**Intelligent Finance Tracker**

This is a full-stack web application that allows users to track their finances using AI-powered natural language transaction entry.

**Features**

* Google OAuth 2.0 for secure sign-in and sign-up.
* AI-powered parsing of transaction strings into structured data.
* Dashboard with financial summaries (income, expenses, savings).
* Visualizations of spending patterns with pie and line charts.
* Full transaction history with create, read, update, and delete (CRUD) functionality.

**Technology Stack**

* **Frontend:** React, Tailwind CSS, Chart.js
* **Backend:** Node.js, Express, MongoDB
* **Authentication:** Google OAuth, JSON Web Tokens (JWT)
* **AI Integration:** OpenAI GPT-4

**Setup and Installation**

1. **Clone the repository:**  
           git clone <repository-url>     cd finance-ai-tracker
2. **Install all dependencies:**  
           npm run install-all
3. **Configure Environment Variables:**  
       - Create a .env file in the backend/ directory and populate it using the .env.example template.  
       - Create a .env file in the frontend/ directory and add your REACT\_APP\_GOOGLE\_CLIENT\_ID.

**how to run the full-stack project** I provided earlier inside **Visual Studio Code (VS Code)**, including exactly where and how to create your .env files.

**Step 1: Open Your Project in VS Code**

1. Launch Visual Studio Code.
2. Click **File > Open Folder...**
3. Navigate to your project root folder finance-ai-tracker and open it.
4. You should see the folder structure with backend/, frontend/, docs/, and other root files.

**Step 2: Install Dependencies**

You need to install dependencies separately for both backend and frontend.

**Backend dependencies:**

1. Open a new terminal in VS Code: **Terminal > New Terminal**.
2. Navigate to the backend folder:

bash    cd backend      
3. Install backend dependencies:

bash    npm install

**Frontend dependencies:**

1. Open a **second terminal tab** (or a new terminal window).
2. Navigate to the frontend folder:

bash    cd frontend      
3. Install frontend dependencies:

bash    npm install

**Step 3: Set up .env Files (Configuration)**

The .env files contain sensitive keys like API keys, database URLs, and third-party client IDs. You must create these yourself in the proper directories.

**Backend .env file:**

1. In VS Code Explorer, right-click the **backend** folder.
2. Click **New File** and name it .env (no filename, just .env).
3. Open this .env file and add:

   PORT=5001    MONGO\_URI=your\_mongodb\_connection\_string    JWT\_SECRET=your\_jwt\_secret    GOOGLE\_CLIENT\_ID=your\_google\_oauth\_client\_id.apps.googleusercontent.com    OPENAI\_API\_KEY=your\_openai\_api\_key

Replace placeholders with your actual credentials.

**Frontend .env file:**

1. Similarly, right-click the **frontend** folder.
2. Click **New File** and name it .env.
3. Add the following as the contents:

   REACT\_APP\_GOOGLE\_CLIENT\_ID=your\_google\_oauth\_client\_id.apps.googleusercontent.com

**Step 4: Start Backend Server**

1. In your VS Code terminal where you're inside the backend folder, run:

bash    npm start

1. You should see output like:

   Backend server running on port 5001    MongoDB Connected...

**Step 5: Start Frontend Server**

1. Switch to your terminal where you're inside the frontend folder.
2. Run:

bash    npm start

1. This will start the React development server, opening a browser to <http://localhost:3000> if it does not open automatically.

**Step 6: Use the Application in Browser**

* Open your browser and go to http://localhost:3000.
* You will see the login page with **"Sign in with Google"**.
* Test logging in with Google OAuth, then use the dashboard features.

**Optional: Start Both Servers with Single Command**

If you want to start both frontend and backend in one command from the project root, use the root-level scripts provided.

1. Open a new terminal at the root (finance-ai-tracker).
2. Run:

bash    npm run dev

This requires concurrently installed and configured in the root package.json. It will run backend and frontend servers together.

**Summary: Location of .env Files**

|  |  |  |
| --- | --- | --- |
| Folder | .env Location | Purpose |
| backend/ | backend/.env | Backend API configuration (DB, keys, secrets) |
| frontend/ | frontend/.env | Frontend environment variables (Google Client ID) |

Make sure **these files are NOT committed to version control** (add .env to .gitignore if you use Git).

If you follow these steps, you will successfully run the full project in VS Code and be able to develop, test, and demonstrate it.

If you have any issues or need help with any step, feel free to ask!

1. **Run the application:**  
           npm run dev      
       The application will be available at http://localhost:3000.

