

Exercice technique

Klaire
Rapport

Address API

Redigé par :
Haroun Bachar Amine

Table des matières

1	Introduction	2
2	Technologies Utilisées	2
2.1	NestJs	2
2.2	Choix NestJS	2
3	Endpoint Implementé	3
3.1	API Externe	3
3.2	api/adresses - POST	3
3.3	api/id/risks - GET	3
3.4	api/adresses - GET	3
3.5	API/adresse	3
4	Test unitaire	4
5	Déploiement et Dockerisation	4
6	Répartition des dossiers	5
6.1	fichier de l'application (src)	5
7	Documentation	5

1 Introduction

L'API adress-api est développée pour interroger, stocker et gérer des adresses en France, ainsi que pour récupérer les risques associés à ces adresses. Elle utilise des API externes pour les données d'adresses et de risques, et stocke les adresses dans une base. Une documentation Swagger est disponible pour faciliter l'utilisation et les tests des endpoints .

2 Technologies Utilisées

NestJS : Framework web pour construire l'application .

2.1 NestJs



Figure 1 – Documentation des API

2.2 Choix NestJS

J'ai choisi NestJS pour développer Address API parce que, même si j'ai déjà bossé avec Django ou Spring , je n'avais jamais codé avec NestJS. C'est un framework JavaScript avec TypeScript. C'était l'occasion parfaite pour me lancer, découvrir un nouveau truc et voir la différences avec les autres framework. Sa structure claire et ses outils (comme TypeORM) m'ont bien aidé pour coder une API REST

3 Endpoint Implementé

3.1 API Externe

Les données d'adresses et de risques sont récupérées via les API suivantes :

API Adresse (Base Adresse Nationale - BAN) :

<https://api-adresse.data.gouv.fr/search/?q=8+bd+du+portlimit=1>

API Géorisques.

Ancien URL(Invalidé) :

https://www.georisques.gouv.fr/api/v3/v1/resultats_rapport_risque?latlon=lon,lat

Nouvelle URL :

https://georisques.gouv.fr/api/v1/resultats_rapport_risque?latlon=lon,lat

3.2 api/adresses - POST

Description : Interroge l'API Adresse (BAN) pour récupérer les détails d'une adresse et la stocker dans la base de données locale.

3.3 api/id/risks - GET

Description : Interroge l'API Géorisques pour récupérer les risques associés à une adresse, en utilisant sa longitude et latitude.

3.4 api/adresses - GET

Description : Liste toutes les adresses stockées dans la base de données locale.

3.5 API/adresse

Documentation swagger : Une documentation des API est disponible depuis l'url : "<http://localhost:8000/api-docs>"

Elle permet de :

Consulter les détails des endpoints (paramètres, body, réponses).
Tester les endpoints directement depuis le navigateur.

Addresses			^
POST	/api/addresses	Créer une nouvelle adresse	▼
GET	/api/addresses	Lister toutes les adresses stockées	▼
GET	/api/addresses/{id}/risks	Récupérer les risques pour une adresse	▼

Figure 2 – Documentation des API

4 Test unitaire

Un test unitaire est implémenté dans `address.service.spec.ts` pour vérifier la méthode `createAddress`. Il s'assure que cette méthode appelle correctement l'API Adresse (BAN), traite la réponse pour extraire les données (comme l'adresse, la latitude et la longitude) et les stocke dans la base de données.

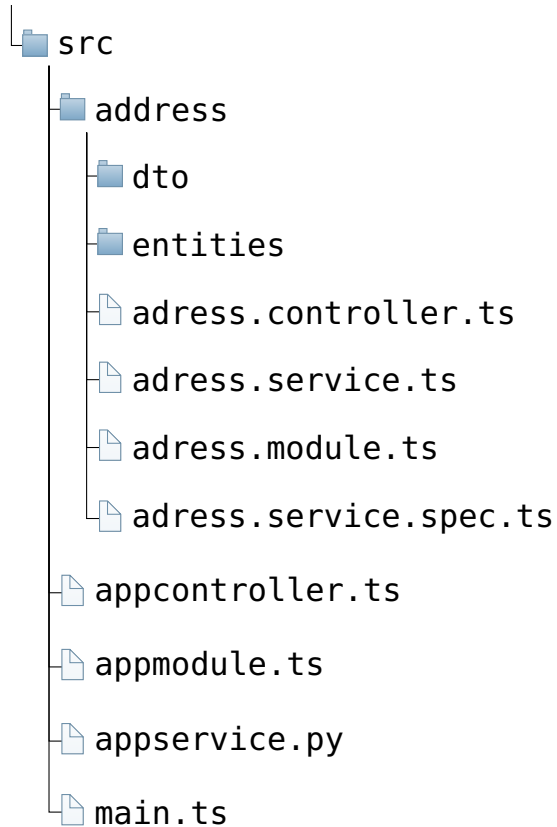
5 Déploiement et Dockerisation

Pour déployer l'application, nous utilisons docker ,un fichier Dockerfile et `docker-compose.yml` sont disponible dans la racine du projet. Afin de lancer l'application avec `docker-compose build` et `docker-compose up`.

6 Répartition des dossiers

6.1 fichier de l'application (src)

Main code source



7 Documentation

Instructions pour Exécuter le Projet : Un fichier README.md a été fourni avec des instructions détaillées

Lien du depot public github :

git clone git@github.com :Amineharoun3/Address_handler-API.git