Documentation API

Table des matières

[I. Gestion des utilisateurs 3](#_Toc174206967)

[1. Inscription 3](#_Toc174206968)

[2. Login 3](#_Toc174206969)

[3. Liste des utilisateurs 4](#_Toc174206970)

[4. Suspension d’un utilisateur 4](#_Toc174206971)

[5. Déconnexion 4](#_Toc174206972)

[II. Gestion des annonces 4](#_Toc174206973)

[1. Création d’une annonce 4](#_Toc174206974)

[2. Modification d’une annonce 5](#_Toc174206975)

[III. Gestion des échanges 6](#_Toc174206976)

[IV. Mapping 6](#_Toc174206977)

# Gestion des utilisateurs

## Inscription

* **URL** : <http://localhost:8080/api/users/register>
* **Méthode** HTTP : POST
* **Authorization :** none
* **Description** : Cet endpoint permet d'inscrire un nouvel utilisateur en fournissant les informations nécessaires telles que le nom d'utilisateur, le nom, le mot de passe, l'email, le numéro de téléphone et l'adresse.
* **Corps de la requête** : Body/raw JSON

{

"username": "john\_doe",

"name": "John Doe",

"password": "password123",

"email": "john@email.com",

"phone": "0330000000",

"address": "Main St"

}

On peut ajouter un paramètre « role » : « ADMIN » si on veut créer un Administrateur.

* **Réponses** :

|  |  |  |
| --- | --- | --- |
| Code | Message | Explication |
| 400 Bad request | Invalid phone number format. | Les numéros de téléphone doivent suivre la forme 03xxxxxxxx ou 3xxxxxxxx ou +2613xxxxxxxx ou 2613xxxxxxxx (valable pour les opérateurs malgaches : 032, 033, 034, 037, 038, 039) |
| 400 Bad request | Invalid email format: xxxxxx | L’email doit être de la forme xxxxxx@xxxxx.xxxxx |
| 200 OK | L’utilisateur créé | Inscription faite |

## Login

* **URL** : <http://localhost:8080/api/auth/login>
* **Méthode HTTP** : POST
* **Authorization :** none
* **Description** : Cet endpoint permet à un utilisateur de se connecter en fournissant son nom d'utilisateur et son mot de passe. Si les informations sont correctes, un jeton d'accès (access token) est retourné.
* **Corps de la requête**: Body/raw JSON

{  
 "username": "fiaritia\_admin",

"password": "mdpprom13"

}

* Réponses :

|  |  |  |
| --- | --- | --- |
| Code | Message | Explication |
| 200 OK | {      "accessToken": "xxx",      "tokenType": "Bearer",      "username": "john\_doe"  } | La valeur de ‘accessToken ‘ sera utilisée comme authentification pour les opérations futures |
| 400 |  |  |

## Liste des utilisateurs

* **URL** : http://localhost:8080/api/users api/page=0&size=10
* **Méthode HTTP** : GET
* **Authorization :** Bearer token (accessToken) obtenu à partir du login
* **Description** : Cet endpoint permet d’obtenir la liste des utilisateurs paginée
* **Corps de la requête**: Params

|  |  |  |
| --- | --- | --- |
| Nom | Valeur | Explication |
| page | Un entier | Le numéro de la page |
| size | Un entier | La taille de la page |

* **Réponses** :

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

## Suspension d’un utilisateur

* **URL** : http://localhost:8080/api/users/{id}/suspend
* **Méthode HTTP** : PUT
* **Authorization :** Bearer token (accessToken) obtenu à partir du login
* **Description** : Cet endpoint permet à l’administrateur de suspendre un utilisateur. Un utilisateur suspendu ne peux plus créer d’annonce ni interagir avec les annonces des autres utilisateurs.
* **Corps de la requête**: none
* Réponses :

|  |  |  |
| --- | --- | --- |
| Code | Message | Explication |
| 401 Unauthorized |  |  |
| 403 Forbidden |  |  |
| 200 OK |  |  |

## Déconnexion

# Gestion des annonces

## Création d’une annonce

* **URL** : http://localhost:8080/api/items/
* **Méthode HTTP** : POST
* **Authorization :** Bearer token (accessToken) obtenu à partir du login
* **Description** : Cet endpoint permet à l’utilisateur connecté de créer une nouvelle annonce
* **Corps de la requête**: Body / form-data

|  |  |  |
| --- | --- | --- |
| Nom | Valeur | Explication |
| title | Chaine de caractères | Le numéro de la page |
| description | Chaine de caractères | La taille de la page |
| category | Chaine de caractères | Fait référence à un enum côté metier. Voici les valeurs possibles : MEUBLE, ELECTRONIQUE, LIVRE, CUISINE, MODE, VETEMENT, SPORT, JOUET, EPICERIE, MUSIQUE, BEAUTE, DECORATION, FITNESS |
| newImages | Type File, peut être multiple | Les images correspondantes à l" |

* Réponses :

|  |  |  |
| --- | --- | --- |
| Code | Message | Explication |
| 401 Unauthorized |  |  |
| 403 Forbidden |  |  |
| 200 OK |  |  |

## Modification d’une annonce

* **URL** : http://localhost:8080/api/items/
* **Méthode HTTP** : PUT
* **Authorization :** Bearer token (accessToken) obtenu à partir du login
* **Description** : Cet endpoint permet à l’utilisateur connecté de modifier une annonce déjà existante
* **Corps de la requête**: Body / form-data

|  |  |  |
| --- | --- | --- |
| Nom | Valeur | Explication |
| title | Chaine de caractères | Le numéro de la page |
| description | Chaine de caractères | La taille de la page |
| category | Chaine de caractères | Fait référence à un enum côté metier. Voici les valeurs possibles : MEUBLE, ELECTRONIQUE, LIVRE, CUISINE, MODE, VETEMENT, SPORT, JOUET, EPICERIE, MUSIQUE, BEAUTE, DECORATION, FITNESS |
| newImages | Type File, peut être multiple | Les images correspondantes à l" |
| imagesToRemove | Tableau de int | Tableau des identifiants des images à supprimer |

* Réponses :

|  |  |  |
| --- | --- | --- |
| Code | Message | Explication |
| 401 Unauthorized |  |  |
| 403 Forbidden |  |  |
| 200 OK |  |  |

# Gestion des échanges

# Mapping