Curso CFP

CFP Programador full-stack

Introducción a Autenticación I

Agenda

- Autenticación
- Implementación en una API
- Manejo de Sesiones
- Secuencia de Login
- Demo en Vivo
- Ejercicios
- Planteo para la próxima clase

Autenticación

- La idea no es verlo en detalle sino dar una noción de qué es, y cómo funciona
- Dentro del manejo de usuarios, una parte del proceso es la *autenticación*
- La idea es verificar que un usuario es quien realmente dice ser
- Por ejemplo si un usuario quiere iniciar sesión con usuario y contraseña, la API verifica si esa combinación de user/pass es auténtica
  - Es decir si existe o es correcta

Implementación en una API (1)

- Normalmente los datos de usuarios y contraseñas se guardan en la base de datos
  - Existen otros enfoques, pero quedan por fuera del alcance de este curso
- La API básicamente lo que hace es comparar lo que le llega en el request, contra lo que figura en la base de datos
  - Si no coinciden → devuelvo un response de error
  - Si coinciden → avisarle al usuario que está autenticado (más adelante vemos cómo hacerlo)
- La forma de mandar un request siempre es mediante un POST (http://localhost:8080/login)
  - En el body poner user/password

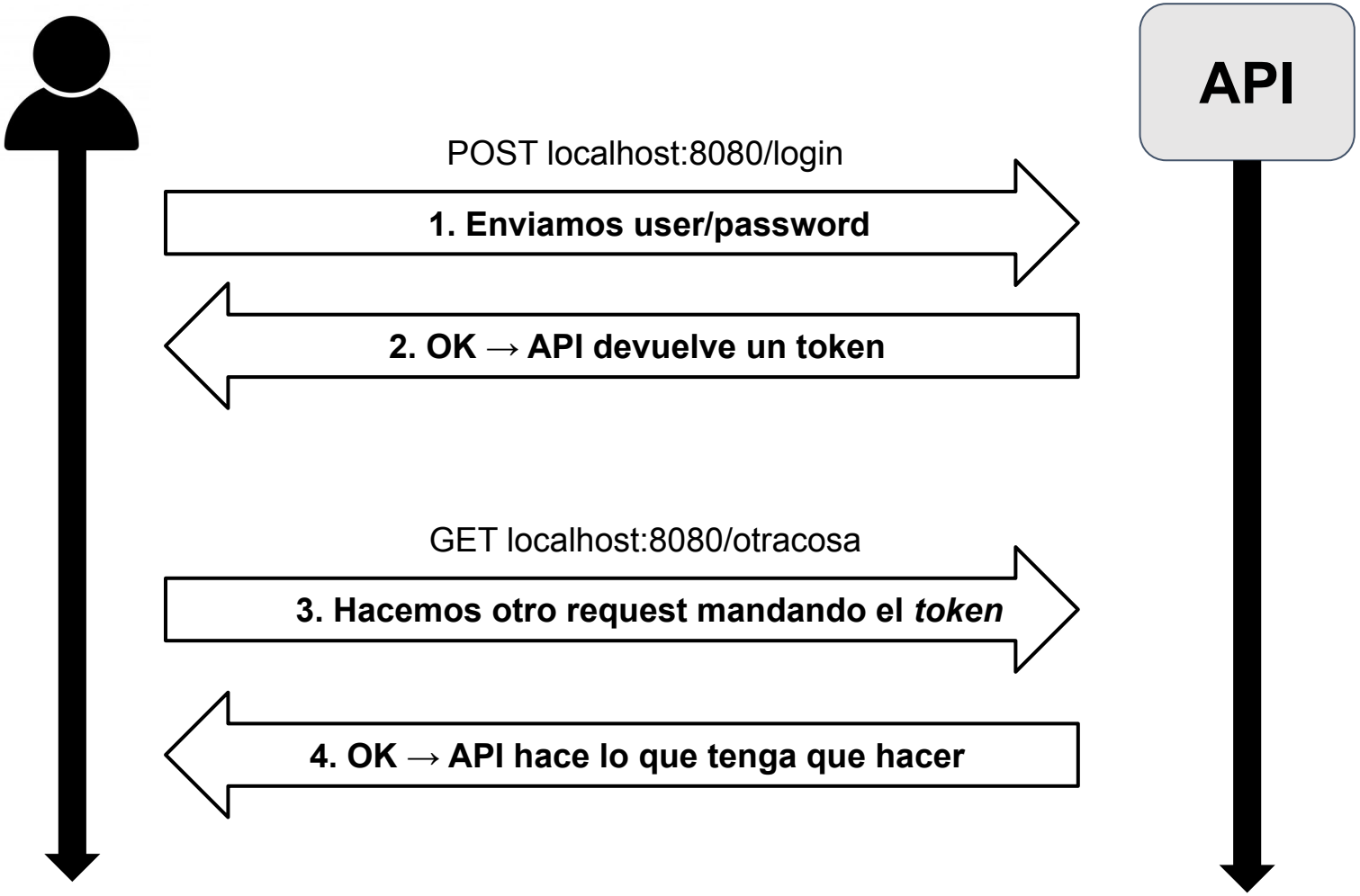
Implementación en una API (2)

