pour créer des API performantes et évolutives

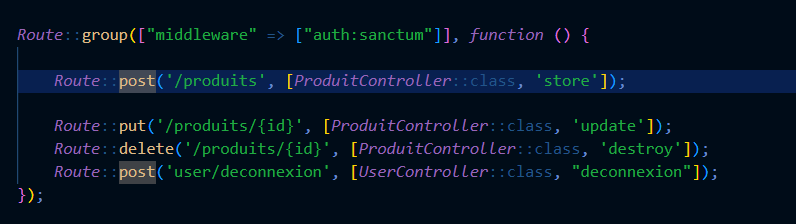
# Les routes de l’API



# La sécurité

Laravel vient avec un système permettant de gérer la notion de sécurité dans le cadre de la création d’API

RESTFUL. Il dispose de middleware notamment le middleware Authenticate qui permet de faire une restriction sur certaines routes nécessitant une authentification.



# Vue JS

Vue.js aussi appelé plus simplement vue, est un Framework JavaScript Open-source utilise pour construire des interfaces utilisateurs et des applications web monopages.

Vue.js a servi dans ce projet de pouvoir créer l’interface utilisateur. Pour plus d’information sur l’utilisation de Vue dans laravel veuillez consulte le lien suivant : [Frontend - Laravel - The PHP Framework For Web Artisans](https://laravel.com/docs/10.x/frontend#using-vue-react).

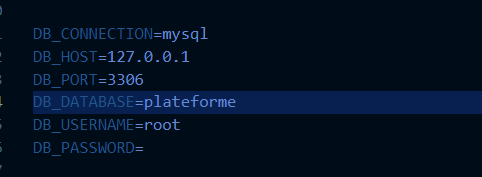
# phpMyAdmin

C’est un système de gestion de base de donnees relationnelles integre dans l’IDE XAMP. Il nous a servi dans ce projet pour gerer la couche d’acces au donnees. L’API utilise ce système pour gerer ses donnees grace a une base de donnees gerer par phpMyAdmin.

# **Deploiment de l’api**

Pour pouvoir utilise ou meme tester l’API vous avez deux options a savoir soit utiliser l’IDE Visual Studio Code qui nous servi comme IDE de developpement pour ce projet, soit utiliser la ligne de commande.

NB : Il faudra dabord configurer la base de donnees, en changeant d’abord le fichier .env contenu dans le repertoire racine de votre projet



Changer << plateforme >> par le nom de votre base de donnee que vous devez créer aussi dans votre système de gestion de base de donnees.

# Visual Studio CODE

Pour tester le projet avec VSCODE, vous pouvez uploader le dossier du projet en cliquant sur :

* File : en haut a gauche
* Choisir l’option Open Workspace from file et choisir le fichier

Une fois le dossier charger vous suivez les etapes suivantes :

* Ouvrir un terminal : en vous rendant l’onglet View puis vous choisissez l’option terminal
* Vous tapez les commandes suivantes :

npm install

npm run dev

php artisan serve

Si vous avez suivi toutes les etapes a la lettre l’API devrait demarrer en local avec un port specifie.

# La ligne de commande

Si vous utilisez la ligne de commande, vous devez vous rendre dans le repertoire ou se trouve le dossier du projet, entrer dans le dossier du projet puis taper les commandes suivantes :

* npm install
* npm run dev
* php artisan serve

Si vous avez suivi toutes les etapes a la lettre l’API devrait demarrer en local avec un port specifie.

# **Les fonctionnalités implémentées**

Les fonctionnalites implementees sont :

1. Authentification de l’utilisateur
2. Catalogues de produits

Par contrainte de temps et a cause de nos ecritures de rapport des projets de fin de cycle, je n’ai pas eu la possiblite d’implementer toutes les fonctionnalites.

Mais nous sommes vient en mesure de realiser toutes les fonctionnalite en temps normale.