Code Test (.Net)

Recruitment Ref. No.: PR/RQ/CTDN/259 *Publication date: 24 April 2024*

Your task is to develop a Library **Management System**. Below is a schema that covers entities like **Books**, **Authors**, **Members**, and **BorrowedBooks**, which will allow for CRUD operations using .Net Core MVC, Repository Pattern, and SQL Server. How ever operators can create updates and delete Authors, Books, Members. Check Borrow History, Authentication Level wise View. These are mandatory, however not limited.

Database Schema

1. **Authors**

- AuthorID(Primary Key, Auto Increment)
- AuthorBio(String)
- AuthorName (String)

2. **Books**

- BookID (Primary Key, Auto Increment)
- Title (String)
- ISBN (String)
- AuthorID (Foreign Key)
- PublishedDate (Date)
- AvailableCopies (Integer)
- TotalCopies (Integer)

3. **Members**

- MemberID (Primary Key, Auto Increment)
- FirstName (String)
- LastName (String)
- Email (String)
- PhoneNumber (String)
- RegistrationDate (Date)

4. **BorrowdBooks**

- BorrowID (Primary Key, Auto Increment)
- MemberID (Foreign Key)
- BookID (Foreign Key)
- BorrowDate (Date)
- ReturnDate (Date)
- Status (String: Borrowed, Returned, Overdue)

Relationships:

- **Books.AuthorID is a foreign key referencing Authors.AuthorID.**
- **BorrowedBooks.MemberID is a foreign key referencing Members.MemberID.**
- **BorrowedBooks.BookID is a foreign key referencing Books.BookID.**

Explanation:

• **Authors**: This table stores information about the authors of the books.

• **Books**: This table stores details about the books, including the title, ISBN, author, publication date, and the number of available and total copies.

• **Members**: This table contains information about the library members, such as their names, email addresses, phone numbers, and registration dates.

• **BorrowedBooks:** This table keeps track of the books that have been borrowed by members, including the borrowing date, return date, and status of the borrowed book (Borrowed, Returned, Overdue).

This schema provides a basic structure to manage books, authors, members, and borrowed books in a library management system. You can expand upon this schema by adding more features/tables such as genres, publishers, or reviews based on your own idea/requirements if you want.

Also, you need to apply authorization, dependency injection in this project.

Project Structure (Deliverables)

1. API Backend
2. Web Frontend
3. WPF Client App
4. Database: MySQL

System can be accessed from Web Client and WPF Client both will relate to API Backend, only API Backend can communicate with Database. Login Interface needs to be protected using required authentication.

Evaluation Criteria:

1. Code Successful Execution
2. Interface Design
3. Deployment Document
4. Code Quality
5. Code Security

Please upload your codebase in GitHub, share a video demonstrating your work & Deployment procedures (Please provide public access so that we can check your video and doc by clicking link, if the content is password protected, please share password during submission. Failure to access link may be result as submission failed).

You are encouraged to write code's manually using auto code from AI and the Web is restricted.

Submission Deadline: 30th April 2024 10:00 am
Submission Email: career@perkyrabbit.com
Submission Email Subject: Code Test Submission*: PR/RQ/CTDN/259 | Name of Candidate*
For Query: Md Kawser Raihan | +8801886000819 (WhatsApp Text Only)