

Diccionario de Datos - Sistema de Gestión de Libros

Entidades

1. Usuario (UserModel)

Atributo	Tipo de Dato	Descripción	Restricciones
id	Long	Identificador único del usuario	PK, Auto-incremental
username	String	Nombre de usuario para acceso al sistema	Único, No nulo
email	String	Correo electrónico del usuario	Único, No nulo
password	String	Contraseña encriptada del usuario	No nulo
theme	String	Preferencia de tema visual	Valor por defecto: "system"

Relaciones:

- Un usuario puede tener uno o más roles (Relación N:M con Role)
- Un usuario puede tener cero o más libros favoritos (Relación 1:N con LibroFavorito)

Tabla en BD: usuarios

2. Rol (Role)

Atributo	Tipo de Dato	Descripción	Restricciones
id	Long	Identificador único del rol	PK, Auto-incremental
name	String	Nombre del rol (ej: ROLE_USER, ROLE_ADMIN)	No nulo

Relaciones:

- Un rol puede estar asignado a múltiples usuarios (Relación N:M con UserModel)

Tabla en BD: roles

3. Libro Favorito (LibroFavorito)

Atributo	Tipo de Dato	Descripción	Restricciones
id	Long	Identificador único del registro	PK, Auto-incremental
usuario_id	Long	ID del usuario que guardó el libro	FK a usuarios.id, No nulo

libroId	String	Identificador del libro en la API externa	No nulo
titulo	String	Título del libro	No nulo
autor	String	Autor del libro	Puede ser nulo
imagenUrl	String	URL de la imagen de portada	Puede ser nulo
fechaGuardado	LocalDateTime	Fecha y hora en que se guardó el libro	Auto-generado

Relaciones:

- Cada libro favorito pertenece a un único usuario (Relación N:1 con UserModel)

Tabla en BD: libros_favoritos

4. Usuario-Rol (Tabla de unión)

Atributo	Tipo de Dato	Descripción	Restricciones
usuario_id	Long	ID del usuario	PK, FK a usuarios.id
rol_id	Long	ID del rol	PK, FK a roles.id

Relaciones:

- Implementa la relación muchos a muchos entre Usuario y Rol

Tabla en BD: usuario_rols

Servicios

1. UsuarioService

Método	Parámetros	Retorno	Descripción
buscarPorUsername	String username	Optional<UserModel>	Busca un usuario por su nombre de usuario
actualizarPerfil	String username, String nuevoNombre, String nuevoEmail	ProfileUpdateResult	Actualiza el perfil de un usuario
registrarUsuario	String usuario, String email, String password	boolean	Registra un nuevo usuario en el sistema

obtenerTodosUsuarios	-	List<UserModel>	Obtiene la lista de todos los usuarios
eliminarUsuario	Long id	void	Elimina un usuario por su ID
actualizarUsuarioAdmin	Long id, String username, String email, String rol	AdminUpdateResult	Actualiza un usuario desde la administración
actualizarTema	String username, String tema	boolean	Actualiza el tema visual del usuario
cambiarPassword	String username, String currentPassword, String newPassword, String confirmPassword	PasswordChangeResult	Cambia la contraseña de un usuario

2. LibroFavoritoService

Método	Parámetros	Retorno	Descripción
obtenerFavoritosPorUsuario	String username	List<LibroFavorito>	Obtiene todos los libros favoritos de un usuario
agregarFavorito	String username, String libroid, String titulo, String autor, String imagenUrl	boolean	Agrega un libro a favoritos
eliminarFavorito	String username, String libroid	boolean	Elimina un libro de favoritos
esFavorito	String username, String libroid	boolean	Verifica si un libro es favorito para un usuario

3. RecomendacionesService

Método	Parámetros	Retorno	Descripción
--------	------------	---------	-------------

obtenerRecomendaciones	String username	List<Map<String, Object>>	Obtiene recomendaciones de libros para un usuario
estaEnFavoritos	List<LibroFavorito> favoritos, String libroid	boolean	Verifica si un libro ya está en la lista de favoritos

Repositorios

1. UserRepository

Método	Parámetros	Retorno	Descripción
findByUsername	String username	Optional<UserModel>	Busca un usuario por su nombre de usuario
findByEmail	String email	Optional<UserModel>	Busca un usuario por su correo electrónico

2. RoleRepository

Método	Parámetros	Retorno	Descripción
findByName	String name	Optional<Role>	Busca un rol por su nombre

3. LibroFavoritoRepository

Método	Parámetros	Retorno	Descripción
findByUsuario	UserModel usuario	List<LibroFavorito>	Busca todos los favoritos de un usuario
findByUsuarioAndLibroid	UserModel usuario, String libroid	Optional<LibroFavorito>	Busca un favorito específico de un usuario
existsByUsuarioAndLibroid	UserModel usuario, String libroid	boolean	Verifica si un libro ya es favorito de un usuario

Enumeraciones

1. ProfileUpdateResult (en UsuarioService)

Valor	Descripción
SUCCESS	Actualización exitosa

EMAIL_IN_USE	El correo electrónico ya está en uso
USERNAME_IN_USE	El nombre de usuario ya está en uso
USER_NOT_FOUND	Usuario no encontrado
ERROR	Error general

2. AdminUpdateResult (en UsuarioService)

Valor	Descripción
SUCCESS	Actualización exitosa
EMAIL_IN_USE	El correo electrónico ya está en uso
USERNAME_IN_USE	El nombre de usuario ya está en uso
USER_NOT_FOUND	Usuario no encontrado
ERROR	Error general

3. PasswordChangeResult (en UsuarioService)

Valor	Descripción
SUCCESS	Cambio de contraseña exitoso
CURRENT_PASSWORD_INCORRECT	La contraseña actual es incorrecta
PASSWORDS_DONT_MATCH	Las nuevas contraseñas no coinciden
ERROR	Error general

Seguridad

1. CustomUserDetails

Clase que implementa la interfaz UserDetails de Spring Security, encapsulando la información del usuario para propósitos de autenticación y autorización.

Método	Retorno	Descripción
getUser	UserModel	Obtiene el modelo de usuario asociado
getTheme	String	Obtiene el tema visual del usuario
getAuthorities	Collection<? extends GrantedAuthority>	Obtiene los roles/autoridades del usuario
getPassword	String	Obtiene la contraseña encriptada

getUsername	String	Obtiene el nombre de usuario
isAccountNonExpired	boolean	Indica si la cuenta no ha expirado
isAccountNonLocked	boolean	Indica si la cuenta no está bloqueada
isCredentialsNonExpired	boolean	Indica si las credenciales no han expirado
isEnabled	boolean	Indica si la cuenta está habilitada

2. CustomUserDetailsService

Implementación de UserDetailsService para cargar detalles de usuario para autenticación y autorización.

Método	Parámetros	Retorno	Descripción
loadUserByUsername	String username	UserDetails	Carga los detalles del usuario por nombre de usuario