

**MICRO PROJECT**

**Academic year**

**2021-22**

**Voice Assistant**

**Program: Computer Engineering Program code:CO6I-B**

**Course: Programming with Python Course code: 22616**



**MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION**

**Certificate**

This is to certify that Mr. /Ms. **Chirag Mangtani, Hiten Dusseja, Sanika Thakur** Roll No. **07-09** **of Sixth** Semester of Diploma in **Computer Engineering** of Institute**, VES POLYTECHNIC (Code: 0004)** has completed the **Micro Project** satisfactorily in Subject –**-Programming with Python (22616)** for the academic year **2021- 2022**. as prescribed in the curriculum.

Place: Mumbai Enrollment No: 1900040145,46,47

Date: Exam Seat No: ………………………….

**Subject Teacher Head of the Department Principal**

**INDEX**

**Academic year: 2021-22 Name of the faculty: Mrs.ShubhangiChintawar**

**Program code: CO6I Course & course code: PWP (22616)**

**Name of the candidate: Chirag Mangtani, Hiten Dusseja, Sanika Thakur**

**Roll No. 07 – 09 Enrollment No. 1900040145,146,147**

|  |  |  |
| --- | --- | --- |
| **Sr.No** | **Content** | **Page No.** |
| 1 | Project Proposal | 01 |
| 2 | Action plan | 02 |
| 3 | Resource required | 02 |
| 4 | Introduction | 03 |
| 5 | Actual Procedure | 04 |
| 6 | Output |  |
| 7 | Skill developed |  |
| 8 | Evaluation sheet |  |

**Annexure – I**

**Micro-Project Proposal**

**Voice Assistant**

**1.0 Aim/Benefits of the Micro-Project**

**This Micro-Project aims at**

* To create a program that takes input from the user’s mic and outputs via user’s mic
* To build a voice assistant that opens app and websites according to user’s demand
* To build a voice assistant that sends emails and messages according to user’s demand

**2.0 Course Outcomes Addressed**

* Display message on screen using python script on IDE [ ✓ ]
* Develop python program to demonstrate use of Operators. [ ✓ ]
* Perform operations on data structure in Python [ ✓ ]
* Develop function for given problem [ ✓ ]
* Design classes for given problem [ ]
* Handle exception [ ✓ ]

**3.0 Proposed Methodology**

* Discussion about topic with guide and among group members
* Literature Survey
* Submission of project proposal
* Information collection
* Analysis of Data
* Compilation of the content
* Discussion about the code
* Representation
* Implementing the code
* Editing and revising the content

**4.0 Action Plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SR. No.** | **Details of activity** | **Planned Start date** | **Planned Finish date** | **Name of Responsible Team Members** |
| **1** | Finalization of project Title and Scope | **15/02/22** | **22/02/22** | **All** |
| **2** | Project Definition | **22/02/22** | **2/03/22** | **All** |
| **3** | Information Collection and Analysis of Data | **2/03/22** | **15/03/22** | **All** |
| **4** | Algorithm and Flowchart | **15/03/22** | **23/03/22** | **All** |
| **5** | Coding | **23/03/22** | **20/04/22** | **All** |
| **6** | Output | **20/04/22** | **26/04/22** | **All** |
| 7 | Report writing | **26/04/22** | **10/05/22** | **All** |
| 8 | Demonstration of project & final submission | **10/5/22** | **18/05/22** | **All** |

**5.0 Resources Required**

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Equipment Name with Broad Specification** | **Remark if any** |
| 1 | Desktop pc – Windows 7 |  |
| 2 | Editor used – Python IDLE, Software – Python 3.4.2 |  |

**Name of Team Members with Roll Nos.**

**1.** Chirag Mangtani - 07

**2.** HitenDusseja - 08

**3.** Sanika Thakur - 09

**Mrs.ShubhangiChintawar**

**Name and Signature of Course Teacher**

**Annexure – II**

**Micro-Project Report**

**Voice Assitant**

1. **Rationale**

A voice assistant that uses user’s voice to take commands and then processes and gives output according to that in audio as well as text. The main focus of the voice assistant is to perform basic functions like searching and playing videos that the user wants, open softwares, websites, etc. It also has functionalities like sending a whatsapp message using whatsapp web and sending gmails. It can search about people and stuff from the Wikipedia api and provide a summary of the person/place/thing to the user. In short, it is a text to speech and speech to text program that performs basic tasks.

1. **Aim/Benefits of the Micro-Project**

**This Micro-Project aims at**

* To create a program that takes input from the user’s mic and outputs via user’s mic
* To build a voice assistant that opens app and websites according to user’s demand
* To build a voice assistant that sends emails and messages according to user’s demand

**3.0 Course Outcomes Addressed**

* Display message on screen using python script on IDE [ ✓ ]
* Develop python program to demonstrate use of Operators. [ ✓ ]
* Perform operations on data structure in Python [ ✓ ]
* Develop function for given problem [ ✓ ]
* Design classes for given problem [ ]
* Handle exception [ ✓ ]

**4.0 Literature Review**

About python:

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small- and large-scale projects

About pyttsx3 module:

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.

We have used this to make our program speak whatever we provide via the user’s speaker.

About speech\_recognition module:

Library for performing speech recognition, with support for several engines and APIs, online and offline.

It supports multiple apis like Google Speech Recognition, Google cloud speech API, Wit.ai, Microsoft Bing Voice Recognition, etc. The one we have used for our project is Google speech recognition.

We have used this module to recognize user’s commands.

About pywhatkit module:

pywhatkit is a Python library for sending WhatsApp messages at a certain time, it has several other features too.

Following are some features of pywhatkit module:

* Send WhatsApp messages.
* Play a YouTube video.
* Perform a Google Search.
* Get information on a particular topic.

