Project Practicum

TED-The Voice Assistant

Done by:

Krishna Teja J

Ravalish Nuguri

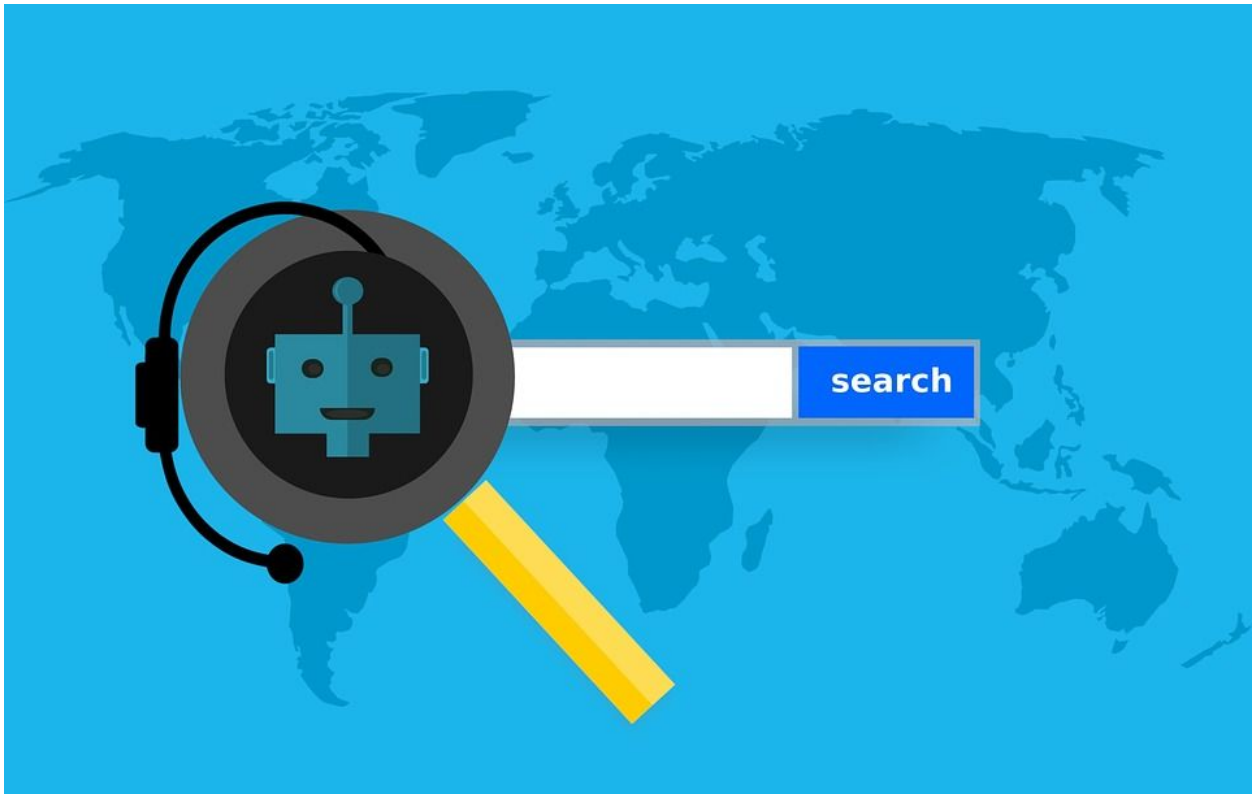
Vaishnavi Bandi

Approved by:

M.Manasa

Assistant Mentor, MSIT

IIIT Hyderabad



Introduction

The advancement in speech recognition and voice technology led to the birth of Apple's Siri, Google Now, Amazon Alexa and Microsoft's Cortana. These devices are off-late taking

over many households, thereby aiding the common man in their daily tasks. The voice assistant we are trying to build is an honest attempt to replicate that technology in its basic and minimalistic form. The voice assistant we are trying to build, TED, achieves a few basic functionalities that will be listed later on in this report.The scope of voice assistance technology and speech recognition go far beyond this report and are truly vast and endless.

