

Login

Este módulo permite iniciar sesión y obtener el *token* usando el estándar de OAuth2PasswordBearer.

Args: form_data (OAuth2PasswordRequestForm, optional): los datos del cliente. Requiere el nombre de usuario y la contraseña. El valor predeterminado es Depends(). db (AsyncSession, opcional): La instancia de la base de datos (automática). El valor predeterminado es Depends(get_db).

Levanta: : *HTTPException*: 403 si las credenciales son inválidas.

Regresa: dict[str, str | int]: El token de tipo portador estándar de json.

grant_type >	Grant Type (string) or Grant Type (null) (Grant Type)
username required	string (Username)
password required	string (Password)
scope	string (Scope) Default: ""
client_id >	Client Id (string) or Client Id (null) (Client Id)
client_secret >	Client Secret (string) or Client Secret (null) (Client Secret)

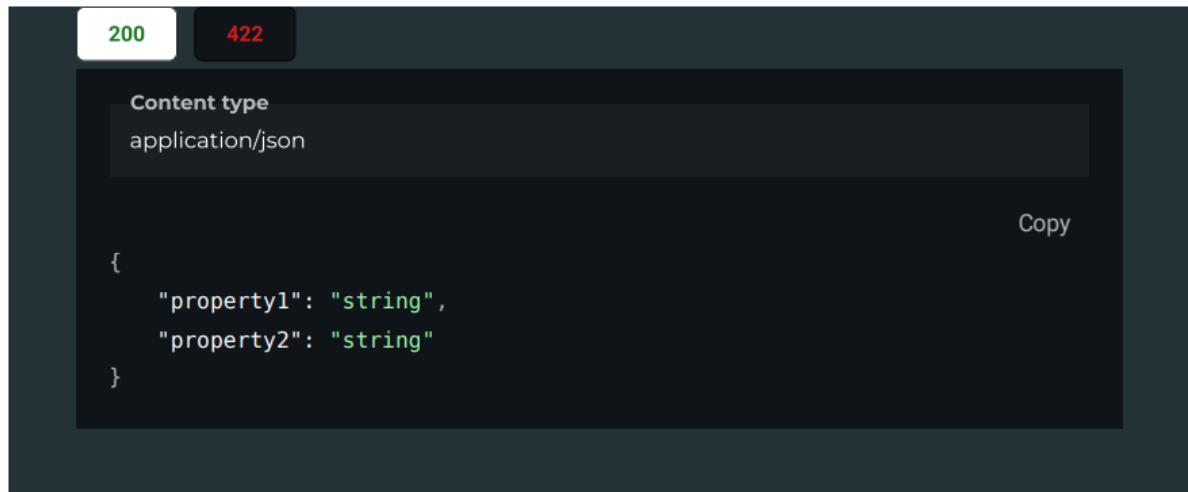
Respuestas

> 200 Successful Response

> 422 Validation Error

POST /token

Response samples



Create user (crear usuario)

`create_user` Crea y registra un nuevo usuario, Usa el correo electrónico como autenticación (no el nombre de usuario).

Args: `data (NewUser)`: Los datos del nuevo usuario. `db (AsyncSession, opcional)`: La sesión de la base de datos. El valor predeterminado es `Depends(get_db)`.

