Project Report
On
Python

PERSONAL ASSISTANT
(J.A.R.V.I.S)

Under the guidance of:
Priti Sardar
from
Indian Cyber Security Solutions

Submitted By-
Aakash Das
Roll No: 1705841
Branch: CSE
Semester: 5th
Kalinga University of Industrial Technology

KALINGA INSTITUTE OF
INDUSTRIAL TECHNOLOGY (KIIT)

Deemed to be University U/S 3 of UGC Act, 1956

Submitted Date:08/06/2019

❖ Table of Contents:

❖ Project Management Plan:

The management plan defines the purpose of the project and its organization.

Overview the project :

It specify, Speech recognition can by done using the Python Speech Recognition module. I make use of Google Speech API because of it's great sound capture quality.

Specifications:

1. Recent Time Speaking
2. Question - Answer Speaking
3. Search Wikipedia with speaking - printing
4. Wi-Fi Checking(ON/OFF)
5. Open Any Website
6. Talking Shutdown
7. Open Any Specific Website by OS
8. Open and Play Movies or Songs From File Directory
9. Open Any Application by OS
10. Email Sending by Command

❖ Project scope and objectives:

I ran into difficulties defining the boundaries of what would be included in the project and what would not. This is because of the open-ended nature of the project. Many different ideas and pieces of functionality had potential to be incorporated into the application. I wanted to be as open as possible when considering features, so it became difficult to say no to different parts of the system. This resulted in pieces of the system that were not developed as completely as they could have been. Future projects should have a clearly defined scope that teams can commit to. The team believes that since the scope was so large, there was no chance of completing everything in the project, which hurt the overall motivation on the project.
Objectives goes to next generation of project by python(text to speech)

❖ Assignment of Roles and Responsibilities:

This .is a personal project of python. I tried my best to do. This project works like a J.A.R.V.I.S or a personal assistant.

A. It works 10 different things in one project like;

B. Recent Time Speaking

C. Question - Answer Speaking

D. Search Wikipedia with speaking - printing

E. Wi-Fi Checking(ON/OFF)

F. Open Any Website

G. Talking Shutdown

H. Open Any Specific Website by OS

I. Open and Play Movies or Songs From File Directory

J. Open Any Application by OS

K. Email Sending by Command

Main Motive: **"10 Specifications = 1 PROJECT"**

❖ Project Schedule:

Each phase of project development will be given a timeline.

This project was completed around in 1 Month.
I implemented each specification in various time of this project and sorted out a shortest way for implementing this project.

❖ Requirements Specifications:

This section will consist of a requirements document outlining the system

- **Installation:-**

a) import pyttsx3:

Pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3.

❖ **pip install pyttsx3**

b) import speech_recognition as sr:

SpeechRecognition 3.8.1 is the library for performing speech recognition, with support for several engines and APIs, online and offline.

❖ **pip install SpeechRecognition**

c) import datetime:

This package provides a DateTime data type, as known from Zope. Unless you need to communicate with Zope APIs, you're probably better off using Python's built-in datetime module.

❖ **pip install DateTime**

d) import wikipedia:

Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia.

❖ **pip install wikipedia**

e) import webbrowser:

By default, It saved in Python Package

f) import os:

By default, It saved in Python Package(os-operating system)

g) import smtplib:

For sending gmail by voice use smtplib

❖ Analysis- Design:

This section will document my analysis-design model.

- Import Packages:

```
import pyttsx3
import speech_recognition as sr
import datetime
import wikipedia
import webbrowser
import os
import smtplib

INFO = '''
        +=======================================+
        |.....JARVIS VIRTUAL INTELLIGENCE.......|
        +---------------------------------------+
        |#Builder: AAKASH DAS                   |
        |#Date: 01/06/2019                      |
        +---------------------------------------+
        '''
print(INFO)
```

- Output:

```
+=======================================+
|.....JARVIS VIRTUAL INTELLIGENCE.......|
+---------------------------------------+
|#Builder: AAKASH DAS                   |
|#Date: 01/06/2019                      |
+---------------------------------------+
```

- Use Of ‘pyttsx3’:

```
engine = pyttsx3.init()            # object creation

""" RATE"""
rate = engine.getProperty('rate')   # getting details of current speaking rate
print (rate)                        # printing current voice rate
engine.setProperty('rate', 125)     # setting up new voice rate


"""VOLUME"""
volume = engine.getProperty('volume')   # getting to know current volume level (min=0 and max=1)
print (volume)                          # printing current volume level
engine.setProperty('volume',1.0)        # setting up volume level  between 0 and 1

"""VOICE"""
voices = engine.getProperty('voices')       # getting details of current voice
engine.setProperty('voice', voices[0].id)   # changing index, changes voices. o for male
#engine.setProperty('voice', voices[1].id)  # changing index, changes voices. 1 for female

def speak(audio):
    engine.say(audio)
    engine.runAndWait()
```

