

Projet API Avancé

Voici une version améliorée et plus claire de votre texte :

Contexte :

Vous êtes recruté en tant que développeur dans une startup qui souhaite lancer un produit appelé

BoîteAnnonces.

Cette plateforme permettra aux utilisateurs de publier des annonces sur leurs pages personnelles, qui pourront être consultées par d'autres utilisateurs. La startup souhaite concevoir à la fois le backend et le frontend de la plateforme.

Fonctionnalités à implémenter pour la première version :

1. **Création de compte utilisateur**
2. **Authentification utilisateur**
 - L'utilisateur pourra s'authentifier de deux manières :
 - a. Après avoir créé un compte. Vérifiez que l'email est unique
 - b. Sans créer de compte, en utilisant les providers OAuth2 suivants : **Google, Twitter et GitHub**.
3. **Publication d'une annonce**
4. **Modification d'une annonce**
5. **Suppression d'une annonce**
6. **Consultation de la liste des annonces**
7. **Consultation des détails d'une annonce**
8. **Déconnexion de l'utilisateur**

Exigences techniques :

- **API Backend :**
 - Toutes les fonctionnalités du backend doivent être accessibles via une API REST et retourner des réponses au format JSON.
 - Les codes de retour HTTP doivent refléter correctement les actions effectuées (exemple : 200 pour une requête réussie, 401 pour une authentification échouée, etc.).
- **Règles d'accès et de sécurité :**
 - Seuls les utilisateurs authentifiés peuvent :
 - Créer, modifier ou supprimer leurs propres annonces.
 - Lister et consulter les détails de leurs annonces.
 - Un utilisateur ne peut pas effectuer plus de **10 requêtes par seconde** pour la création d'annonces.
- **Caractéristiques des données :**
 - Une annonce doit contenir un **titre**, une **description** et une **image (jpeg ou png)**.
 - Un utilisateur doit disposer des informations suivantes : **nom, prénom, email et mot de passe**.
- **Gestion des fonctionnalités avancées :**
 - Mise en place de la gestion de la **modification concurrente** : si un utilisateur modifie une annonce depuis deux sessions différentes, ces actions doivent être correctement gérées.
 - La **liste des annonces** doit intégrer une gestion du **cache** : aucune requête supplémentaire à la base de données ne doit être effectuée si les données n'ont pas changé pour l'utilisateur courant.
 - Après une déconnexion, il ne doit plus être possible d'utiliser le token de session.

- **Authentification :**
 - L'authentification, quel que soit le mode choisi (compte classique ou OAuth2), doit utiliser le **JSON Web Token (JWT)**.
 - Le JSON Web Token doit être stocké dans les cookies pour éviter une fuite de données
- **Livraison :**

Le projet doit être accessible depuis votre dépôt github

- **Documentation :**
 - Un fichier **README** doit accompagner le projet. Il devra contenir :
 - Une présentation des auteurs.
 - Les étapes nécessaires pour configurer et lancer l'application en local.

Cette première version de l'application se concentrera uniquement sur les fonctionnalités et les exigences listées ci-dessus.

Date butoir : le 31 Janvier 2025 à 23h59