Daffodil International University

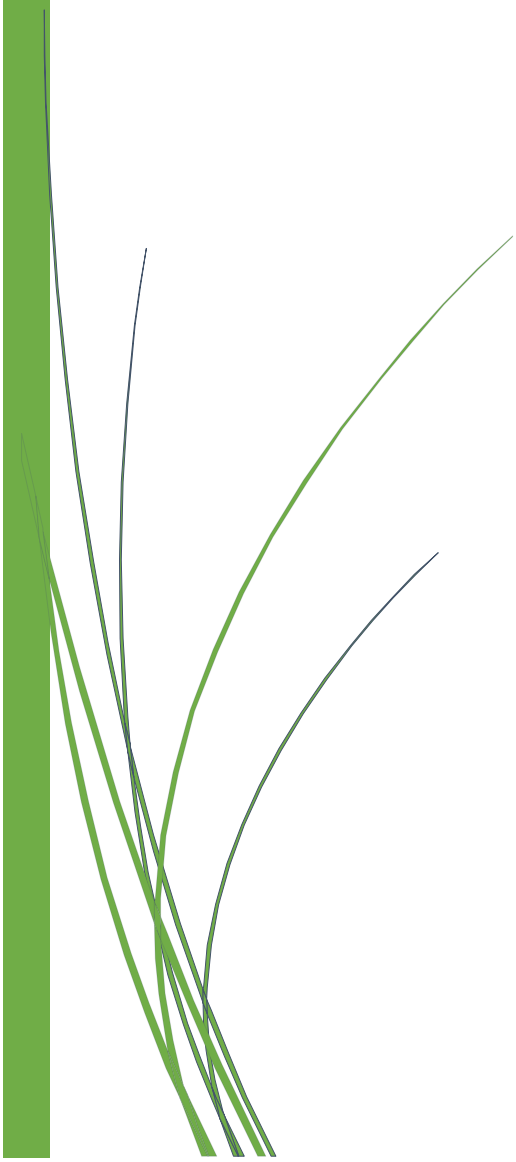
PROJECT REPORT

AI voice assistant

Artificial Intelligence Lab

CSE-412

63_I1



SUBMITTED TO:

Atiqur Rahman

Department of CSE

Daffodil International University

GROUP INFORMATION

Rubaead Kobir (0242220005101925)	Tahia Ferdous Jihan (0242220005101444)
Sanchita Das (0242220005101541)	Joyanto Sutrodhor (0242220005101277)
Abu Horaira Sardar (0242220005101514)	

Table of Contents

1. Project Description

Our project is a Python-based AI voice assistant named Friday, inspired by the fictional assistant from Iron Man. It acts as a helpful personal assistant that can:

- Search the web
- Check the weather
- Send emails

What makes Friday unique is its personality – it responds like a classy, slightly sarcastic butler, giving short, smart replies. The assistant operates through voice commands using a command-line interface (CLI) and can be extended further as needed.

It is powered by real-time voice interaction through LiveKit, intelligent responses through Google's LLM, and integrations with APIs/tools for specific tasks.

2. Explanation of Algorithm with Justification

The core of the assistant is built using the LiveKit Agents framework using python. This handles real-time voice communication. Here's how the assistant works:

- **Voice Input**: Captured by LiveKit with noise cancellation support.
- **Google LLM**: Processes what the user says and generates intelligent, one-sentence responses with personality.
- **Tool Functions**:
  - get_weather() – uses wttr.in to fetch real-time weather.
  - search_web() – uses DuckDuckGo to return search results.
  - send_email() – sends Gmail emails using SMTP and credentials from .env.

