Coding Exercise: Flight Search Web Application

Overview

Build a full-stack web application that allows users to sign in and search for flight details using the ADSBDB API. All external API calls must be executed on the server side.

Tech Stack

- Front-end: React
- Back-end: Node.js with Express OR .NET Core Web API
- Database: Any relational or NoSQL database (SQLite, PostgreSQL, MongoDB)

Functional Requirements

1. User Authentication

- Create a user table or collection in the database
- Implement a login endpoint
- Store passwords securely using hashing
- Protect flight-related API endpoints so only authenticated users can access them

2. Server-Side API Integration

Base URL: https://www.adsbdb.com/

- GET /api/flights/stats
  Calls the ADSBDB stats API and returns a list of popular flight searches

- GET /api/flights/search?query=...
  Uses a selected popular search to retrieve flight details from ADSBDB

Note: The client must never call the ADSBDB API directly.

3. Front-End (React)

- Login page with username/email and password
- Flight search page displaying popular searches
- Clicking a popular search displays flight results

Suggested Development Steps

1. Set up back-end project
2. Create database schema and test users
3. Implement authentication
4. Integrate ADSBDB API on the server
5. Build React login page
6. Build popular searches and results UI
7. Protect routes using authentication

Bonus (Optional)

- User registration
- Logout
- Error and loading states
- Server-side caching

Deliverables

- Front-end and back-end source code
- README explaining setup and assumptions