Abstract:

A voice assistant or Intelligent Personal Assistant is a software that performs tasks by taking voice input from the user utilising complex speech recognition techniques and respond via speech. The commands given by the user can be anything from being very obvious to being extremely abstract. The sole motivation for implementing this project was to understand speech recognition packages present in python and also how the google voice recognizer API analyses the voice input and converts it into text. After conversion to text we search for keywords in the tet and achieve the various functions that are listed in this report later on.
pyttsx(Python-Text-To-Speech) has been used to let the system communicate with the user .

Goal Of The Project:

This project aims at replicating some basic functions of the various voice assistants that are already present. The project aims to achieve the stated functions with utmost accuracy.

Functions:

The project aims at achieving the following :

1.Open any website the user asks for
2.Give us the weather conditions at current location and elsewhere
3. Crack a random joke
4.Send a mail
5.Greet the user according to the time of the day
6.Give the user any information he asks for.
7.Play a song of user's choice.

Coding and Implementation:

Start by importing all the required libraries :

```
import speech_recognition as sr
import os
import sys
import re
import yagmail
import webbrowser
import smtplib
import youtube_dl
import requests
import subprocess
from pyowm import OWM
from urllib.request import urlopen
import json
import pyttsx3
from bs4 import BeautifulSoup as soup
import wikipedia
import pafy
from time import strftime
```

For our voice-assistant to perform all the above-discussed features, we have to code the logic of each of them in one method.So our first step is to create the method which will interpret user voice response:

```
def myCommand():
    "listens for commands"
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print('Say something...')
        r.pause_threshold = 1
        r.adjust_for_ambient_noise(source, duration=1)
        audio = r.listen(source)
    try:
        command = r.recognize_google(audio).lower()
        print('You said: ' + command + '\n')
    except sr.UnknownValueError:
        print('....')
        command = myCommand();
    return command
```

Next, create a method that will convert text to speech.

```
def tedResponse(audio):
    "speaks audio passed as argument"
    print(audio)
    print()
    engine = pyttsx3.init()
    for line in audio.splitlines():
        engine.say(audio)
        engine.runAndWait()
```

Now create a loop to continue executing multiple commands. Inside the method assistant() passing user command(myCommand()) as parameters.

```
while True:
    assistant(myCommand())
```

Our next step is to create multiple if statements corresponding to each of the features. So let us see how to create these small modules inside if statement for each command.

1. Open any website in the browser.

You can open any website just be saying “open website.com” or “open website.org”.For example: “Please open facebook.com” or “Hey, can you open linkedin.com” like this you can ask Ted to open any website for you.

How it works : *If you have said the word* ***open*** *in your command then it will search for website name in the user command using re.search(). Next, it will append the website name to https://www. and using* ***webbrowser*** *module the complete URL gets opened in the browser.*

```
if 'open' in command.split(" "):
    reg_ex = re.search('open (.+)', command)
    if reg_ex:
        domain = reg_ex.group(1)
        print(domain)
        url = 'https://www.' + domain + '.com'
        webbrowser.open(url)
        tedResponse('The website you have requested has been opened for you .')
        tedResponse('Is there anything else you want me to do?')
        assistant(myCommand)
    else:
        pass
```

2. Send Email.

You can also ask your desktop assistant to send an email.

