Fullstack Developer Interview Project

# Project Description

Build a real-time airport digital display that shows rotating Departures, Arrivals, and Advertisement Screens. Each screen rotates every 5 seconds and fetches data from the backend. Use Socket.IO to simulate real-time updates (e.g., flight delays or gate changes).

# Technology Stack

Frontend: Nuxt 3 + Nuxt UI + TailwindCSS

Backend: NestJS (with Socket.IO support)

Data: From socket io socket

Realtime: Socket.IO

State Management: ref / reactive / composables

Routing: Nuxt pages directory

Animations: watchEffect, setInterval, CSS transitions

Language: TypeScript

# Backend Tasks (NestJS)

1. Setup NestJS server with CORS and JSON body parser.

2. Create REST API Endpoints:

* - GET /api/departures
* - GET /api/arrivals
* - GET /api/ads

3. Create a WebSocket Gateway to emit `flight\_update` events simulating real-time updates.

# Frontend Tasks (Nuxt 3 + Nuxt UI)

1. Build individual components:

* - ArrivalScreen.vue
* - DepartureScreen.vue
* - AdScreen.vue
* - Clock.vue (real-time clock)
* - FlightCard.vue

2. Implement a screen cycle that rotates every 5 seconds using setInterval.

3. Fetch data using `useFetch()` or `useAsyncData()`.

4. Integrate Socket.IO for live updates.

5. Use TailwindCSS for responsive design.

6. Show loading and error states.

# Bonus Features (Optional)

- Flight search/filter bar

- Dark mode toggle

- Unit testing with Vitest + Vue Test Utils

# Evaluation Criteria

- Code Quality and Modularity

- Correctness of API Integration

- Real-time Update Integration

- UI/UX Clarity and Responsiveness

- Screen Switching Stability

# Delivery Instructions

- Submit via GitHub repository with README.md

- Optional: Provide a live preview (Vercel for frontend, Render/Railway for backend)

- Include code comments and documentation