

# TITULO DEL PROYECTO

BLOGAPP:  
PLATAFORMA WEB  
CON API REST Y FRONTEND

**CURSO: DESARROLLO DE APLICACIONES WEB I**

**CICLO: V**

**SECCION:**

**SEMESTRE: 2025 – 02**

**PROFESOR: Franklin Ruiz Asto Leon**

**COORDINADOR:**

**Leonardo Raphael Pereyra Pacheco (i202401202)**

**INTEGRANTES**

- 1. Sebastian Daniel Lopez Huaman (i202213150)**
- 2. Anthony Aldair Espinoza Perez (i202400317)**

# RESUMEN

El proyecto BlogApp es una API REST desarrollada con Spring Boot que permite la gestión de usuarios, publicaciones y comentarios. Su funcionalidad incluye la creación, consulta, listado y eliminación de estos elementos, proporcionando endpoints estructurados y documentados con Swagger.

La API utiliza Spring Boot, MySQL y Hibernate (JPA) para garantizar una arquitectura modular y escalable. Su diseño RESTful facilita la integración con cualquier frontend o aplicación externa que requiera gestionar contenido de blogs de manera eficiente y segura.

# INTRODUCCIÓN

En la actualidad, la gestión de contenido digital es fundamental para la interacción en línea, especialmente en plataforma de blogs, donde los usuarios pueden compartir publicaciones y generar discusiones a través de comentarios. Para facilitar esta dinámica, se requiere una infraestructura backend eficiente que permita administrar estos elementos de manera estructurada y accesible. Este proyecto no solo ofrece una solución para la gestión de blogs, sino que también establece una base sólida para futuras mejoras, como la implementación de autenticación y autorización, optimización de consultas y mejoras en la seguridad de los datos.

# DIAGNOSTICO

1

El desarrollo de la API REST de BlogApp ha sido exitoso, validado mediante pruebas con Postman y Swagger UI, garantizando respuestas consistentes y una interacción adecuada con la base de datos.

2

La arquitectura modular permite futuras mejoras como la integración de autenticación y optimización de consultas.

3

Sin embargo, se pueden mejorar aspectos como la seguridad con JWT, la gestión de errores y el rendimiento en consultas con paginación.

4

En general, la API es eficiente, bien estructurada y lista para integrarse con frontend u otras aplicaciones.

# JUSTIFICACIÓN DEL PROYECTO:

BlogApp es una plataforma que facilita la creación y gestión de blogs, utilizando Spring Boot y Thymeleaf. Su objetivo es ofrecer una solución sencilla para bloggers y usuarios, promoviendo la interacción y el intercambio de ideas.

## Beneficiarios directos:

- Creadores de contenido (bloggers): Usuarios que desean una plataforma fácil de usar para crear, editar y gestionar sus publicaciones.
- Usuarios de la plataforma: Personas interesadas en consumir contenido, participar en debates y seguir temas de su interés.

## Beneficiarios indirectos:

- Desarrolladores: Interesados en aprender y aplicar tecnologías modernas como Spring Boot y Thymeleaf.
- Empresas de marketing y publicidad: Que pueden utilizar la plataforma para promocionar productos o Servicios mediante contenido patrocinado o colaboraciones con bloggers.
- Comunidad en general: Personas que buscan información, entretenimiento y aprendizaje a través del contenido generado por los usuarios.



# OBJETIVOS

Lograr una plataforma web de blogs eficiente en términos de funcionalidad, usabilidad y diseño, que permita a los usuarios crear, editar y gestionar contenido de manera efectiva.

## OBJETIVOS ESPECÍFICOS:

### MEJORAR LA FUNCIONALIDAD DE REGISTRO Y AUTENTICACIÓN DE USUARIOS.

El objetivo es implementar un sistema seguro para crear cuentas y acceder a la plataforma, con un tiempo de registro menor a 3 segundos y una tasa de éxito en autenticación del 90%. Se utilizará Spring Security y se realizarán pruebas de funcionalidad. La implementación debe completarse en 2 meses.

### OPTIMIZAR LA INTERFAZ DE CREACIÓN Y EDICIÓN DE BLOGS

El objetivo es mejorar la creación y edición de publicaciones, garantizando accesibilidad en dispositivos móviles y de escritorio. Se medirá la satisfacción del usuario, buscando que el 90% pueda crear y editar blogs sin ayuda. Se aplicará diseño responsivo y pruebas de usabilidad. La optimización debe completarse en 3 meses.

### INCORPORAR UNA SECCIÓN DE COMENTARIOS INTERACTIVOS EN LAS PUBLICACIONES.

El objetivo es integrar una sección de comentarios en cada publicación para fomentar la interacción entre usuarios. Se medirá el número de comentarios y la tasa de interacción, con la meta de alcanzar 100 comentarios diarios en publicaciones populares. Se usará Spring Boot y Thymeleaf para crear el sistema, y la implementación debe completarse en 4 meses.

