|  |
| --- |
|  |
| Documentation technique de l’API open-mind |
|  |

|  |
| --- |
| Corentin Piquerez  Juillet 2024 |



Table des matières

[1. Présentation de l’API 2](#_Toc171099441)

[2. Base de données 2](#_Toc171099442)

[3. Routes et appels 2](#_Toc171099443)

[3.1 Documentation en ligne 2](#_Toc171099444)

[3.2 Récupération d’un token de connexion 2](#_Toc171099445)

[3.3 Ajout d’un token de connexion 3](#_Toc171099446)

[3.4 Récupération des compétitions (toutes, la prochaine, la dernière) 3](#_Toc171099447)

[3.5 Ajout d’une compétition 4](#_Toc171099448)

[3.6 Modification d’une compétition 4](#_Toc171099449)

[3.7 Suppression d’une compétition 4](#_Toc171099450)

[3.8 Récupérer une image 4](#_Toc171099451)

[3.9 Récupérer des images 4](#_Toc171099452)

[3.10 Ajouter une image 5](#_Toc171099453)

[3.11 Récupération d’un texte 5](#_Toc171099454)

[3.12 Modification d’un texte 5](#_Toc171099455)

# Présentation de l’API

L’ API a été développée pour le site web open-mind. Cette API est une API JSON (appels et réponses) développée avec NodeJS. Elle permet de gérer la connexion administrateur, les compétitions, les images, l’actualité du club ainsi que tous les textes du site. Le but est que le site puisse évoluer grâce à une interface administrateur connectée directement à cette API. Cette API permet donc de récupérer des informations à afficher pour les visiteurs et de modifier ces informations via l’interface administrateur.

URL de l’API : {url}/api

Site web utilisé pour apprendre à se servir de NodeJS : <https://blog.postman.com/how-to-create-a-rest-api-with-node-js-and-express/#3>

# Base de données

Cette API est connectée à une base de données MONGODB possédant 3 tables, une stockant toutes les compétitions (type, date, lieux, résultat), une stockant les images (uniquement nom de l’image), et une stockant les différents textes du site.

# Routes et appels

## Documentation en ligne

**GET** /api/doc

## Statut de l’API

GET /api/status

## Connexion à la partie administrateur

**POST** /api/login

Paramètres :

{

    "email":" test@test.fr ",

    "password":"test"

}

Réponses :

**200 OK : Connexion réussie**

{

    "token":"0sqlf56sf6789...",

    "message":"Connexion réussie"

}

**403 Unauthorized : login ou mot de passe incorrect (pas dans la base de données)**

{

    "error":"Connexion échouée : identifiant ou mot de passe incorrect"

}

**404 Page not found : dans le cas où un paramètre serait manquant / mal passé**

{

    "error":"Information(s) manquante(s)"

}

## Ajout d’un compte de connexion

**POST** /api/login/new

Paramètres :

{

    "admin-email":"test@test.fr",

    "admin-password":"test",

    "new-admin-email":"test2@test.fr",

    "new-admin-password":"test2",

}

Réponses :

## Récupération des compétitions (toutes, la prochaine, la dernière)

**GET** /api/tournament/{type}

Type peut valoir les valeurs ci-dessous :

* all -> pour récupérer toutes les compétitions
* next -> pour récupérer la prochaine compétition
* last -> pour récupérer la dernière compétition

Exemple : **GET** /api/tournament/all

Réponses :

## Ajout d’une compétition

**POST** /api/tournament

Paramètres

{

    "title":"Championnat de France",

    "date":"1970-01-1",

    "location":"PARIS",

    "result":"1e place"

}

Le champ "result" n’est pas obligatoire et peut être modifié avec un autre appel API.

Réponses :

## Modification d’une compétition

**PUT** /api/tournament/{id}

Paramètres :

{

    "title":"Championnat de France",

    "date":"1970-01-1",

    "place":"PARIS",

    "result":"1e place"

}

Tous les champs sont optionnels, il en faut juste un au minimum.

Réponses :

## Suppression d’une compétition

**DELETE** /api/tournament/{id}

Réponses :

## Récupérer une image

**GET** /api/image/{id}

Réponses :

## Récupérer des images

**GET** /api/image/all

Réponses :

## 3.10 Ajouter une image

**POST** /api/image

Paramètres :

{

    "image":"...",

}

Réponses :

## Récupération d’un texte

**GET** /api/text/{id}

Réponses :

## Modification d’un texte

**PUT** /api/text/{id}

Paramètres :

{

    "text":"<p>Example</p>",

}

Réponses :