TITLE OF PROJECT REPORT

Desktop Assistant Using Python

‘YUDHISHTHIR’

A REPORT ON PROJECT BASED LEARNING

(SEMESTER –II)

Submitted by-

NAME OF THE CANDIDATE

Prasad Unecha

FIRST YEAR ENGINEERING

Society for Computer Technology and Research’s

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

, PUNE – 43

A.Y. 2020-21

Abstract

It all started in 1961 when IBM introduced IBM shoebox,1st digital speech recognition tool, it recognized 16 words and digits . It is 21 CE, Artificial intelligence and machine learning is being infused in today's voice assistants. Understanding the basics of python and different modules in python we started working on our own voice assistant –**Yudhishthir .**
We imported several modules like pyttsx3, web browser, speech recognition ,tkinter ,smtplib ,pyaotugui , beautifulsoup, time etc.
AI-based Voice assistants are the operating systems that can recognize human voice and respond via integrated voices. This voice assistant will gather the audio from the microphone and then convert that into text, later it is sent through 'Sapi5' - Microsoft developed API. It allows use of speech recognition and 'speech synthesis' within windows application, then that audio is played using play sound package of python programming Language. We incorporated different commands into our voice assistant like certificate generator, finding location of place,spam bot etc. Results are limited, incomplete and unorganized. As a conclusion, the role of assistant and its use in the various sectors are still at an early stage of research and more studies need to address this topic.

ACKNOWLEDGEMENT

I would like to express our special thanks of gratitude to our teacher Professor Usha Polina mam as well as our Hod – Professor E.M Reddy sir who gave us the golden opportunity to do this wonderful project -Creating a desktop assistant, which also helped us in doing a lot of Research and we came to know about so many new things we are really thankful to them. Furthermore we would also like to thank our friends who helped us a lot in finalizing this project within the limited time frame.

Place: PUNE Name of Student (in Capital) &Sign

TABLE OF CONTENT

1]Introduction

A) What is personal voice assistant?

- An intelligent **virtual assistant** (IVA) or intelligent **personal assistant** (IPA) is a software agent that can perform tasks or services for an individual based on commands or questions. Some **virtual assistants** are able to interpret human **speech** and respond via synthesized **voices**.

B) Concept-

- Based on specific commands, sometimes called intents, spoken by the user, voice assistants can return relevant information by listening for specific keywords and filtering out the ambient noise.
- Voice assistants use Artificial Intelligence and Voice recognition to accurately and efficiently deliver the result that the user is looking for. While it may seem simple to ask a computer to set a timer, the technology behind it is fascinating.

Voice recognition-

- **Voice recognition** works by taking an analog signal from a user voice and turning it into a digital signal. After doing this, the computer takes the digital signal and attempts to match it up to words and phrases to recognize the user intent. To do this, the computer requires a database of pre-existing words and syllables in a given language to be able to closely match the digital signal with.

Artificial Intelligence-

- These systems can improve upon themselves using a process known as machine learning.

C] Objectives-

- The main goal is to provide people a quick and easy way to have their questions answered.

Let us take an example of a person working in an IT sector:

Before getting dressed for the day he can ask assistant for the current weather forecast.

Assistant alerts him when he should probably leave for work based on current traffic conditions along his typical route and how long it will take him to get there.

During his morning process assistant tells him his known agenda for the day. Then assistant gives him the news assistant knows he will be interested in seeing. Assistant also lets him know how the financial market items, sports teams and other special interests that he is tracking.

While he is driving assistant tells him when an SMS comes in and lets him to respond without taking his eyes off the road or his hands of the wheel. Assistant also connects him to people he needs to speak with while he is in transit.

If he has got any appointments assistant will let him know when he needs to leave Etc.

2] Python learning and technical skills-

A] PYTHON LEARNING:

Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (Particularly, procedural), object-oriented and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library

B] Programmed design:

When you make your app's tasks available through assistant, you have several opportunities to customize the user experience. At a fundamental level, you customize the flow and functionality of the everyday tasks and actions you support to implement your business requirements. To reinforce this functionality throughout the user experience, you can write dialog that reflects the style and tone of your company's communications and design custom UI that incorporates your app's visual style into the Siri interface

C] Design a Great Voice Experience

- A great voice interface helps people feel confident they'll get the results they want, even when they're not sure what they can say.
- With a custom intent, your app helps people perform a task that assistant doesn't know about yet, which results in a different type of support for the voice experience. Custom intents give you additional opportunities to customize conversational dialog, but also require people to create and speak a precise phrase to trigger the interaction.
- As a designer, you have several ways to enhance both types of conversational experiences and help people specify what they want without engaging in lengthy exchanges.

