

# ProjetDPiscine

Documentation de l'API de ProjetDPiscine.

## Authentication

Toutes les requêtes liées à la gestion de la connexion/création de compte.

### GET Auth

localhost:8080/auth

Renvoie la page HTML de connexion/Inscription.

Example Request	Auth
<pre>curl --location --request GET 'localhost:8080/auth'</pre>	
Example Response	200 OK
<div><div>Body</div><div>Header (10)</div></div> <pre>&lt;!DOCTYPE html&gt; &lt;!--suppress HtmlFormInputWithoutLabel, JSDeprecatedSymbols --&gt; &lt;html lang="en"&gt;   &lt;head&gt;     &lt;meta charset="UTF-8"&gt;     &lt;title&gt;Title&lt;/title&gt;     &lt;link rel="stylesheet" href="../../css/app.css"&gt;     &lt;!--&lt;script src="../../controllers/controllerAuthentication.js"&gt;&lt;/script&gt;--&gt;</pre> <div>View More</div>	

localhost:8080/auth

Connecte l'utilisateur avec les informations qu'il a saisis.

BODY urlencoded

username

username

Username de l'utilisateur qui essaye de se connecter.

password

password

Mot de passe de l'utilisateur qui essaye de se connecter.

Example RequestAuth

curl --location --request POST 'localhost:8080/auth' \

--data-urlencode 'username=username' \

--data-urlencode 'password=password'

Example Response200 OK

BodyHeader (7)

OK

# Events

Toutes les requêtes liées aux évènements.

GET Events

http://localhost:8080/events/

Example Request

Events

```
curl --location --request GET 'http://localhost:8080/events/'
```

Example Response

200 OK

BodyHeader (7)

```
[
  {
    "user": "DPiscine",
    "id": 0,
    "title": "On va à la piscine",
    "date": "2022-10-20T13:00:00.000Z",
    "duration": "120",
    "description": "On nage 2km",
    "importance": "Très important"
  }
]
```

View More

GET Day

http://localhost:8080/events/day?date=YYYY-MM-JJ

Récupérer tous les évènements d'un jour.

PARAMS

**date**  
YYYY-MM-JJ  
Année-Mois-Jour

Example Request

Day

```
curl --location --request GET 'http://localhost:8080/events/day?date=2022-10-20'
```

Example Response

200 OK

```
[
  {
    "user": "DPiscine",
    "id": 0,
    "title": "On va à la piscine",
    "date": "2022-10-20T13:00:00.000Z",
    "duration": "120",
    "description": "On nage 2km",
    "image": "Tête de requête"
```

View More

### GET Week

http://localhost:8080/events/week?date=YYYY-MM-JJ

Récupérer tous les évènements d'une semaine.

### PARAMS

date

YYYY-MM-JJ

Example Request

Week

```
curl --location --request GET 'http://localhost:8080/events/week?date=2022-10-20'
```

Example Response

200 OK

Body

Header (7)

```
[
  {
    "user": "DPiscine",
    "id": 0,
    "title": "On va à la piscine",
    "date": "2022-10-20T13:00:00.000Z",
    "duration": "120",
    "description": "On nage 2km",
    "image": "Tête de requête"
```

View More

http://localhost:8080/events/month?date=YYYY-MM

Récupérer tous les évènements d'un mois.

PARAMS

date

YYYY-MM

Année-Mois

Example Request

Month

```
curl --location --request GET 'http://localhost:8080/events/month?date=2022-10'
```

Example Response

200 OK

Body

Header (7)

[  
 {  
 "user": "DPiscine",  
 "id": 0,  
 "title": "On va à la piscine",  
 "date": "2022-10-20T13:00:00.000Z",  
 "duration": "120",  
 "description": "On nage 2km",  
 "image": "Tête d'éléphant"  
 }  
]

View More

GET Year

http://localhost:8080/events/year?date=YYYY

Récupérer tous les évènements d'une année.

PARAMS

date

YYYY

Example Request

Year

```
curl --location --request GET 'http://localhost:8080/events/year?date=2022'
```

Example Response

200 OK

BodyHeader (7)

```
[
  {
    "user": "DPiscine",
    "id": 0,
    "title": "On va à la piscine",
    "date": "2022-10-20T13:00:00.000Z",
    "duration": "120",
    "description": "On nage 2km",
    "importance": "Très important"
  }
]
```

View More

POST Add

http://localhost:8080/events/add

Permet d'ajouter un évènement.

BODY urlencoded

title

title

Nom de l'évènement à ajouter.

date

YYYY-MM-DDTHH:mm:00.000Z

Date de l'évènement à ajouter. YYYY pour l'année, MM pour le mois, DD pour le jour HH pour l'heure et mm pour les minutes.

duration

120

Durée de l'évènement en minutes

https://documenter.getpostman.com/view/20927448/2s84DkUQs2

Description

Description de l'évènement.

