



AIRLINE CUSTOMER SUPPORT - MULTI-AGENT, POLICY-COMPLIANT ASSISTANT

ASWATH A
MUKHIL ADINATH P
SURYA S
SWADHI K

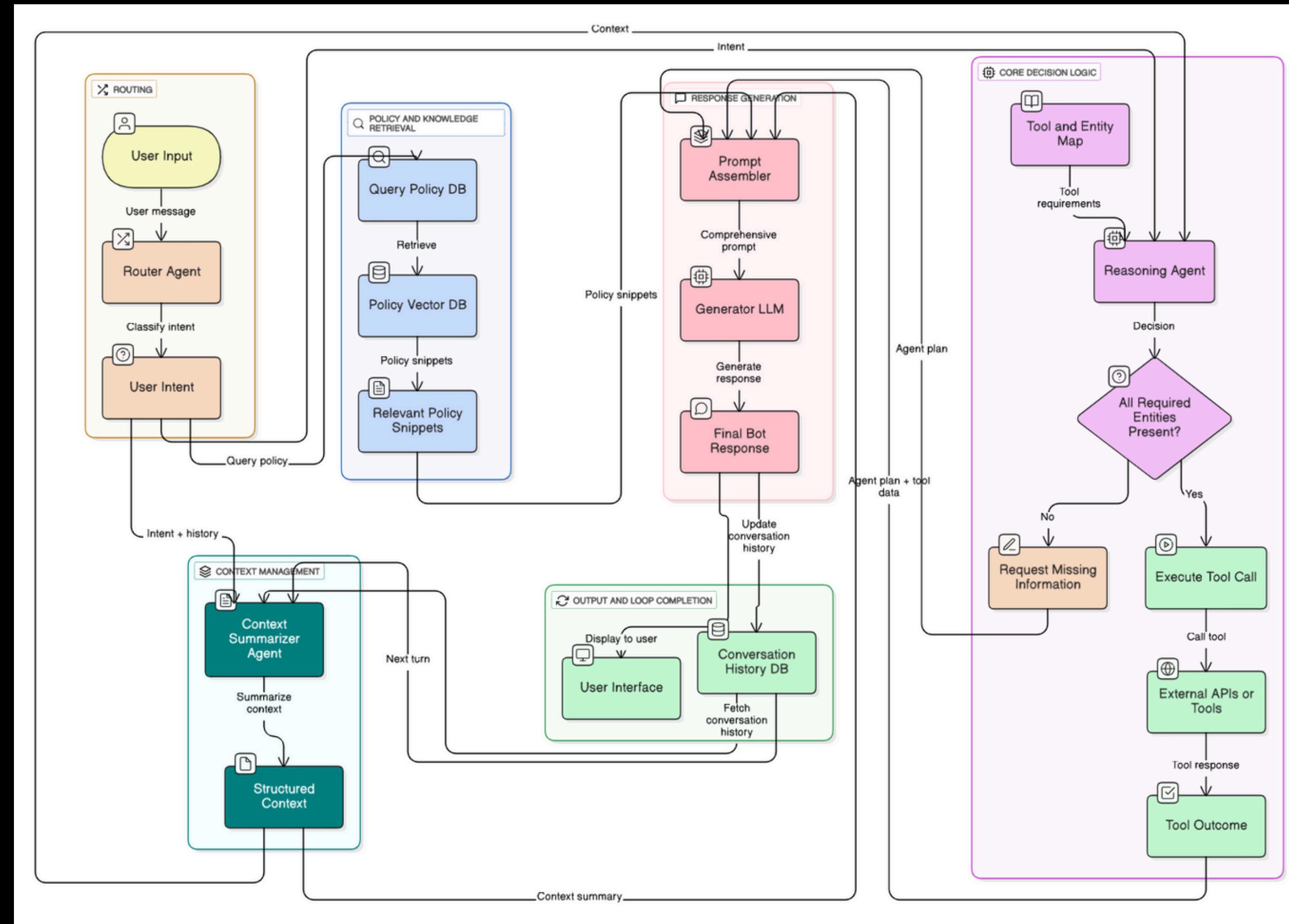
PROBLEM STATEMENT

- The task is to build an airline customer support chatbot that generates natural, human-like responses to customer messages.
- It must handle any input while staying strictly on-topic to airline-related queries.
- The system should understand user intent, maintain conversation context, and support multi-turn interactions.
- It must adhere to strict airline-specific quality control and policy guidelines in every response.
- Overall, the goal is to ensure accurate, empathetic, and policy-compliant communication that enhances customer experience.

