AI-Powered Conversational Chatbot for Research Papers and Resumes using n8n

Developed by: Vijaykumar Mangore

Under the guidance of Mr. Saxon K Shah

1. Abstract

This project presents the development of an AI-powered Conversational Chatbot that allows users to upload documents such as resumes or research papers and interactively ask questions about their content. The chatbot integrates a Streamlit-based frontend with an n8n backend workflow leveraging Google Gemini (PaLM) for Natural Language Understanding. The system extracts text from uploaded files, interprets user queries, and returns accurate, document-based responses without external data dependency. This project demonstrates the power of Low-Code AI Orchestration combined with modern Generative AI for intelligent document analysis and automation.

3. Introduction

With the increasing volume of digital documents, manually analyzing and extracting insights from resumes, research papers, and reports has become time-consuming. The AI Conversational Chatbot automates this process, enabling users to query any uploaded file conversationally. The system consists of a Streamlit frontend that handles user interactions and file uploads, and an n8n backend that automates workflow, connects to Google Gemini for text processing, and returns contextual responses. This integration provides a scalable, user-friendly solution for educational institutions, HR departments, and research organizations to analyze document content in real-time.

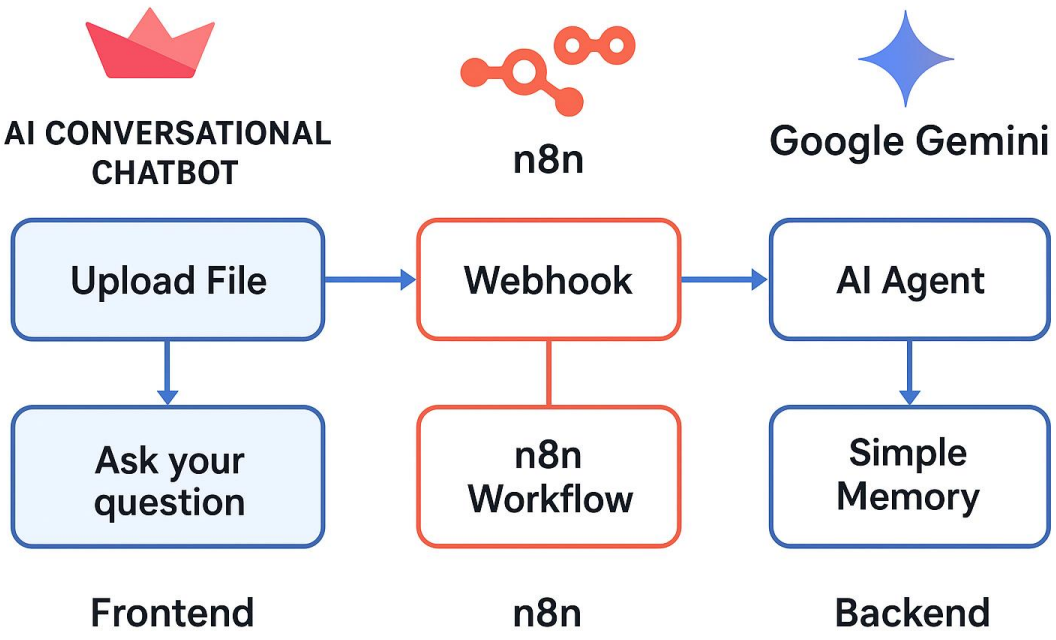
3. Objective

The objectives of this project are:

1. To design a chatbot interface capable of answering questions based solely on uploaded documents.
2. To integrate n8n with Google Gemini AI for backend automation and response generation.
3. To ensure accurate, context-specific answers without hallucinations.
4. To demonstrate a low-code AI orchestration architecture for scalable deployment.

4. System Architecture Diagram

Below is the architecture diagram showing the interaction between Streamlit, n8n, and Google Gemini.



5. Technology Stack

Layer	Technology / Tool	Purpose
Frontend	Streamlit	Web UI for user interaction and chat interface
Backend	n8n	Workflow automation and orchestration
AI Model	Google Gemini (PaLM)	Natural language processing and reasoning
Memory	LangChain Buffer Window	Maintains conversational context
File Handling	n8n Extract from File Node	Reads uploaded PDF/DOC/CSV files

Communication	HTTP Webhooks	Data exchange between Streamlit and n8n
Language	Python	Used for Streamlit application

6. Implementation Details

Frontend (Streamlit): Users upload files and input questions in chat format. The frontend sends data to n8n webhook via POST request. The responses are displayed using chat interface.