**importance**

Normale

Importance de l'évènement (Normale, Important ou Très important).

Example Request

Add

```
curl --location --request POST 'http://localhost:8080/events/add' \
--data-urlencode 'title=title' \
--data-urlencode 'date=2022-10-20T15:00:00.000Z' \
--data-urlencode 'duration=120' \
--data-urlencode 'description=Description' \
--data-urlencode 'importance=Normale'
```

Example Response

200 OK

Body

Header (7)

OK

**DEL DeleteAll**

http://localhost:8080/events/delete/all

Requête supprimant tous les évènements du serveur.

Example Request

DeleteAll

```
curl --location --request DELETE 'http://localhost:8080/events/delete/all'
```

Example Response

200 OK

Body

Header (7)

## GET Event

http://localhost:8080/event

Permet d'afficher le modal de modification d'un évènement.

Example Request

Event

```
curl --location --request GET 'http://localhost:8080/event'
```

Example Response

404 Not Found

BodyHeader (8)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Error</title>
  </head>
  <body>
    <pre>Error: ENOENT: no such file or directory, stat &#39;C:\Users\Amaroke\WebstormProjects\Pro
  </body>
</html>
```

View More

## GET EventID

http://localhost:8080/event/0

Permet de récupérer un évènement depuis son id.

Example Request

EventID

```
curl --location --request GET 'http://localhost:8080/event/0'
```





Body Header (7)

```
{
  "user": "DPiscine",
  "id": 0,
  "title": "On va à la piscine",
  "date": "2022-10-20T13:00:00.000Z",
  "duration": "120",
  "description": "On nage 2km",
  "importance": "Très important"
}
```

## POST ModifyID

http://localhost:8080/modifyEvent/0

Permet de modifier un évènement selon son ID.

### BODY urlencoded

#### title

title

Nom de l'évènement après modification.

#### date

YYYY-MM-DDTHH:mm:00.000Z

Date après modification.

#### duration

120

Durée après modification.

#### description

Description

Description après modification.

#### importance

Normale

Importance après modification.



## Example Request

ModifyID

```
curl --location --request POST 'http://localhost:8080/modifyEvent/0' \  
--data-urlencode 'title=title' \  
--data-urlencode 'date=2022-12-09T20:00:00.000Z' \  
--data-urlencode 'duration=120' \  
--data-urlencode 'description=Description' \  
--data-urlencode 'importance=Normale'
```

## Example Response

200 OK

Body

 Header (7)

OK

**DEL DeleteID**

http://localhost:8080/deleteevent/0

Permet de supprimer un évènement selon son ID.

## Example Request

DeleteID

```
curl --location --request DELETE 'http://localhost:8080/deleteevent/0'
```

## Example Response

200 OK

Body

 Header (7)

OK

# Users

Toutes les requêtes liés à la gestion des comptes utilisateurs.

POST Add

http://localhost:8080/users/add

Requête permettant l'ajout d'un nouvel utilisateur.

BODY urlencoded

usernameSignUp

username  
L'username choisit par l'utilisateur qui s'inscrit.

passwordSignUp

password  
Le mot de passe choisit par l'utilisateur qui s'inscrit.

name

name  
Le nom de l'utilisateur qui s'inscrit.

firstName

firstName  
Le prénom de l'utilisateur qui s'inscrit.

Example Request

AddNewUser

```
curl --location --request POST 'http://localhost:8080/users/add' \
--data-urlencode 'usernameSignUp=username' \
--data-urlencode 'passwordSignUp=password' \
--data-urlencode 'name=name' \
--data-urlencode 'firstName=firstName'
```

Example Response

200 OK

Body

Header (7)

OK



## DEL DeleteAll

```
http://localhost:8080/users/delete/all
```

Requête supprimant tous les utilisateurs du serveur.

Example Request

DeleteAll

```
curl --location --request DELETE 'http://localhost:8080/users/delete/all'
```

Example Response

200 OK

Body

Header (7)

OK

## Easter egg

Easter egg lié au code HTTP 418.

## GET Coffee

```
http://localhost:8080/coffee
```

Petit Easter egg lié au code HTTP 418.

Example Request

Coffee

```
curl --location --request GET 'http://localhost:8080/coffee'
```

Body

Header (7)

I'm a Teapot

# Unknow Endpoint

Toutes les requêtes non traité par l'API termine ainsi.

## GET Unknow Endpoint

localhost:8080/UNKNOWN

UNKNOWN correspond à n'importe quel chemin non reconnu par l'API.

Example Request

Unknow Endpoint

```
curl --location --request GET 'localhost:8080/UNKNOWN'
```

Example Response

404 Not Found

Body

Header (7)