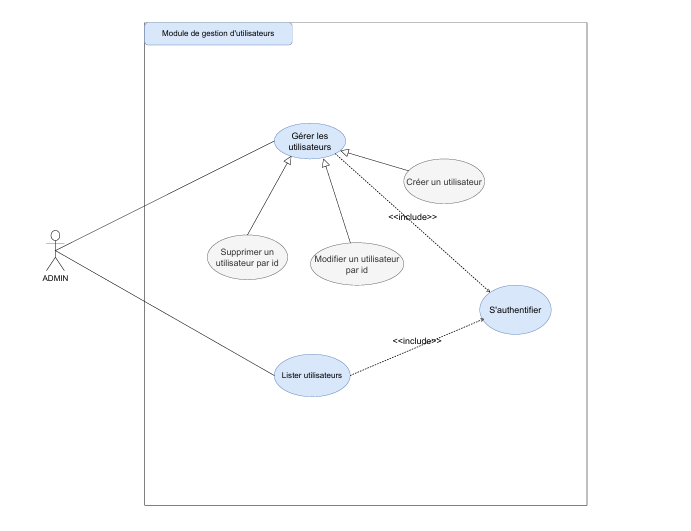
**Readme Projet OWA**

**Réalisé par :**

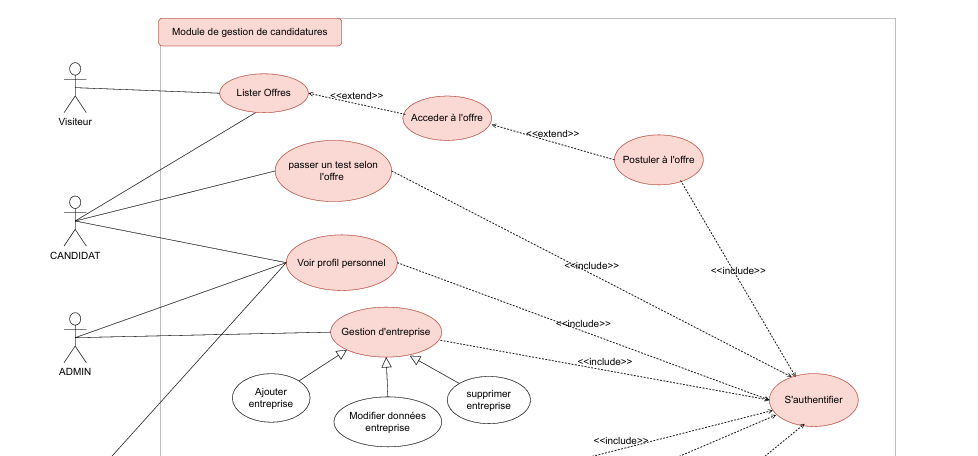
* **Benharrats Wail**
* **El Hassouni Otmane**
* **El Khantour Aymane**

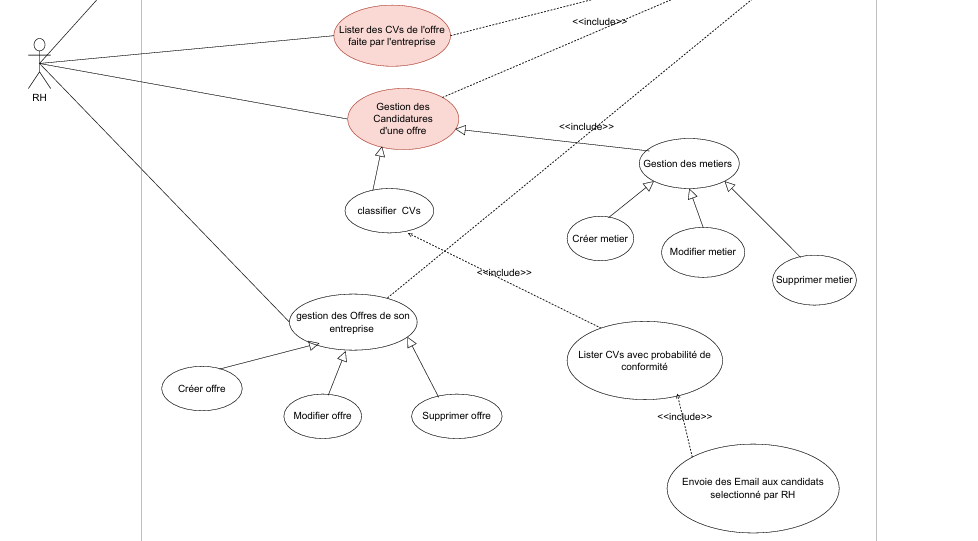
**Plan**

1. **Conception**
   1. **Diagramme de use cases pour chaque module**
2. **Architecture technique.**
3. **Partie AI.**
   1. **Fonctionnalités**
   2. **Données sur dataset**
   3. **Données sur modèle**
4. **Difficultés rencontrées et comment on les a surmontés**
5. **Outils utilisés**
6. **Conception**
   1. **Diagramme de use cases pour chaque module**
      1. **Module de gestion d’utilisateur :**