# DESARROLLO DEL PROYECTO:



# USUARIO - CONTROLLER

# AUTENTICACION DE USUARIO

The screenshot displays a REST client interface with a dark theme. The top bar shows the URL `http://localhost:8080/public/api/auth/login`. Below this, the request method is set to **POST**. The request body is a JSON object with the following structure:

```
{
  "username": "nrojas",
  "password": "12345"
}
```

The response status is **200 OK** with a response time of **325 ms** and a body size of **902 B**. The response body is a JSON object containing an access token, a refresh token, and an expiration time:

```
{
  "data": {
    "accessToken": "eyJhbGciOiJIUzUxMiJ9.eyJyY2xlcYI6WyJVVU0VSIiwidU9ERVJBVE9SI0sIm5hbWUiOiJ0b2VsaWEgUm9qYXMiLCJzdWIiOiJucm9qYXMiLCJpYXQiOiJE3NTY2NDU2MDQsImV4cCI6MTc1NjY0NjUwNH0.pM--sdJyvHxIy9B41q2rJjz-04zSDyWrr8QAaMZiGsWfMawgZtIiX6Rsb3YLLr02WbzGa8tjc-YxZZf8L5Y-9Q",
    "refreshToken": "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJucm9qYXMiLCJpYXQiOiJE3NTY2NDU2MDQsImV4cCI6MTc1NjY0NjUwNH0.zhk6_7B7nqNpj_6wPFXXSEwSIqTDZARmR7dtsd1eAnC6k2qS2bhVIS3JvcBGEyRQiycuI1CDcVl0jlpW6PipGA",
    "expiresIn": 900
  }
}
```

# REGISTRAR USUARIOS

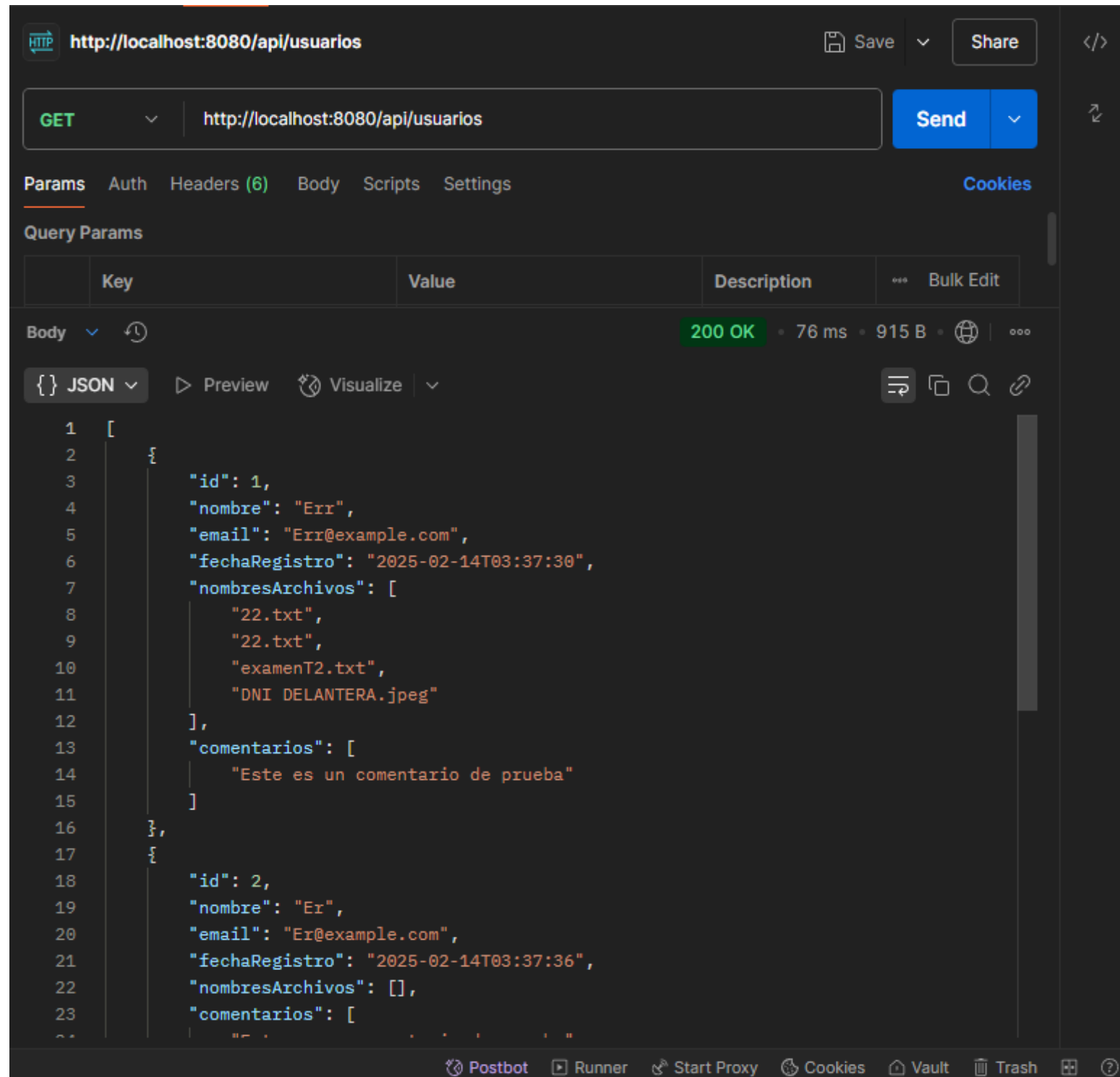
The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8080/api/usuarios/registrar`
- Method:** `POST`
- Body (raw):**

```
1 {  
2   "nombre": "Alfonso",  
3   "email": "Alfonso@example.com",  
4   "password": "123456"  
5 }  
6
```
- Status:** `200 OK` (131 ms, 308 B)
- Body (JSON):**

```
1 {  
2   "id": 4,  
3   "nombre": "Alfonso",  
4   "email": "Alfonso@example.com",  
5   "password": "123456",  
6   "fechaRegistro": "2025-02-19T20:12:41.5837351",  
7   "publicaciones": null  
8 }
```

