E-Commerce Chatbot with Login & Chat History

An advanced e-commerce chatbot simulation, featuring user authentication, persistent chat history, and dynamic product information retrieval. This project was developed as part of the Uplyft.ai Internship Case Study, demonstrating practical full-stack development skills.

S **by Swapnil Gite**

Project Overview & Problem Statement

The core problem addressed by this project is the simulation of a smart e-commerce chatbot designed to enhance user experience by providing product search capabilities, personalized activity tracking, and seamless retention of conversation history across multiple sessions. This initiative aimed to create a robust and user-centric solution for the Uplyft.ai Internship Case Study in June 2025.

Key Features Implemented

User Authentication

Implemented a secure login and logout system for static users.

Product Search

Chatbot returns detailed product information from a `products.json` file.

Persistent Chat History

Individual chat histories are stored in `chat_history.json` and auto-loaded upon user login.

Separated Architecture

Frontend and backend are designed as distinct, decoupled modules.

Local Hosting

Project is hosted locally via a Flask backend for easy deployment.

Technology Stack

`</>`

Frontend

HTML, CSS, JavaScript for a responsive and intuitive user interface.

Backend

Python and Flask, with Flask-CORS for secure cross-origin requests.

Data Storage

Lightweight JSON files simulate database operations for user data.
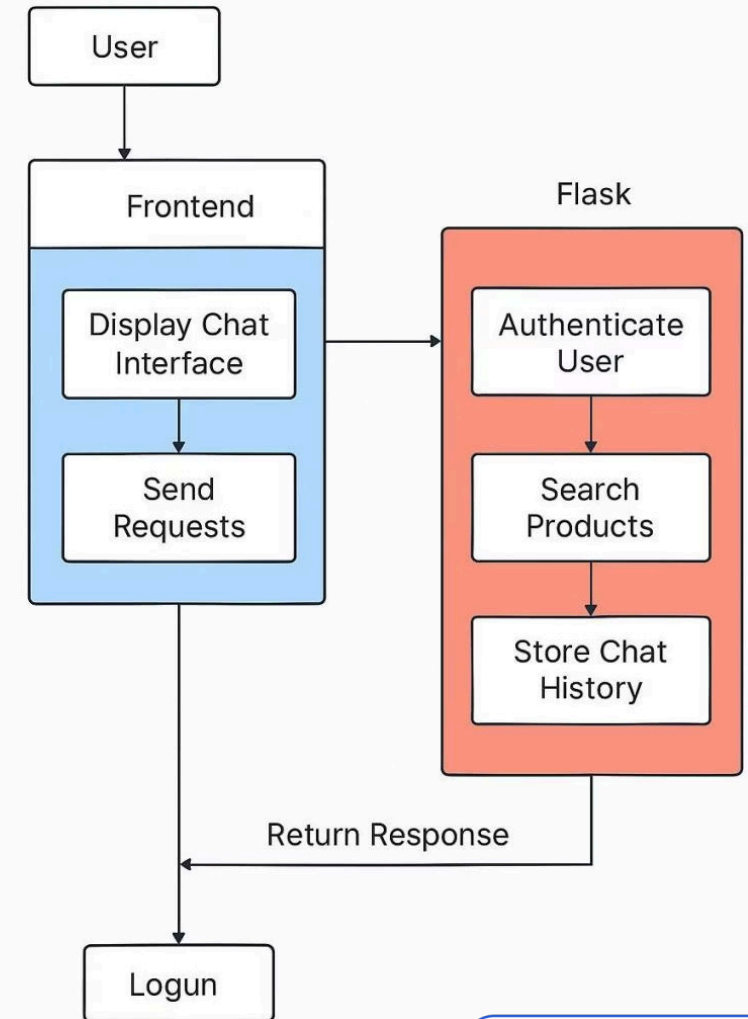
Development Tools

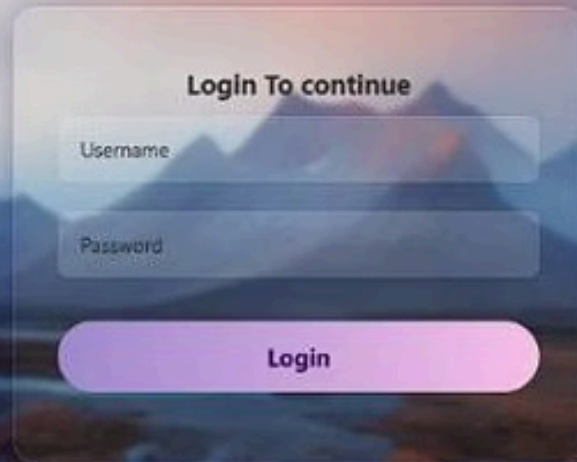VS Code for development and GitHub for version control and collaboration.

Architecture & Data Flow

The system architecture separates the frontend, which handles user interaction, from the Flask backend, which manages business logic and data persistence. API routes such as /login, /chat, and /history/ facilitate seamless communication between these layers. The backend interacts directly with JSON files, serving as a lightweight, file-based database for storing and retrieving user and chat data.

E-Commerce Chatbot – Project Flow

User → Frontend

Frontend: Display Chat Interface → Send Requests

Flask: Authenticate User → Search Products → Store Chat History

Return Response

Logun

User Interface & Interaction

The chatbot's intuitive user interface allows for easy product inquiries and displays chat history clearly. Users can log in to access their past conversations and seamlessly continue their interactions. The system only responds with product names pre-stored in the products.json{" "} file, ensuring data integrity and consistency.

Key Learnings & Takeaways

Full-Stack Development

Gained comprehensive experience in building and integrating frontend and backend systems.

API Interaction

Deepened understanding of how frontend and backend communicate through RESTful APIs, particularly with JavaScript's Fetch API.

Data Persistence

Learned to manage manual user authentication and persist user data using JSON files.

Project Management

Enhanced skills in project structuring, GitHub for version control, and professional documentation.

Conclusion & Future Outlook

This e-commerce chatbot project provided invaluable practical experience in full-stack web development, bridging theoretical knowledge with real-world application. It underscored the importance of seamless frontend-backend integration and robust data management. I am grateful to Uplyft.ai for this challenging and insightful internship opportunity, which has significantly boosted my problem-solving abilities and prepared me for collaborative team environments.

View on GitHub