Personal Agentic AI Chatbot - Setup Guide

This document provides a step-by-step guide to set up and run the Personal Agentic AI Chatbot project. It explains how to configure your environment (Pipenv, pip+venv, or Conda), prepare API keys, and run the backend (FastAPI) and frontend (Streamlit).

# 1. Project Overview

This project is a Personal Agentic AI Chatbot that enables users to interact with multiple LLM providers (OpenAI, Groq, Llama, Mistral) through a unified interface.  
  
Purpose:  
- Switch between multiple providers/models at runtime  
- Customize chatbot behavior with system prompts  
- Integrate real-time web search (Tavily)  
- Provide a modular backend + frontend template for GenAI applications

# 2. Environment Setup

You can set up the environment using one of three methods: Pipenv, pip+venv, or Conda.

## 2.1 Using Pipenv

1. Install Pipenv (if not already installed):  
 pip install --user pipenv  
  
2. Install dependencies:  
 pipenv install  
  
3. Activate the environment:  
 pipenv shell  
  
4. Run project commands:  
 pipenv run uvicorn backend:app --host 127.0.0.1 --port 9999 --reload  
 pipenv run streamlit run frontend.py  
  
5. (Optional) Export requirements.txt:  
 pipenv lock -r > requirements.txt

## 2.2 Using pip and venv

1. Create a virtual environment:  
 python -m venv venv  
  
2. Activate the environment:  
 macOS/Linux: source venv/bin/activate  
 Windows (PowerShell): venv\Scripts\Activate.ps1  
  
3. Install dependencies:  
 pip install -r requirements.txt  
  
4. Run project commands:  
 uvicorn backend:app --host 127.0.0.1 --port 9999 --reload  
 streamlit run frontend.py  
  
5. Deactivate when done:  
 deactivate

## 2.3 Using Conda

1. Create a Conda environment (Python 3.11 recommended):  
 conda create --name myenv python=3.11  
  
2. Activate the environment:  
 conda activate myenv  
  
3. Install dependencies:  
 pip install -r requirements.txt  
  
4. Run project commands:  
 uvicorn backend:app --host 127.0.0.1 --port 9999 --reload  
 streamlit run frontend.py  
  
5. Deactivate when done:  
 conda deactivate

# 3. Running the Application

Phase 1: Create AI Agent  
 python ai\_agent.py  
  
Phase 2: Setup Backend with FastAPI  
 uvicorn backend:app --host 127.0.0.1 --port 9999 --reload  
 Swagger docs available at http://127.0.0.1:9999/docs  
  
Phase 3: Setup Frontend with Streamlit  
 streamlit run frontend.py  
 UI available at http://localhost:8501

# 4. Important Notes

- Keep the backend running in a separate terminal while using the frontend.  
- Ensure a .env file exists in the project root with valid API keys:  
 OPENAI\_API\_KEY=your\_openai\_key  
 GROQ\_API\_KEY=your\_groq\_key  
 TAVILY\_API\_KEY=your\_tavily\_key