BESLOGIC

FULL STACK TECHNICAL TEST

BESLOGIC

Full Stack Application Test

Goal: Create a full-stack application using Angular and .NET Web API that allows users to register new aircrafts.

Angular requirements:

1. Create an Angular application using Angular 11 or higher.
2. Implement a navigation bar with links to the following pages: Home, Aircrafts, Add Aircraft.
3. Use Angular Material to style the navigation bar and add icons for each link.
4. Implement a Home page that displays a welcome message and a brief description of the application.
5. Implement an Add Aircraft page that displays a form for users to enter the details of a new aircraft.
6. Use Angular Material to style the form and add validation for required fields.
7. Implement an Aircrafts page that displays a table of all registered aircrafts, showing the ModelName, SerialNumber, RegistrationNumber, RegistrationStatus and RegistrationDate.
8. Use Angular Material to style the table.
9. Implement an Aircraft Details page that displays the details of a specific aircraft.
10. Use Angular Material to style the page and display the aircraft's ModelName, SerialNumber, RegistrationNumber, RegistrationStatus, and RegistrationDate.
11. (Optional) Implement a basic user authentication system using JWT tokens.

.NET requirements:

1. Create a .NET Web API project that uses .NET Core 3.1 or higher.
2. Implement a data model for registered aircrafts that includes the following properties: ModelName, SerialNumber, RegistrationNumber, RegistrationStatus, and RegistrationDate.
3. Use Entity Framework Core to create a database schema and perform data operations for the registered aircrafts.
4. Implement a controller that exposes RESTful endpoints for the following operations:
   - GET /api/aircrafts: Retrieves a list of all registered aircrafts.
   - GET /api/aircrafts/{id}: Retrieves the details of a specific aircraft by ID.
   - POST /api/aircrafts: Adds a new registered aircraft to the database.
   - PUT /api/aircrafts/{id}: Updates the details of a specific registered aircraft by ID.
   - DELETE /api/aircrafts/{id}: Deletes a specific registered aircraft by ID.
5. (Optional) Implement basic authentication using JWT tokens to secure the endpoints.