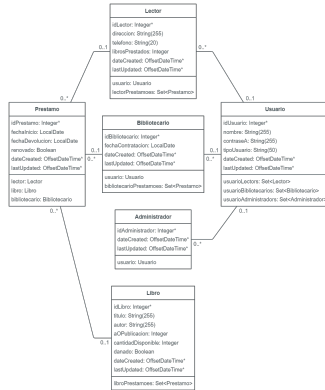


Biblioteca con JPA e Hibernate

Alejandro Domínguez Rábano



1. Introduction y objetivos

En este proyecto hemos intentado simular una biblioteca mediante spring haciendo uso de Hibernate y JPA. El objetivo era mostrar en una interfaz web una "biblioteca" con libros, bibliotecarios y lectores, los cuales podrán ejecutar acciones dependiendo de sus permisos, como por ejemplo añadir y eliminar libros o hacer un seguimiento de los préstamos.

El proyecto se ha diseñado y escrito en java y empaquetado con maven. También se han añadido varias dependencias, basándose principalmente en springboot. La estructura se ha creado haciendo uso de Bootify.

2. Modelo de datos y esquema de base de datos

El esquema de base de datos modela una aplicación para una biblioteca con varios tipos de usuarios, libros y préstamos. La tabla Usuario almacena información básica del usuario, como su nombre, contraseña y tipo de usuario (Lector, Bibliotecario o Administrador). La tabla Lector está vinculada a la tabla Usuario y almacena detalles específicos de los lectores, como dirección, teléfono y la cantidad de libros prestados. Las tablas Bibliotecario y Administrador están también vinculadas a la tabla Usuario y contienen información particular sobre bibliotecarios y administradores. La tabla Libro representa los libros de la biblioteca con detalles como título, autor, año de publicación, cantidad disponible y estado de daño. La tabla Prestamo registra la información de los préstamos, incluyendo el lector, libro, bibliotecario, fechas de inicio y devolución, y un indicador de si el préstamo ha sido renovado.

3. Base de datos e implementación

Para la base de datos, una vez diseñado e implementado el modelo de datos y el esquema de la misma por medio de bootify, hemos implementado una base de datos relacional con PostgreSQL, a la cual hemos añadido las columnas requeridas para nuestra biblioteca.

Las clases creadas para las entidades han sido mapeadas a la base de datos mediante JPA e Hibernate, de esta forma, se nos permite interactuar con ella.

4. Diseño de la aplicación

Para diseñar nuestra biblioteca e implementar nuestras funcionalidades ha sido conveniente una interfaz web, para que el usuario interactúe con la aplicación. La idea era que el usuario seleccionara la clase de usuario que está accediendo, ya sea un lector para devolver un libro, un bibliotecario para hacer un seguimiento de un préstamo o un administrador. Una vez dentro de la biblioteca se implementan métodos para cada acción (añadir un libro, eliminar un bibliotecario, etc.).

Para que nuestra interfaz se guarde y se cree correctamente hemos creado un docker, y mediante el puerto en el que está vinculado el docker accedemos con nuestro localhost en una terminal.

5. Conclusiones

Este proyecto ha presentado un reto bastante amplio con el que hemos asentado una importante base para futuros proyectos de carácter similar.

No cabe duda de que supone un gran aprendizaje para un programador inexperto en el campo de la programación concurrente.