

Tamoor Haider Aslam  
Progress Report Number: 5

Note: Like last week, this week I and Nushkan again did peer programming sessions utilizing LangChain in our chat system and streaming the data. Commits are pushed by both of us but mainly by me.

### Work Logs

Date	Hours	Description of Work Done
March 31	2.5	Designed and configured LangChain AgentExecutor with streaming callback support.
April 1	3	Integrated custom tools (search, arithmetic, forecasting) into the AI assistant pipeline.
April 2	2.5	Implemented custom QueueCallbackHandler for real-time streaming responses.
April 3	3	Developed structured prompt templates for agent reasoning and multi-step tool use.
April 4	3	Enabled async tool handling and tested agent responses across multiple use cases.
April 5	2.5	Finalized architecture, improved output clarity, and reviewed feedback handling logic.
April 6	2	Documented the agent framework and updated files for GitHub commit and reporting.

### Work Completed This Week

This week, I focused on advancing the **Asklytics AI Agent** by building a robust LangChain-based assistant that handles real-time data analytics and query resolution.

Using **LangChain's AgentExecutor** and **OpenAI's GPT-4o-mini**, I implemented streaming agent responses powered by a custom QueueCallbackHandler, allowing the chatbot to deliver partial, dynamic outputs in real-time.

Additionally, I developed and registered multiple custom tools including search (via SerpAPI), arithmetic operations, and time series forecasting, all designed to be triggered contextually during a conversation. I also created a structured **ChatPromptTemplate** to guide the agent's reasoning, allowing it to think step-by-step when using multiple tools in a single query.

Key achievements:

- Integrated **LangChain streaming agents** for real-time, asynchronous responses.
- Built custom tools with async support for arithmetic, forecasting, and web search.
- Developed a structured prompting system for agent-driven reasoning.
- Enhanced user interaction flow and tool execution sequencing.

One challenge was ensuring smooth concurrent execution of multiple tools without blocking, which I addressed using asynchronous routines and refined prompt formatting. Another key focus was making the chatbot scalable and maintainable for future extensions

### **GitHub Repository Check-In**

The following file was updated in the repository under asklytics/agent\_logic:

- **asklytics** – Includes LangChain streaming agent, custom tool integration, and prompt refactor.

This progress significantly improves Asklytics' user interaction quality and responsiveness, paving the way for intelligent and personalized analytics support.