Task 2: Voice Assistant

Objective

Create a custom voice assistant using Python that can automate tasks and personalize responses according to user needs.

Methodology

1. Setup Environment: Install necessary libraries (SpeechRecognition, pyaudio, pyttsx3).

2. Initialize Components: Set up the speech recognizer and text-to-speech engine.

3. Define Functions: Create functions for speaking, listening, and running the assistant.

4. Implement Command Handling: Add functionality to respond to various commands.

5. Test the Assistant: Ensure the assistant works as expected and can handle different inputs.

Implementation

1. Environment Setup:

• Install necessary libraries:

pip install SpeechRecognition pyaudio pyttsx3

2. Voice Assistant Code:

import speech\_recognition as sr

import pyttsx3

# Initialize the recognizer and text-to-speech engine

recognizer = sr.Recognizer()

tts\_engine = pyttsx3.init()

def speak(text):

"""Convert text to speech."""

tts\_engine.say(text)

tts\_engine.runAndWait()

def listen():

"""Listen for user input and return it as text."""

with sr.Microphone() as source:

print("Listening...")

audio = recognizer.listen(source)

try:

text = recognizer.recognize\_google(audio)

print(f"You said: {text}")

return text

except sr.UnknownValueError:

print("Sorry, I did not understand that.")

return ""

except sr.RequestError:

print("Sorry, my speech service is down.")

return ""

def voice\_assistant():

"""Main function to run the voice assistant."""

speak("Hello, how can I assist you today?")

while True:

command = listen().lower()

if "stop" in command:

speak("Goodbye!")

break

elif "your name" in command:

speak("I am your custom Python voice assistant.")

elif "time" in command:

from datetime import datetime

now = datetime.now().strftime("%H:%M:%S")

speak(f"The current time is {now}")

else:

speak("I'm sorry, I don't understand that command.")

# Run the voice assistant

if \_\_name\_\_ == "\_\_main\_\_":

voice\_assistant()

Explanation

1. Initialization: Import the required libraries and initialize the speech recognizer and text-to-speech engine.

2. Functions:

• speak(text): Converts text to speech.

• listen(): Listens for user input and converts it to text.

• voice\_assistant(): Main function to run the assistant, providing responses to user commands.

3. Commands: The assistant responds to specific commands such as asking for its name or the current time. It listens for the word “stop” to terminate.

Results

Run the script, and the voice assistant will start listening for your commands. You can interact with it by speaking into your microphone.