How it works : *If you have said the word* ***email*** *in your command then the bot will ask for recipient, If my response is rajat, the bot will use python' s smtplib library. The* ***smtplib module*** *defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP Sending mail is done with Python's smtplib using an SMTP server. First it will initiate gmail SMTP using* ***smtplib.SMTP()****, then identify the server using* ***ehlo()*** *function, then encrypting the session using* ***starttls(),*** *then login to your mailbox using* ***login()****, and sending the message using* ***sendmail().***

```
if 'email' in command.split(" "):
        tedResponse('What should I say to him?')
        content = myCommand()
        mail = smtplib.SMTP('smtp.gmail.com', 587)
        mail.ehlo()
        mail.starttls()
        mail.login('your_email_address', 'your_password')
        mail.sendmail('sender_email', 'receiver_email', content)
        mail.close()
        tedResponse('Email has been sent successfuly. You can check your inbox.')
        tedResponse('Is there anything else you want me to do?')
```

3. Tells you the current weather and temperature of almost any city.

Ted can also tell you the weather, maximum and minimum temperature of any city around the world. The user just needs to say something like "what is the current weather in London" or "tell me the current weather in Delhi".

How it works: *If you have said the phrase current weather in your command then it will search for city name using re.search(). I have used pythons pyowm library to get the weather of any city. get_status() will tell you about the weather condition like haze, cloudy, rainy etc and get_temperature() will tell you about the max and min temperature of the city.*

```
if 'weather' in command.split(" "):
    reg_ex = re.search('weather (.*)', command)
    if reg_ex:
        l = command.split(" ")
        place = l[-1]
        city = place
        owm = OWM(API_key='ed7aca10668bd47a09e666615cecc68c')
        obs = owm.weather_at_place(city)
        w = obs.get_weather()
        k = w.get_status()
        x = w.get_temperature(unit='celsius')
        tedResponse('Current weather in %s is %s. The maximum temperature is %0.2f and the minim
        tedResponse('Is there anything else you want me to do?')
```

4. Tells you the current time.

"Ted can you tell me the current time ?" or "what is the time now ?" and Ted will tell you the current time of your timezone.

How it works : *By importing the datetime module from python .*

```
if 'time' in command.split(" "):
    import datetime
    now = datetime.datetime.now()
    tedResponse('Current time is %d hours %d minutes' % (now.hour, now.minute))
```

5. Greetings/ leave

Say " hello/yo/whats up/hi Ted" to greet your voice assistant or when you want the program to terminate say something like "shutdown Ted" or "Ted please shutdown" etc.

How it works : *If you have said the word* ***hello*** *in your command, then depending on the time of the day, the bot will greet the user. If the time is more than 12 noon, the bot will respond "Hello Sir. Good afternoon", likewise if the time is more than 6 ck pm, the bot will respond "Hello Sir. Good evening. And when you give command as shutdown,* ***sys.exit()*** *will be called to terminate the program.*

```
if 'hello' or 'hi' or 'yo' or 'whats up?' in command.split(" "):
    day_time = int(strftime('%H'))
    if day_time < 12:
        tedResponse('Hello !!. Good morning')
    elif 12 <= day_time < 18:
        tedResponse('Hello !!. Good afternoon')
    else:
        tedResponse('Hello !!. Good evening')
```

6. Play you a song

This feature allows your voice bot to play your desired song. The user will say “Ted play me a song”, the bot will ask “What song shall I play Sir?”. Just say the name of the song and Ted will play the song from youtube

How it works :*If you have said the phrase* ***play me a song*** *in your command, then it will ask you what video song to play, it then looks it up on youtube and then plays the song.*

```
if 'song' in command.split(" "):
    mysong = myCommand()
    if mysong:
        flag = 0
        url = "https://www.youtube.com/results?search_query=" + mysong.replace(' ', '+')
        webbrowser.open(url)
```

7. Tells you the latest news feeds.

Ted can also tell you the latest news update. The user just has to say “Ted what are the top news for today ?” or “tell me the news for today”.

