Library Application Report

Beyza Komiş

B231202072

13/12/2024

Web Programming Project

Title: Development of a library management system using ASP.NET Core Identity and Entity Framework

Introduction

This projects aims to manage a simple library application that can perform CRUD operations , authentication, layouts, viewbag/viewdata, tag helpers, forms, and validations.

The system allows user to login and the students to register , and additional functions that differ and resemble according to the users.

The application consists of 3 controllers , 17 views Including the default views of the application.

Project Overview

Technologies Used :

- ASP.NET Core Identity for user authentication and role management.
- SQLite for database storage.
- Entity Framework Core for data access.
- ASP.NET MVC for the web interface.

Features

User Registiration and Authentication

- Registration integrates with ASP.NET Core Identity to create users and assign roles.
- Students can register and login.
- The librarians can login they aren't suitable for registiration

Student Specific Features

- Additional properties for users are implemetnted.
- Students can borrow books and return books
- Can leave and list reviews.
- View the list of books
- View the catagorized book list.

Librarian Specific Features

- Additional properties for users are implemented.
- Librarians can perform CRUD operations like;
- Adding book, genre ,author and book category
- Updating and Deleting books.

Both of the Users

- Can search the books
- List the books
- Can View their User Info
- List the reviews

Implementation Details

For the users

- The User entity İnherits from the IdentityUser and the Student and Librarian is inheriting from the User entity.

Contollers

- 3 controllers were enough to handle the application
- Home Controller that first directs to the Account Controller then the Dashboard Controller.
- The Account Controllers handles the login and registiration actions
- The Dashboard controller has the actions, contains the main functions for the application.

Total Models

- User
- Student
- Librarian
- Author
- Category
- Genre
- Review

SeedData

- Created multiple seed data files
- To populate the application before hand.
- There are Views for almost all the actions.

Challenges faced

A lot of errors were faced such as ;

- SQLite Error 19 due to UNIQUE constraint violations in AspNetUsers.
- SqliteException: SQLite Error 19: 'UNIQUE constraint failed:AspNetUsers.NormalizedUserName'.
- Unable to track an entity of type 'User' because its primary key property 'Id' is null.
- An exception occurred in the database while saving changes for context type 'Project.Data.LibraryDbContext'.
- SqliteException: SQLite Error 19: 'FOREIGN KEY constraint failed'.
- The model item passed into the ViewDataDictionary is of type 'Project.Models.Book', but this ViewDataDictionary instance requires a model item of type 'System.Collections.Generic.IEnumerable1
- The request matched multiple endpoints.
- Student field is required.
- System.NullReferenceException: 'Object reference not set to an instance of an object.'

And a lot more but resolved all of them.

Conclusion

This project successfully demonstrates how to build a simple Library Application using ASP.NET Core Identity. It highlights key concepts such as role-based access control, custom user attributes, and integration with a relational database.

Potential future improvements.

Future enhancements include adding more roles, improving the UI/UX, and expanding functionalities like notifications or advanced analytics for student performance.

References:

- Youtube Identity core Tutorials
- StackOverflow
- ChatGPT
- Professors Documents