# Guía de pruebas de API con Postman

## Configuración inicial

1. Descarga e instala Postman desde [postman.com](https://www.postman.com/downloads/)

2. Abre Postman y crea una nueva colección llamada "Sports Management API"

3. Dentro de esta colección, crea una nueva variable de entorno llamada `base\_url` con el valor `http://localhost:8000/api`

4. Crea otra variable de entorno llamada `token` (la llenaremos más tarde)

## Pruebas de endpoints

### 1. Registro de usuario (POST /register/)

- Método: POST

- URL: {{base\_url}}/register/

- Headers:

- Content-Type: application/json

- Authorization: Token {{admin\_token}}

- Body (raw JSON):

```json

{

"username": "newuser",

"password": "password123",

"email": "newuser@example.com",

"role": "PLAYER"

}

```

- Guardar la respuesta del token en la variable de entorno `token`

### 2. Inicio de sesión (POST /login/)

- Método: POST

- URL: {{base\_url}}/login/

- Headers:

- Content-Type: application/json

- Body (raw JSON):

```json

{

"username": "newuser",

"password": "password123"

}

```

- Guardar la respuesta del token en la variable de entorno `token`

### 3. Cierre de sesión (POST /logout/)

- Método: POST

- URL: {{base\_url}}/logout/

- Headers:

- Authorization: Token {{token}}

### 4. Listar usuarios (GET /users/)

- Método: GET

- URL: {{base\_url}}/users/

- Headers:

- Authorization: Token {{admin\_token}}

### 5. Obtener detalles de usuario (GET /users/{id}/)

- Método: GET

- URL: {{base\_url}}/users/1/

- Headers:

- Authorization: Token {{admin\_token}}

### 6. Actualizar usuario (PUT /users/{id}/)

- Método: PUT

- URL: {{base\_url}}/users/1/

- Headers:

- Content-Type: application/json

- Authorization: Token {{admin\_token}}

- Body (raw JSON):

```json

{

"username": "updateduser",

"email": "updated@example.com",

"role": "COACH"

}

```

### 7. Eliminar usuario (DELETE /users/{id}/)

- Método: DELETE

- URL: {{base\_url}}/users/1/

- Headers:

- Authorization: Token {{admin\_token}}

## Notas importantes

- Reemplaza `{{admin\_token}}` con un token válido de administrador.

- Para las operaciones que requieren permisos de administrador, asegúrate de usar el token de un usuario con rol de administrador.

- Los IDs de usuario en las URLs (como en /users/1/) deben ser reemplazados con IDs válidos de tu base de datos.