Spring Security - Gestión de Roles

 **Ejercicio:** Implementar un sistema de gestión de roles de usuario donde cada usuario puede tener uno o más roles asignados, utilizando Spring Security para asegurar los endpoints de la API y JWT como mecanismo de autenticación y autorización.

 **Descripción del ejercicio:**

* **Creación de roles y usuarios:**
  + Implementar un endpoint de la API que permita la asignación de roles (ROLE\_ADMIN, ROLE\_USER, ROLE\_CUSTOMER) a usuarios existentes.
  + Cada rol tendrá un nombre único y una descripción.
* **Autenticación y autorización:**
  + Implementar un endpoint de registro (/registro) que permita a los usuarios registrarse proporcionando un nombre de usuario, correo electrónico y contraseña.
  + Implementar un endpoint de inicio de sesión (/login) que autentique a los usuarios utilizando Spring Security y genere un token JWT como respuesta.
  + El token JWT debe contener la información del usuario autenticado y los roles asociados.
* **Protección de endpoints:**
  + Configurar Spring Security para proteger los endpoints de la API basados en los roles asignados a cada usuario.
  + Ejemplo: Un usuario con el rol ROLE\_ADMIN puede acceder a todos los endpoints protegidos, mientras que un usuario con el rol ROLE\_USER solo puede acceder a ciertos endpoints específicos.

 **Tareas a realizar:**

* Implementar las entidades de dominio (Usuario, Rol) y sus relaciones utilizando JPA.
* Configurar Spring Security para utilizar JWT como mecanismo de autenticación.
* Implementar los controladores REST para manejar las operaciones de creación de roles, asignación de roles a usuarios, registro de usuarios y autenticación.
* Implementar la lógica de negocio y las capas de persistencia necesarias para mantener la coherencia de los roles y usuarios en la base de datos.

Esquema base de datos