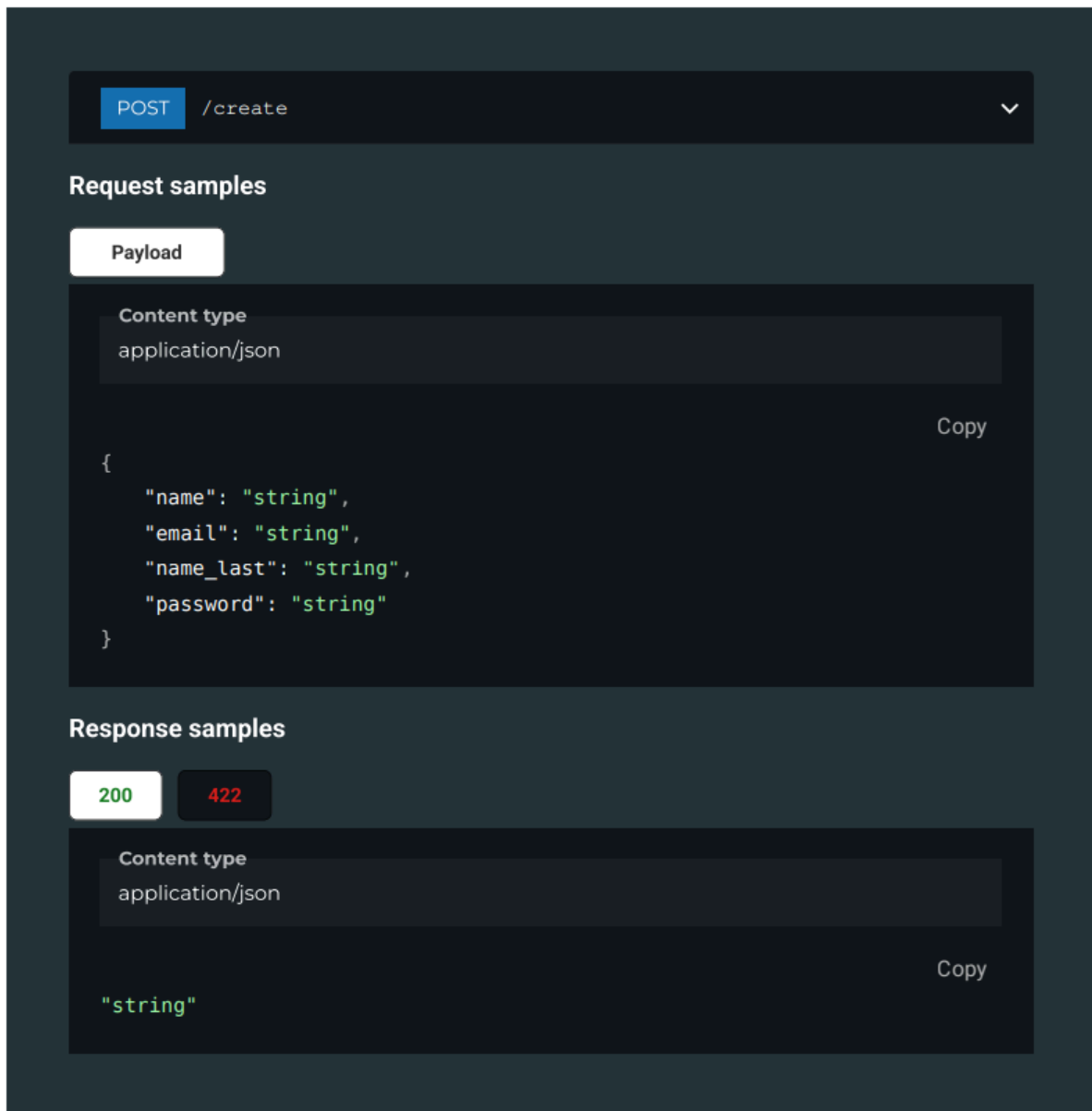
Regresa: `str`: Un mensaje de “éxito”.

<div>name</div> <div>required</div>	string (Name)
<div>email</div> <div>required</div>	string (Email)
<div>name_last</div> <div>required</div>	string (Name Last)
<div>password</div> <div>required</div>	string (Password)

Respuestas

> 200 Successful Response

> 422 Validation Error



Get Data (Obtener datos)

`get_data` obtiene los datos del usuario a desplegar en el *front-end*.

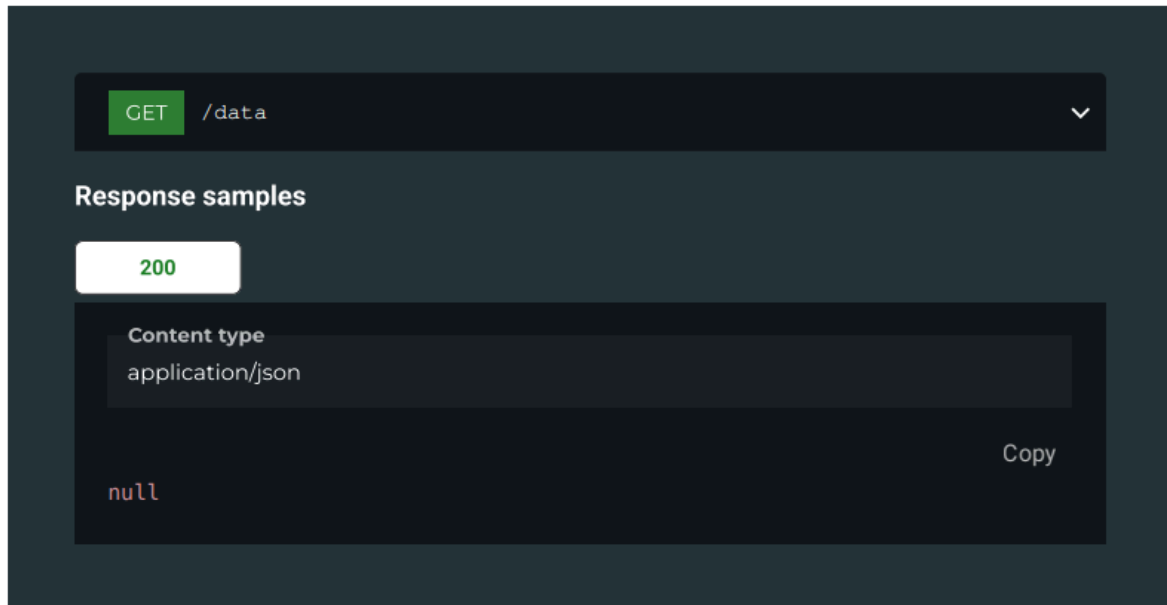
Args: `db` (`AsyncSession`, opcional): La sesión de la base de datos. El valor predeterminado es `Depends(get_db)`. `current_user` (`dict`, opcional): El token de portador requerido. El valor predeterminado es `Depends(get_current_user)`.

