

Gestion Intelligente des Ressources Humaines (RH)

I. Fonctionnalités de l'Application

1. Gestion des Utilisateurs

- Implémentez un système de création, modification et suppression des comptes utilisateurs.
- Les utilisateurs doivent avoir des rôles différents : **admin**, **recruteur**, **candidat**.
- Question : *Comment mettre en place la gestion des rôles dans Django (permissions, groupes, décorateurs d'accès) ?*

2. Gestion des Candidatures

- Permettez aux candidats d'ajouter leurs candidatures (CV et lettres de motivation).
- Les recruteurs et administrateurs doivent pouvoir consulter et supprimer les candidatures.
- Question : *Comment gérer l'upload sécurisé de fichiers (CV en PDF/Word) dans Django ?*

3. Analyse des Candidatures avec IA

- Intégrez un modèle de **traitement du langage naturel (NLP)** basé sur **BERT** ou **GPT**.
- Le modèle doit être capable d'extraire des informations clés (compétences, expériences, formation).
- Chaque CV doit recevoir un **score de pertinence** par rapport à un poste donné.
- Question : *Comment intégrer un modèle NLP pré-entraîné (HuggingFace, spaCy, etc.) dans Django ?*

4. Notifications

- Implémentez un système de notification par email pour informer :
 - Les candidats du statut de leur candidature,
 - Les recruteurs lorsqu'une nouvelle candidature est reçue.
- Question : *Quels paramètres faut-il configurer pour envoyer des emails sécurisés dans Django ?*

5. Tableau de Bord

- Créez une interface de statistiques : nombre de candidatures, scores moyens, répartition par compétences, etc.
 - Question : *Quels outils Django ou bibliothèques front-end (Chart.js, Recharts) utiliser pour afficher des statistiques dynamiques ?*
-

II. Modèle d'IA

1. Préparation des données

- Téléchargez et explorez le dataset [Resume Dataset](#).
- Question : *Comment prétraiter des CV textuels pour extraire compétences, formations et expériences ?*

2. Entraînement du modèle

- Entraînez un modèle NLP (BERT, DistilBERT ou GPT-like) pour classifier et scorer les CV.
- Question : *Quelles métriques utiliser pour évaluer la performance du modèle (F1-score, précision, rappel) ?*

3. Intégration dans Django

- Ajoutez une API interne qui prend un CV en entrée et renvoie un score + une recommandation.
 - Question : *Quelle architecture Django (views, serializers, celery pour tâches asynchrones) convient pour appeler un modèle IA ?*
-

III. Données

- Utilisez un jeu de données de CV (anonymisés).
 - Question : *Quels formats de fichiers choisir pour stocker les CV et les extractions du modèle (PDF, JSON, PostgreSQL avec champ JSONField) ?*
-

IV. Meilleures Pratiques

1. Sécurité

- Protégez l'application contre les attaques XSS, CSRF, injections SQL.
- Question : *Quels middlewares Django activer pour renforcer la sécurité ?*

2. Performance

- Mettez en cache les résultats fréquents, optimisez les requêtes SQL.
- Question : *Comment utiliser `select_related` et `prefetch_related` pour améliorer les performances ?*

3. Maintenance

- Écrivez des tests unitaires et d'intégration.
- Question : *Comment tester un modèle IA intégré dans une application Django ?*

4. Déploiement

- Conteneurisez l'application avec **Docker**.
- Question : *Quels avantages concrets apporte Docker pour un projet Django ?*

5. Scalabilité

- Utilisez un load balancer et configurez un auto-scaling.
 - Question : *Quelles solutions cloud (AWS, Azure, GCP) permettent une scalabilité horizontale de Django + IA ?*
-

V. Questions Finales (Synthèse)

1. Comment implémenter la gestion des utilisateurs avec rôles (admin, recruteur, candidat) dans Django ?
2. Quels mécanismes Django assurent la sécurité des mots de passe des utilisateurs ?
3. Quelles étapes suivre pour entraîner un modèle NLP d'analyse de CV ?
4. Comment intégrer un modèle d'IA (BERT/GPT) dans un projet Django ?
5. Quelles pratiques améliorent la performance d'une application Django en production ?
6. Comment sécuriser une application Django contre les attaques CSRF et XSS ?
7. Quels sont les avantages de l'utilisation de Docker pour le déploiement ?
8. Comment mettre en place un pipeline CI/CD pour automatiser le déploiement de Django ?