 

**Desarrollo web en entorno servidor  
Acceso a datos**

**Actividad integradora 3EV**

**Desarrollo de una aplicación Web con Spring Boot**

**13**

**Servicios para aplicación móvil**

**CONTENIDO**

[1.- Objetivos 2](#__RefHeading___Toc1397_3455024786)

[2.- Requisitos 2](#__RefHeading___Toc1399_3455024786)

[3.- Tareas a realizar 2](#__RefHeading___Toc1401_3455024786)

[3.1.- Desarrollo de servicios de aplicación móvil NO RELACIONADOS con autenticación 2](#__RefHeading___Toc1403_3455024786)

# 1.- Objetivos

* Desarrollar una serie de servicios REST / JSON para integración con una aplicación Android.
* Creación de un sistema de usuarios y roles para la aplicación
* Implementación de seguridad basada en JWT con tokens de autorización y de refresco. Sólo para los servicios de la aplicación móvil.

# 2.- Requisitos

Haber completado todas las partes anteriores de la actividad, incluida la migración a base de datos MariaDB

# 3.- Tareas a realizar

La actividad se ha diseñado para realizarla en un orden concreto:

1. Desarrollo de servicios NO relacionados con autenticación. Estos servicios serán los relacionados con:
   1. Búsquedas / listados de productos – Con paginación
   2. Añadir productos al carro de compra
   3. Eliminar productos al carro de compra
   4. Vaciar el carro de compra
2. Desarrollo de un sistema de usuarios
   1. Creación de tabla en BD para usuarios, y mapeo de la entidad JPA correspondiente.
3. Desarrollo de servicios y otros componentes para autenticación de usuarios usando JWT:
   1. Login y generación de tokens de autorización y de refresco.
   2. Refresco de token de autorización.
   3. Logout / revocación de token.
4. Adaptación del sistema de carro de compra para que sea multiusuario.

## 3.1.- Desarrollo de servicios de aplicación móvil NO RELACIONADOS con autenticación

Todos los servicios para la aplicación móvil, que no tengan que ver con autenticación o autorización se expondrán a través de URL que comiencen con /api/app/v1. Se proponen unas URL para estos servicios. No es obligatorio respetarlas, pero es aconsejable, para que sea más fácil el soporte y ayuda del profesor.

Los servicios a desarrollar serán:

* GET /api/app/v1/products/find. Búsqueda paginada de productos.
  + Recibirá los siguientes parámetros (siempre como @RequestParam):
    - search: Cadena de texto para buscar el producto. Se puede usar para buscar en el nombre y la descripción del producto, pero se puede buscar, además, en cualquier otro atributo. Opcional.
    - cat: código de categoría por el que se desea filtrar los productos. Opcional, si no se recibe, no se filtrará por categoría.
    - sortBy: nombre del atributo de producto por el que se quiere ordenar. Si no se recibe, se ordenará por el nombre del producto.
    - sortDir: dirección (ascendente o descendente) de la ordenación. Si no se recibe, se asume que es ascendente.
    - pageNumber: número de página que se solicita. Si no aparece se asume que es la primera página.
    - pageSize: tamaño de la página. Si no se recibe, se asume que es 10.
  + Devolverá:
    - Una página (puede usarse Page de Spring) de DTO con la información del producto.
    - Cada producto debe tener, como mínimo:
      * Nombre
      * Descripción
      * Precio
      * URL de una imagen del producto.
* GET /api/app/v1/cart. Devuelve el carro de compra, con todos los productos del carro. Incluirá el total del carro (importe y número de productos)
* POST /api/app/v1/cart/{productId}. Añadir una unidad de producto al carro de compra.
* POST /api/app/v1/cart/{productId}/{count}. Añadir count unidades de producto al carro de compra.
* DELETE /api/app/v1/cart/{productId}. Añadir un producto al carro de compra.
* DELETE /api/app/v1/cart. Vacía el carro de compra.

PENDIENTE DE AMPLIACIÓN