

VIRTUAL MASTER USING PYTHON

NAME- AYUSHI ANKITA RAKSHIT

DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING,

ASANSOL ENGINEERING COLLEGE, WEST BENGAL, INDIA.

Email: - ayushiarakshit@gmail.com

ABSTRACT

Virtual Master is a virtual assistant or a software agent like 'Ok Google', 'Hi Alexa' or 'Hello Siri' who can understand natural language voice commands and complete the specified task for the user. This paper is a technical brief on Virtual Master and its applications. Major modules needed for this Virtual Master are also described.

CONTENT

A Virtual Master can tell the current date, current time, responds back with a random greeting, can return information on a person and many more things. To perform these tasks, Virtual Master must have three basic skills:

1. Word Processing Skill
2. Good Oral communication
3. Quick thinking and effective decision making

This is really an interesting thing and it is possible using Python, because Python offers a good major library. Using Python version 3.9.0 and PyCharm IDE commands will be given to the system.

The modules needed for this Virtual Master are as follows:

1. *PyAudio (version 0.2.11):*

PyAudio is a basic Python library which is used to play and record audio. It provides Python bindings for PortAudio, the cross-platform audio I/O library. Command to install this module:

pip install PyAudio

2. *SpeechRecognition (version 3.8.1):*

It is used to convert audio into text for processing. Command to install this module:

pip install SpeechRecognition

3. gTTS (version 2.2.1):

gTTS is the Google Text to Speech API which converts the text into audio and can be saved as a mp3 file. Command to install this module:

pip install gTTS

4. Wikipedia (version 1.4.0):

It is a Python library that makes it easy to access and parse data from Wikipedia. Command to install this module:

pip install Wikipedia

Full Procedure:

- Create a new project file in PyCharm IDE in virtual environment.
- *Import the libraries:*

1.import speech_recognition as sr

2.import os

3.from gtts import gTTS

4.import datetime

5.import warnings

6.import calendar

7.import random

8.import Wikipedia

- *Ignore any kind of warning messages:*
warnings.filterwarnings('ignore')

- *Record audio and return it as a string:*

```
def recordAudio():  
  
    r=sr.Recognizer()  
    with sr.Microphone() as source:  
        print('say something...!')  
        audio= r.listen(source)  
  
    data = ''  
    try:  
        data = r.recognize_google(audio)
```

```

print('You said: '+data)
except sr.UnknownValueError:
print('Google Speech Recognition could not understand audio,
unknown error')
except sr.RequestError as e:
print('request results from Google Speech Recognition Service
error'+ e)
return data

```

- *Get the response of the Virtual Master:*

```

def assistantResponse(text):

print(text)

#convert text to speech
myobj = gTTS(text= text, lang= 'en', slow=False)

#save converted audio
myobj.save('assistant_response.mp3')

#play the converted file
os.system('start assistant_response.mp3')

```

- *A function wake word:*

```

def wakeword(text):
    WAKE_WORDS=['ok computer', 'okay lappy']

    text = text.lower()
    for phrase in WAKE_WORDS:
    if phrase in text:
    return True

    return False

```

- *Give current date:*

```

def getDate():

    now = datetime.datetime.now()
    my_date = datetime.datetime.today()
    weekday = calendar.day_name[my_date.weekday()]
    monthnum =now.month
    daynum = now.day

    month_name=['January', 'February', 'March', 'April', 'May', 'June', '
July', 'August', 'September', 'October', 'November', 'December']
    originalnum =
    ['1st', '2nd', '3rd', '4th', '5th', '6th', '7th', '8th', '9th', '10th', '
11th', '12th', '13th', '14th', '15th', '16th'
, '17th', '18th', '19th', '20th', '21st', '22nd', '23rd', '24th', '25th'
, '26th', '27th', '28th', '29th', '30th', '31st']
    return 'Today is '+weekday+' '+ month_name[monthnum-1]+'
'+originalnum[daynum-1]+'.'

```

- *Get back a random greeting:*

```

def greeting(text):
    greeting_input = ['hii', 'hello', 'hey']
    greeting_response = ['hii', 'hello', 'hey there']
    for word in text.split():
    if word.lower() in greeting_input:

```

```
return random.choice(greeting_response) + '.'

return ''
```

- *Get details about a famous person:*

```
def getPerson(text):
    wordlist = text.split()

    for i in range(0, len(wordlist)):
        if i+3 <= len(wordlist)-1 and wordlist[i].lower() == 'who' and
            wordlist[i+1].lower() == 'is':
            return wordlist[i+2] + ' ' + wordlist[i+3]
```

```
while True:
```

```
    text = recordAudio()
    response = ''
```

