

Ejercicio Práctico

Autenticación con JWT: Protegiendo la API

Después de completar la integración con Firestore, Sabrina y Matías regresan con un nuevo desafío.

"La API ahora maneja datos en la nube, pero ¿qué pasa si cualquier persona intenta acceder a ellos? Es momento de dar el siguiente paso y proteger nuestra aplicación."



"En aplicaciones reales, los datos deben estar resguardados. Para ello, utilizaremos JSON Web Tokens (JWT), un estándar seguro para manejar autenticación en APIs."

Misión:

1. Instalar y configurar JWT:

- Agrega las dependencias **jsonwebtoken** y **body-parser** a tu proyecto.
- Crea las variables de entorno necesarias para almacenar tu clave secreta.
- Implementa la lógica para generar un token JWT cuando un usuario inicie sesión correctamente.

2. Middleware de acceso:

- Crea una función middleware que intercepte las peticiones antes de que lleguen a los controladores.
- Extrae el token desde el header Authorization y verifica su validez utilizando la lógica de JWT.
- Si el token es válido, permite el acceso; de lo contrario, devuelve un error de autenticación.

3. Implementar la Validación de Usuarios

- Aplica el middleware en las rutas que requieran autenticación (por ejemplo, rutas para obtener o modificar datos en Firestore).
- Prueba el flujo enviando peticiones con y sin tokens válidos para asegurarte de que el sistema funcione correctamente.



"Cuando completes este reto, nuestra API será mucho más segura y profesional. ¡Ahora estás protegiendo tu aplicación como un verdadero desarrollador backend!"