# OBTENER USUARIOS



HTTP <http://localhost:8080/api/usuarios> Save Share

GET <http://localhost:8080/api/usuarios> Send

Params Auth Headers (6) Body Scripts Settings Cookies

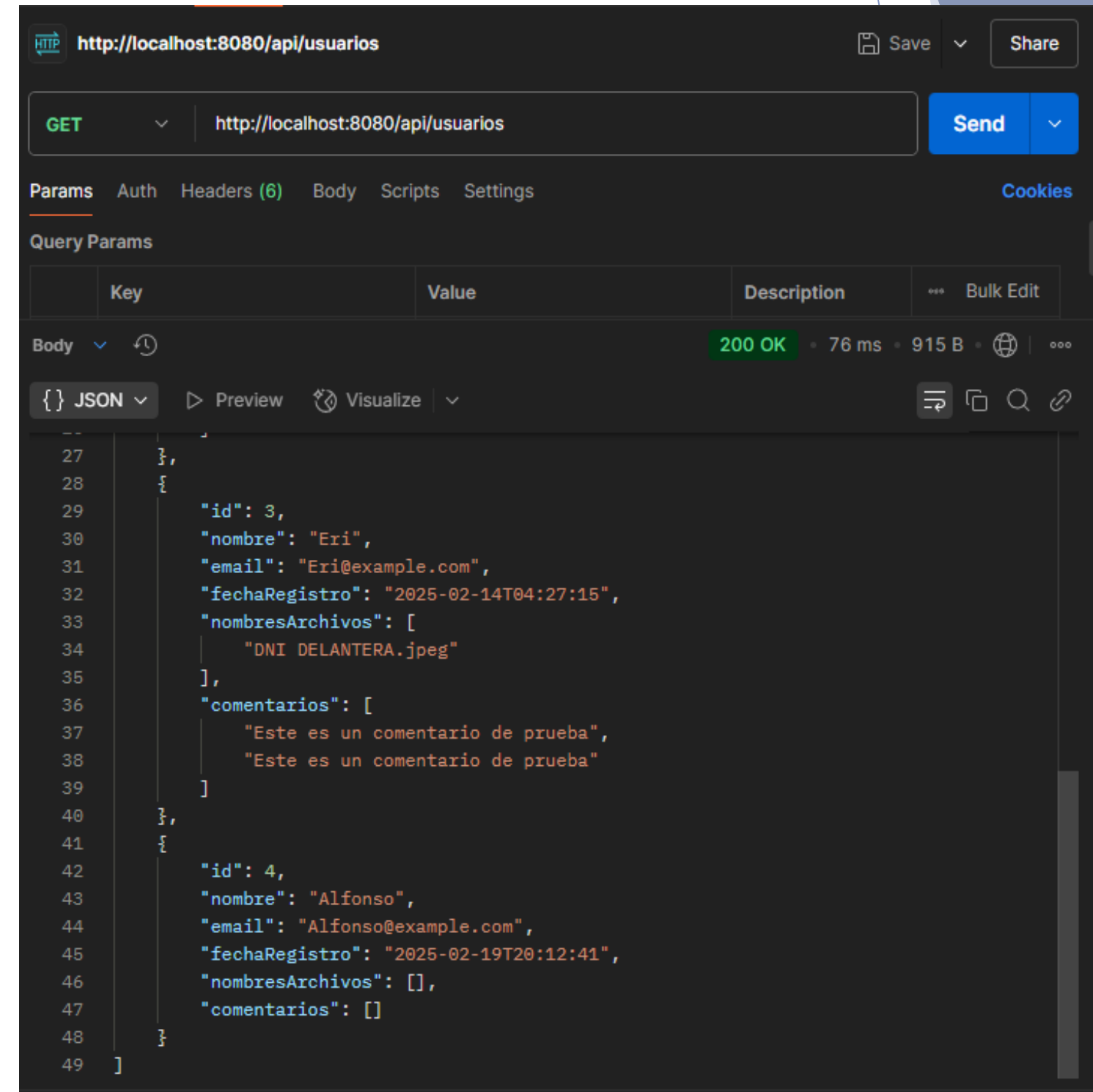
Query Params

Key	Value	Description	Bulk Edit
-----	-------	-------------	-----------

Body 200 OK • 76 ms • 915 B Visualize

JSON Preview

```
1 [
2   {
3     "id": 1,
4     "nombre": "Err",
5     "email": "Err@example.com",
6     "fechaRegistro": "2025-02-14T03:37:30",
7     "nombresArchivos": [
8       "22.txt",
9       "22.txt",
10      "examenT2.txt",
11      "DNI DELANTERA.jpeg"
12    ],
13    "comentarios": [
14      "Este es un comentario de prueba"
15    ]
16  },
17  {
18    "id": 2,
19    "nombre": "Er",
20    "email": "Er@example.com",
21    "fechaRegistro": "2025-02-14T03:37:36",
22    "nombresArchivos": [],
23    "comentarios": [
```



HTTP <http://localhost:8080/api/usuarios> Save Share

GET <http://localhost:8080/api/usuarios> Send

Params Auth Headers (6) Body Scripts Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
-----	-------	-------------	-----------

Body 200 OK • 76 ms • 915 B Visualize

JSON Preview

```
27 },
28 {
29   "id": 3,
30   "nombre": "Eri",
31   "email": "Eri@example.com",
32   "fechaRegistro": "2025-02-14T04:27:15",
33   "nombresArchivos": [
34     "DNI DELANTERA.jpeg"
35   ],
36   "comentarios": [
37     "Este es un comentario de prueba",
38     "Este es un comentario de prueba"
39   ]
40 },
41 {
42   "id": 4,
43   "nombre": "Alfonso",
44   "email": "Alfonso@example.com",
45   "fechaRegistro": "2025-02-19T20:12:41",
46   "nombresArchivos": [],
47   "comentarios": []
48 }
49 ]
```