D]Understanding different modules in python

- **Subprocess: -** This module is used for getting system subprocess details which are used in various commands that is Shutdown, Sleep, etc. This module comes built-in with Python.

- **Wolfram Alpha: -** It is used to compute expert-level answers using Wolfram's algorithms, knowledgebase and AI technology. To install this module, type the below command in the terminal.

#pip install wolframaplha

- **Pyttsx3: -** This module is used for the conversion of text to speech in a program it works offline. To install this module, type the below command in the terminal.

pip install pyttsx3

An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3.Engine instance

- During construction, the engine initializes a pyttsx3.driver.DriverProxy object responsible for loading a speech engine driver implementation from the pyttsx3.drivers module.

- **Tkinter :-** This module is used for building GUI and comes inbuilt with Python. This module comes built-in with Python.

- **Wikipedia: -** As we all know Wikipedia is a great source of knowledge. we have used the Wikipedia module to get information from Wikipedia or to perform a Wikipedia search. To install this module, type the below command in the terminal.

#pip install wikipedia

- **Speech Recognition: -** Since we're building an application of voice assistant, one of the most important things in this is that your assistant recognizes your voice (means what you want to say/ ask). To install this module, type the below command in the terminal.

- The first component of speech recognition is, of course, speech. Speech must be converted from physical sound to an electrical signal with a microphone, and then to digital data with an analog-to-digital converter. Once digitized, several models can be used to transcribe the audio to text.

pip install SpeechRecognition

- **Web browser: -** To perform Web Search. This module comes built-in with Python.

- **Ecapture:-** To capture images from your Camera. To install this module, type the below command in the terminal.

#pip install ecapture

- **Pyjokes:-** Pyjokes is used for collection Python Jokes over the Internet. To install this module, type the below command in the terminal.

#pip install pyjokes

- **Datetime: -** Date and Time is used to showing Date and Time. This module comes built-int with Python.

The **datetime** module supplies classes for manipulating dates and time

While date and time arithmetic are supported, the focus of the implementation is on efficient attribute extraction for output formatting and manipulation.

Date and time objects may be categorized as “aware” or “naive” depending on whether they include time zone information or not.

Constants:

1.) datetime.MINYEAR

2.) datetime.MAXYEAR

- **Twilio: -** Twilio is used for making call and messages. To install this module, type the below command in the terminal.

#pip install twilio

OS module:

- The OS module in Python provides functions for interacting with the operating system. OS comes under Python's standard utility modules. This module provides a portable way of using operating system dependent functionality. The *os* and *os.path* modules include many functions to interact with the file system.
- It is possible to automatically perform many operating system tasks. The OS module in Python provides functions for creating and removing a directory (folder), fetching its contents, changing and identifying the current directory, etc.
- You first need to import the os module to interact with the underlying operating system. So, import it using the import os statement before using its functions

- **BeautifulSoup:** Beautiful Soup is a library that makes it easy to scrape information from web pages. To install this module, type the below command in the terminal.

#pip install beautifulsoup4

3] Implementation and Description of work:

A]Engine

We have defined engine as a variable

pyttsx3 is a text-to-speech conversion library in Python .

'Sapi 5 :

- Microsoft developed speech API.

- Helps in synthesis and recognition of voice.

What Is Voice Id?

- Voice id helps us to select different voices.
- voice[0].id = Male voice
- voice[1].id = Female voice

```
engine = pyttsx3.init('sapi5')
voices = engine.getProperty('voices')
# print(voices[0].id)
engine.setProperty('voice', voices[0].id)
```

B]Speak

We have defined **speak()** as a function**.** This function will take audio as an argument, and then it will vocalize it.

```
def speak(audio):
    engine.say(audio)
    engine.runAndWait()
    # getting details of current speaking rate
    rate = engine.getProperty('rate')
    # printing current voice rate
    # setting up new voice rate
    engine.setProperty('rate', 150)
```

C] Wish me:

- The team has defined a wishme() function that will make the voice assistant , wish or greet the user according to the time of computer.
- For providing accurate current time to our assistant, Import of datetime module has been done

```
def wishMe():
    hour = int(datetime.datetime.now().hour)
    if hour >= 0 and hour < 12:
        speak("Good Morning")

    elif hour >= 12 and hour < 18:
        speak("Good Afternoon")
    else:
        speak("Good Evening")
    speak(" my name is yudhishthir. hello boss , how may i help you?")
```

D]Take Command

- With the help of the takeCommand() function, assistant will return a string output by taking microphone input from the user.