- *Check for response if wake word is spoken:*

```
if(wakeword(text) == True):
```

```
    response= response + greeting(text)
```

```
#Virtual Master will response if 'date' word is spoken.
```

```
if('date' in text):
    get_date = getDate()
    response = response + ' ' +get_date
```

```
#Virtual Master will response if 'time' word is spoken.
```

```
if('time' in text):
    now=datetime.datetime.now()
    meridian =''
```

```
if now.hour>=12:
    meridian = 'p.m'
hour = now.hour - 12
```

```
else:
    meridian= 'a.m'
hour=now.hour
```

```
if now.minute<10:
    minute= '0'+str(now.minute)
```

```
else:
    minute= str(now.minute)
```

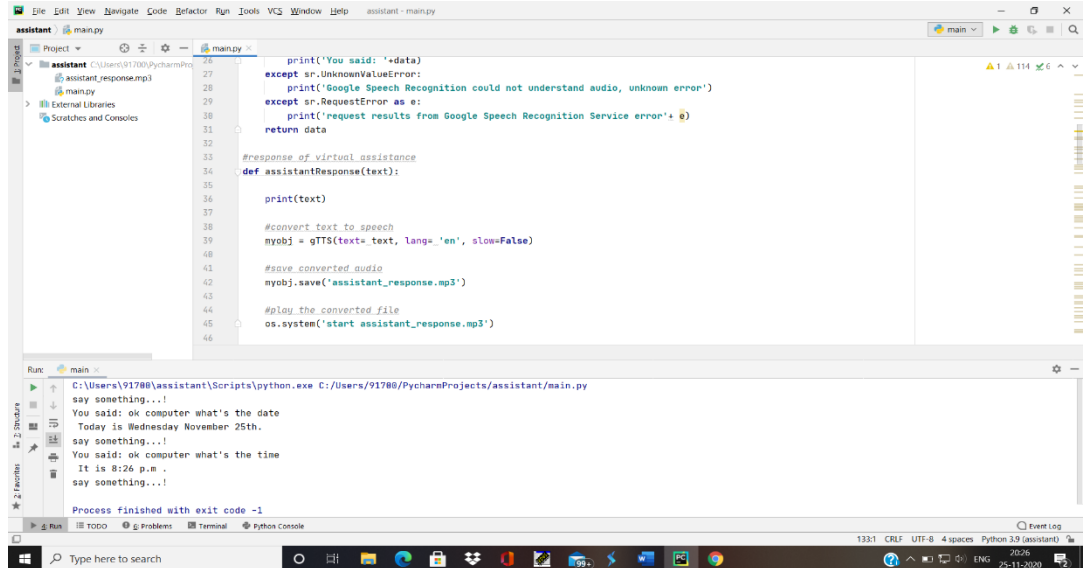
```
    response = response + ' ' + 'It is ' +str(hour) + ':' +
minute+ ' ' +meridian+ ' .'
```

```
#Virtual Master will response if the word 'who is' is spoken.
```

```
if('who is' in text):
    person=getPerson(text)
    wiki = wikipedia.summary(person, sentences=2)
    response = response + ' ' + wiki
```

assistantResponse (response)

- After this, save the program and run it.
- Pictorial Demonstration of Virtual Master:



The screenshot shows a Python IDE with a project named 'assistant'. The main.py file contains the following code:

```
26 print('You said: '-data)
27 except sr.UnknownValueError:
28     print('Google Speech Recognition could not understand audio, unknown error')
29 except sr.RequestError as e:
30     print('request results from Google Speech Recognition Service error'+ e)
31 return data
32
33 #response of virtual assistance
34 def assistantResponse(text):
35
36     print(text)
37
38     #convert text to speech
39     myobj = gTTS(text= text, lang= 'en', slow=False)
40
41
42     #save converted audio
43     myobj.save('assistant_response.mp3')
44
45     #play the converted file
46     os.system("start assistant_response.mp3")
```

The Run console shows the following output:

```
C:\Users\91788\assistant\Scripts\python.exe C:/Users/91788/PycharmProjects/assistant/main.py
say something...!
You said: ok computer what's the date
Today is Wednesday November 25th.
say something...!
You said: ok computer what's the time
It is 8:26 p.m .
say something...!
Process finished with exit code -1
```

APPLICATIONS

Virtual Master can be integrated into many types of platforms. The list of some platforms where this Virtual Master are used and can be used are as follows: -

- In appliances, cars and wearable technologies.
- In devices like smart speakers such as Amazon Echo, Google Home, etc.
- Built into a mobile operating system (OS) as Apple's Siri, in desktop OS Microsoft's Cortana and in smartphones independent of OS as Samsung's Bixby.
- Used for instant messaging apps like Facebook.
- Within mobile apps such as Dom from Domino's pizza.

CONCLUSION

In this paper, main idea of Virtual Master or Virtual Assistant and modules required for this are described. Then the procedure, how to make this Virtual Master and its application that providing opportunities in various domains are discussed. Thus, efforts are continuously made to use the benefits of this technology as far as possible.

REFERENCES

- www.geeksforgeeks.com
- www.wikipedia.com
- www.outsourceworkers.com

Copyright protected @ ENGPAPER.COM and AUTHORS



<https://www.engpaper.com>