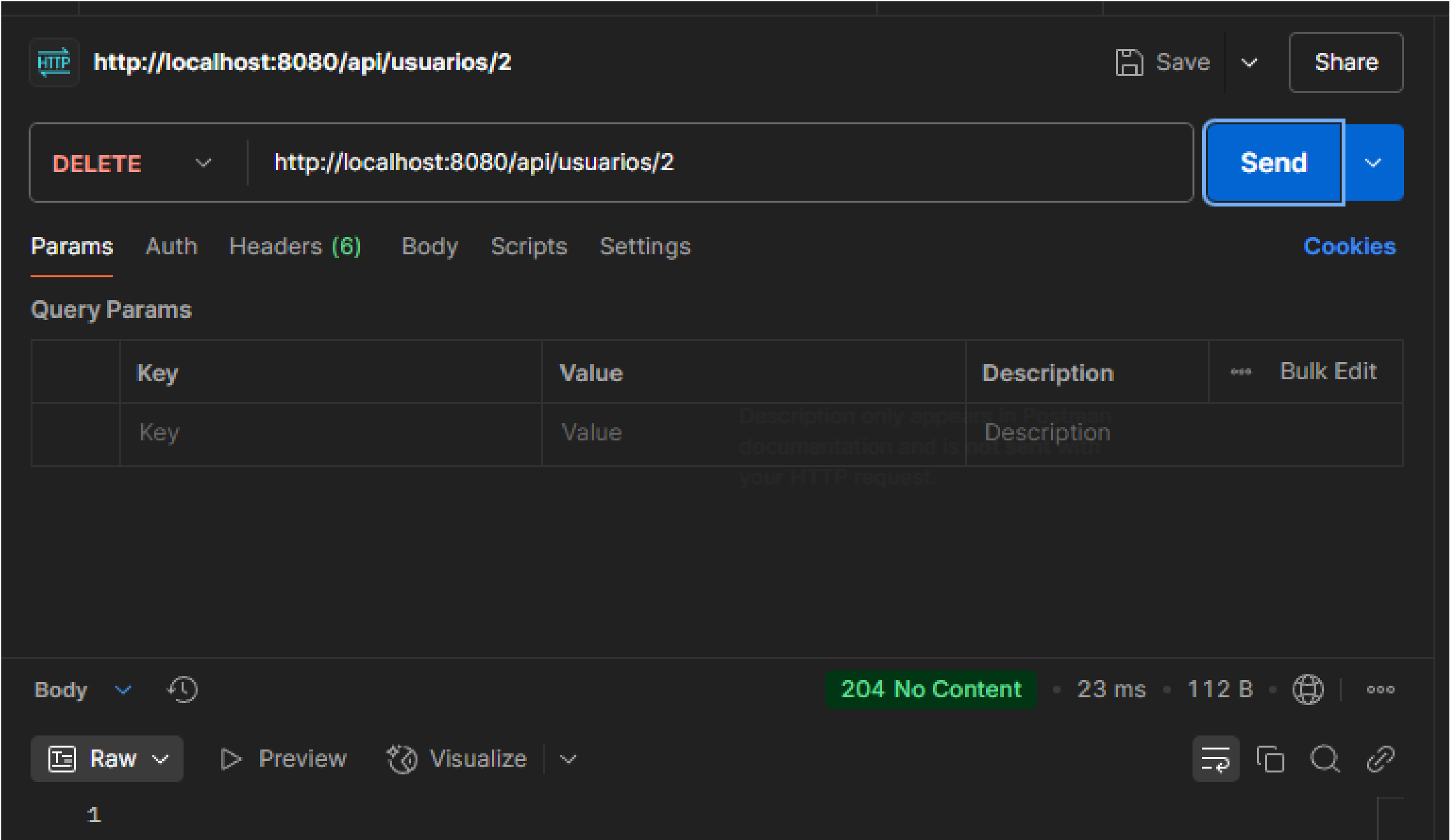
# OBTENER USUARIOS POR ID

The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8080/api/usuarios/3`
- Method:** `GET`
- Status:** `200 OK` (16 ms, 376 B)
- Response Body (JSON):**

```
1 {
2   "id": 3,
3   "nombre": "Eri",
4   "email": "Eri@example.com",
5   "fechaRegistro": "2025-02-14T04:27:15",
6   "nombresArchivos": [
7     "DNI DELANTERA.jpeg"
8   ],
9   "comentarios": [
10    "Este es un comentario de prueba",