How it works : *If you have said the phrase* ***news for today*** *in your command then it will scrape data using Beautiful Soup from Google News RSS() and read it for you. For convenience I have set number of news limit to 8.*

```
if 'news' in command.split(" "):
    try:
        news_url="https://news.google.com/news/rss"
        Client=urlopen(news_url)
        xml_page=Client.read()
        Client.close()
        soup_page=soup(xml_page,"xml")
        news_list=soup_page.findAll("item")
        for news in news_list[:8]:
            tedResponse(news.title.text.encode('utf-8'))
            break
        tedResponse('Is there anything else you want me to do?')
        assistant(myCommand)
    except Exception as e:
        tedResponse(e)
```

8. Tells you about almost anything you ask.

Your bot can fetch details of almost anything you ask her. Like “Ted tell me about Google” or “Please tell me about Supercomputers” or “please tell me about the Internet”. So as you can see you can ask about almost anything.

How it works : *If you have said the phrase* ***tell me about*** *in your command then it will search for the keyword in the user command using re.search(). Using pythons wikipedia library it will search for that topic and extract first 500 characters(if you dont specify the limit the bot will read the whole page for you). Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia.*

```
if 'about' in command.split(" "):
    reg_ex = re.search('about (.*)', command)
    try:
        if reg_ex:
            topic = reg_ex.group(1)
            ny = wikipedia.page(topic)
            ny = ny.content[:500].encode('utf-8')
            tedResponse(ny)
            sys.exit()
    except Exception as e:
            tedResponse(e)
```

