Lab Assignment #5

Library System Design

Description

Brooks Library wants you do develop a new inventory system for them. Here is what the director of the library said about what he wants from this system, and how it should work:

"*It's quite simple actually, we just need a system that can be used for checking in and checking out books and DVDs. The system should obviously keep a list of the available books and DVDs. My employees need to be able to add new books and DVDs to the list when we get new ones. They also need a way to search for the location of the book/DVD in the library, this should be done by typing in the title or authors name. Sometimes books get lost and DVDs get damaged, so we would also need a way to remove a book or DVD from the system.*

Oh, and also we obviously need to keep track in the system whatever a book or DVD has been checked out and by who….. (talks about how much he hates when books get damaged or lost)….. we are also thinking about getting some skateboards and scooters that student can check out during the spring quarter but haven't got an okay yet…" – Not a Director Of Brooks Library

Additional information provided by the Director after the meeting.

- Each book has a title, author(s), publication year, ISBN number and a call number.
- Each DVD has a title, year, and a call number.
- People who can check out items have a name and student ID number.

Details

Create a UML class diagram for such a system and implement it in Java. Remember to keep things as abstract as possible, reading the directors requests you might have noticed that they might want to carry more than just books and DVDs in the future. Your design must be able to accommodate changes in the future without a full system redesign.

Implementation Details

You need to use efficient data structures to store the items in inventory. Your code needs to be commented using javadoc comments and be nicely formatted. There should be line breaks in the code when needed. For the final submission there should not be any dead code or commented out testing code. Classes, methods, and variables need to be named according to the Java naming conventions.

Write a driver class to test out your application. During testing you can create a few users and add a few DVDs and books into the system. Then try to check out a specific book under a specific user. Refer to the lecture slides and have a look at Ricks Guitar Store system for inspiration.

Your interface can be standard input/output in the console. For testing, the scenarios can be hardcoded in and there is no need for any input from the user.

Submission

You need to submit a zip file of your entire project and an UML class diagram.

Rubric

Total 100 points:

30 points - formatted, commented and clean code.

20 points - UML class diagram

50 points - Object Oriented Java implementation