

Table des matières

1. [Endpoints et méthodes HTTP](#)
2. [Données envoyées par le frontend et réponses du backend](#)
3. [Modèles de données](#)

Endpoints et méthodes HTTP

Authentification

- **POST /login/** : Authentifier un utilisateur.
- **POST /logout/** : Déconnecter un utilisateur.

Utilisateur

- **GET /users/** : Récupérer la liste des utilisateurs.

Tournois

- **POST /create_tournament/** : Créer un nouveau tournoi.
- **POST /player_joined_tournament/** : Permettre à un joueur de rejoindre un tournoi.
- **POST /start_tournament/** : Démarrer un tournoi.
- **POST /finish_tournament/** : Terminer un tournoi.

Amis

- **POST /send_friend_request/** : Envoyer une demande d'ami.
- **POST /accept_friend_request/** : Accepter une demande d'ami.
- **POST /refuse_friend_request/** : Refuser une demande d'ami.

Jeu

- **POST /start_game/** : Commencer une partie.
- **POST /update_score/** : Mettre à jour le score.
- **POST /finish_game/** : Terminer une partie.
- **POST /cancel_game/** : Annuler une partie.

Données envoyées par le frontend et réponses du backend

Les échanges de données entre le frontend et le backend sont au format JSON

POST /login/

- **Données envoyées par le frontend :**

```
{
  "username": "string",
  "password": "string"
}
```

- Exemple de réponse du backend :

```
{
  "status": "success",
  "message": "User logged in"
}
```

POST /create_tournament/

- Données envoyées par le frontend :

```
{
  "name": "string"
}
```

- Exemple de réponse du backend :

```
{
  "status": "tournament_created",
  "tournament_id": "integer",
  "name": "string"
}
```

POST /player_joined_tournament/

- Données envoyées par le frontend :

```
{
  "tournament_id": "integer",
  "alias": "string"
}
```

- Exemple de réponse du backend :

```
{
  "status": "player_joined",
  "tournament_id": "integer",
  "player_id": "integer",
  "alias": "string",
  "order_of_turn": "integer"
}
```

POST /start_tournament/

- **Données envoyées par le frontend :**

```
{  
  "tournament_id": "integer"  
}
```

- **Exemple de réponse du backend :**

```
{  
  "status": "tournament_started",  
  "tournament_id": "integer",  
  "start_date": "datetime"  
}
```

POST /finish_tournament/

- **Données envoyées par le frontend :**

```
{  
  "tournament_id": "integer",  
  "winner_id": "integer"  
}
```

- **Exemple de réponse du backend :**

```
{  
  "status": "tournament_finished",  
  "tournament_id": "integer",  
  "end_date": "datetime",  
  "winner_id": "integer"  
}
```

POST /cancel_tournament/

- **Données envoyées par le frontend :**

```
{  
  "tournament_id": "integer"  
}
```

- **Exemple de réponse du backend :**

```
{
  "status": "tournament_canceled",
  "tournament_id": "integer"
}
```

- **Erreur possible :**

```
{
  "status": "error",
  "message": "Cannot cancel a started tournament"
}
```

POST /send_friend_request/

- **Données envoyées par le frontend :**

```
{
  "to_user_id": "integer"
}
```

- **Exemple de réponse du backend :**

```
{
  "status": "friend_request_sent",
  "request_id": "integer"
}
```

POST /accept_friend_request/

- **Données envoyées par le frontend :**

```
{
  "request_id": "integer"
}
```

- **Exemple de réponse du backend :**

```
{
  "status": "friend_request_accepted",

```

```
"request_id": "integer"
}
```

POST /refuse_friend_request/

- **Données envoyées par le frontend :**

```
{
  "request_id": "integer"
}
```

- **Exemple de réponse du backend :**

```
{
  "status": "friend_request_refused",
  "request_id": "integer"
}
```

POST /start_game/

- **Données envoyées par le frontend :**

```
{
  "opponent_id": "integer"
}
```

- **Exemple de réponse du backend :**

```
{
  "game_id": "integer",
  "status": "started"
}
```

POST /update_score/

- **Données envoyées par le frontend :**

```
{
  "game_id": "integer",
  "player_id": "integer",
  "new_score": "integer"
}
```

- **Exemple de réponse du backend :**

```
{
  "status": "score_updated",
  "game_id": "integer"
}
```

POST /finish_game/

- **Données envoyées par le frontend :**

```
{
  "game_id": "integer",
  "winner_id": "integer"
}
```

- **Exemple de réponse du backend :**

```
{
  "status": "finished",
  "game_id": "integer",
  "winner": "integer"
}
```

POST /cancel_game/

- **Données envoyées par le frontend :**

```
{
  "game_id": "integer"
}
```

- **Exemple de réponse du backend :**

```
{
  "status": "canceled",
  "game_id": "integer"
}
```

Modèles de données

Utilisateur (User)

```
{
  "id": "integer",
  "username": "string",
  "creation_date": "datetime",
  "langue": "string",
  "avatar": "string",
  "friends": ["list of user ids"],
  "groups": ["list of group ids"],
  "user_permissions": ["list of permission ids"]
}
```

Jeu (Game)

```
{
  "id": "integer",
  "player1": "integer",
  "player2": "integer",
  "player1_score": "integer",
  "player2_score": "integer",
  "status": "string",
  "winner": "integer",
  "created_at": "datetime"
}
```

Tournoi (Tournament)

```
{
  "id": "integer",
  "name": "string",
  "is_started": "boolean",
  "start_date": "datetime",
  "end_date": "datetime",
  "winner": "integer"
}
```

Participation (Participate)

```
{
  "id": "integer",
  "player": "integer",
  "tournament": "integer",
  "order_of_turn": "integer",
}
```

```
"alias": "string"  
}
```

Amis (Friendship)

```
{  
  "id": "integer",  
  "id_user_1": "integer",  
  "id_user_2": "integer"  
}
```