****

* + 1. **Module de gestion de candidatures :**

****

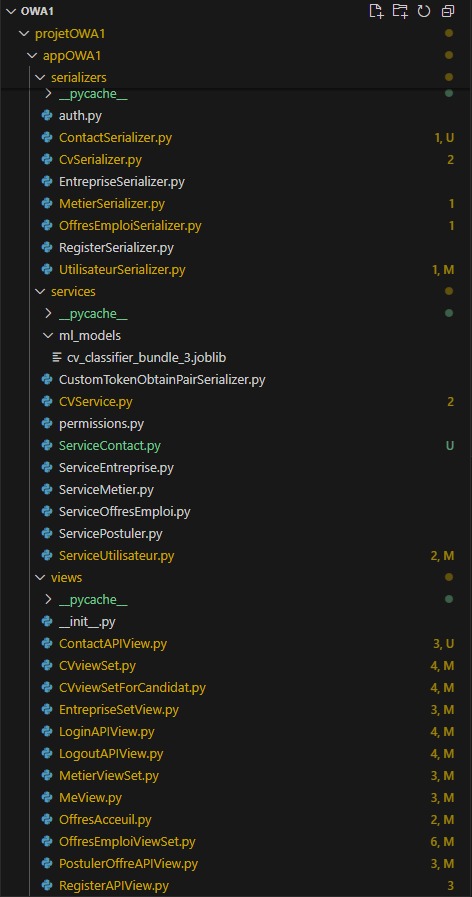
****

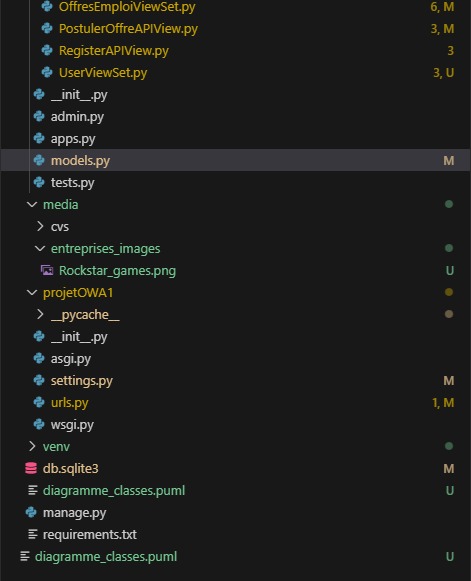
1. **Architecture technique**

**L’architecture technique contient 2 services : Backend et front-end**

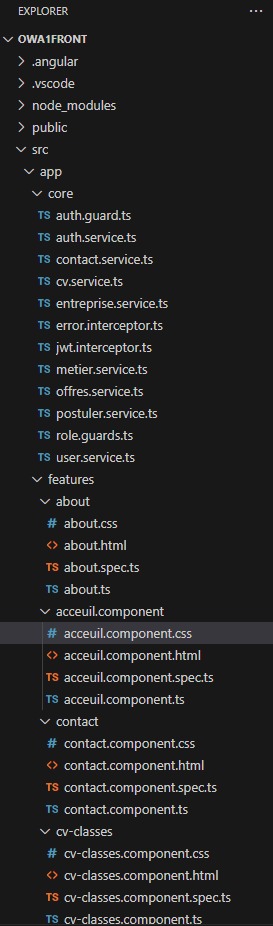
**Sachant que :**

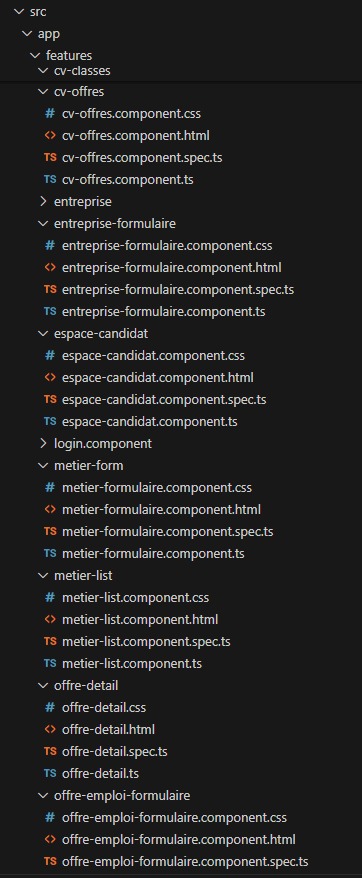
* **Le Backend : Fait avec le Framework Django de python et contient les modeles de base de données , ml\_model qui contient , les services les vues (les APIs Rest) et la base de données :**