- Una vez que el usuario inicia sesión, hay que hacer que el resto de los endpoints requieran de un usuario logueado
- Una posible forma de hacerlo es que para cada request, haya que mandar user/pass → no es la forma más elegante
  - Por cada request habría que hacer un acceso a la base de datos → feo!! 😛
  - Tiene que haber una forma de que el usuario tenga que iniciar sesión una sola vez

Manejando Sesiones

- Cuando iniciamos sesión de forma exitosa, la API nos retorna un string largo y raro → *token*
- Este token es nuestro *permiso* para poder usar el resto de los endpoints de la API
  - Es decir que el resto de los endpoints antes de hacer su trabajo, chequean que el request tenga un token
  - Si no tenemos el token, la API recibe el request pero no nos va a dejar usar la funcionalidad que ofrece
- Imaginarse una tarjeta para fichar en el trabajo, o para abrir una puerta
  - La tarjeta es el mecanismo que tiene el sensor para saber que estamos *autorizados*
  - Sin la tarjeta, no hay manera de abrir la puerta

Secuencia de Login



Demo en Vivo

- Hacer con Postman un POST a un endpoint, para así obtener un token
  - Armar una API de prueba
- Usar ese token para pegarle a otro endpoint
- Hacer pruebas con un token diferente

Curso CFP

CFP Programador full-stack

Ejercicios

Ejercicios - Parte 1

Implementar un login básico 🤪

- Un endpoint POST que recibe en el body la combinación username/password
- Usando TypeORM, crear una tabla “usuario” que tenga como columnas usuario y contraseña
- La API debe chequear que esa combinación coincida lo que haya en la base de datos
- En caso de que no exista, la API debe devolver un mensaje de error
- En caso de que exista, devolver un texto fijo
  - Este texto fijo lo vamos a usar como *token*

Ejercicios - Parte 2

Implementar un endpoint básico

- La funcionalidad básica: que retorne un JSON con un “hola mundo”
- El requisito es que el “hola mundo” se debe devolver **solamente** si en el body del request le pasamos el *token* que obtuvimos en el otro endpoint

Planteo para la próxima clase

- Lo que vimos fue un manejo *muy* básico de usuarios
- Para todos los usuarios estamos devolviendo el mismo token
- Estaría bueno poder manejar diferentes tipos de usuarios → ejemplo: user, admin, etc.
- El contenido del token lo podemos ver directo desde Postman → problemas de seguridad
  - Necesitaríamos que el token esté encriptado
- Existe un mecanismo de tokens muy utilizado actualmente → JSON Web Token (JWT)