We have used this module to send whatsapp messages and play youtube videos.

About smtplib module:

The smtplib module defines an SMTP client session object that can be used to send mail to any internet machine with an SMTP or ESMTP listener daemon.

We have used this module to send emails

About pyautogui module:

PyAutoGUI is essentially a Python package that works across Windows, MacOS X and Linux which provides the ability to simulate mouse cursor moves and clicks as well as keyboard button presses.

We have used this module to send whatsapp messages typed by pywhatkit

About API:

API stands for “application programming interface.” An API is essentially a set of rules that dictate how two machines talk to each other. Some examples of API-based interactions include a cloud application communicating with a server, servers pinging each other, or applications interacting with an operating system.

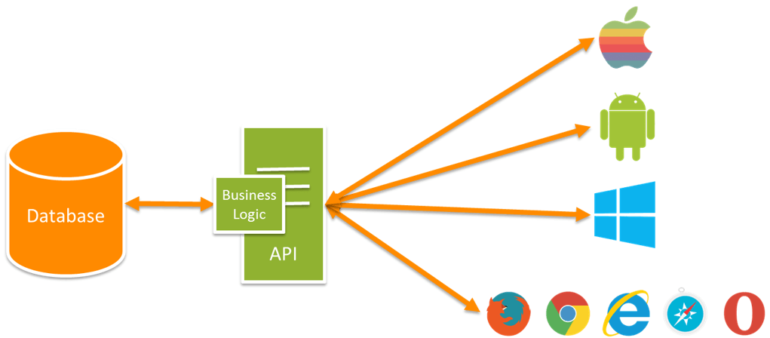


Fig 4.1: Working of an API

About modules in python:

In Python, Modules are simply files with the “. py” extension containing Python code that can be imported inside another Python Program. In simple terms, we can consider a module to be the same as a code library or a file that contains a set of functions that you want to include in your application

About functions in python:

A function is a block of reusable code that is used to perform a specific action. The advantages of using functions are: Reducing duplication of code. Decomposing complex problems into simpler pieces. Improving clarity of the code.

**5.0 Actual Procedure Followed.**

Algorithm:

Step 1: Start

Step 2: Import all the modules that are needed

Step 3: Greet the user

Step 4: Take the command from the user

Step 5: Check what does the command have and do the following accordingly:

Step 5.1: if command has 'who is' then search on wikipedia and return the summary

Step 5.2: else if command has 'open' in it, then open the specified software or website

Step 5.3: else if command has 'joke' in it, then fetch a joke and play it on user's speaker

Step 5.4: else if command has 'play' in it, then check from the following

Step 5.4 a: if command has 'play music' or 'play songs' in it then play a random song from the user's directory

Step 5.4 b: else open youtube and play the specified video/channel/artist

Step 5.5: elseif command has 'current time' in it, then tell the current time

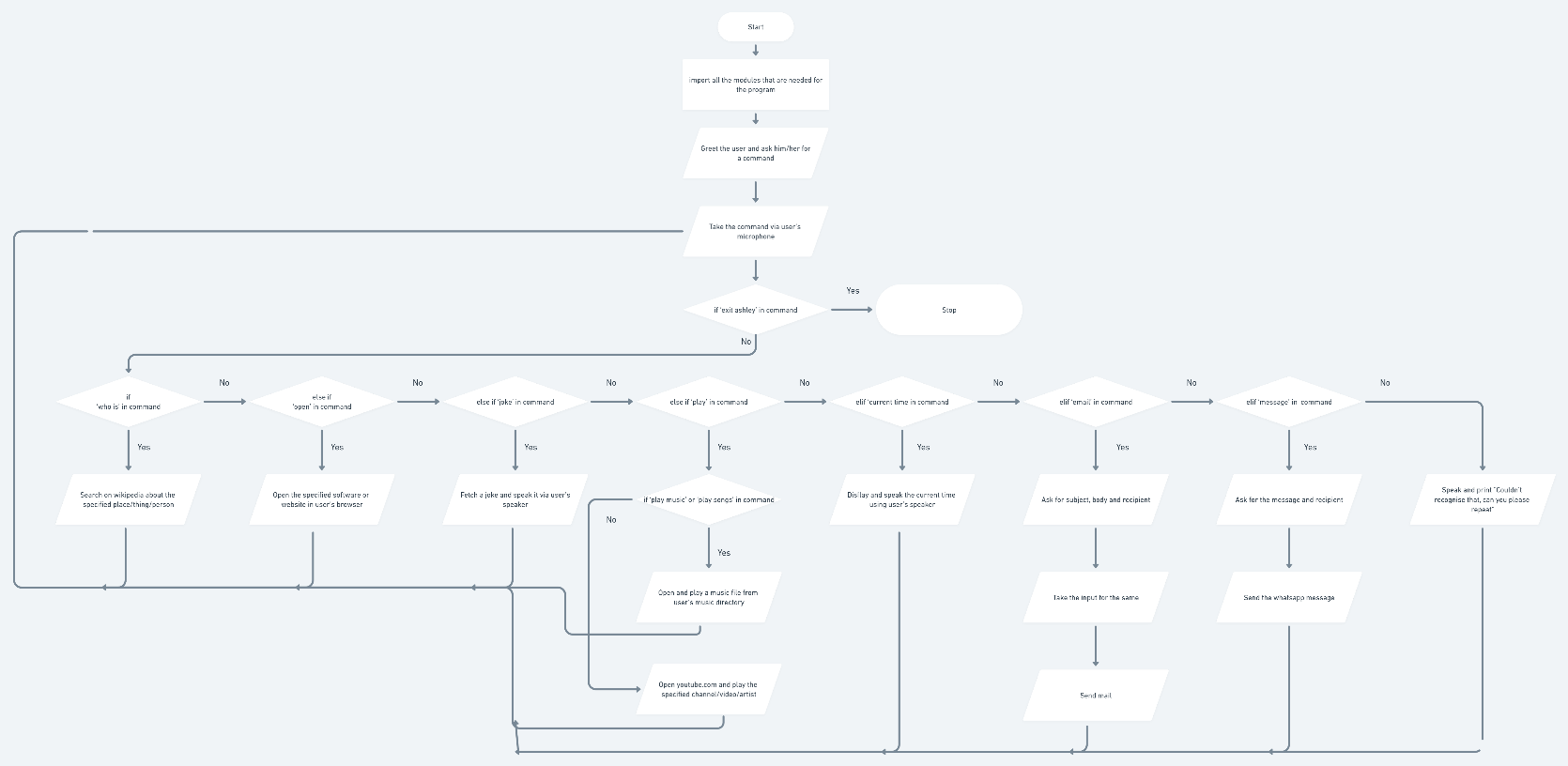
Step 5.6: else if command has 'email' in it, then ask for subject,body and recipient of the mail and send the mail accordingly