11    "Este es un comentario de prueba"
12  ]
13 }
```
- Query Params Table:**

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		



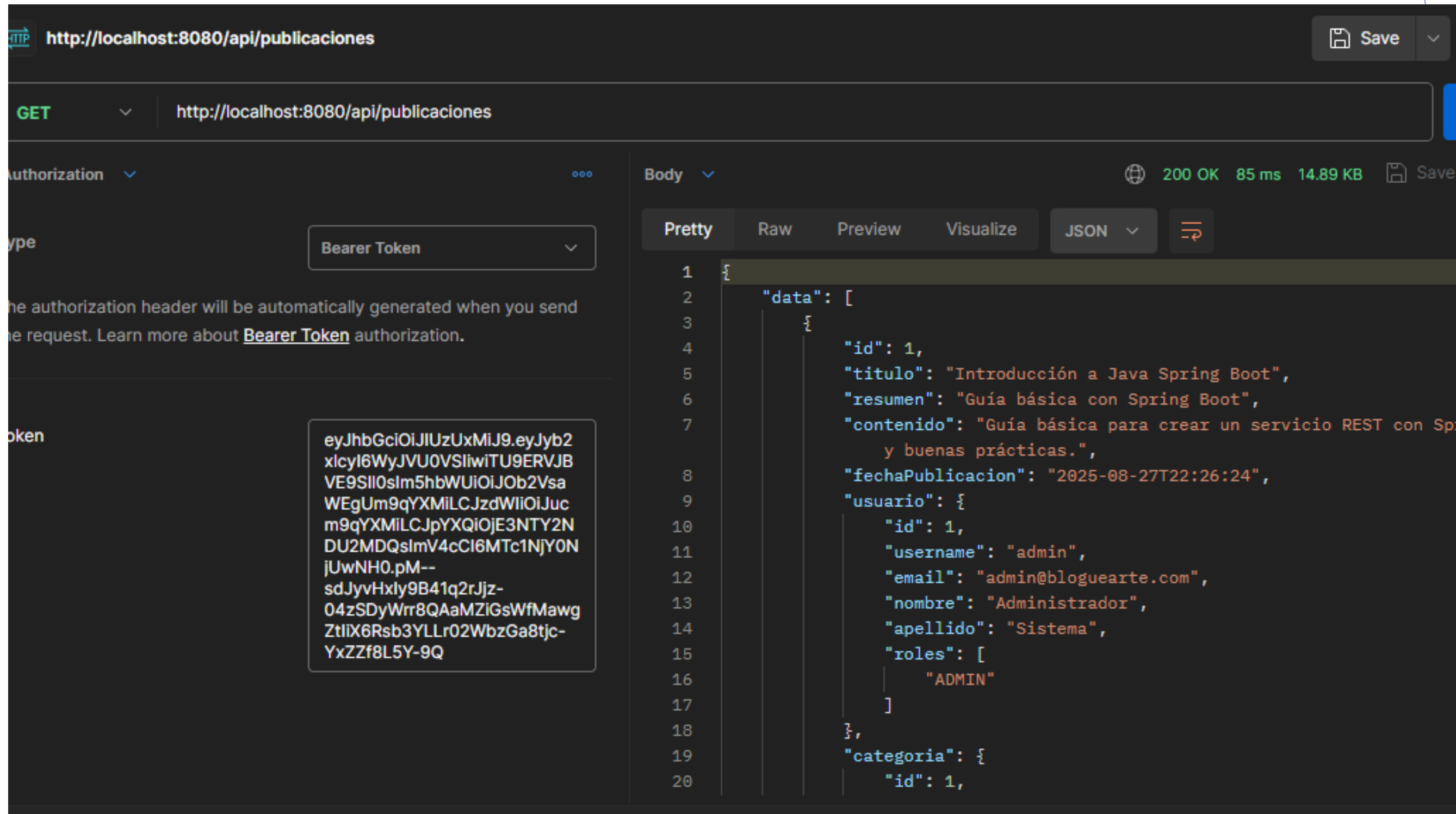
# ELIMINAR USUARIO

	id	nombre	email	password	fecha_registro
▶	1	Err	Err@example.com	123456	2025-02-14 08:37:30
	3	Eri	Eri@example.com	123456	2025-02-14 09:27:15
	4	Alfonso	Alfonso@example.com	123456	2025-02-20 01:12:41
⚙	NULL	NULL	NULL	NULL	NULL