SOLUTION OVERVIEW - MULTI-AGENT ARCHITECTURE

- The system uses a **multi-agent (api) architecture**, where each agent performs a specialized task for efficient and intelligent conversation handling.
- A **Router Agent** classifies user intent and filters off-topic messages to ensure the chatbot stays within the airline domain.
- A **Context Manager** maintains structured conversation summaries, enabling accurate **multi-turn and context-aware interactions**.
- A Reasoning Agent **retrieves relevant airline policies using RAG** (Retrieval-Augmented Generation) and executes necessary tools or APIs.
- A Response Assembler combines retrieved facts and policies to generate natural, empathetic, and policy-compliant replies with **low latency and high generalizability**.

WORKFLOW



ROUTER AGENT

➡ PURPOSE:

- The Router Agent classifies the user's message based on its intent—whether it's related to booking, status, baggage, refunds, or off-topic questions.

➡ FUNCTIONALITY:

- Off-topic queries are filtered, and the bot stays focused on airline-related issues. It directs the flow to the appropriate next agent or service.

➡ OUTCOME:

- Ensures the conversation remains on-topic and efficient, improving user experience.

CONTEXT MANAGEMENT LAYER

➡ PURPOSE:

- The Context Manager stores and organizes the relevant conversation history, ensuring the bot can recall essential details like booking references and user preferences across interactions.

➡ FUNCTIONALITY:

- Summarizes the conversation into structured memory, reducing unnecessary verbosity and improving efficiency in subsequent queries.

➡ OUTCOME:

- Enables smooth, multi-turn conversations and accurate context tracking, ensuring the bot remembers past interactions and user goals.

RAG (RETRIEVAL-AUGMENTED GENERATION)

- RAG integrates external knowledge, pulling relevant airline policies from a Vector Database.
- Functionality: Retrieves relevant policy snippets (e.g., baggage rules, cancellations) to inject into the response, ensuring compliance with airline standards.
- Outcome: Enhances accuracy and ensures that every response is grounded in current policies, offering both correctness and empathy.



PROMPT ASSEMBLER LAYER

➡ PURPOSE:

- The Prompt Assembler Layer is responsible for structuring and formatting user queries into a well-defined prompt. It ensures the correct context and intent are passed to the AI model for generating accurate and relevant responses.

➡ FUNCTIONALITY:

- It dynamically assembles detailed prompts by integrating user input, structured context, and relevant policies. This allows the model to provide responses that are not only contextually accurate but also aligned with the business logic.

➡ OUTCOME:

- optimizes the model's understanding of the conversation, ensuring the AI can effectively respond to user queries while maintaining consistency in interactions.

TECH STACK

→ STREAMLIT
→ PYTHON
→ FASTAPI
→ PYDANTIC
→ UVICORN
→ CHROMA DB

→ NUMPY
→ PANDAS
→ TRANSFORMERS
→ TORCH
→ SCIKIT LEARN
→ REQUESTS

USER EXPERIENCE

- **Natural Guidance:** Asks only for missing info and confirms before taking action.
- **Streaming Responses:** The chatbot provides real-time responses, reducing wait time and enhancing the user experience.
- **Quick-Reply Chips:** Interactive options for faster, efficient interactions.
- **Off-Topic Handling:** Gracefully redirects off-topic queries with polite refusals.





DEMO VIDEO

A screenshot of a web browser window titled "Conversation AI Demo" showing a conversational AI interface. The browser address bar shows "localhost:3001". The main content area displays a "Chat with the Bot" section where the bot says "Hello! Welcome to Airline Assistant. How can I help you today?". Below this is a text input field with the placeholder "What would you like to do?". To the right, there is a large play button icon. A sidebar on the right lists "Inner Workings", "Intent Detected", "Context Summary (Your Part)", and "Policies Retrieved". The entire interface is framed by a decorative border made of glowing neon lines.

Conversation AI - Demo

A demonstration of a multi-agent conversational workflow.

Chat with the Bot

Bot: Hello! Welcome to Airline Assistant. How can I help you today?

What would you like to do?

Inner Workings

This panel shows the real-time state of the AI's 'brain'.

Intent Detected

Context Summary (Your Part)

Policies Retrieved

CONCLUSION

This multi-agent, policy-compliant system ensures a smooth, efficient, and natural customer support experience. By leveraging intelligent routing, context management, and real-time policy retrieval, it provides accurate, empathetic responses while adhering to airline-specific guidelines. The system's ability to handle multi-turn conversations and gracefully manage off-topic queries enhances user satisfaction, making it scalable across various industries.



FUTURE IMPLEMENTATION

Future Implementation

- **Personalization:** Tailor responses based on user profiles and preferences.
- **Multilingual Support:** Expand to support multiple languages for global accessibility.
- Emotion Detection: Integrate sentiment analysis for more empathetic interactions.
- Tool Integration: Add real-time tools like flight tracking and service recommendations.
- Cross-Industry Use: Adapt the system for other sectors by replacing tools and policies.



THANK YOU

WWW.REALLYGREATSITE.COM