Step 5.7: else if command has 'message' in it, then ask for the body and recipient of the message and send a whatapp message

Step 5.8: else if not even single condition of the above are true, speak "Couldn't recognise that, can you please repeat"

Step 6: Stop if the user says exit ashley

Flowchart:



Code:

import pyttsx3

import speech\_recognition as sr

import wikipedia

import webbrowser

import os

import random

import datetime

import smtplib

import pyjokes

import pywhatkit

from email.message import EmailMessage

import time

import pyautogui

import keyboard as k

from datetime import datetime

mymail = "ankitdsouza15@gmail.com"

mypassword = "13318990@Hiten"

engine = pyttsx3.init('sapi5')

voices = engine.getProperty('voices')

engine.setProperty('voice', voices[1].id)

emaillist = {'personal': "ankitdsouza15@gmail.com",

'college': "co2019.hiten.dusseja@ves.ac.in",

'alternate': "xyzabc13318990@gmail.com",

'prank': "prenkmaster1331@gmail.com"}

message\_recipient\_dict = {'hello': "+919881010419"}

def speak(sentence):

engine.say(sentence)

engine.runAndWait()

def sendemail(to, content):

server = smtplib.SMTP('smtp.gmail.com', 590)

server.ehlo()

server.starttls()

# file = open("password.txt", 'r')

server.login(mymail, mypassword)

# server.sendmail('ankitdsouza15@gmail.com', to, content)

server.send\_message(content)

server.close()

speak(f"Sent an email to {to} ")

def sendmesage(phnnumber, message\_sent):

now = datetime.now()

current\_time = now.strftime("%H:%M")

hour = int(current\_time[:2])

minu = int(current\_time[3:5])

print("Current Time =", hour, " ", minu)

if(minu == 59 or minu == 60):

minu = 0

hour = hour + 1

else:

minu = minu+1

print("Time after 1 min =", hour, " ", minu)

pywhatkit.sendwhatmsg(phnnumber, message\_sent, hour, minu)

pyautogui.click(1050, 950)

time.sleep(1)

k.press\_and\_release('enter')

def greetme():

speak("Hello there This is Ashley,how can I help you?")

def takeCommand():

r = sr.Recognizer()

with sr.Microphone() as source:

print("Listening..")

r.pause\_threshold = 0.9

r.phrase\_threshold = 0.4

# r.adjust\_for\_ambient\_noise(source, duration=1)

audio = r.listen(source)

try:

print("Recognizing")

query = r.recognize\_google(audio)

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

query = query.lower()

if 'exit ashley' in query:

speak("Exiting Ashley")

exit()

else:

query = query.replace("ashley", "")

except Exception as e:

print("Couldn't recognize that, can you please repeat")

speak("Couldn't recognize that, can you please repeat")

return "none"

return query

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

greetme()

while True:

query = takeCommand().lower()

if 'who is' in query:

try:

print("Searching....wait")

query = query.replace("wikipedia", "")

query = query.replace("according to", "")

query = query.replace("who is", "")

query = query.replace("what is", "")

results = wikipedia.summary(query, sentences=1)

speak("So wikipedia says ")

print(results)

speak(results)

except Exception as e:

speak("Couldn't find that, sorry!")

print("Couldn't find that")

elif 'open browser' in query:

browserpath = "C:\\Program Files (x86)\\Google\\Chrome\\Application\\chrome.exe"

speak("opening chrome")

print("Opening chrome..")

os.startfile(browserpath)

elif 'open instagram' in query:

speak("opening instagram")

print("Opening instagram...")

webbrowser.open('instagram.com')

elif 'open youtube' in query:

speak("starting youtube")

print("Opening youtube...")

webbrowser.open('youtube.com')

elif 'open github' in query:

speak("opening github")

print("Opening github..")

webbrowser.open('github.com')

elif 'open google' in query:

speak("opening google")

print("Opening google...")

webbrowser.open('google.com')

elif 'joke' in query:

joke = pyjokes.get\_joke()

speak(joke)

print(joke)

elif 'play' in query:

if 'play music' in query:

print("Playing music...")

speak("Playing music")

music\_dir = "D:\\College docs\\SEM 6\\python\\Microproject\\music"

songslist = os.listdir(music\_dir)

songlistlen = len(songslist) - 1

r = random.randint(0, songlistlen)

os.startfile(os.path.join(music\_dir, songslist[r]))

elif 'play songs' in query:

print("Playing music...")

