

A. How-To Manual

This section explains the system requirements, installation, and configuration needed to run the Airport Departure Planner. These steps are only required if you are running the system locally for development or testing. If you are using the deployed version, no setup is required.

System Requirements

Frontend Requirements

- Node.js (v18 or newer recommended)
- npm (comes with Node.js)
- A modern web browser (Chrome, Edge, Firefox)

Backend Requirements (Local Only)

- Java JDK (17 recommended)
 - Maven
 - Apache Tomcat (9.x)
 - Google Maps API key with required APIs enabled
-

Running the Deployed Version (No Setup Required)

If you are not running the system locally, you can access the fully deployed version of the application here:

<https://cs-370-final-project.vercel.app/>

This version already connects to the deployed backend and does not require any local installation.

How to Run the Frontend Locally

Requirements

- Node.js (v18+ recommended)
 - npm
-

Installation

1. Open a terminal.
2. Navigate to the frontend directory:

```
cd frontend
```

3. Install dependencies:

```
npm install
```

Run Development Server

Start the frontend development server:

```
npm run dev
```

Once the server starts, the application will be available at:

<http://localhost:5173>

Open this URL in a browser to use the application locally.

Backend Configuration (Local Only)

When running locally, the frontend communicates with a locally deployed backend running on Apache Tomcat.

Required Environment Variable

A Google Maps API key must be set as an environment variable so the backend can access Google APIs securely.

Example:

```
GOOGLE_MAPS_API_KEY=your_api_key_here
```

This key is required for:

- Places API (address autocomplete)
- Geocoding API
- Routes API (traffic-aware travel time)
- Weather API (forecast-based delays)

The API key is never exposed to the frontend.