Final Project - Movie Checkout System

(100 Points) Create a movie checkout system:

Your system should be able to: (40 points)

- switch between different customers (customers can be hard coded in the database)
- add new movies
- display all the movies in the database
- search for a movie using keyword in movie title or description
- checkout a movie for a particular customer
- return a movie for a particular customer
- display the checkout history for a particular customer
- keep track of the number of available copies and only allow customers to checkout a movie when there are copies available
- use REST API to retrieve a movie by title
- use REST API to retrieve checkout history by customerId

Your project should demonstrate the successful implementation of the following knowledge: (40 points)

- entity class design
- database table structure and relationships
- system setup (Maven dependencies, dispatcher servlet, database connection)
- dependency injection
- Spring MVC
- Hibernate ORM mapping
- Spring REST
- 3 layer architecture

Presentation, UI design, and coding habit (20 points):

- You should use a clear and audible voice in your presentation. The presentation should be around 5 minutes long and demonstrates all features of the project and also includes a short code review describing how the database and program are structured, the purpose of each class/package and how they work together.
- UI design should be intuitive, easy to use and professional
- Code should be clean, understandable and well-organized. Appropriate indentation, whitespace, and meaningful identifier names should be used throughout the project.

Database design hint:
Customer and Movie should have a “many-to-many” relationship. Instead of using many-to-many mapping, I used one-to-many between Customer and Checkout, and another

one-to-many between Movie and Checkout. “checkout” table is the one that joins customer and movie.