- WishMe At First Of Program:

```
def WishMe():
    hour=int(datetime.datetime.now().hour)
    if hour>=0 and hour<12:
        speak("Good Morning Sir!")
    elif hour>=12 and hour<18:
        speak("Good Afternoon Sir!")
    else:
        speak("Good Evening Sir!")
speak("Sir Please Tell Me How may I help You..")
```

- Command Taking:

```
def takeCommand():
    r = sr.Recognizer()
    with sr.Microphone() as source:
        audio = r.adjust_for_ambient_noise(source)
        print("Listening...")
        r.pause_threshold=1
        audio = r.listen(source)
                                                    # interprete audio (Google Speech Recognition)
    try:
        print("Recognizing...")
        query=r.recognize_google(audio)
        print(f"User said:{query}\n")
    except exception as e:
        print(e)
        print("Say that again please...")
        return "None"
    return query
```

- Question - Answer Speaking:

```
  elif ('goodbye') in query:
      ans=['Goodbye Sir', 'Jarvis powering off in 3, 2, 1, 0']
      speak(ans)
  elif ('hello') in query or ('hi') in query:
      ans=['Wellcome to Jarvis virtual intelligence project. At your service sir.']
      speak(ans)

  elif ('thanks') in query or ('thank you') in query:
      ans=['You are wellcome', 'no problem']
      speak(ans)

  elif query == ('jarvis'):
      ans=['Yes Sir?', 'What can I doo for you sir?']
      speak(ans)

  elif  ('how are you') in query or ('and you') in query or ('are you okay') in query:
      ans=['Fine thank you']
      speak(ans)

  elif  ('*') in query:
      ans=['Be polite please']
      speak(ans)
```

- Search Wikipedia with speaking - printing:

```
elif 'wikipedia' in query:
    speak('Searching wikipedia...')
    query=query.replace("Wikipedia"," ")
    results=wikipedia.summary(query,sentences=2)
    speak('According to wikipedia')
    print(results)
    speak(results)
```

- Open Stream App:

```
elif 'open stream' in query:
    codePath='D:\stream\Steam.exe'
    os.startfile(codePath)
```

- Wi-Fi Connecting:

```
elif ('wi-fi') in query:
    REMOTE_SERVER = "www.google.com"
    ans=['We are connected']
    speak(ans)
```

- Open Any Website:

```
elif ('.com') in query:
    rand = ['Opening' + query]
    Chrome = ("C:/Program Files (x86)/Google/Chrome/Application/chrome.exe %s")
    speak(rand)
    webbrowser.get(Chrome).open('http://www.'+ query)
    print ('')
```

- Open Sleep Mode:

```
elif ('sleep mode') in query:
    ans=['good night']
    speak(ans)
    os.system('rundll32.exe powrprof.dll,SetSuspendState 0,1,0')
```

- Play Movies from Directory:

```
elif 'play movies' in query:
    movie_dir='D:\IMPdwnld\Desktop\movie'
    movie=os.listdir(movie_dir)
    print (movie)
    os.startfile(os.path.join(movie_dir,movie[0]))
```

- Recent Time Speaking:

```
if 'time' in query:
    strTime=datetime.datetime.now().strftime("%H:%M:%S")
    speak(f"Sir, the time is {strTime}")
```

- Taking Command in Lower-Case:

```
if __name__=="__main__":
    WishMe()
    #while True:                # for many command
    if 1:                       # for one command
        query=takeCommand().lower()
```

- Send Email:

```
elif 'email' in query:
    try:
        speak("What should I say: ")
        content=takeCommand()
        to='aakshdas276@gmail.com'
        sendEmail(to,content)
        speak("Sir,Email has been sent!")

    except Exception as e:
        print(e)
        speak('Sorry my friend !! I am not able to send this email')
```

- Email Server Setting:

```
def  sendEmail(to,content):
    server = smtplib.SMTP('smtp.gmail.com', 587)
    server.ehlo()
    server.starttls()
    server.login('your email.id','password')
    server.sendmail('email id',to,content)
    server.close()
```

❖ Implementation-Test Documentation:

In this part; I implemented and tested all specifications together.