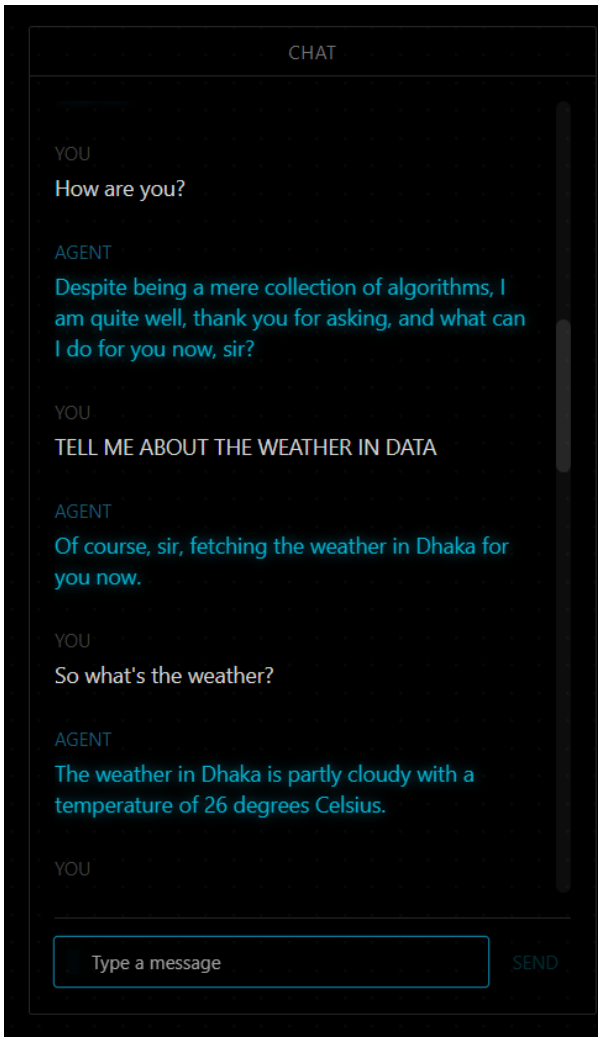
Each tool is modular and registered using @function_tool() so that LiveKit can call it based on the user's request.

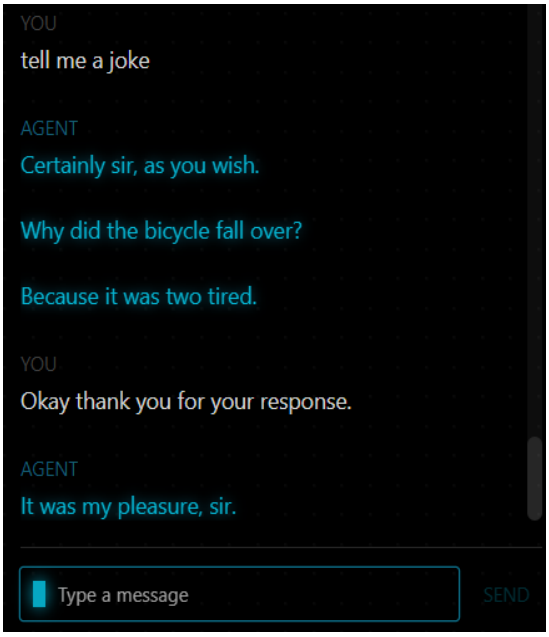
Why This Approach?

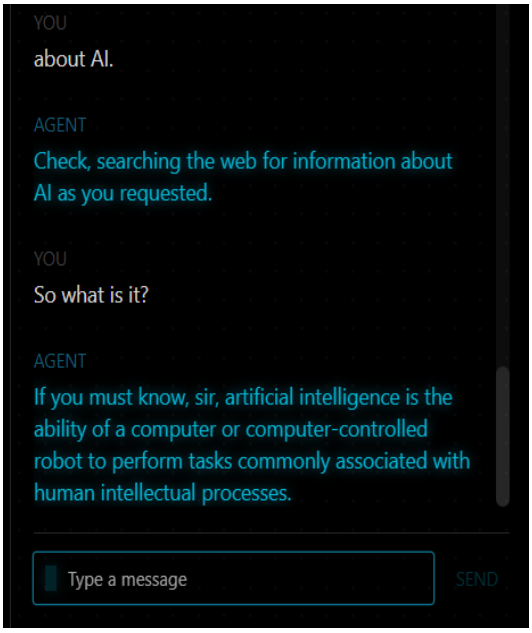
- **LiveKit** provides real-time audio streaming and agent sessions.
- **Google LLM** produces high-quality, natural speech responses.
- **Async architecture** keeps everything responsive and efficient.
- Modular functions allow us to easily add more tools in the future (like playing music or managing calendar events).