speak("playing music")

music\_dir = "D:\\College docs\\SEM 6\\python\\Microproject\\music"

songslist = os.listdir(music\_dir)

songlistlen = len(songslist) - 1

r = random.randint(0, songlistlen)

os.startfile(os.path.join(music\_dir, songslist[r]))

else:

query = query.replace("play", "")

speak(f"Playing {query}")

print(f"Playing {query}")

pywhatkit.playonyt(query)

elif 'current time' in query:

crnttime = datetime.datetime.now().strftime("%I:%M %p")

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

print(f"The current time is {crnttime}")

elif 'open torrent' in query:

tpath = "C:\\Users\\Komal\\AppData\\Roaming\\uTorrent\\uTorrent.exe"

speak("Opening torrent")

os.startfile(tpath)

elif 'hello' in query:

try:

speak("What should I put in the body of the mail?")

content = takeCommand()

to = "xyzabc13318990@gmail.com"

sendemail(to, content)

except Exception as e:

print(e)

elif 'send a mail' in query:

try:

content = EmailMessage()

# speak("What should be the subject of the mail?")

# to set the subject

query = 'none'

while query == 'none':

speak("What should I put in the subject of the mail?")

query = takeCommand()

content['Subject'] = query

# to set the body

query = 'none'

while query == 'none':

speak("What should I put in the body of the mail?")

query = takeCommand()

content.set\_content(query)

# to set the recipient

query = 'none'

content['From'] = "ankitdsouza15@gmail.com"

while query == 'none':

speak("Who should be the recipient?")

rec = takeCommand().lower()

query = rec

try:

to = emaillist[rec]

except Exception as e:

print(e)

query = 'none'

content['To'] = to

# to send the actual email

sendemail(to, content)

except Exception as e:

print(e)

elif 'search' in query:

query = query.replace("search", "")

query = query.replace("about", "")

query = query.replace("for", "")

speak(f"searching {query}")

pywhatkit.search(query)

elif 'message' in query:

message\_tobe\_sent = 'none'

recipient = 'none'

while(message\_tobe\_sent == 'none'):

speak("What should be the message?")

message\_tobe\_sent = takeCommand()

while(recipient == 'none'):

speak("Who should I send the message to?")

recipient\_key = takeCommand()

try:

recipient = message\_recipient\_dict[recipient\_key]

except Exception as e:

print(e)

recipient = 'none'

if(recipient == 'none'):

print("Can't find that person, please try again")

try:

sendmesage(recipient, message\_tobe\_sent)

except Exception as e:

print(e)

elif 'send an email' in query:

try:

content = EmailMessage()

# speak("What should be the subject of the mail?")

# to set the subject

query = 'none'

while query == 'none':

speak("What should I put in the subject of the mail?")

query = takeCommand()

content['Subject'] = query

# to set the body

query = 'none'

while query == 'none':

speak("What should I put in the body of the mail?")

query = takeCommand()

content.set\_content(query)

# to set the recipient

query = 'none'

content['From'] = "ankitdsouza15@gmail.com"

while query == 'none':

speak("Who should be the recipient?")

rec = takeCommand().lower()

query = rec

try:

to = emaillist[rec]

except Exception as e:

print(e)

query = 'none'

content['To'] = to

# to send the actual email

sendemail(to, content)

except Exception as e:

print(e)

else:

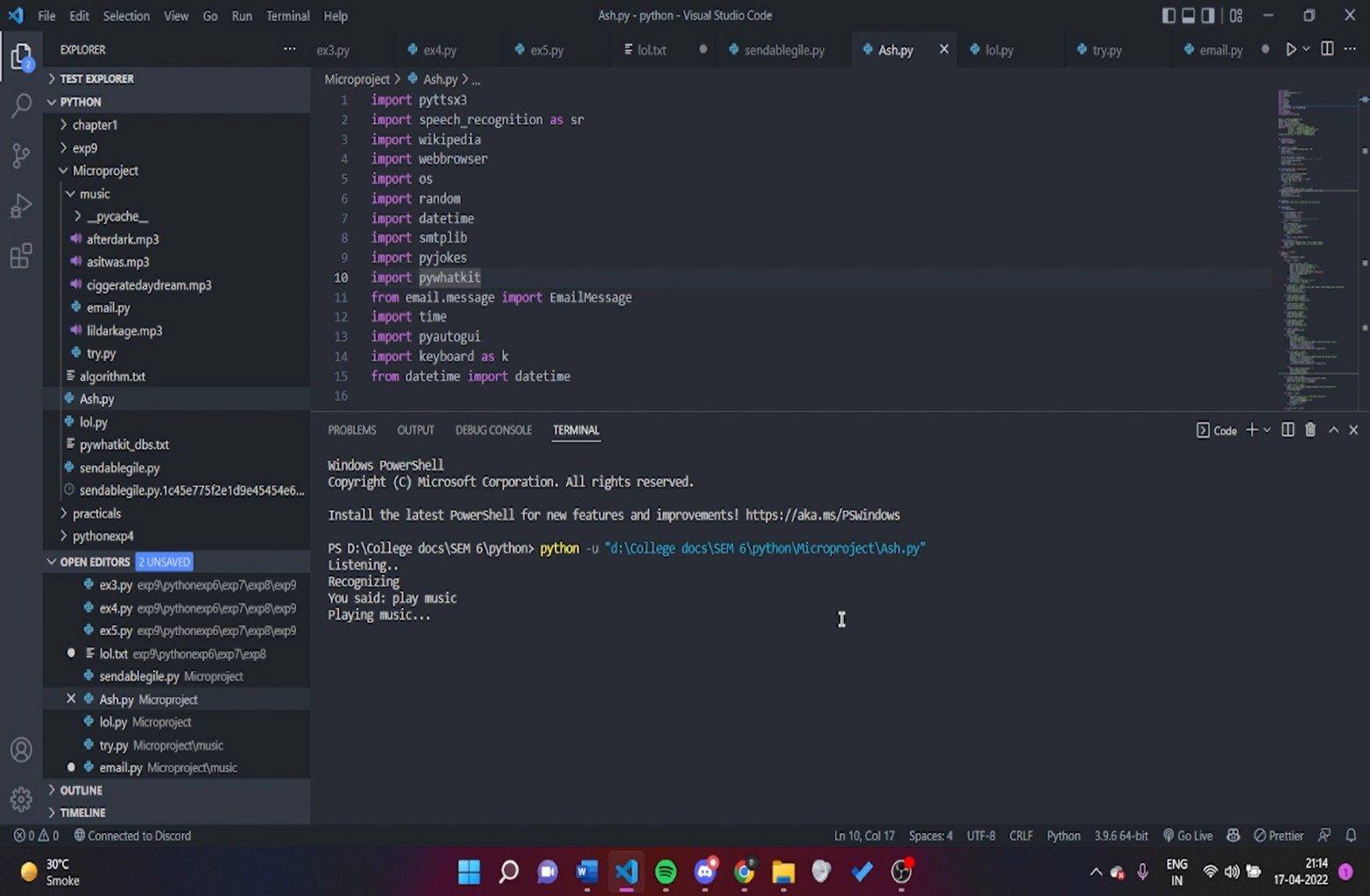
if query != 'none':

