Expense Tracker App - Project Hierarchy

This document outlines the project hierarchy based on Layered Architecture pattern.

Layered Architecture divides the application into layers: Presentation, Application, and Data.

# Presentation Layer (Frontend)

This layer handles user interface and interactions.

Location: src/

- App.jsx: Main application component

- main.jsx: Entry point

- index.css: Global styles

- App.css: Application styles

- components/:

- Dashboard/: Dashboard component and styles

- Expense/: Expense management component and styles

- Income/: Income management component and styles

- SignIn/: User sign-in component and styles

- SignUp/: User sign-up component and styles

- ForgotPassword/: Password recovery components and styles

- Other components: DataTable, Form, InputField, Layout, Navbar, Routes

# Application Layer (Backend)

This layer contains business logic and API endpoints.

Location: backend/

- main.py: FastAPI application and API routes

- models.py: Data models (SQLAlchemy)

- database.py: Database connection and session management

- requirements.txt: Python dependencies

# Data Layer

This layer manages data storage and retrieval.

- expense\_tracker.db: SQLite database file

- Database schema defined in models.py

# Additional Configuration Files

- package.json: Frontend dependencies and scripts

- vite.config.js: Vite configuration

- eslint.config.js: ESLint configuration

- .gitignore: Git ignore rules

- README.md: Project documentation

- TECHNICAL\_DOCUMENTATION.txt: Technical details

- Other files: .env, validations.txt, etc.