# PUBLICACIÓN- CONTROLLER

# LISTAR TODAS LAS PUBLICACIONES

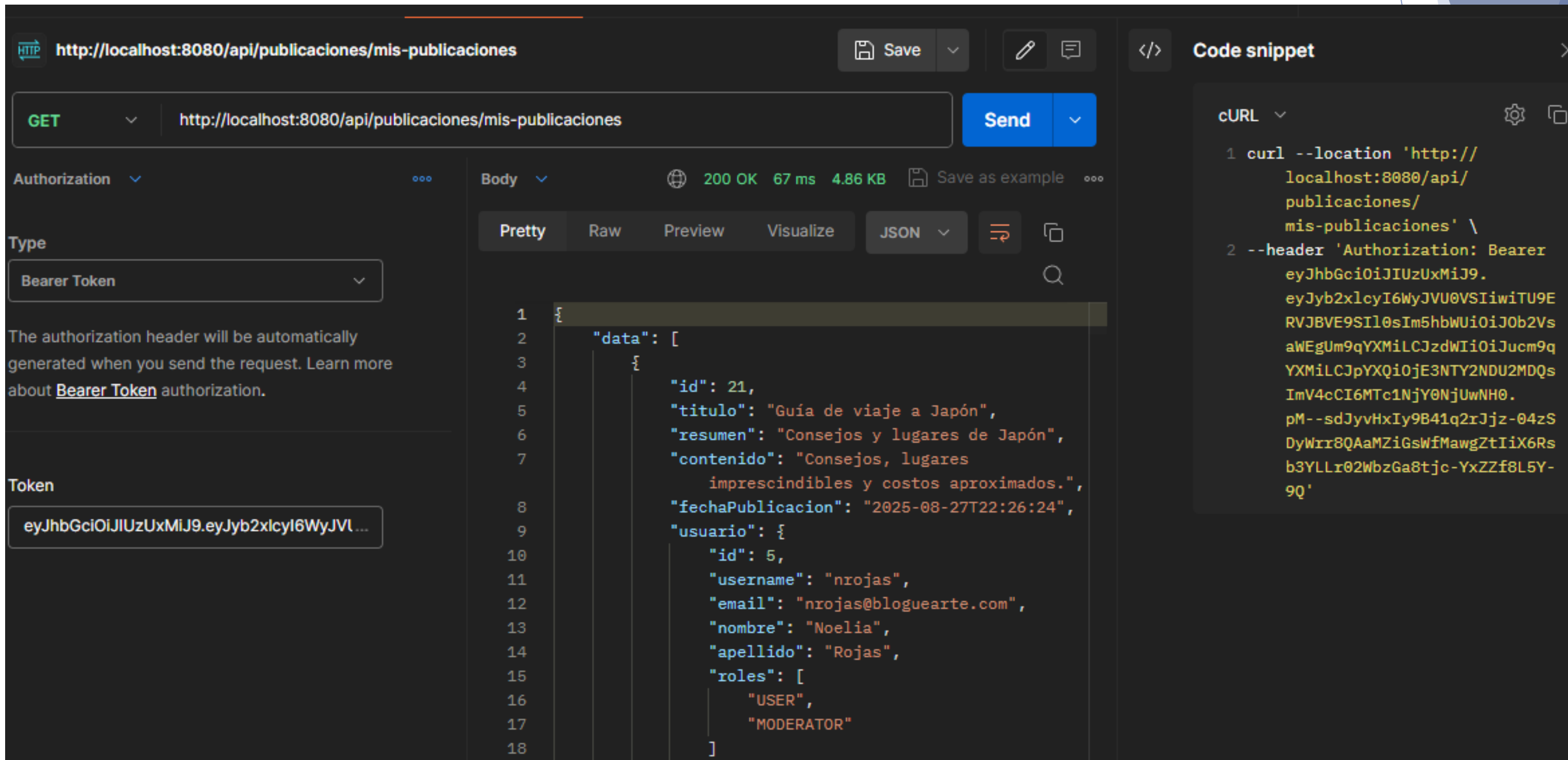


The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8080/api/publicaciones`
- Method:** `GET`
- Authorization:** Bearer Token
- Response Status:** `200 OK`, `85 ms`, `14.89 KB`
- Response Body (JSON):**

```
{
  "data": [
    {
      "id": 1,
      "titulo": "Introducción a Java Spring Boot",
      "resumen": "Guía básica con Spring Boot",
      "contenido": "Guía básica para crear un servicio REST con Spring Boot y buenas prácticas.",
      "fechaPublicacion": "2025-08-27T22:26:24",
      "usuario": {
        "id": 1,
        "username": "admin",
        "email": "admin@bloguearte.com",
        "nombre": "Administrador",
        "apellido": "Sistema",
        "roles": [
          "ADMIN"
        ]
      },
      "categoria": {
        "id": 1,
```

