

Coding Round Submission

Agentic RAG Chatbot for Multi-Format Document QA using MCP

GitHub Link: <https://github.com/rak-shi>

Rakshitha Valipireddy
rakshithavalipireddy@gmail.com

Date : 8 July 2025

Tech Stack

Content Used

✓ Languages & Frameworks

- Python
- Gradio

✓ Vector Store:

- FAISS

✓ Embeddings:

- Hugging Face sentence-transformers

✓ LLM:

- OpenAI GPT

✓ Parsing Libraries:

- PyMuPDF
- python-docx
- python-pptx
- pandas

✓ Environment:

- Google Colab

Agent-Based Architecture with MCP

- **IngestionAgent** (parsing)
- **RetrievalAgent** (embedding + retrieval)
- **LLMResponseAgent** (format + GPT call)

Model Context Protocol (MCP)

- JSON structured message passing
- sender, receiver, type, trace_id, payload

Block Diagram

```
User (Gradio UI)  
  ↓  
CoordinatorAgent  
  ↓  
IngestionAgent → RetrievalAgent → LLMResponseAgent → GPT → User
```

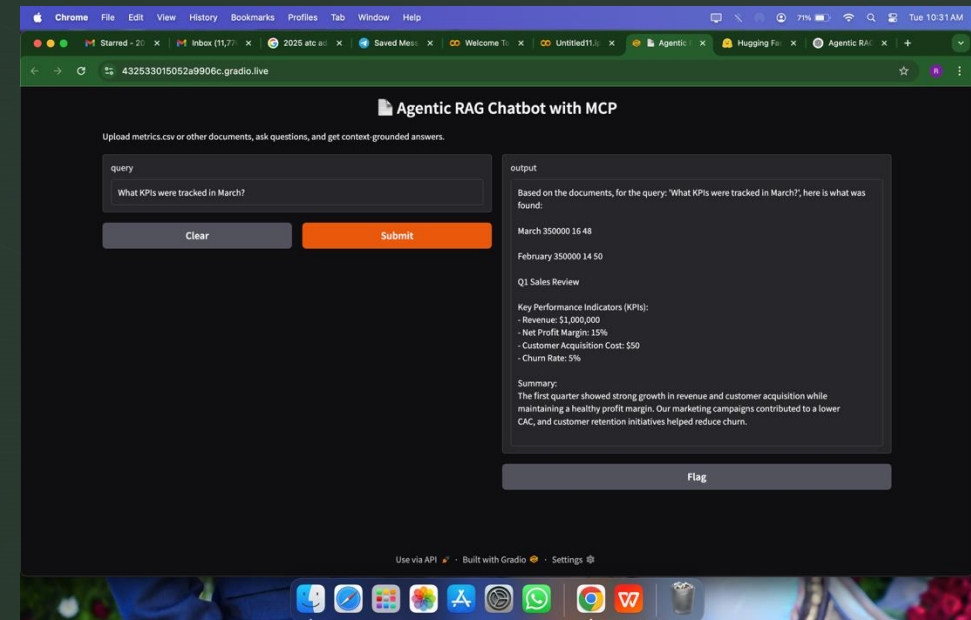
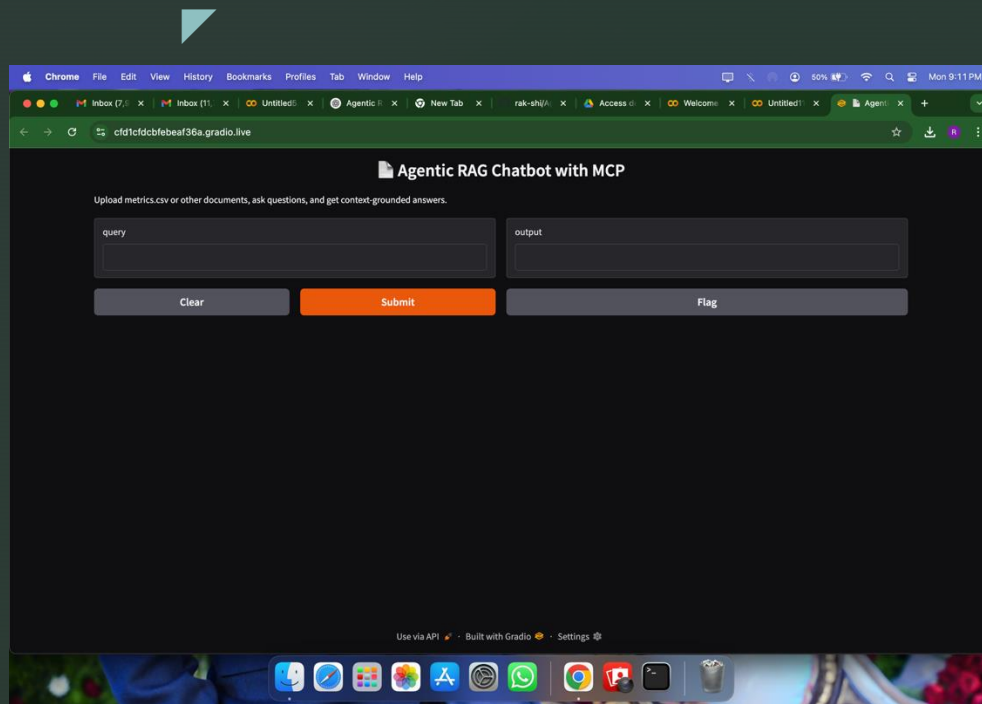
System Flow

User uploads files (PDF, DOCX, etc.)

- ✓ User query (e.g., "What KPIs were tracked in Q1?")
- ✓ Coordinator triggers:

- IngestionAgent → parse
- RetrievalAgent → retrieve chunks
- LLMResponseAgent → query GPT
- ✓ Output: GPT answer + relevant snippets.

UI Screenshots



Challenges & Future Scope

✓ Challenges Faced:

- Parsing diverse file formats
- Managing context chunking for retrieval
- Keeping system lightweight for Colab

✓ Future Scope:

- REST API endpoints
- Multi-turn memory
- Highlight source in GPT answers
- Offline embedding generation for cost reduction