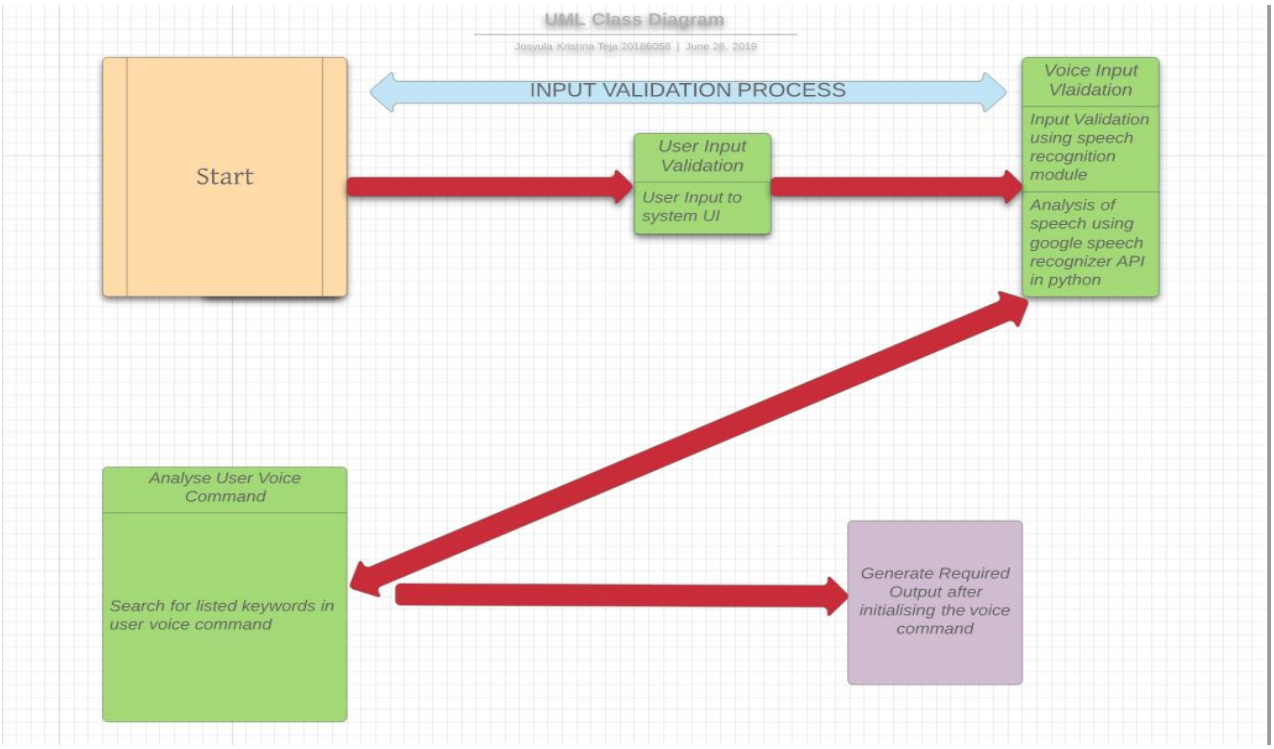
9.Cracks a random joke

Ted can crack a random joke by retrieving the required info as json format. For this the json package has been imported.

```
if 'joke' in command.split(" "):
    res = requests.get('https://icanhazdadjoke.com/',headers={"Accept":"application/json"})
    if res.status_code == requests.codes.ok:
        tedResponse(str(res.json()['joke']))
    else:
        tedResponse('oops!I ran out of jokes')
```

Project UML/Flowchart :

The flow of control/visual understanding of the project can be better understood with an example of the below flowchart:



Project Screenshots:

Attached below are the screenshots of the code that is utilised to run the voice assistant.

```
def myCommand():
    "listens for commands"
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print('Say something...')
        r.pause_threshold = 1
        r.adjust_for_ambient_noise(source, duration=1)
        audio = r.listen(source)
    try:
        command = r.recognize_google(audio).lower()
        print('You said: ' + command + '\n')
    except sr.UnknownValueError:
        print('....')
        command = myCommand();
    return command
```

```
if 'news' in command.split(" "):
    try:
        news_url="https://news.google.com/news/rss"
        Client=urlopen(news_url)
        xml_page=Client.read()
        Client.close()
        soup_page=soup(xml_page,"xml")
        news_list=soup_page.findAll("item")
        for news in news_list[:8]:
            tedResponse(news.title.text.encode('utf-8'))
            break
        tedResponse('Is there anything else you want me to do?')
        assistant(myCommand)
    except Exception as e:
        tedResponse(e)
```

```
if 'open' in command.split(" "):
    reg_ex = re.search('open (.+)', command)
    if reg_ex:
        domain = reg_ex.group(1)
        print(domain)
        url = 'https://www.' + domain + '.com'
        webbrowser.open(url)
        tedResponse('The website you have requested has been opened for you .')
        tedResponse('Is there anything else you want me to do?')
        assistant(myCommand)
    else:
        pass
```

```
if 'song' in command.split(" "):
    mysong = myCommand()
    if mysong:
        flag = 0
        url = "https://www.youtube.com/results?search_query=" + mysong.replace(' ', '+')
        webbrowser.open(url)
```

```
def tedResponse(audio):
    "speaks audio passed as argument"
    print(audio)
    print()
    engine = pyttsx3.init()
    for line in audio.splitlines():
        engine.say(audio)
        engine.runAndWait()
```

```
if 'time' in command.split(" "):
    import datetime
    now = datetime.datetime.now()
    tedResponse('Current time is %d hours %d minutes' % (now.hour, now.minute))
```

```
if 'weather' in command.split(" "):
    reg_ex = re.search('weather (.*)', command)
    if reg_ex:
        l = command.split(" ")
        place = l[-1]
        city = place
        owm = OWM(API_key='ed7aca10668bd47a09e666615cecc68c')
        obs = owm.weather_at_place(city)
        w = obs.get_weather()
        k = w.get_status()
        x = w.get_temperature(unit='celsius')
        tedResponse('Current weather in %s is %s. The maximum temperature is %0.2f and the minim
        tedResponse('Is there anything else you want me to do?')
```

```
while True:
    assistant(myCommand())
```

Conclusion:

This project has taught the intricacies involved in voice assistance technology and also how to analyse voice input to achieve various functions. The advancement in speech recognition and NLP techniques has motivated us to use the google speech recognizer to analyse the voice input and convert it to text.

References:

The project has been developed with constant help and aid from stack overflow and also the python documentation of various modules and packages.