- Time Speaking:

```
+========================================+
|.....JARVIS VIRTUAL INTELLIGENCE.......|
+----------------------------------------+
|#Builder: AAKASH DAS                    |
|#Date: 01/06/2019                       |
+----------------------------------------+

200
1.0
Listening...
Recognizing...
User said:time
```

- Question-Answer Speaking:

```
+========================================+
|.....JARVIS VIRTUAL INTELLIGENCE.......|
+----------------------------------------+
|#Builder: AAKASH DAS                    |
|#Date: 01/06/2019                       |
+----------------------------------------+

125
1.0
Listening...
Recognizing...
User said:goodbye

Listening...
Recognizing...
User said:hello

Listening...
Recognizing...
User said:thanks
```

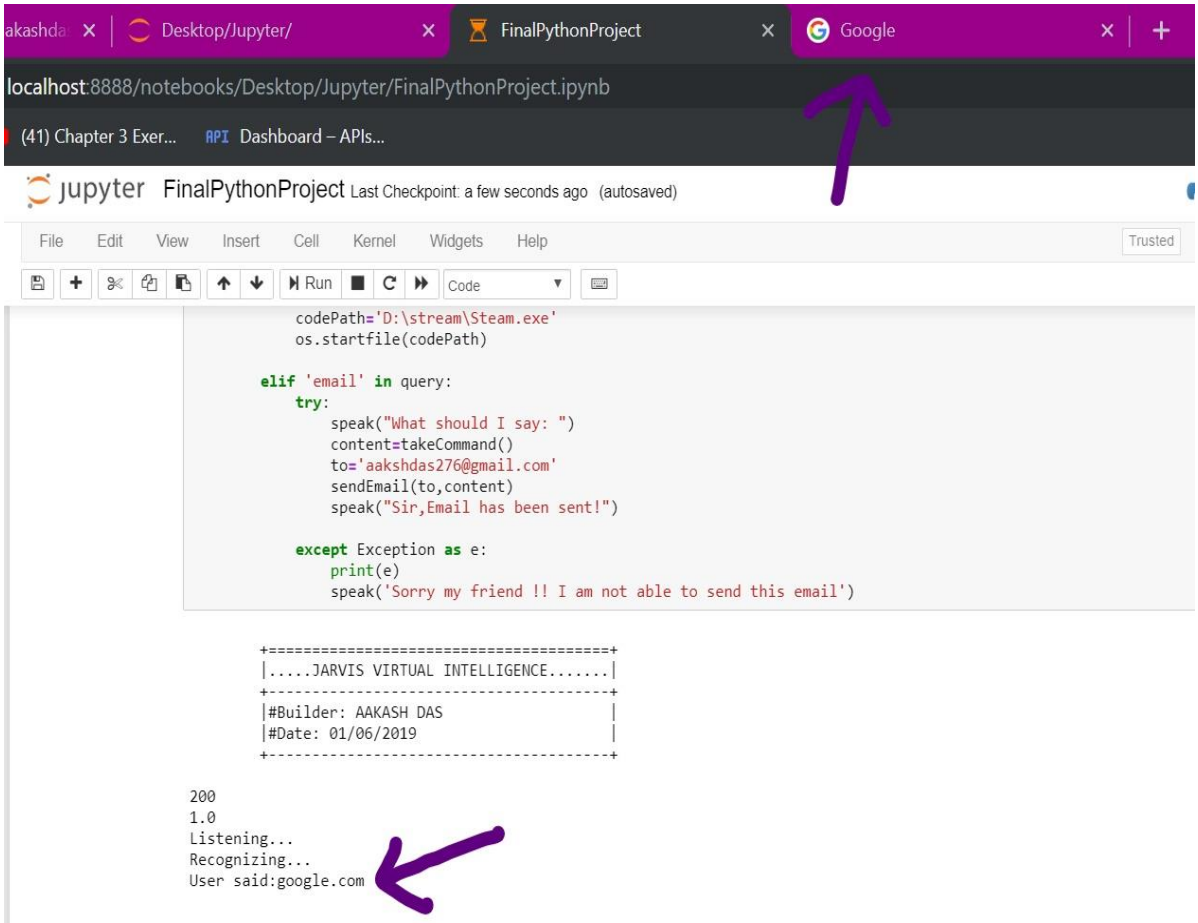
- Wikipedia Testing:

```
+====================================+
|.....JARVIS VIRTUAL INTELLIGENCE.....|
+------------------------------------+
|#Builder: AAKASH DAS                |
|#Date: 01/06/2019                   |
+------------------------------------+

200
1.0
Listening...
Recognizing...
User said:Shahrukh Khan according Wikipedia

Shah Rukh Khan (born Shahrukh Khan; 2 November 1965), also known by the initialism SRK, is an Indian actor, film producer, a
nd television personality. Referred to in the media as the "Badshah of Bollywood", "King of Bollywood" and "King Khan", he h
as appeared in more than 80 Bollywood films, and earned numerous accolades, including 14 Filmfare Awards.
```

