Playlist Passeport - App Summary

What it is

A travel web app that lets users create trips, pin memories on a map, and build playlists from the songs tied to those memories. Frontend is React; backend is Node/Express with PostgreSQL and JWT auth.

Who it is for

Travelers who want a map-based journal of memories plus a music soundtrack per trip.

What it does

- Register and log in with email/password (JWT auth).
- Create, view, and delete trips; see finished vs ongoing status.
- Add memories with a location, a selected song, photos, and coordinates.
- Search songs via Deezer and preview tracks.
- Drop and view map pins on Leaflet; search addresses via Nominatim.
- Upload photos to backend storage and view them in memories.
- Finish a trip to generate a Deezer-based playlist and view previews.

How it works (repo-based)

- Frontend: React + Vite, pages Home/TripDetails/Login/Register, AuthContext for token state, Axios API adds Bearer token.
- Map + geo: react-leaflet with OpenStreetMap tiles; Nominatim search from the frontend for coordinates.
- Backend: Express routes /auth, /trips, /memories, /songs, /playlists, /uploads; JWT middleware protects routes.
- Data: PostgreSQL via pg pool; models for users, trips, memories, photos, playlists, and songs.
- External: Deezer API for search + recommendations; Multer saves uploaded photos to /uploads and returns URLs.
- Data flow: UI -> Axios -> Express -> Postgres; Deezer -> backend -> playlist; uploads -> backend -> URL -> UI.

How to run (minimal)

- Backend: cd playlist-passport-backend; set .env with PGHOST, PGPORT, PGDATABASE, PGUSER, PGPASSWORD, PORT, JWT_SECRET; npm install; npm run dev (or npm start).
- Frontend: cd playlist-passport-frontend; npm install; npm run dev. API base URL is http://localhost:5005 in src/services/api.js.

Not found in repo

- Database schema/migrations/seed data.
- Deployment configs (Vercel/Render) and Cloudinary/Spotify integration beyond README mention.