Voice Assistant Project Documentation

Introduction

The Voice Assistant project, named "Sifra," is an AI-powered application designed to assist users with various tasks using voice commands. It integrates speech recognition, text-to-speech, web automation, and API integrations to perform functions such as fetching news, retrieving weather updates, managing applications, and interacting with OpenAI's ChatGPT. This project was developed as part of a placement exam submission, demonstrating expertise in Python, automation, and natural language processing (NLP).

Libraries Used

The project utilizes several Python libraries to enhance its functionality:

- **pyttsx3**: Converts text to speech
- **speech_recognition**: Converts spoken language into text
- **selenium**: Automates web interactions (YouTube searches, Wikipedia lookups)
- **requests**: Fetches data from APIs (weather, news, jokes, facts)
- **randfacts**: Generates random facts
- **datetime**: Retrieves and formats the current date and time
- **openai**: Enables ChatGPT responses
- **psutil**: Manages and terminates running applications
- **tkinter**: Creates a graphical user interface for user interaction
- **webdriver_manager**: Ensures compatibility with the latest ChromeDriver

Key Features

1. **Voice Interaction**: Recognizes user commands and responds using synthesized speech.
2. **Information Retrieval**: Fetches news headlines and Wikipedia data.
3. **Weather Updates**: Provides real-time weather details based on user-input city names.
4. **Application Management**: Opens and closes applications like Notepad, Chrome, and Calculator.
5. **Joke & Fact Generator**: Fetches random jokes and facts for user engagement.
6. **ChatGPT Integration**: Retrieves AI-generated responses for user queries.
7. **Graphical User Interface (GUI)**: Implements a Tkinter-based interface for ease of use.

Technical Architecture

- **Speech Processing**:
    - Uses speech_recognition to capture and transcribe user input.

    - Converts text to speech using pyttsx3.
- **Web Automation**:
    - Uses selenium to interact with YouTube and Wikipedia.
- **API Integration**:
    - Fetches news from NewsAPI, weather data from OpenWeatherMap, and jokes from a public API.
- **Modular Design**:
    - Implements functions to handle different tasks such as get_weather, get_news, and chatgpt_response.
- **GUI Interface**:
    - Uses tkinter to create a chat window for displaying responses.

Main Challenges Faced & Solutions Implemented

1. **Speech Recognition Accuracy**
    - **Challenge**: Background noise and unclear pronunciations affected recognition quality.
    - **Solution**: Used adjust_for_ambient_noise() to improve recognition efficiency.
2. **Handling API Errors**
    - **Challenge**: Network issues or incorrect API keys caused failures.
    - **Solution**: Implemented exception handling to manage API request failures and provide fallback responses.
3. **Web Automation Stability**
    - **Challenge**: Changes in YouTube's HTML structure caused failures in video search.
    - **Solution**: Used dynamic XPath selection to ensure robustness.
4. **Efficient Response Time**
    - **Challenge**: Delays in processing multiple voice commands.
    - **Solution**: Optimized speech recognition by adjusting energy threshold values.

Conclusion

The "Sifra" Voice Assistant project showcases proficiency in AI-powered automation, NLP, and API integrations. It demonstrates the ability to build an interactive, user-friendly assistant with advanced features, making it a strong submission for the placement exam.

About the Developer

- **Name**: Viki Kumar
- **Role**: Developer
- **Project**: Voice Assistant
- **Institution**: Lovely Professional University
- **Email**: vikikumar0505@gmail.com
- **Submission for**: Placement Exam