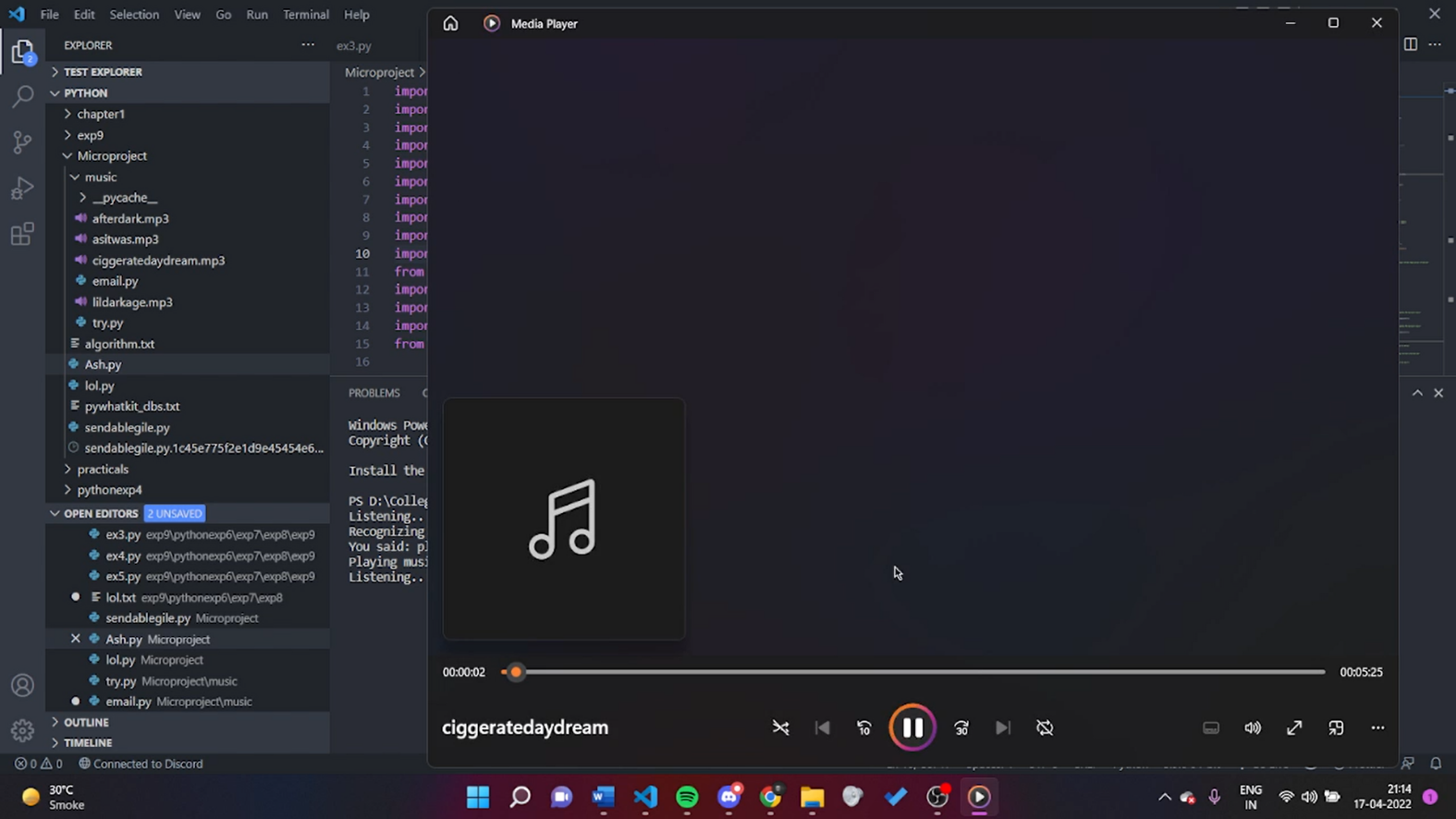
speak("I am not able to do that, sorry!")

