Projet Data Pipeline & API Django - JobTech

Thelma LUM

Mai PHUONG

1. Introduction

Ce projet vise à construire une chaîne de traitement de données (pipeline) pour agréger, nettoyer, unifier et exposer des offres d'emploi tech issues de différentes sources (Adzuna, Glassdoor, StackOverflow, GitHub, etc.), en fournissant un accès simple via une API Django/PostgreSQL.

2. Architecture générale

- Collecte de données multi-source via des scripts de scraping/API.
- Unification et nettoyage des fichiers bruts JSON/CSV (harmonisation des champs : entreprise, pays, secteur, salaire, compétences, etc.
- Export d'un fichier CSV unique prêt pour l'import dans PostgreSQL/Django.
- Stockage en base PostgreSQL (modèle Django avec ArrayField pour les compétences).
- API REST sécurisée par token, permettant de filtrer/interroger les données (salaires, compétences, etc.)

3. Modèle de données final

company	country	sector	title	salary_min	salary_max	skills
...	...	...	...	...	...	["python", ...]

- company : nom de l'entreprise
- country : pays (harmonisé en anglais)
- sector : secteur d'activité (harmonisé)
- title : intitulé du poste
- salary_min / salary_max : bornes du salaire (float, annuelles)
- skills : liste des compétences au format JSON

4. API – Exemple d'appel

Exemple de commande CURL :

```
curl.exe -H "Authorization: Token VOTRE_TOKEN"
"http://localhost:8000/api/v1/salary-
daily/?country=Belgium&skill=python"
```

Réponse :

```
{
  "date": "2025-07-04",
  "count": 10,
  "median": 51000.0,
  "distribution": [5,2,1,0,1,0,0,0,0,1]
}
```

- median : salaire médian des offres correspondant
- distribution : histogramme des salaires sur 10 classes

5. Problèmes rencontrés & solutions

Problèmes de format dans les fichiers CSV/JSON

Erreurs de parsing (Error tokenizing data. C error: Expected...) lors de l'union ou du nettoyage.

- **Cause** : Fichiers sources hétérogènes (séparateurs différents, colonnes absentes, caractères spéciaux).
- **Solution** : Scripts robustes avec gestion des exceptions, harmonisation des séparateurs, mapping systématique des colonnes clés.

Mapping sector/pays difficile

Colonnes sector/industry/secteur pas toujours cohérentes, pays absents ou en plusieurs langues.

- **Cause** : Multiplicité des sources et des formats (anglais/français/italien/allemand/etc.).
- **Solution** : Harmonisation via fonctions de mapping, passage systématique des champs en anglais, suppression des valeurs aberrantes ou inconnues.

Gestion des NaN, listes et doublons

Valeurs manquantes, erreurs TypeError: unhashable type: 'list' lors des déduplications.

- **Solution** : Normalisation de toutes les listes (skills), nettoyage préalable avant tout traitement, suppression des doublons en convertissant les listes en chaînes pour la comparaison.

6. Résultat obtenu

- Pipeline automatisé de la collecte à l'exposition API.
- Données cleans, filtrées, accessibles sous PostgreSQL.
- Endpoints API permettant la visualisation en temps réel des salaires médians par pays, skill, secteur, etc.

7. Commandes d'utilisation

- Initialisation :
  python -m venv venv
  pip install -r requirements.txt

- Migration et lancement serveur :
  python manage.py migrate
  python manage.py runserver

- Import des données cleans en base :
  python jobapi/import_csv.py

- Tester l'API :
  curl -H "Authorization: Token xyz" "http://localhost:8000/api/v1/salary-

daily/?country=FR&skill=Python"