Backend (n8n): Workflow nodes include Webhook, Extract from File, AI Agent, Google Gemini model, Memory buffer, and Respond nodes.

Data Flow: User uploads document → n8n extracts text → Gemini processes → n8n responds → Streamlit displays AI output.

Workflow Summary

Platform –n8n

Frontend Integration- Streamlit via Webhooks

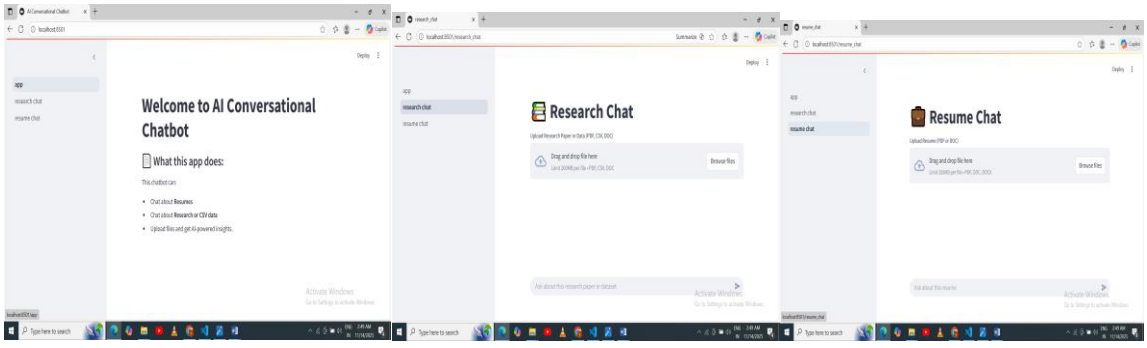
Model- Google Gemini

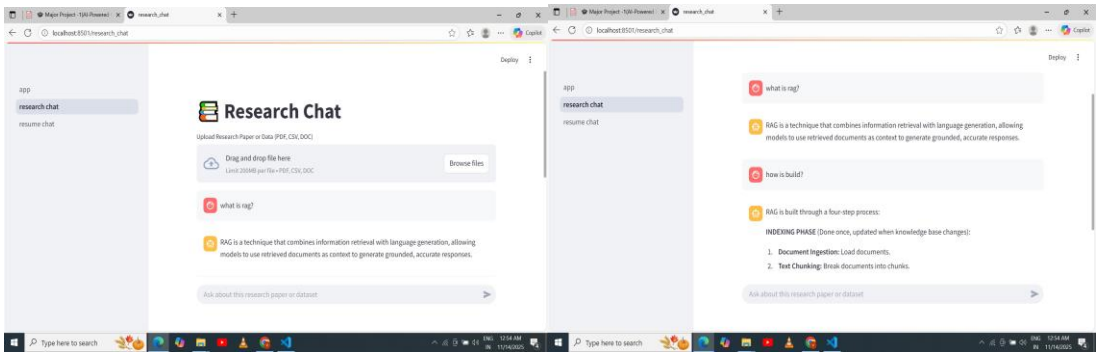
Workflow Status- Inactive

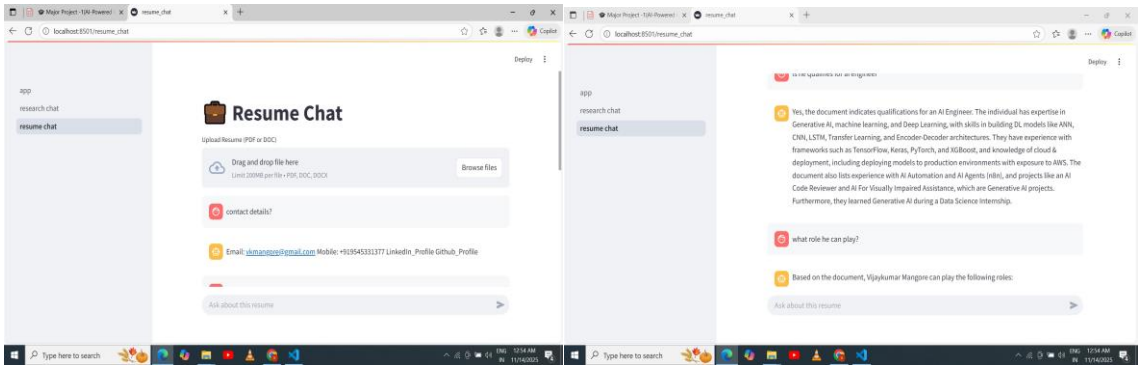
Output- Text-based response to Streamlit Frontend

7. Frontend Screenshots

(Homepage, Resume Chat, Research Chat, Conversation Examples).

