Online Auction Application

Technology

Web APIs (Backend) – .Net / .Net Core

UI (Frontend) – Angular

Database – MSSQL Server

Database ORM – Entity Framework

Total Hours

4 Hours

Tools Required

- Visual Studio 2022
- Visual Studio Code
- SQL Server Management Studio
- Git
- Google Chrome / Mozilla Firefox

Problem Statement

Develop an Online auction application. The application should allow users to participate in online auctions to buy and sell products.

Must Have Features

Product Listing

- Allow users to list products they want to sell in the auction.
- Users provide details such as ○ Product Name ○ Description ○ Starting price ○ Auction duration (in hours)
    - Product Category
    - Reserved Price (Minimum price for selling the product)
- Implement validation checks to ensure required fields are filled and starting price/reserved price is a valid numeric value.

Bidding & Auction:

- Users can place bids on products they want to buy.
- The application ensures that bids are higher than the current highest bid.
- Users can search auctions based on product names or categories
- User should be able to filter or sort the search list based on ○ Price ○ Time remaining

Note

- The application manages ongoing auctions, displaying product details, current highest bids, and time remaining.
- Signup functionality is not required. Add Normal users and Admin user data into the database directly using the seed method.
- Both Admin and normal users can login using email and password.
- Feel free to add any other fields if found necessary. The fields given are just for reference.
- Implement user authentication to ensure only logged in users should be able to participate in the auctions.

Good to Have (Bonus Functionality)

- The admin can view and delete auctions and suspend or ban users if needed.
- The admin can view the previous history of user participation, including the auctions they have participated in, the products they have bought or sold, and their bidding or offer history.

Guidelines

- Form validations should be implemented wherever needed.
- Exception Handling should be implemented to handle errors.
- Routes should have proper authorization to restrict access based on user roles.
- Mention relevant assumptions taken while implementing assignment.
- Standard coding guidelines and clean coding practices should be strictly followed.
- Feel free to enhance the user experience such that the application should be responsive and user-friendly.
- Follow N-Layer or Clean Architecture to develop Web APIs.

- You can refer official documentation for any syntax/code issue but downloading project from internet, cloning from GitHub or taking reference from StackOverflow/ChatGPT is strictly prohibited. (Angular - https://angular.io/) (.Net - https://learn.microsoft.com/en-us/docs/)
- Do not include Node Modules folder while uploading/submitting the assignment.

Deliverables / Expectations

- Web Application (UI) developed using Angular.
- Restful Web APIs developed using .Net/.Net Core.
- Database Backup File (.bak).
- Readme file to setup the application.

Evaluation Criteria

- Functional and Non-Functional requirements completion.
- Bonus Functionality.
- Clean coding practices and standard coding guidelines.
- Overall Application structure and DB-Design.