

Documentation des endpoints de l'API – JobApp Backend

Ce document regroupe **les modèles (DTO)** puis **les endpoints**, en indiquant pour chacun **ce qui est envoyé** et **ce qui est reçu**.

1) Modèles (DTO)

1.1 RegisterRequest (corps envoyé en création/mise à jour)

- username (*String*)
- email (*String*)
- password (*String*)
- role (*String*) — valeurs attendues : **STUDENT, COMPANY, ADMIN**
- firstName (*String*)
- lastName (*String*)
- address (*String*)
- companyName (*String*)
- phoneNumber (*String*)

1.2 UserResponse (corps reçu)

- id (*Long*)
- username (*String*)
- email (*String*)
- role (*String*)
- firstName (*String*)
- lastName (*String*)
- address (*String*)
- companyName (*String*)
- phoneNumber (*String*)

1.3 ApplicationResponse (modèle de réponse)

- id (*Long*) — identifiant de la candidature

- offerTitle (*String*) — titre de l'offre associée
- companyName (*String*) — nom de l'entreprise ayant publié l'offre
- appliedAt (*LocalDateTime*) — date/heure de soumission

1.4 AuthRequest (*request*)

- username (*String*)
- password (*String*)

1.5 AuthResponse (*response*)

- token (*String*)

1.6 ChangePasswordRequest (*request*)

- username (*String*)
- oldPassword (*String*)
- newPassword (*String*)

1.7 OfferRequest (*request*)

- title (*String*)
- description (*String*)
- logoUrl (*String*, *optionnel*)
- websiteUrl (*String*, *optionnel*)

1.8 OfferResponse (*response*)

- id (*Long*)
- title (*String*)
- description (*String*)
- logoUrl (*String*, *optionnel*)
- websiteUrl (*String*, *optionnel*)
- createdAt (*LocalDateTime*)
- companyName (*String*)
- applied (*boolean*) — indique si l'étudiant courant a postulé / a déjà postulé

2) Endpoints par contrôleur

2.1 Administration des utilisateurs — /admin/users

1) Lister tous les utilisateurs

GET /admin/users

- **Envoi** : —
- **Réception** : List<UserResponse>

2) Créer un utilisateur

POST /admin/users

- **Envoi** : **Body** → RegisterRequest
- **Réception** : UserResponse

3) Mettre à jour un utilisateur

PUT /admin/users/{id}

- **Envoi** : **Path** → id (*Long*) ; **Body** → RegisterRequest
- **Réception** : UserResponse

4) Supprimer un utilisateur

DELETE /admin/users/{id}

- **Envoi** : **Path** → id (*Long*)
- **Réception** : (*corps vide*)

2.2 Candidatures étudiant — /applications/student

1) Lister les candidatures d'un étudiant

GET /applications/student/{studentId}

- **Envoi** : **Path** → studentId (*Long*)
- **Réception** : List<ApplicationResponse> (*JSON*)

2.3 Authentification — /auth

1) Authentification (login)

POST /auth/login

- **Envoi** : **Body** → AuthRequest

- **Réception : Body** → AuthResponse

2) Inscription (register)

POST /auth/register

- **Envoi : Body** → RegisterRequest
- **Réception : Body** → UserResponse

3) Mot de passe oublié

POST /auth/forgot-password

- **Envoi : Query/Form** → email (*String*)
- **Réception : Body** → String (*message de confirmation*)

4) Changer le mot de passe

PUT /auth/change-password

- **Envoi : Body** → ChangePasswordRequest
- **Réception : Body** → String (*message de confirmation*)

2.4 Offres (entreprise) — /company/offers

1) Créer une offre

POST /company/offers

- **Envoi : Body** → OfferRequest ; **Auth/Contexte** → userId (claim ou query)
- **Réception : Body** → OfferResponse

2) Lister les offres de l'entreprise

GET /company/offers

- **Envoi : Auth/Contexte** → userId (claim ou query)
- **Réception : Body** → List<OfferResponse>

3) Mettre à jour une offre

PUT /company/offers/{id}

- **Envoi : Path** → id (*Long*) ; **Body** → OfferRequest ; **Auth/Contexte** → userId (claim ou query)
- **Réception : Body** → OfferResponse

4) Supprimer une offre

DELETE /company/offers/{id}

- **Envoi : Path** → id (*Long*) ; **Auth/Contexte** → userId (claim ou query)

- **Réception** : *(corps vide)*

2.5 Offres (étudiant) — /offers

1) Lister toutes les offres disponibles (pour l'étudiant authentifié)

GET /offers

- **Envoi** : *(contexte d'authentification)* → utilisateur courant via Principal
- **Réception** : **Body** → List<OfferResponse>

2) Obtenir le détail d'une offre

GET /offers/{id}

- **Envoi** : **Path** → id (*Long*)
- **Réception** : **Body** → OfferResponse

3) Postuler à une offre (upload CV + lettre de motivation)

POST /offers/{id}/apply (*Content-Type : multipart/form-data*)

- **Envoi** :
 - **Path** → id (*Long*)
 - **Form-data** → cv (*MultipartFile*), motivation (*MultipartFile*)
 - *(contexte d'authentification)* → utilisateur courant via Principal
- **Réception** : *(corps vide — statut succès)*