**AI Design Tool Credits**

The UI/UX was inspired by modern financial apps like Mint and Credit Karma, with component design and layout assisted by AI code generation tools. change this into headings in bold and dark and normal matter in another form and change font

Certainly! Here is your content reformatted with **bold dark headings** and the normal body text in a clear, readable style with appropriate font styling indications. This is suitable for Markdown or documentation styling:

This is a full-stack web application that allows users to track their finances using AI-powered natural language transaction entry.

**Features**

* Google OAuth 2.0 for secure sign-in and sign-up.
* AI-powered parsing of transaction strings into structured data.
* Dashboard with financial summaries (income, expenses, savings).
* Visualizations of spending patterns with pie and line charts.
* Full transaction history with create, read, update, and delete (CRUD) functionality.

**Technology Stack**

* **Frontend:** React, Tailwind CSS, Chart.js
* **Backend:** Node.js, Express, MongoDB
* **Authentication:** Google OAuth, JSON Web Tokens (JWT)
* **AI Integration:** OpenAI GPT-4

**Setup and Installation**

**1. Clone the repository**

git clone <repository-url>  
cd finance-ai-tracker

**2. Install all dependencies**

npm run install-all

**3. Configure Environment Variables**

* Create a .env file inside the backend/ directory and populate it using .env.example as a template.
* Create a .env file inside the frontend/ directory and add your REACT\_APP\_GOOGLE\_CLIENT\_ID.

**Step 1: Open Your Project in VS Code**

1. Launch Visual Studio Code.
2. Click **File > Open Folder...**
3. Navigate to your project root folder finance-ai-tracker and open it.
4. You should see the folder structure with backend/, frontend/, docs/, and other root files.

**Step 2: Install Dependencies**

You need to install dependencies separately for both backend and frontend.

**Backend dependencies:**

1. Open a new terminal in VS Code (Terminal > New Terminal).
2. Navigate to the backend folder:

cd backend

1. Install backend dependencies:

npm install

**Frontend dependencies:**

1. Open a second terminal tab (or a new terminal window).
2. Navigate to the frontend folder:

cd frontend

1. Install frontend dependencies:

npm install

**Step 3: Set up .env Files (Configuration)**

The .env files contain sensitive keys like API keys, database URLs, and third-party client IDs. You must create these yourself in the proper directories.

**Backend .env file:**

1. In VS Code Explorer, right-click the **backend** folder.
2. Click **New File** and name it .env (no filename, just .env).
3. Open this .env file and add:

PORT=5001  
MONGO\_URI=your\_mongodb\_connection\_string  
JWT\_SECRET=your\_jwt\_secret  
GOOGLE\_CLIENT\_ID=your\_google\_oauth\_client\_id.apps.googleusercontent.com  
OPENAI\_API\_KEY=your\_openai\_api\_key

Replace placeholders with your actual credentials.

**Frontend .env file:**

1. Similarly, right-click the **frontend** folder.
2. Click **New File** and name it .env.
3. Add the following content:

REACT\_APP\_GOOGLE\_CLIENT\_ID=your\_google\_oauth\_client\_id.apps.googleusercontent.com

**Step 4: Start Backend Server**

1. In the VS Code terminal where you are inside the backend folder, run:

npm start

1. You should see output similar to:

Backend server running on port 5001  
MongoDB Connected...

**Step 5: Start Frontend Server**

1. Switch to your terminal where you’re inside the frontend folder.
2. Run:

npm start

1. This will start the React development server and open your browser (usually at <http://localhost:3000>).

**Step 6: Use the Application in Browser**

* Open your browser and go to http://localhost:3000.
* You will see the login page with **Sign in with Google**.
* Test logging in with Google OAuth, then use the dashboard features.

**Optional: Start Both Servers with Single Command**

If you want to start both frontend and backend in one command from the project root:

1. Open a new terminal at project root (finance-ai-tracker).
2. Run:

npm run dev

This requires concurrently to be installed and configured in root package.json. This command runs backend and frontend servers together.

**Summary - Location of .env Files**

|  |  |  |
| --- | --- | --- |
| Folder | .env Location | Purpose |
| backend/ | backend/.env | Backend API config (DB, keys, secrets) |
| frontend/ | frontend/.env | Frontend environment variables (Google Client ID) |

Make sure **these .env files are NOT committed to version control** (add .env to .gitignore).

If you follow these steps, you will successfully run the full project in Visual Studio Code and be able to develop, test, and demonstrate it!

Feel free to ask if you want me to help with any specific part.