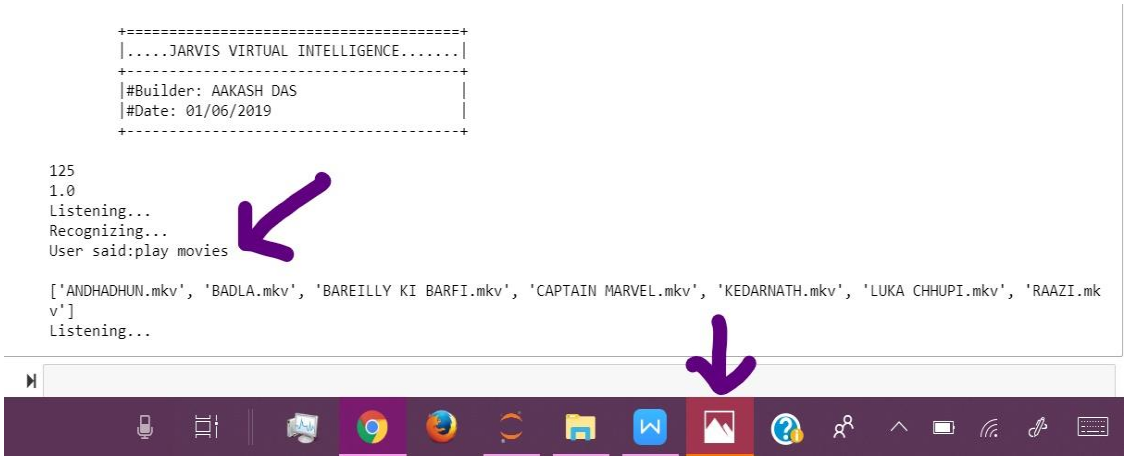
- Wi-Fi Testing:

```
+=======================================+
|.....JARVIS VIRTUAL INTELLIGENCE.......|
+---------------------------------------+
|#Builder: AAKASH DAS                   |
|#Date: 01/06/2019                      |
+---------------------------------------+

125
1.0
Listening...
Recognizing...
User said:Wi-Fi
```

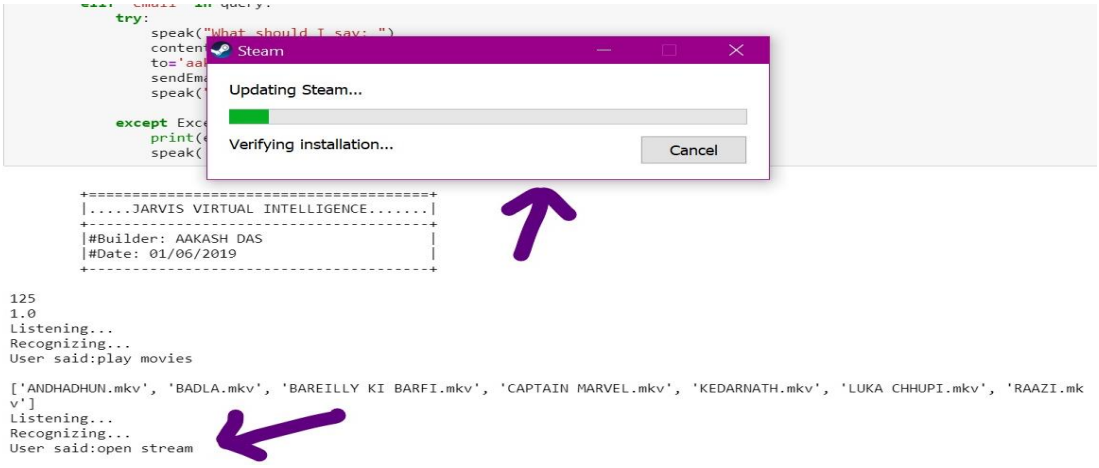
- Open Any WebSite Testing:



- Play Movies From Directory:



- Open Stream(App) by Command Testing:



- Send Email Testing:

❖ References:

I completed this project under the guidance of: **Priti Sardar** Ma'am *from* **Indian Cyber Security Solutions** and also helped from Internet specially **Git Hub** website.

❖ Progress reports:

❖ Time Logs: