The Superior University, Lahore

(Fall 2023)

Course Title:	AI Lab				Course Code:		Credit	3
Instructor:	Mr Rasikh					BSDS		
Semester:			F23		BSDSM-3A	Date:	30 / 12 / 24	
Time Allowed:					Maximum Marks:			
Student’s	Afzaal Mustafa				Roll no :	BSDS-037		

Project Report: Voice Assistant - "Parlina"

1. Introduction

This project involves the creation of a Python-based voice assistant named "Parlina." The assistant is designed to perform various tasks such as retrieving information from Wikipedia, playing music, opening websites, sending emails, providing the current time, and more. By leveraging natural language processing and speech recognition, the assistant interacts with users through voice commands.

---

2. Objectives

- To develop a user-friendly voice assistant capable of performing routine tasks.
- To integrate APIs and libraries to enhance the assistant's functionality.
- To demonstrate the use of Python in building interactive systems.

---

3. Features

1. Personalized Greetings:
    - Based on the time of day, Parlina greets the user with "Good Morning," "Good Afternoon," or "Good Evening."
    - Prompts users for their name and addresses them personally during the session.
2. Speech Recognition:
    - Converts voice input to text using the speech_recognition library.
3. Wikipedia Integration:
    - Searches for and summarizes information from Wikipedia.
4. Web Browser Control:
    - Opens popular websites like YouTube, Google, and Stack Overflow via voice commands.
5. Music Player:
    - Randomly selects and plays music from a specified directory.
6. Email Sending:
    - Sends emails via SMTP, allowing voice input for content.

7. Time Announcement:
   - Provides the current time upon request.
8. News Updates:
   - Fetches top news headlines using the NewsAPI.
9. WolframAlpha Integration:
   - Performs mathematical calculations and answers queries using the WolframAlpha API.
10. Customized Responses:
    - Answers queries such as "Who made you?" or "What's your name?" with pre-programmed responses.

---

4. Technology Stack

1. Programming Language:
   - Python
2. Libraries and Frameworks:
   - speech_recognition: For voice-to-text conversion.
   - pyttsx3: For text-to-speech conversion.
   - wikipedia: For fetching data from Wikipedia.
   - wolframalpha: For calculations and advanced queries.
   - webbrowser: To open URLs.
   - smtplib: For email functionality.
   - ecapture: For camera functionality (e.g., taking photos).
   - urllib and BeautifulSoup: For web scraping and data retrieval.
   - datetime: For time-based features.
   - os and shutil: For file and system operations.

5. Workflow

1. Initialization:
    - The assistant initializes the text-to-speech engine (pyttsx3) and greets the user.
2. User Input:
    - Listens to voice commands using a microphone (speech_recognition).
    - Recognizes the command using Google's speech recognition API.
3. Task Execution:
    - Based on the parsed command, it performs the corresponding task:
        - Searching Wikipedia.
        - Playing music.
        - Opening websites.
        - Performing calculations.
        - Fetching news.
4. Response:
    - Provides auditory feedback using pyttsx3 and visual feedback via the terminal.

6. Challenges and Limitations

1. Accuracy:
    - Speech recognition depends on background noise and microphone quality.
2. Email Security:
    - Requires Gmail’s "less secure apps" feature to be enabled, which is not recommended for production.

3. Dependency on APIs:
    - Reliance on external APIs like WolframAlpha and NewsAPI may lead to failures if the keys are invalid or quotas are exceeded.
4. Limited Natural Language Understanding:
    - The assistant is rule-based and does not handle complex conversational flows.

< ---------------------------- Good Luck ------------------------------------->