Repository / ZIP File Contents

This repository contains all files required for the CS370 Final Project, including source code, documentation, diagrams, and supporting materials. Below is a brief explanation of what each folder and file is used for.

Main Folders

.vscode/

Contains VS Code workspace and editor settings used during development. This folder is optional and only affects the development environment.

Final Project Code/

This is the main project code directory. It contains both the backend and frontend source code.

Inside this folder:

- backend/
  Contains the Java backend application deployed on Apache Tomcat.
  This includes:
    - Java servlets (such as TripEstimateServlet)
    - Google API integration logic (Places, Routes, Geocoding, Weather)
    - Validation and request handling logic
      The backend is built into a WAR file using Maven.
- frontend/
  Contains the React frontend built with Vite.
  This includes:
    - UI components (TripForm, ResultsPanel, Layout, App)
    - Client-side validation
    - API request logic (estimate.js, places.js)
      The frontend runs as a development server locally or is deployed separately.

- deploy-backend.ps1
  A PowerShell script used to automate backend deployment steps on Windows.

- start.sh
  A shell script used to start or configure the backend environment on Unix-based systems.

Documentation Files

README.md

Main project README file. Provides an overview of the project, features, setup instructions, and usage details.

---

6. How to Manual.pdf

User-facing how-to manual explaining how to run and use the system, including local and deployed usage.

---

9. Deployment Guide.pdf

Describes backend deployment steps, required software versions (Java, Tomcat), and build instructions.

---

12. Configuration Guide.pdf

Explains configuration responsibilities for system administrators and users, including API key setup and environment configuration.

Known Limitations (1).pdf

Lists known system limitations such as restricted airport support and approximation-based timing logic.

Requirement Specification Document.pdf

Contains the original system requirements, functional expectations, and constraints defined for the project.

Test Plan.pdf

Outlines the testing strategy, test cases, and validation steps used to verify system functionality.

Final Project Validation Traceability Matrix.xlsx

Maps project requirements to implementation and testing artifacts to demonstrate requirement coverage.

Honor Statement and Roles.pdf

Describes team member roles and includes the academic honesty statement.

Diagrams and Design Artifacts

CabAirportTracker_Architecture.drawio.png

Architecture diagram showing the high-level structure of the system, including frontend, backend, and external APIs.

UMLDiagram.drawio.png

UML diagram illustrating system classes, responsibilities, and relationships.

Project Management Artifacts

Sprints Burndown Charts/

Contains burndown charts used to track sprint progress throughout development.

---

interviews/

Contains interview notes or materials collected during the project planning or requirements gathering phase.

Other Files

package-lock.json

Generated automatically by npm. Locks frontend dependency versions to ensure consistent builds.