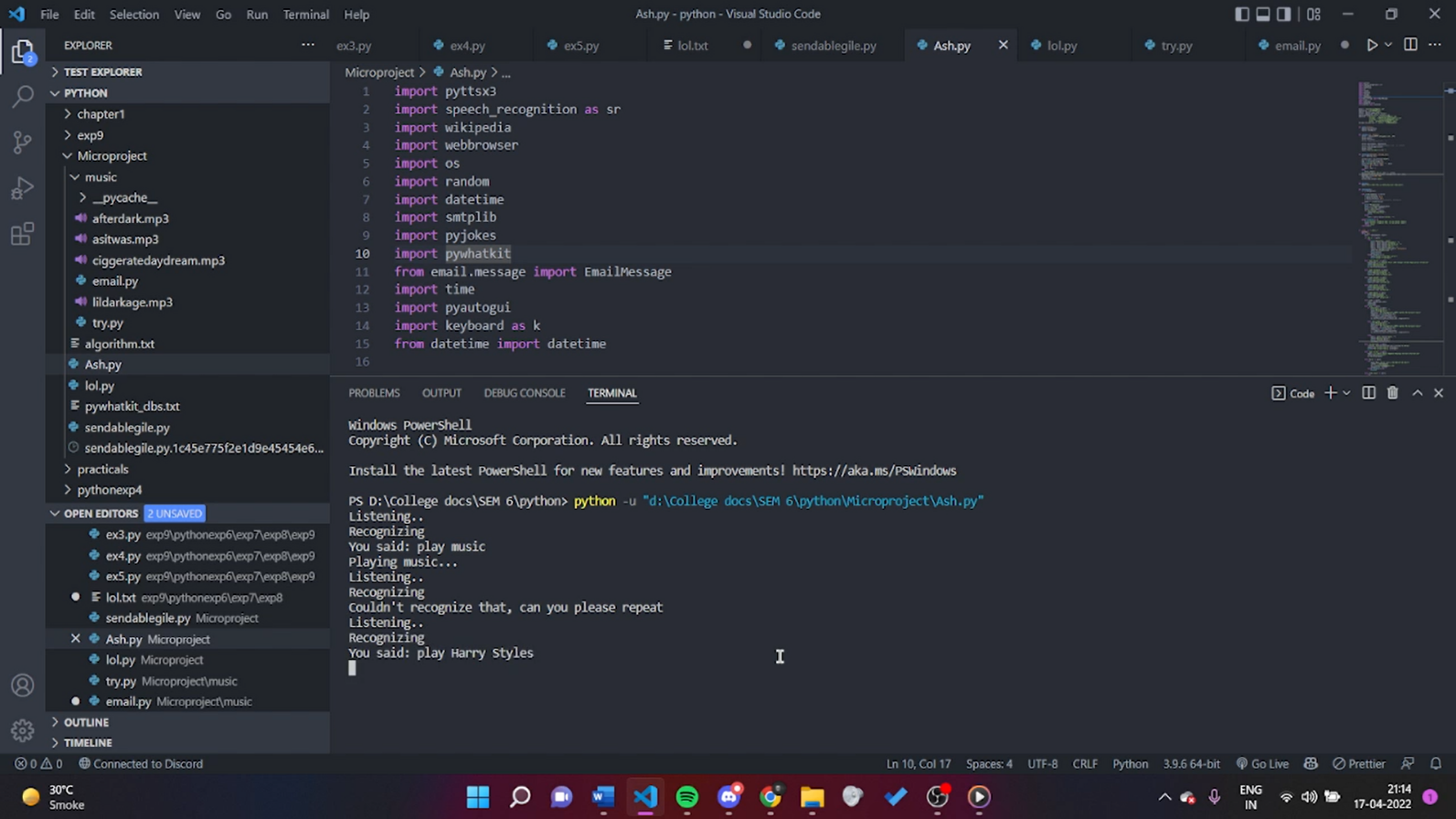
**6.0 Actual Resources Used**

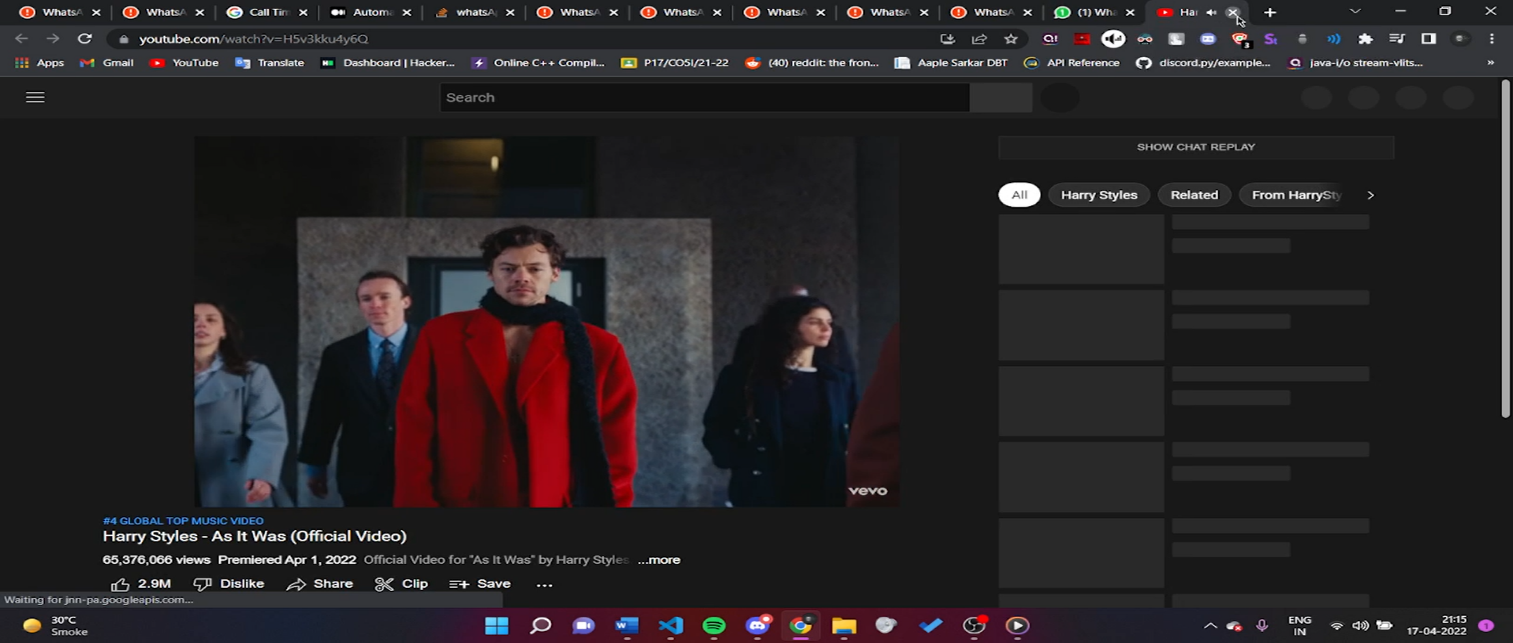
|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Equipment Name with Broad Specification** | **Remark         if any** |
| 1 | Desktop pc – Windows 7 |  |
| 2 | Editor used – Python IDLE, Software – Python 3.4.2 |  |

