

# FS.FakeTwitter API

## Overview

## Security

Type	Name	Scopes
apiKey	ApiKey	
apiKey	Bearer	

## Paths

### **POST** /api/Auth/login

#### Responses

Code	Description	Links
200	Success	No Links

### **POST** /api/Follow Permite a un usuario seguir a otro usuario.

#### Responses

Code	Description	Links
200	Follow realizado correctamente	No Links
400	Datos inválidos  <i>Content</i> application/json	No Links

### **GET** /api/Follow/followers/{userId} Obtiene la lista de usuarios que siguen a un usuario determinado.

#### Parameters

Type	Name	Description	Schema
path	userId <i>required</i>	ID del usuario del que se desea obtener los seguidores.	string

#### Responses

Code	Description	Links
200	Lista de seguidores obtenida correctamente  <i>Content</i> <b>application/json</b>	No Links
404	Usuario no encontrado  <i>Content</i> <b>application/json</b>	No Links

## **GET /api/Follow/following/{userId}** Obtiene la lista de usuarios a los que sigue un usuario determinado.

### Parameters

Type	Name	Description	Schema
path	<b>userId</b> <i>required</i>	ID del usuario del que se desea obtener los seguidos.	string

### Responses

Code	Description	Links
200	Lista de seguidos obtenida correctamente  <i>Content</i> <b>application/json</b>	No Links
404	Usuario no encontrado  <i>Content</i> <b>application/json</b>	No Links

## **POST /api/Tweet** Publica un nuevo tweet.

### Responses

Code	Description	Links
200	Tweet creado correctamente  <i>Content</i> <b>application/json</b>	No Links
400	Contenido inválido  <i>Content</i> <b>application/json</b>	No Links

## ***GET /api/Tweet/user/{userId}*** Obtiene todos los tweets publicados por un usuario.

### Parameters

Type	Name	Description	Schema
path	<b>userId</b> <i>required</i>	ID del usuario.	string

### Responses

Code	Description	Links
200	Tweets obtenidos correctamente  <i>Content</i> <b>application/json</b>	No Links
404	Usuario no encontrado  <i>Content</i> <b>application/json</b>	No Links

## ***GET /api/Tweet/timeline/{userId}*** Obtiene el timeline (tweets) de los usuarios que sigue un usuario específico.

### Parameters

Type	Name	Description	Schema
path	<b>userId</b> <i>required</i>	ID del usuario que consulta el timeline.	string

### Responses

Code	Description	Links
200	Timeline obtenido correctamente  <i>Content</i> <b>application/json</b>	No Links
404	Usuario no encontrado  <i>Content</i> <b>application/json</b>	No Links

## **GET /api/User** Obtiene todos los usuarios registrados

### Responses

Code	Description	Links
200	Success	No Links

## **POST /api/User** Crea un nuevo usuario.

### Responses

Code	Description	Links
200	Success	No Links

## **GET /api/User/{id}** Obtiene un usuario por su ID.

### Parameters

Type	Name	Description	Schema
path	id <i>required</i>	Identificador del usuario.	string (uuid)

### Responses

Code	Description	Links
200	Success	No Links

## **PUT /api/User/{id}** Actualiza un usuario existente.

### Parameters

Type	Name	Description	Schema
path	id <i>required</i>	ID del usuario a actualizar.	string (uuid)

### Responses

Code	Description	Links
200	Success	No Links

## **DELETE /api/User/{id}** Elimina un usuario por ID (soft delete).

### Parameters

Type	Name	Description	Schema
path	id <i>required</i>	ID del usuario a eliminar.	string (uuid)

#### Responses

Code	Description	Links
200	Success	No Links

# Components

## Schemas

### CreateUserCommand

#### Properties

Name	Description	Schema
username <i>optional</i>	<i>nullable</i>	string
email <i>optional</i>	<i>nullable</i>	string

### FollowUserCommand

#### Properties

Name	Description	Schema
followerId <i>optional</i>	<i>nullable</i>	string
followId <i>optional</i>	<i>nullable</i>	string

### LoginRequest

#### Properties

Name	Description	Schema
email <i>optional</i>	<i>nullable</i>	string
password <i>optional</i>	<i>nullable</i>	string
twoFactorCode <i>optional</i>	<i>nullable</i>	string

Name	Description	Schema
twoFactorRecoveryCode <i>optional</i>	<i>nullable</i>	string

## PostTweetCommand

### Properties

Name	Description	Schema
userId <i>optional</i>	<i>nullable</i>	string
content <i>optional</i>	<i>nullable</i>	string

## ProblemDetails

### Properties

Name	Description	Schema
type <i>optional</i>	<i>nullable</i>	string
title <i>optional</i>	<i>nullable</i>	string
status <i>optional</i>	<i>nullable</i>	integer (int32)
detail <i>optional</i>	<i>nullable</i>	string
instance <i>optional</i>	<i>nullable</i>	string

## UpdateUserCommand

### Properties

Name	Description	Schema
id <i>optional</i>		string (uuid)
username <i>optional</i>	<i>nullable</i>	string
email <i>optional</i>	<i>nullable</i>	string