Levanta: *HTTPException*: 403 Usuario no autorizado

Regresa: `UserData`: La información correspondiente al usuario.

Respuestas

> 200 Successful Response



Get Leaderboard (Obtener tabla de clasificación)

`get_leaderboard` regresa la tabla de clasificación para el top 20 de jugadores.

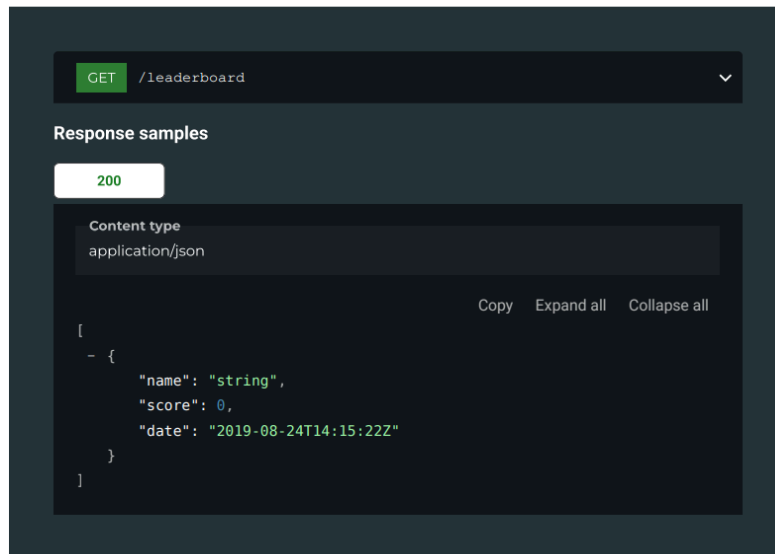
Args: Argumentos: `db` (`AsyncSession`, opcional): La sesión de base de datos (Fastapi). El valor predeterminado es `Depends(get_db)`. `Current_user` (`dict`, opcional): El token del usuario. El valor predeterminado es `Depends(get_current_user)`.

Levanta: *`HTTPException: 403`* No autorizado

Regresa: `list[LeaderboardEntry]`: Una lista de las entradas de la tabla de clasificación en orden.