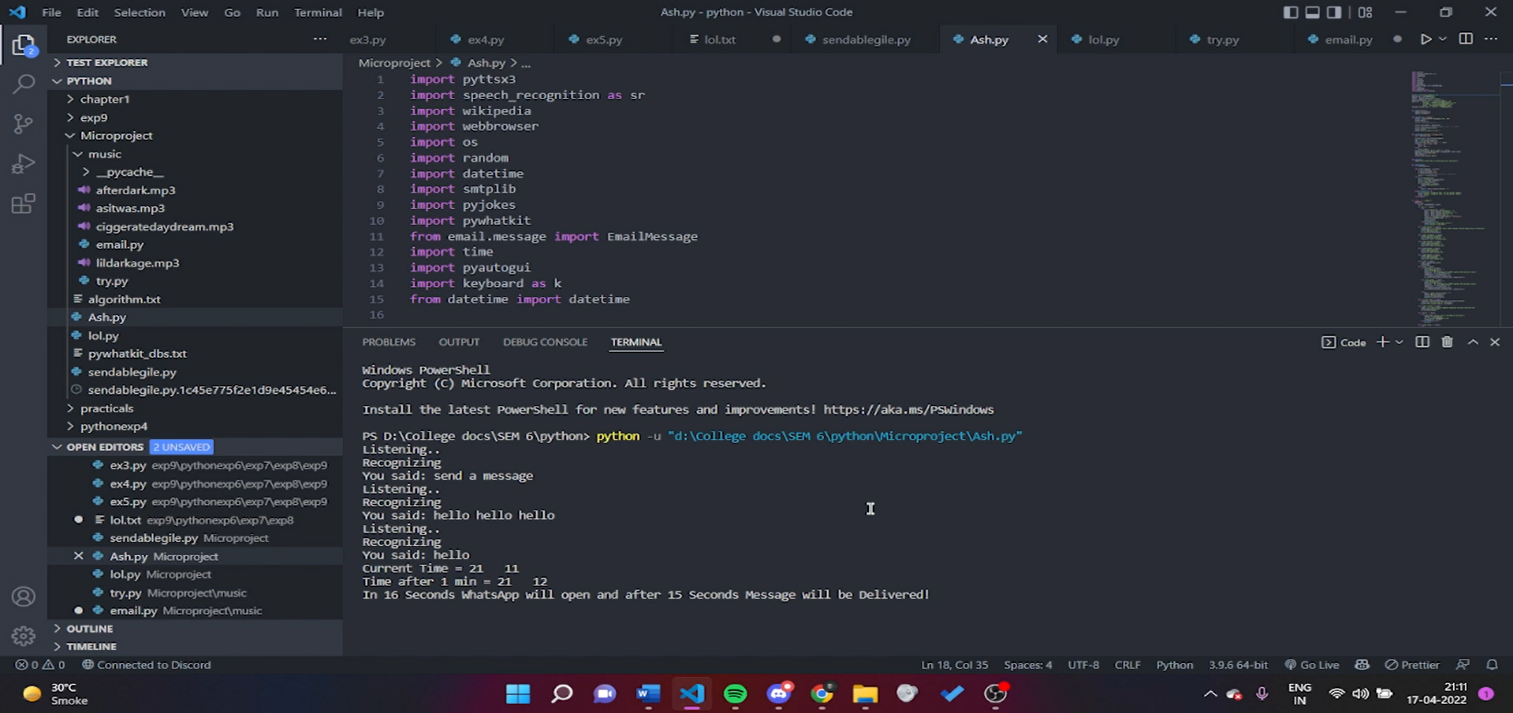
**7.0 Outputs of the Micro-Projects**

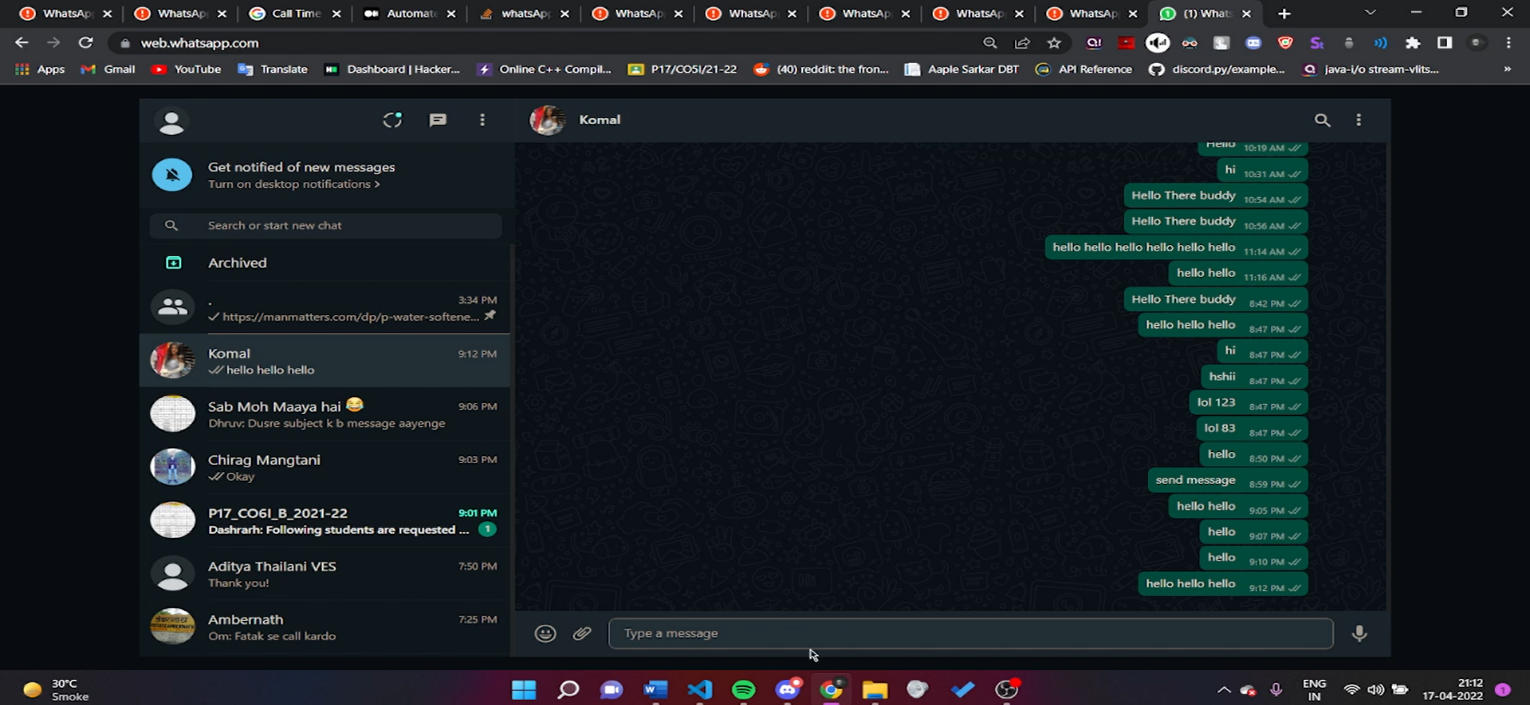












1. **Skill Developed / learning out of this Micro-Project**
2. We learnt how to use python for solving real world problems.
3. We learnt how to use functions in python.
4. We learnt how to use loops for practical usage in python.
5. We learnt how to handle exceptions and errors in python.
6. We understood and implemented various modules and libraries that are available in python.
7. We understood how to use text to speech and speech to text modules in python.
8. We understood the importance of working in team and listening to each other's ideas and views to further improve our project and our knowledge.
9. We have also learnt to coordinate with the team members and support each other for successful completion of the project.
10. We learnt to follow the deadlines and complete our work within specified time frame.

**9.0 Application of this Micro-Project**

* The developed micro-project can be used to make stuff easier while using computer.
* It can be used to send automated whatsapp messages without typing.
* It can be used to play desired music with just a single voice command via microphone.
* It can be used to open various applications and website easily with just a single voice command via microphone.
* In short, the major application is to get user’s command via microphone instead of a keyboard or a mouse which makes it easier for the user to use a lot of things on the computer.

**Annexure – III**

**Suggested Rubric for Assessment of Micro-Project**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S. No.** | **Characteristic to be assessed** | **Poor**  **( Marks 1 - 3 )** | **Average**  **( Marks 4 - 5 )** | **Good**  **( Marks 6 - 8 )** | **Excellent**  **( Marks 9- 10 )** |
| **1** | Relevance to the course | Relate to very few LOs | Related to some LOs | Take care of at-least one CO | Take care of more than one CO |