# LISTAR PUBLICACIONES POR USUARIO



# OBTENER PUBLICACIONES POR ID

The screenshot displays a REST client interface with the following components:

- URL Bar:** Shows the endpoint `http://localhost:8080/api/publicaciones/5`.
- Method:** Set to `GET`.
- Authorization:** A Bearer Token is provided: `eyJhbGciOiJIUzUxMiJ9.eyJyYb2xlcyl6WyJ'...`.
- Response:** A `200 OK` status with a response time of `24 ms` and a size of `1.07 KB`. The body is formatted as JSON and contains the following data:

```
{
  "data": {
    "id": 5,
    "titulo": "Patrones de diseño en Backend",
    "resumen": "Patrones de arquitectura modernos",
    "contenido": "Un repaso por los principales patrones usados en arquitecturas modernas.",
    "fechaPublicacion": "2025-08-27T22:26:24",
    "activo": true,
    "usuario": {
      "id": 1,
      "username": "admin",
      "email": "admin@bloguearte.com",
      "passwordHash": "$2a$10$MTc9fMP43wx6M3P3AC159.ol6TBsx02ceMSXeAwYTB1CVjTwm6RbS",
      "nombre": "Administrador",
      "apellido": "Sistema",
      "activo": true,
      "roles": [
        {
          "id": 1,
```

# ACTUALIZAR PUBLICACION

HTTP

http://localhost:8080/api/publicaciones/mis-publicaciones/8

Save

PUT

http://localhost:8080/api/publicaciones/mis-publicaciones/8

Body

rawJSONBeautify

```
1 {
2   "titulo": "Guía de viaje a Japón 5000",
3   "resumen": "Consejos y lugares de Japón 5000",
4   "contenido": "Consejos, lugares imprescindibles y
5               costos aproximados 5000.",
6   "categoria": {
7     "id": 4
8   },
9   "activo": true
10 }
```

Body

PrettyRawPreviewVisualizeJSON

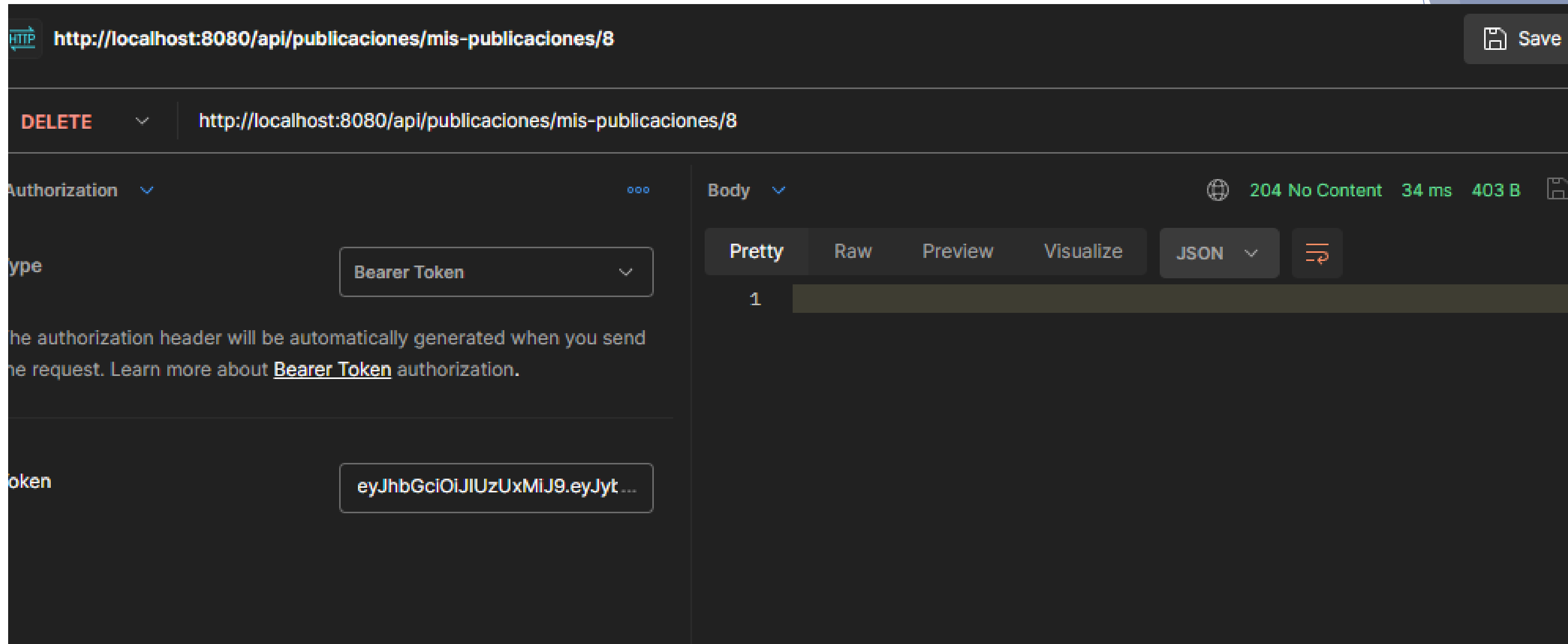
```
1 {
2   "data": {
3     "id": 8,
4     "titulo": "Guía de viaje a Japón 5000",
5     "resumen": "Consejos y lugares de Japón 5000",
6     "contenido": "Consejos, lugares imprescindibles y costos apro
7                 5000.",
8     "fechaPublicacion": "2025-08-27T22:26:24",
9     "activo": true,
10    "usuario": {
11      "id": 2,
12      "username": "jlopez",
13      "email": "jlopez@bloguearte.com",
14      "passwordHash": "$2a$10$MTc9fMP43wx6M3P3AC159.
15                    ol6TBsx02ceMSXeAwYTB1CVjTwm6RbS",
16      "nombre": "Jose",
17      "apellido": "Lopez",
18      "activo": true,
19      "roles": [
20        {
21          "id": 2,
```

