**Library Management System Project Documentation**

**Project Overview**

**Project Title:**

Library Management System

**Project Description:**

The Library Management System is an advanced desktop application developed using the .NET framework, specifically leveraging C# and XAML. The primary aim of this software is to elevate and streamline library operations, offering a comprehensive set of features. These include a Welcome Page, Login Page, Registration Page, Book Record Insertion Page, and Return Page. The desktop application is designed to cater to the nuanced needs of both library administrators and patrons.

**Classes**

The system architecture is implemented using C# and XAML, with four core classes serving distinct roles within the application:

**1. User Class:**

- Centrally manages the representation of individuals interacting with the system.

- Key attributes include UserID (Primary Key), Username, Password (hashed for security using C#), UserType, providing crucial insights into user roles within the library management context.

**2. Book Class:**

- A fundamental component in the management of the library's extensive book collection.

- Essential attributes encompass BookID (Primary Key), Title, Author, Genre, ISBN, and Availability, offering a comprehensive view of each book's details and current availability status.

**3. Transaction Class:**

- Responsible for meticulously recording and tracking book transactions, covering both checkouts and returns.

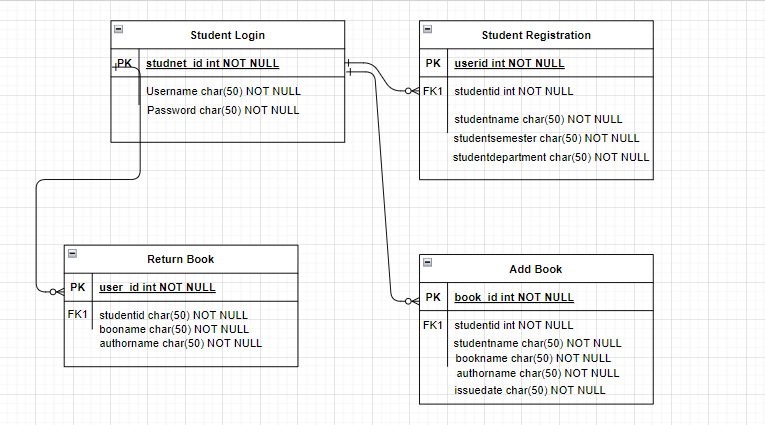
- Key attributes such as TransactionID (Primary Key), UserID (Foreign Key), BookID (Foreign Key), TransactionType, and TransactionDate facilitate a detailed log of user interactions with library resources.

**4. Registration Class:**

- Serves as a repository for storing vital information collected during user registration.

- Attributes include UserID (Primary Key), FullName, Email, ContactNumber, and RegistrationDate, providing a rich dataset for user management and engagement analysis.

**Database Design (ERD)**

****

**Key Points:**

**- User Class:**

- The UserID serves as a primary key in the User table, establishing relationships with various library transactions, and acts as a foreign key in the Book, Transaction, and Registration tables.

**- Book Class:**

- BookID is the primary key in the Book table, with the UserID acting as a foreign key referencing the User table, ensuring a relational link between users and their associated books.

**- Transaction Class:**

- TransactionID serves as the primary key in the Transaction table, while UserID and BookID act as foreign keys, facilitating a comprehensive log of user interactions and relationships with library resources.

**- Registration Class:**

- UserID serves as the primary key in the Registration table, providing a unique identifier for each user, ensuring data integrity and effective user management.

**GUI Design**

The graphical user interface (GUI) is crafted using XAML to provide a visually appealing and user-friendly experience, tailored for desktop application deployment:

**- Visual Appeal:**

- Engaging design with background images and colors.

- Rounded textboxes for a modern and polished appearance.

**- DataGrids:**

- Strategically placed well-formatted and user-friendly DataGrids enhance usability.

**Login and Signup Functionality**

**Login Process:**

1. Users enter credentials (Username and Password) using the C# application.

2. The system validates the input using secure C# methods.

3. If valid, users are seamlessly redirected to the appropriate page; otherwise, an error message promptly communicates the issue.

**Signup Process:**

1. Users provide comprehensive registration details (FullName, Email, ContactNumber, Username, Password) via the C# application.

2. Passwords are securely hashed using C# for data security.

3. Registration information is efficiently stored in the Registration table.

**Functionality**

**- Book Record Insertion:**

- Users can effortlessly add new books with relevant details using the desktop application.

- Data is systematically stored in the Book table.

**- Return Page:**

- The desktop application facilitates the return of books and records transactions in the Transaction table.

**Documentation**

**Code Quality:**

- The source code, developed in C#, is meticulously organized, thoroughly commented, and adheres to best practices.

**Database Design:**

- The ERD demonstrates a clear and effective use of primary and foreign keys in the context of the .NET framework.

**GUI Design:**

- The XAML-based GUI features engaging design elements and adheres to desktop application design principles.

**Functionality:**

- Successful implementation of login and signup functionality, seamlessly integrated with the desktop application.

- Proper functioning of book record insertion and return features.

**- Database Design:**

- Clear and effective use of primary and foreign keys in the ERD, integrated into the .NET framework.

**- GUI Design:**

- Effective use of XAML for an engaging and user-friendly desktop application interface.

**- Functionality:**

- Successful implementation of login and signup functionality, seamlessly integrated into the .NET framework.

- Proper functioning of book record insertion and return features.

**THE END**