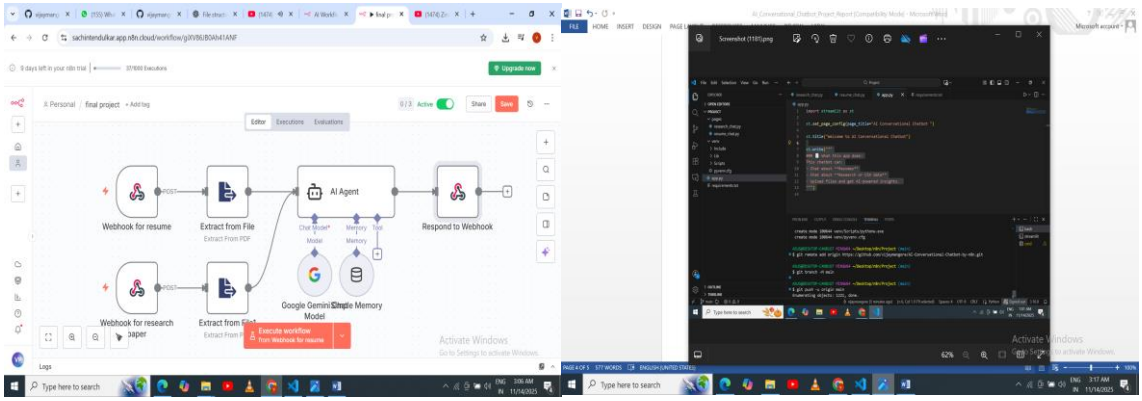


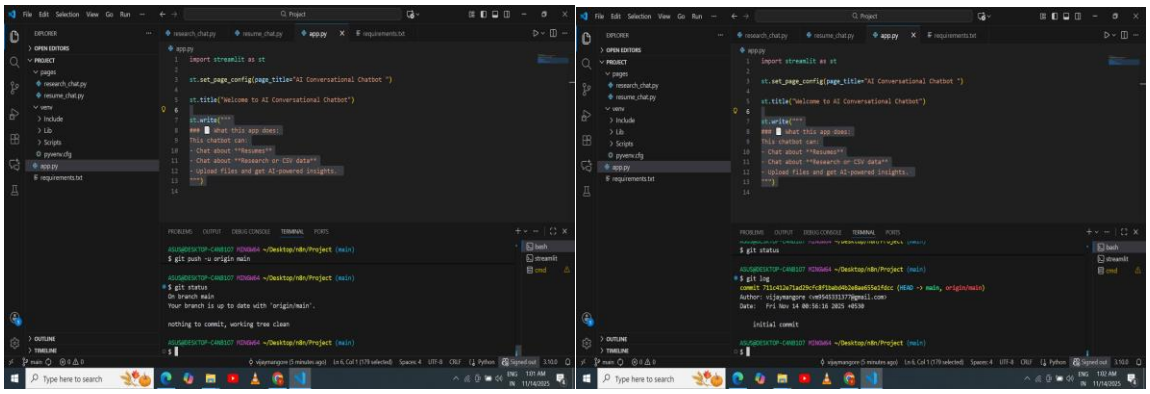




8. n8n Workflow Screenshots

(Webhook node, Extract from File node, AI Agent node, Respond node).





9. Challenges Faced & Solutions

1. File parsing issues → Solved by structured extraction in n8n.
2. API timeout → Optimized payload and used memory buffer.
3. Hallucination in answers → Added strict prompt constraint.

10. Conclusion

The AI Conversational Chatbot demonstrates how low-code tools and generative AI can work together to automate document understanding. The integration of Streamlit and n8n with Google Gemini provides an efficient, scalable, and user-friendly solution for real-world document analysis tasks. Future improvements can include OCR, multi-language support, and database-backed chat memory.

11. References

1. Streamlit Documentation — https://docs.streamlit.io/
2. n8n Documentation — https://docs.n8n.io/
3. Google Gemini API — https://ai.google.dev/gemini-api

Application Link: AI Conversational Chatbot

Github Link: Github