id	titulo	resumen	contenido	fecha_publicaci
7	Cómo hacer pan casero	Pan desde cero en casa	Explicación detallada para hacer ...	2025-08-27 22:
8	Postres con chocolate	Ideas dulces con cacao	Ideas dulces para los amantes d...	2025-08-27 22:
9	La ciencia detrás del ...	Cómo se tuesta y pr...	Un análisis de cómo se tuesta y ...	2025-08-27 22:
10	Recetas vegetariana...	Platos saludables sin ...	Platos fáciles, saludables y delici...	2025-08-27 22:
11	Introducción a las der...	Qué son las derivadas	Explicación sencilla de qué son y ...	2025-08-27 22:
12	Álgebra lineal aplicada	Casos prácticos de ál...	Casos prácticos de álgebra en pr...	2025-08-27 22:

id	titulo	resumen	contenido	fecha_p
7	Cómo hacer pan casero	Pan desde cero en casa	Explicación detallada para hacer ...	2025-08-
8	Guía de viaje a Japón 5000	Consejos y lugares de Japón 5000	Consejos, lugares imprescindibile...	2025-08-
9	La ciencia detrás del café	Cómo se tuesta y prepara café	Un análisis de cómo se tuesta y ...	2025-08-
10	Recetas vegetarianas rápidas	Platos saludables sin carne	Platos fáciles, saludables y delici...	2025-08-
11	Introducción a las derivadas	Qué son las derivadas	Explicación sencilla de qué son y ...	2025-08-
12	Álgebra lineal aplicada	Casos prácticos de álgebra	Casos prácticos de álgebra en pr...	2025-08-

publicaciones 2 x





# ELIMINAR PUBLICACION

Result Grid					
Filter Rows:					
	id	titulo	resumen	contenido	fecha_publicacion
	4	Guía rápida de Docker	Imágenes y contenedores Docker	Veremos cómo crear imágenes y ...	2025-08-27 22:...
	5	Patrones de diseño en Backend	Patrones de arquitectura modernos	Un repaso por los principales pat...	2025-08-27 22:...
	6	10 recetas fáciles de pasta	Recetas italianas rápidas	Recopilación de recetas italianas ...	2025-08-27 22:...
	7	Cómo hacer pan casero	Pan desde cero en casa	Explicación detallada para hacer ...	2025-08-27 22:...
	9	La ciencia detrás del café	Cómo se tuesta y prepara café	Un análisis de cómo se tuesta y ...	2025-08-27 22:...
	10	Recetas vegetarianas rápidas	Platos saludables sin carne	Platos fáciles, saludables v delici...	2025-08-27 22:...

publicaciones 3 x



# CONCLUSIONES:

EL PROYECTO BLOGUEATE LOGRÓ CUMPLIR CON LOS OBJETIVOS PROPUESTOS, COMO LA CREACIÓN DE UNA PLATAFORMA DE BLOGS FUNCIONAL, SEGURA Y FÁCIL DE USAR.

LA INTEGRACIÓN CON TECNOLOGÍAS COMO SPRING BOOT Y ANGULAR PERMITIÓ UNA CONSTRUCCIÓN EFICIENTE TANTO EN EL BACKEND COMO EN EL FRONTEND, ASEGURANDO UNA PLATAFORMA ROBUSTA.

LA IMPLEMENTACIÓN DE UN SISTEMA DE REGISTRO, LA CREACIÓN Y EDICIÓN DE PUBLICACIONES, Y LA INTERACCIÓN CON COMENTARIOS SON CARACTERÍSTICAS CLAVE QUE CONTRIBUYEN A UNA EXPERIENCIA COMPLETA PARA EL USUARIO.

¡MUCHAS  
GRACIAS!