Respuestas

> 200 Successful Response



GET /leaderboard

Response samples

200

Content type
application/json

Copy Expand all Collapse all

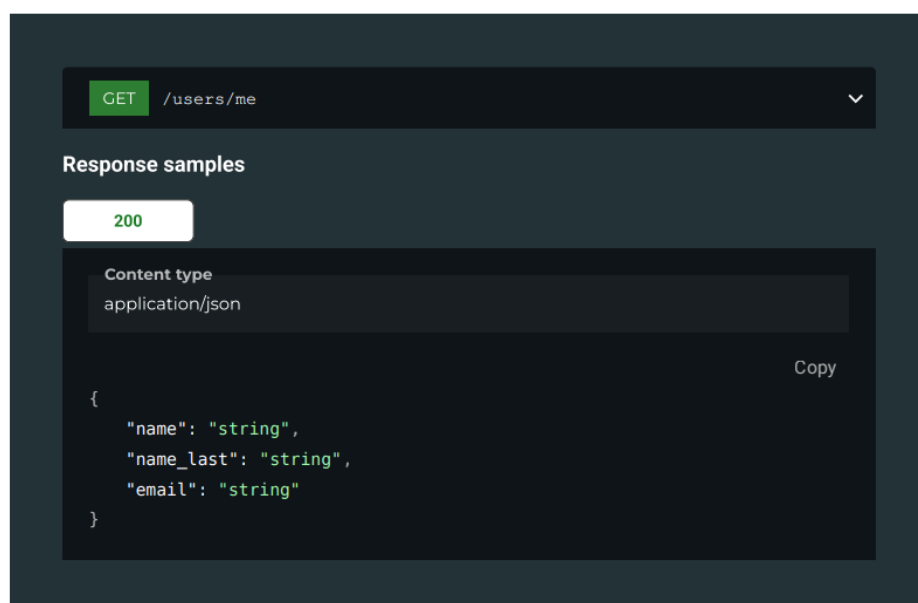
```
[
  - {
    "name": "string",
    "score": 0,
    "date": "2019-08-24T14:15:22Z"
  }
]
```

Read Users Me (Leer el perfil del usuario)

Devuelve el perfil mínimo del usuario actual (nombre, apellido, correo electrónico).

Respuestas

> 200 Successful Response



GET /users/me

Response samples

200

Content type
application/json

Copy

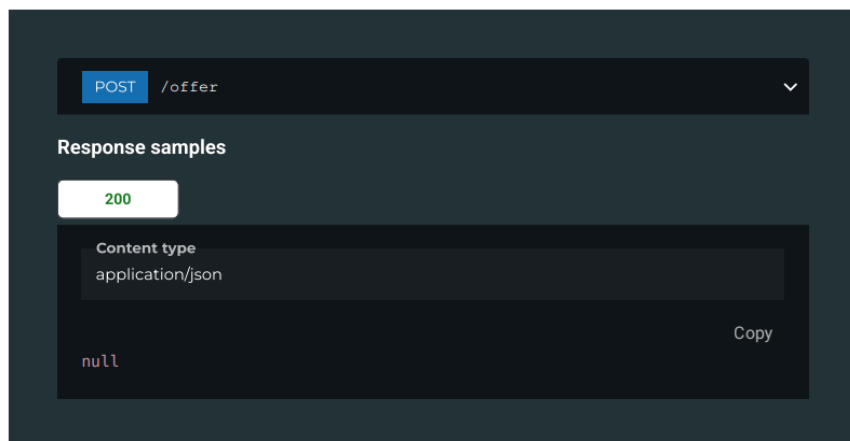
```
{
  "name": "string",
  "name_last": "string",
  "email": "string"
}
```

Offer

Crea una nueva sesión para la transmisión de vídeo y las acciones del usuario. Devuelve un *session_id* para que el cliente lo use en conexiones *WebSocket*.

Respuestas

> 200 Successful Response

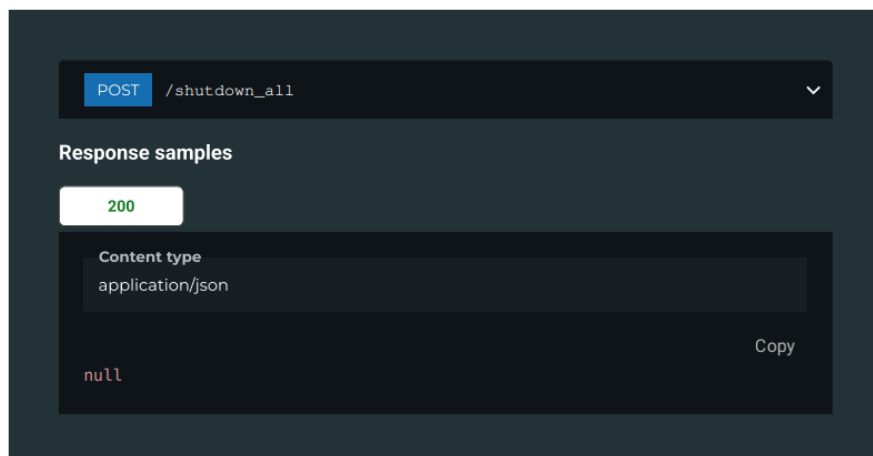


Shutdown All

Cierra a la fuerza todas las conexiones entre pares activos y limpia todas las sesiones.

Respuestas

> 200 Successful Response



Root (Raíz)

Verificación de estado predeterminado del *endpoint*.

Respuestas

> 200 Successful Response

GET /

▼

Response samples

200

Content type

application/json

Copy

null