Web Patterns – CA1 – 30% - The DAO Pattern & Internationalization

This CA covers internationalization of applications and database interaction using the DAO pattern.

You have been tasked with writing a basic library system. There are two types of user for this library: admins and members. Your system should provide the following functionality for **members**:

1) Registering with the library (a member must provide more than just a username and password to register).
2) Logging in to your library account (by providing your username and password).
3) Viewing the details of all books in the library.
4) Viewing the details of all loans currently active for you.
5) Viewing the details of all loans you have made since joining the library.
6) Borrowing a copy of a book in the library (if there are no copies available in the library, this will not succeed). A member can only borrow one copy of any title in a single transaction. However, once they return their copy of a book, they can borrow it again in the future.
7) Returning a book you currently have on loan.
8) Logging out.

In addition to general member functionality **excluding registration** (i.e. admins should also be able to perform functions 2-8 listed above), the system should also provide the following functionality only to admin users:

1) Adding a book to the library.
2) Increasing/decreasing the number of copies of a book in the library stock by x amount.
3) **Disabling** a member from the library (Note: An admin cannot disable another admin from the library).

You are required to:

1. Create a MySQL database to hold all of the required information for the library system to function in a well-designed fashion. Primary and foreign keys should be included, and the database should include reasonable levels of information, not merely the minimum required to allow the system to function.
2. Write the **front-end interface** for the system – this MUST be based on the command-line. Your interface should be internationalized with support for two languages.
3. Write the set of Java classes that control access to the database & provide the program's functionality. These classes MUST implement the DAO pattern for all database access & you MUST create and use Data Transfer Objects (DTOs).
4. Write a set of Junit tests for all table-specific DAO classes (no need for DTOs to be tested)
5. Write Javadoc comments for all DAO methods as well as any other non-DTO code. These should include information on return types, parameters, method functionality and the meaning of any potential exceptions that can be thrown.

Marks will be awarded as follows:

- Implementation of the DAO pattern, construction of DAO classes & provision of database functionality - 40%
- Creation and use of DTOs – 5%
- Database design and population of tables with dummy information – 10%
- Program interface – 10%
- Internationalization – 10%
- Junit tests – 15%
- Javadocs – 10%

Submission instructions:

Upload a zipped copy of your **Netbeans** project to Moodle. This zipped file **MUST** also include a file containing the SQL statements used to create and populate your MySQL database. If this file is not included, your project **will not be marked**.

This project is a **GROUP** submission. Every class submitted must contain a statement at the top indicating the names of all three group members. Each member must write **at least one DAO class, the JavaDocs for the methods within that class and their JUnit tests entirely themselves**, and the author of each class should be clearly stated (using Javadoc notation) at the top of that class's file.

Deadline: 6th November 2020.