- Before defining the takeCommand() function, we need to install a module called **speechRecognition**

```python
def takeCommand(ask=False):
    # it takes microphone input from the user and returns string output

    r = sr.Recognizer()
    with sr.Microphone() as source:
        if ask:
            print(ask)
        print("Listening....")
        r.pause_threshold = 1
        r.energy_threshold = 500
        audio = r.listen(source)

    try:
        print("Recognizing....")
        query = r.recognize_google(audio, language='en-in')
        print("user said:", query)

    except Exception as e:
        # print(e)

        print("say that again please....")
        return "None"
    return query
```

#We will create a main() function, and inside this main() Function, we will call our speak function.

E] Search from Wikipedia

- For searching on Wikipedia domain, we need to install and import the Wikipedia module into our program.
- In the below code, we have used an if statement to check whether Wikipedia is in the user's search query or not.
- If Wikipedia is found in the user's search query, then two sentences from the summary of the Wikipedia page will be converted to speech with the speak function's help.

```
if 'wikipedia' in query:
    speak('searching Wikipedia....')
    query = query.replace("wikipedia", "")
    results = wikipedia.summary(query, sentences=1)
    speak("According to wikipedia")
    print(results)
    speak(results)
```

F] Time

ctime() converts a **time** expressed in seconds to a string representing local **time.**

```
elif 'time' in query:
    print(ctime())
    speak(str('time is'+ ctime()))
```

G] Search in google

Opening Google in a web-browser using elif statement and module used is web browser.

```
elif 'google' in query:
    speak('opening google')
    webbrowser.open("google.com")
```

H] Certificate Generator

- If we get certificate as a query , then Yudhishthir does sends the certificate via mail to the respective person.
- Here we import MIMEMultipart this module is used to support transfer of single or multiple text and non text attachment.
- Encoder is used here for speedy and accurate operation. By using this we can send same message to multiple people at a time.

```
elif 'certificate' in query:
    from os import close
    import smtplib
    from email.mime.multipart import MIMEMultipart
    from email.mime.text import MIMEText
    from email.mime.base import MIMEBase
    from email import encoders
    import csv
    from PIL import Image, ImageDraw, ImageFont
    import pandas as pd
```

I] Calculator

- If Yudhishithir gets calculator in query calculator opens on screen and we can perform simple mathematical calculations on it.
- There is a lot of scope to improve it and we are working on it to perform diffcult calculations also by using this calculator**.**

```
elif 'calculator' in query:
    speak('opening calculator')

    def click(event):
        global scvalue
        text = event.widget.cget("text")
        print(text)
        if text == "=":
            if scvalue.get().isdigit():
                value = int(scvalue.get())
            else:
                value = eval(screen.get())
```

J] Sending Mail:

What is smtplib?

- Simple Mail Transfer Protocol (SMTP) is a protocol that allows us to send emails and route emails between mail servers. An instance method called **sendmail** is present in the SMTP module. This instance method allows us to send an email. It takes 3 parameters:
- **The sender:** Email address of the sender.
- **The receiver:** T Email of the receiver.
- **The *message:*** A string message which needs to be sent to one or more than one recipient.

```
elif 'mail' in query:
    import smtplib
    import speech_recognition as sr
    import pyttsx3
    from email.message import EmailMessage
```

K] Spam bot

- If we ask Yudhishthir to spam then it can send a particular message which is saved in data csv and can be sent number of times as we mention it in the code.
- It can be used as greeting and away messages in similar apps activity.

```
elif 'spam' in query:
    speak('spamming')
    import pyautogui
    import webbrowser as wb
    import time
```

L]Finding the location

- Here we have used web browser module and defined url link where we find the location of the required place.
- The required location is opened on Google maps.

```
elif 'find location' in query:
    # to find location of any place
    speak('What is the location ?')
    location = takeCommand('What is the location ?')
    url = 'https://google.nl/maps/place/' + location + '/&amp;'
    webbrowser.get().open(url)
    print('Here is the location of ' + location)
    speak('Here is the location of' + location)
```