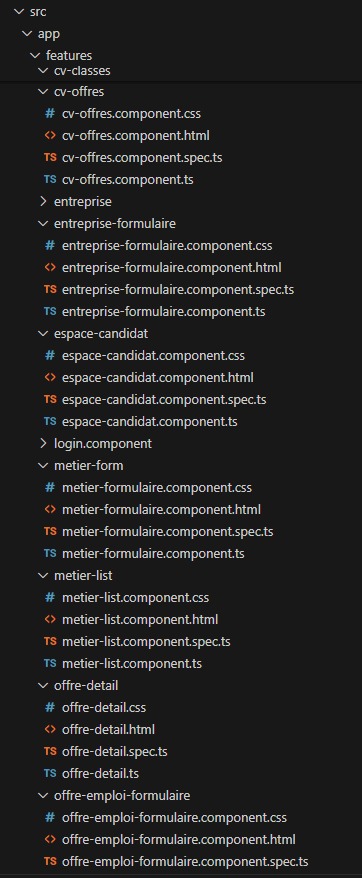


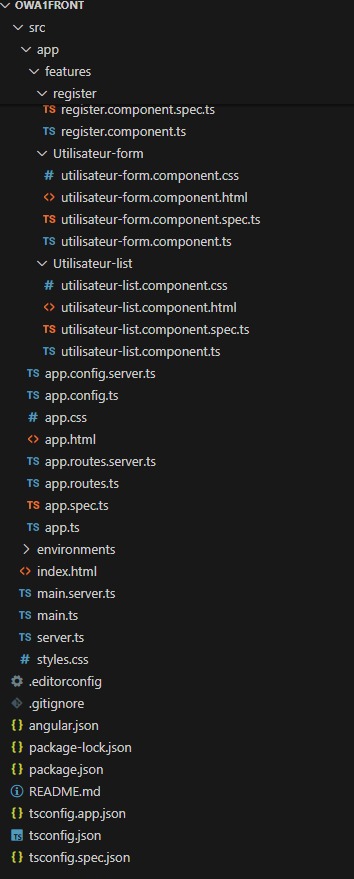


* **Le Frontend : fait avec le Framework Angular et contient les couches services pour communication http des données , features pour les component d’Angular :**









1. **Partie AI**
   1. **Fonctionnalités**

**Au niveau de cette partie on va détailler les fonctionnalité AI :**

* **Fonctionnalité 1 : classification des CV selon métiers tel qu’on a entrainé le modèle du machine Learning Logistic Regression ou Régression logistique sur une dataset qui contient les CVs sous forme de texte brut comme variable d’entré et le métier comme variable de sortie, le but c’est d’entrer un CV et pouvoir savoir le métier correspondant parmi les métiers existants dans notre ensemble de données.**
* **Fonctionnalité 2 : génération des testes techniques concernant une offre d’emploi en utilisant une API Open AI** 
  1. **Données sur dataset**

**La dataset contient comme champs :**

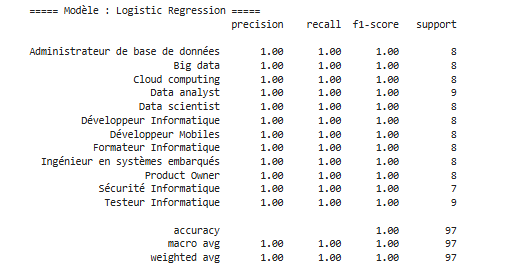
* **Textes bruts des CVs**
* **Les métiers correspondants**

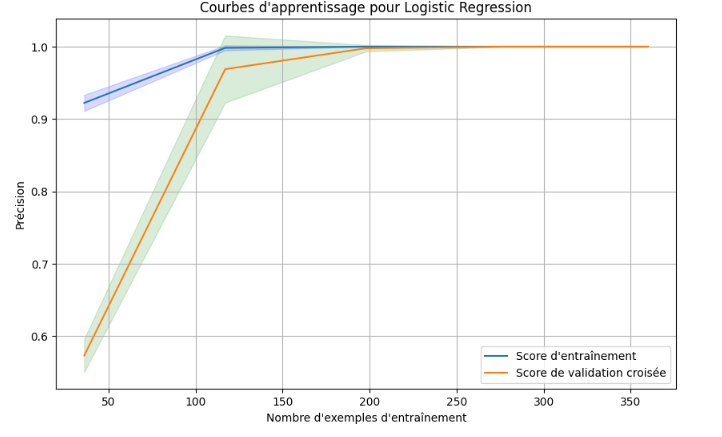
**La collecte des données consiste à faire un web scrapping depuis le site web :** [**https://entreprise.francetravail.fr/**](https://entreprise.francetravail.fr/)

**Puis après avoir collecter les données on les a par la suite nettoyés en éliminant des CVs qui expriment mal le métier, et en reconstruisant des CVs pour qu’ils puissent être conformes au métier.**

* 1. **Données sur modèle**

**Le modèle qu’on a utilisé c’est Logistic Regression , voila les résultat qu’il nous a donné :**

****

****

1. **Difficultés rencontrées et comment on les a surmontés**

**Généralement la difficulté majeure qu’on a rencontré dans ce projet est la collecte de données tel qu’à chaque fois on collecte les données on remarque que les données ne sont pas entrainable ce qui fait il est nécessaire de refaire encore une fois la collecte jusqu’à ce qu’on a eu l’idée de nettoyé et modifier les données courants pour qu’ils puissent répondre au besoin des modeles du ML et soient bien entrainables.**

1. **Outils utilisés**

**Concernant l’IA :**

* scikit-learn Version: 1.2.2
* joblib Version: 1.4.2

**Concernant le logiciel :**

* **Django 5.2.1**
* **Angular 20.0.1**