**Laravel Project Documentation: Airports and Tickets System**

This document provides an overview of your Laravel-based project for managing airports and ticket purchases. The system includes core user functionalities, ticket management, and administrative features, built with Laravel Jetstream + Inertia and Blade for the backend/frontend, and an optional Vue.js/Next.js integration with Tailwind CSS implementation and with PostgreSQL database.

Features

* User Authentication and Management
* Login: Users can log in using their registered credentials
* Register: Users can create an account to access the system
* Email verification is required for certain functionalities
* Realtime verification emails sent using Gmail
* Profile Management
* Users can view and update their profile information
* Users can see a list of their purchased tickets with detailed specifics

Homepage and Flight Search

* Search Bar: Users can filter their search by source and destination airports and flight date (optional)
* Flight Listings:
  + Displays available flights based on user search.
  + Logged-in, verified users can see a "Buy Ticket" button.
  + Non-logged-in or non-verified users cannot see or access the purchase functionality.

Ticket Management

* Purchase Restrictions:
  + Only logged-in and email-verified users can purchase tickets.
  + Users can buy up to 3 tickets per flight.
* Users can view all purchased tickets with full details in their profile section.

During the project’s task implementation, we’ve faced several issues, the first of which was connecting Laravel Jetstream Inertia with Nextjs. As it had its own Vue.js implemented in it, we had to reimplement the project with Laravel Breeze + Nextjs, but here we had to setup CORS and Sanctum, for which after many attempts, we couldn’t manage to connect frontend to the backend (User Login/Registration was impossible for implemented project, we even tried official documentation with GitHub repository [breeze-and-next;](https://laravel.com/docs/11.x/starter-kits#breeze-and-next) But still we weren’t able to resolve the problem even though Postman managed log in with no problem, frontside still had Error419 which couldn’t not be resolved for us unfortunately)  
  
This is The Second Implementation Which Includes NextJS:  
<https://github.com/SakhelaTheInvincible/Airport2.0>

Here, as mentioned functionality works well, but Login/Registration is an unresolved problem.  
For Features, we use Nextjs Typescript with Tailwind CSS for frontside, and for backend we use Laravel Breeze with API only.