

# Seguridad básica: refactorización

Implementa la seguridad de tu API Rest con Spring Boot

# Algunos cambios

- Para poder integrar la seguridad con el funcionamiento de la aplicación necesitamos hacer algunos cambios, y añadir alguna funcionalidad.
- Usualmente, suele implementarse un *endpoint* para obtener los datos de usuario (perfil). Podría estar en */user/me*.
  - Debe devolver los datos del usuario que esté actualmente autenticado.

# Algunos cambios en Pedidos

- Obtener todos
  - Un usuario con rol ADMIN puede obtener todos los pedidos
  - Un usuario con rol USER puede obtener **todos sus pedidos**.
- Nuevo pedido
  - Debemos asociar, como cliente que realiza el pedido, el usuario que actualmente está **autenticado**.

# Reto

- Si no lo has hecho el resto de cursos sobre API REST con Spring Boot, puedes completar este proyecto
- *PedidoController*
  - Modificar un pedido (PUT)
    - Añadir/Eliminar una línea de pedido
    - Modificar la cantidad de una línea de pedido
  - Eliminar un pedido (DELETE)
    - Solo lo puede hacer el dueño o un ADMIN

# Reto

- Todos los métodos de controlador en
  - *LoteController*
    - Modificar un lote (PUT)
    - Eliminar un lote (DELETE)

## Reto 2

- Añade un nuevo rol, llamado CONTENT\_MANAGER
  - Inserta un usuario de ejemplo con dicho rol
  - Permite que pueda realizar GET, POST y PUT sobre los productos y POST y PUT sobre lote.