

AI AGENT CHATBOT SYSTEM

1. Project Overview

This project is a modern, full-stack conversational AI chatbot system designed to deliver intelligent, interactive responses using advanced language models and agent frameworks. The system combines a robust Python backend, an agentic reasoning engine, and a user-friendly web-based frontend to provide a seamless chat experience for end users.

2. Architecture

Layer	Technology/Library Used	Purpose
Frontend	Streamlit	Interactive web UI for chatting with the AI agent
Backend	FastAPI, Uvicorn, Pydantic	API endpoints, validation, and server hosting
AI Agent	LangGraph, LangChain, Groq, OpenAI, Tavily Search	Conversational intelligence, tool-augmented reasoning
Utilities	Requests, Pipenv, dotenv	HTTP requests, environment management, configuration.

3. Key Features

- **Multi-Model Support:** Users can select between Groq and OpenAI as providers, with models such as gpt-4o-mini, llama-3.3-70b-versatile, and mixtral-8x7b-32768.
- **Tool-Augmented Reasoning:** The agent uses external tools (e.g., Tavily Search) for real-time web information, improving answer accuracy and relevance.
- **Custom System Prompts:** Users can define instructions for the AI agent, tailoring its behavior for various use cases.
- **Modern Web UI:** The Streamlit-based frontend offers a chat-like interface, easy model selection, prompt customization, and robust error handling.
- **Robust Backend:** FastAPI ensures high performance and easy API documentation, while Pydantic provides strong data validation and type checking.

4. Technology Stack

Category	Tools/Libraries	Description
Language Models	Groq, OpenAI	Advanced LLMs for conversation
Agent Framework	LangGraph, LangChain	Enables tool use and reasoning chains
Search Tool	Tavily Search	Real-time web search capability
Backend API	FastAPI, Uvicorn, Pydantic	API endpoints, server, and validation
Frontend	Streamlit	Interactive chat UI
Utilities	Requests, Pipenv, dotenv	HTTP requests, dependency/env management

5. Workflow

- **User Interaction:** The user accesses the Streamlit web app, selects a model/provider, configures the system prompt, and submits a query.
- **Frontend-to-Backend:** Streamlit sends the query and settings as a POST request to the FastAPI backend.
- **Backend Processing:** FastAPI validates the request using Pydantic and forwards it to the agent logic.
- **Agentic Reasoning:** The agent selects the appropriate LLM and tools, processes the query (using web search if enabled), and generates a response.
- **Response Delivery:** The backend returns the agent's response to the frontend, which displays it in the chat window and updates conversation history.

6. Example Code Structure

- `ai_agent.py`: Handles model selection, tool integration, and agentic reasoning logic.
- `backend.py`: Defines the FastAPI app, request validation, and endpoint logic.
- `frontend.py`: Implements the Streamlit UI, handles user input, and displays responses.

7. Security and Best Practices

- **Environment Variables:** API keys and sensitive data are managed via environment variables and not hardcoded, ensuring security.
- **Error Handling:** Both backend and frontend provide user-friendly error messages and robust exception management.
- **Extensibility:** The modular design allows for easy addition of new models, tools, or UI features.

8. Sample Use Cases

- Personal research assistant with real-time web search

AI Chatbot Agents

Create and interact with smart AI agents using LangGraph and FastAPI!

Define your AI Agent:

Act as a Research Assistant

Select Provider:

☒ Groq
☐ OpenAI

Select Groq Model:

llama-3.3-70b-versatile

☒ Allow Web Search

Enter your query:

What are the latest advancements in artificial intelligence this year?

Ask Agent!

Agent Response

Final Response: Based on the search results, the latest advancements in artificial intelligence this year include:

1. Apple's entry into generative AI
2. Unprecedented achievements in robotics and art
3. The rise of generative AI (GenAI) as a powerful productivity driver
4. Breakthroughs in GenAI that have redefined its capabilities
5. The impact of GenAI on commercial and government implementations of the technology
6. Innovations that reflect a clear shift toward embedding GenAI more deeply to drive efficiency and resilience in critical sectors like healthcare, IT, retail, and defense.

These advancements suggest that 2024 has been a transformative year for artificial intelligence, with significant progress and developments that will continue to shape the future of AI in 2025 and beyond.

- Educational tutor for interactive learning sessions

AI Chatbot Agents

Create and interact with smart AI agents using LangGraph and FastAPI!

Define your AI Agent:

Act as an Educational Tutor

Select Provider:

☐ Groq
☒ OpenAI

Select OpenAI Model:

gpt-4o-mini

☒ Allow Web Search

Enter your query:

Can you explain the concept of photosynthesis in simple terms?

Ask Agent!

Agent Response

Final Response: Photosynthesis is the process that plants, algae, and some bacteria use to turn sunlight into energy. Here's how it works in simple terms:

1. **Sunlight:** Plants capture sunlight using a green pigment called chlorophyll, mainly found in their leaves.
2. **Carbon Dioxide:** Plants take in carbon dioxide from the air through tiny openings in their leaves called stomata.
3. **Water:** Roots absorb water from the soil.
4. **Making Food:** Using the energy from sunlight, plants combine carbon dioxide and water to create glucose (a type of sugar) and oxygen. This process mainly happens in the leaves.
5. **Oxygen Release:** As a byproduct of making food, plants release oxygen, which is what animals and humans breathe.

In summary, photosynthesis is how plants make their own food using sunlight, carbon dioxide, and water, while also producing oxygen that we need to breathe.

9. Summary Table

Feature	Benefit
Multi-provider LLMs	Flexible, powerful AI responses
Web search integration	Up-to-date, factual answers
Customizable prompts	Tailored agent behavior
Easy-to-use UI	Accessible for all users
Modular backend	Maintainable and extensible codebase
Secure configuration	Protects API keys and sensitive settings

10. Conclusion

This project demonstrates a robust, extensible AI chatbot system that leverages state-of-the-art language models, agent frameworks, and a modern UI to deliver intelligent, interactive conversations. Its modular design, security practices, and user-centric features make it suitable for a wide range of real-world applications.