

Title: Secure Container Management System

Description: Develop a secure web-based application that allows users to manage and track container shipments, including handling PDF file uploads for container documentation. Ensure that authentication and authorization controls are in place, and file uploads are restricted by size and format.

Requirements:

1. Project Setup:

- Create a new ASP.NET MVC project.
- Implement user authentication and authorization to control access.
- Use Entity Framework to interact with an SQL Server database.

2. Database Design:

- Set up an SQL Server database.
- Design a model for container information, including:
 - Container ID (auto-generated)
 - Container Number (text)
 - Shipment Date (date)
 - Origin Port (text)
 - Destination Port (text)
 - Status (dropdown with options such as In Transit, At Dock, Delivered)
 - File Upload (for container documentation in PDF format)
- Use Entity Framework for data access and querying.

3. Container Management Features:

- Create a form to add new containers, including:
 - Input fields for container details.
 - An option to upload a PDF file (max 5 MB).
 - Server-side validation to ensure all required fields are filled and that the file is a valid PDF under the size limit.
- Save the PDF file in a folder on the server and store the file path in the database.

4. File Upload Handling:

- Implement functionality that:
- Restricts file uploads to PDF format only.
- Enforces a maximum file size of 5 MB.
- Handles any errors during file upload gracefully (e.g., invalid file type, file too large).

5. Container List View:

- Create a page to view a list of containers with:
 - Container Number, Shipment Date, Origin, Destination, and Status.
 - A download link to access the container's documentation file.

6. Authorization:

- Ensure that only authenticated users can add or view containers.
- Use role-based authorization for restricting access to specific functionality.

7. Error Handling and Best Practices:

- Implement proper error handling throughout the application.
- Ensure validation for form inputs, including file uploads.
- Follow coding best practices including separation of concerns, clean architecture, and efficient database querying.

Deliverables:

- Full ASP.NET MVC solution.
- SQL Server database with seeded data.
- Demonstration of working container management functionality.