M]Checking price of my laptop

If we a set particular price at which we want to buy a laptop whenever the price comes within that range Yudhishthir will send us a mail regarding the same.

```
elif 'Check My laptop Price' in query:
    speak('checking price')
    print('Checking....')
    import requests
    from bs4 import BeautifulSoup
    from email_alert import alert_system
    from threading import Timer
```

N] Game 1

This code is modified as a fun activity. We can play Tic Tac Toe game .We can play this game with two players also.

```
elif 'games(tik tak toe)' in query:
    import time
    import random

    # welcome script
    time.sleep(1.5)
    print('----------------------------------Welcome to Tic-Tac-Toe----------------------------------')
    speak('Welcome to Tic-Tac-Toe')
    time.sleep(0.5)
    print('This game is made by Yudhishthir\n')
    speak('This game is made by Yudhishthir')
```

O] Game 2

- When we ask Yudhishthir to play rock paper scissors it opens the new window to play stone paper scissor game.
- This is also part of fun activity.

```
elif 'games(rock paper scissor' in query:
    from games import rps
    speak('playing game')
    print('playing game')
    rps()
```

P] Weather Report

- When we ask Yudhishthir to tell the weather report of particular city at a particular time it tells us the temperature, wind speed and the overall weather report at that time.
- With the help of this feature, we can get instant weather report which can be very useful.

```
elif 'Check weather (any city in any country)' in query:
    import requests
    from pprint import pprint
    def weather_data(query):
        res=requests.get('http://api.openweathermap.org/data/2.5/weather?'+query+'&APPID=b35975e18dc9
        return res.json();
    def print_weather(result,city):
        print("{}'s temperature: {}°C ".format(city,result['main']['temp']))
```

Q] Security Surveillance

- It is a video surveillance system that detects the movement of an object and produces an alarming sound
- It is beneficial for security surveillance at any store or for personal use.

```
def security():
    import cv2
    import winsound
    cam = cv2.VideoCapture(0)
    while cam.isOpened():
        ret, frame1 = cam.read()
        ret, frame2 = cam.read()
        diff = cv2.absdiff(frame1, frame2)
        gray = cv2.cvtColor(diff, cv2.COLOR_RGB2GRAY)
```

(variable) frame2: Any

R] Exit

The following image represents the query which is used to exit from the code.

```
elif 'bye' in query:
    speak('thank you boss')
    exit()
```

4]

CONCLUSION

- A personalized virtual desktop assistance can be made with the knowledge of python , its module (text to speech…) using various functions and adding a little bit of creativity to it.
- Understanding of modules and functions is necessary for implementation of various commands and using the voice assistant for different purpose.
- A person's daily task can be performed smartly and efficiently with the use of personalized virtual desktop assistant.

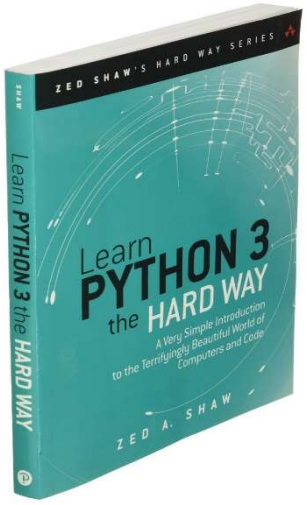
5]

FUTURE SCOPE

- We will be launching an application in the near future suitable for all operating systems.
- The application will contain the code and data of our virtual desktop assistant and will have access to large number of applications.
- A desire to create a smart home project using automation for home usage appliances through voice assistance.

REFERENCES :

1. Reference from geeksforgeeks – geeksforgeeks.com
2. Reference from GitHub – github.com



3. Learn python the hard way by Zed A. Shaw-
4. Research paper study:
5. RESEARCH PAPER BY:

1)IRJET (INTERNATIONAL RESEARCH JOURNAL OF ENGINEERING TECHNOLOGY)

IMPORTANT POINTS FROM THE RESEARCH PAPERS:

The overall system design consists of following phases:

a. Data collection in the form of speech.
b. Voice analysis and conversion to text
c. Execute Python script
d. Generating speech from the processed text output

2) RESEARCH PAPER:

Voice Assistants and Smart Speakers in Everyday Life and in Education George by

TERZOPOULOS, Maya SATRATZEMI