Ce backend utilise plusieurs technologies et bibliothèques pour garantir une gestion fluide et sécurisée des utilisateurs. **FastAPI** est le framework central qui permet de créer cette API, offrant une validation de données automatisée et une documentation interactive avec **Swagger**. Pour gérer la sécurité des échanges entre le frontend et le backend, **CORSMiddleware** permet l'activation des CORS, autorisant ainsi des requêtes depuis des domaines externes comme un frontend local ou un autre serveur.

Le **package Pydantic** est utilisé pour la validation des données des utilisateurs. Dans ce cas, le modèle **Utilisateur** garantit que les informations comme le nom, l'email et le mot de passe respectent les formats appropriés. Cette validation permet de réduire les erreurs en cas de données incorrectes ou malformées.

Le mot de passe est sécurisé avec **Passlib**, qui permet de le hacher avant de l’enregistrer, protégeant ainsi les informations sensibles des utilisateurs. Lors de la connexion, l’API génère un **JWT (JSON Web Token)** avec la bibliothèque **PyJWT**, ce qui permet d’assurer une authentification sécurisée, sans nécessiter de conserver l’état côté serveur.

Le serveur est exécuté avec **Uvicorn**, un serveur ASGI ultra-rapide et performant compatible avec FastAPI, qui assure la gestion des requêtes HTTP. L'option **--reload** permet de recharger automatiquement le serveur lors de modifications du code pour simplifier le développement. Enfin, l’API est conçue pour faciliter l’inscription et la connexion des utilisateurs tout en garantissant une gestion sécurisée des mots de passe et des tokens d'authentification.