



FTTH PHOTO API

language repo not found

languages repo not found repo size repo not found license repo not found

[À propos](#) | [Informations](#) | [Fonctionnalités](#) | [Technologies](#) | [Exigences](#) | [Configuration de la BDD](#) |
[Démarrage rapide](#) | [Licence](#) | [Auteur](#)

À propos

FTTH-API-PHOTO est une API RESTful développée avec Symfony, destinée à gérer les photos associées aux commandes, aux prestations et aux interventions dans un système de gestion de réseau FTTH (Fiber to the Home). Cette API permet de récupérer et de mettre à jour des photos en fonction des références de commandes, de prestations et d'interventions.

La fonctionnalité d'envoi d'image est en cours de développement.

Informations

J'ai décidé de réaliser le test technique en utilisant Symfony, car c'est un framework PHP très puissant et que je maîtrise bien. J'ai également intégré API Platform pour bénéficier de fonctionnalités utiles telles que la génération automatique de documentation, ainsi que du soutien de la communauté.

J'ai abordé le projet étape par étape, en commençant par une base simple avec peu de fonctionnalités. Ensuite, j'ai progressivement complexifié le projet en ajoutant les différentes fonctionnalités et contraintes attendues.

En raison de la programmation des différents composants en anglais, certains noms ont été choisis pour éviter l'utilisation de noms réservés par le système. Voici les traductions :

FR => EN

- COMMANDE_INTERNE => INTERNAL_COMMAND
- INTERVENTION => OPERATION
- PRESTATION => SERVICE

Exemple de référence d'une commande interne : CMD-INT-2024-FFBE609643 Exemple de référence d'une prestation : PREST-2024-FFBE609643 Exemple de référence d'une intervention : INTERV-2024-FFBE609643

J'ai supposé que lors de la création d'une "commande", une référence de commande interne est générée pour le service au sein de l'entreprise, ainsi qu'une référence de prestation pour le client.

Si vous souhaitez que les références de prestation et de commande interne ne soient pas générées simultanément, il suffit de désactiver la fonction `validateReferences()` dans la classe `Command.php`.

✨ Fonctionnalités

- ✓ CRUD des commandes internes;
- ✓ CRUD des prestations;
- ✓ CRUD des interventions;
- 📁: CRUD des images en utilisant les endpoints Photo;
- ✓ Génération automatique des références des commandes internes;
- ✓ Génération automatique des références des prestations;
- ✓ Génération automatique des références des interventions ;
- ✓ Récupération des photos avec la référence d'une commande;
- ✓ Récupération des photos avec la référence d'une prestation;
- ✓ Récupération des photos avec la référence d'une intervention;
- ✓ Documentation API Plateform généré;
- ✓ Gestion des erreurs;
- ✓ Traduction des actions des différents endpoints;\

🚀 Technologies

Voici la liste des outils et technologies utilisés pour ce projet.

- [PHP v8.1.29](#)
- [MariaDB v8.0.31](#)
- [Symfony v6.4.10](#)
- [API Plaform v3](#)
- [Composer v2.7.7](#)
- [Git v2.40.0](#)
- [Symfony CLI v5.10.2](#)

✅ Exigences

Pour faire fonctionner ce projet, vous devez avoir les outils suivants installés :

- PHP
- MariaDB
- Symfony
- API Platform
- Composer
- Git

Pour les versions spécifiques de ces outils, veuillez vous référer à la section [Technologies](#).

:gear: Configuration de la base de données

Avant de démarrer le projet, vous devez configurer l'URL de la base de données.

1. Ouvrez le fichier `.env` à la racine du projet.
2. Modifiez la ligne suivante pour correspondre à l'URL de votre base de données :

```
DATABASE_URL="mysql://user:password@127.0.0.1:3306/your_database_name"
```

Remplacez `user`, `password`, `127.0.0.1`, `3306`, et `your_database_name` par vos propres informations de connexion.

3. Sauvegardez le fichier `.env` après avoir apporté les modifications nécessaires.
4. Par défaut, le projet est configuré pour l'environnement de développement.



Démarrage rapide

Suivez ces étapes pour lancer rapidement le projet sur votre machine locale :

```
# Clonez le projet depuis le dépôt GitHub.
$ git clone https://github.com/AFelix20100/ftth-photo-api

# Accédez au dossier du projet.
$ cd ftth-photo-api

# Installez les dépendances du projet avec Composer.
$ composer install

# Créez la base de données
$ php bin/console doctrine:database:create

# Exécutez les migrations pour mettre à jour la structure de la base de données.
$ php bin/console doctrine:migrations:migrate

# Chargez les fixtures pour peupler la base de données avec des données de test.
$ php bin/console doctrine:fixtures:load

# Lancez le serveur web avec Symfony CLI.
$ symfony server:start

# Le serveur sera accessible à l'adresse suivante : http://localhost:8000
```



Licence

Ce projet est sous licence MIT.

Fait par [Félix-Vincent ARTHUR](#)

[Revenir en haut](#)