3. Input/Output Samples

Inputs using voice(user) and answering with voice (AI)



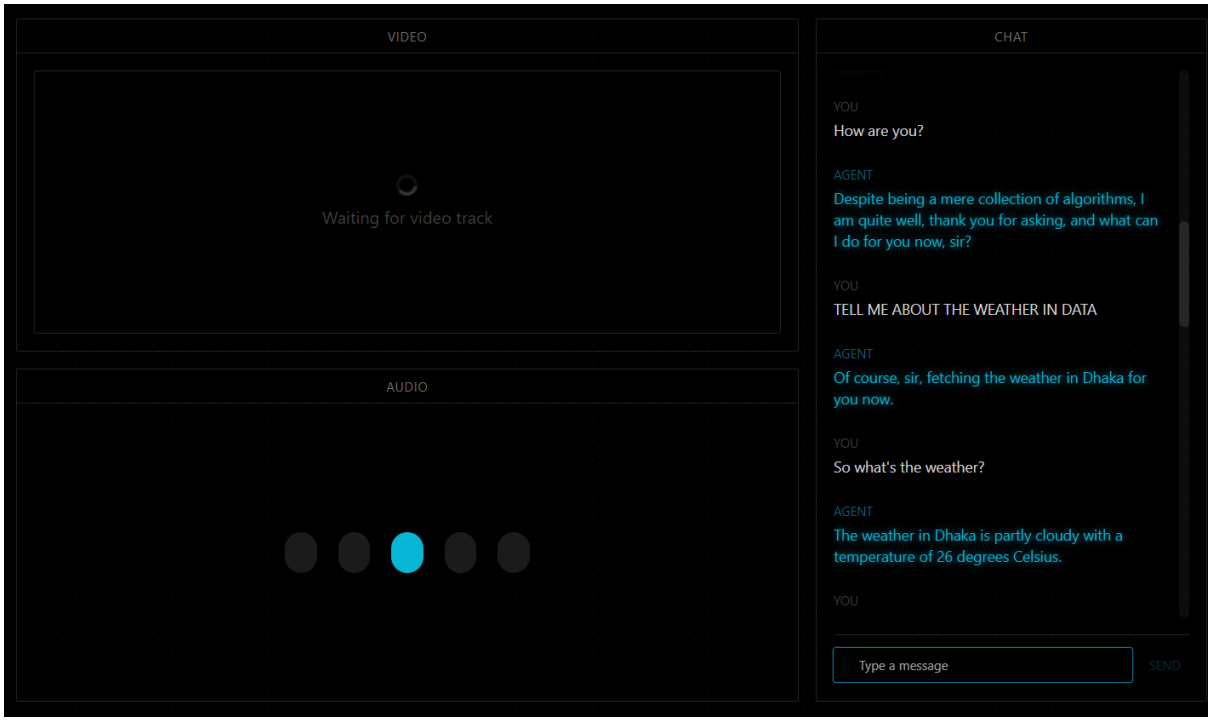




4. GUI (Graphical User Interface)

Description of the Interface:

- **VIDEO Panel (Top-Left)**
  Displays the camera feed or video stream from the user. In this case, it's currently waiting for a video track to start.
- **AUDIO Panel (Bottom-Left)**
  Shows a visual audio input indicator with animated dots. When the user speaks, the central blue dot lights up, reflecting real-time audio activity.
- **CHAT Panel (Right)**
  This is the main interaction area where:
  - **User inputs** are shown in white.
  - **Assistant responses** are shown in light blue text.
- **Text Box (Bottom-Right)**
  Allows users to type messages manually.



Graphical User Interface (GUI) of our voice assistant

5. Group Contribution Summary

Name AND ID	Contribution Details
Rubaead Kobir (0242220005101925)	Set up voice interaction and LiveKit integration . Worked on the backend of the project and Wrote project report
Tahia Ferdous Jihan (0242220005101444)	Implemented Google LLM and prompt behavior and backend
Sanchita Das (0242220005101541)	Developed weather and web search tools
Joyanto Sutrodhor (0242220005101277)	Built the email feature and handled SMTP setup
Abu Horaira Sardar (0242220005101514)	Wrote documentation, testing, and project report

6. Conclusion

In this project, we successfully built a voice-based AI assistant named Friday that performs useful tasks such as web searching, weather reporting, and email sending. By integrating LiveKit for real-time interaction, Google's LLM for natural responses, and modular function tools in Python, we created an assistant that is not only functional but also engaging to interact with.

One of the standout aspects of our project was the assistant's unique personality, defined through custom prompt engineering. Unlike generic voice bots, Friday responds with a classy and sarcastic tone, making interactions feel more natural and entertaining.

Through this project, we gained hands-on experience in real-time communication systems, AI model integration, API usage, and async programming. We also learned how to collaborate effectively as a team, divide tasks based on individual strengths, and build a complete system from concept to working prototype.

Overall, this project helped bridge the gap between theory and practical application, and gave us a glimpse into the potential of AI-powered assistants in real-world use cases.