| **2** | Literature review /information collection | Not more than two sources very old reference | At-least 5 relevant sources, at least 2 latest | At –least 7 relevant sources, most latest | About 10 relevant sources, most latest |
| **3** | Completion of the Target as per project proposal | Completed less than 50% | Completed 50 to 60% | Completed 60 to 80% | Completed more than 80 % |
| **4** | Analysis of Data and representation | Data neither organized nor presented well | Sufficient and appropriate enough data generated but not organized and not presented well. | Sufficient and appropriate enough data generated which is organized and but not used. | Enough data collected and sufficient and presenting data. |
| 5 | Quality of Prototype/Model | Incomplete Programming code | Just assembled and some codeis not functioning well. | Well assembled and functioning parts. But no creativity in design and use of graphics function | Well assembled with proper functioning parts.. Creativity in design and use of graphics function |
| **6** | Report Preparation | Very short, Details about methods, and conclusions omitted, some details are wrong | Nearly sufficient and correct details about methods, and conclusion. but clarity is not there in presentation. | Detailed, correct and clear description of methods and  Conclusions. Sufficient Graphic Description. | Very detailed, correct, clear description of methods, and conclusions. |
| **7** | **Presentation of the micro project** | Major information is not included, information is not well organized. | Includes major information but not well organized and not presented well | Includes major information and well organized but not presented well | Well organized, includes major information ,well presented |
| **8** | **Defense** | Could not reply to considerable number of question. | Replied to considerable number of questions but not very properly | Replied properly to considerable number of question. | Replied most of the questions properly |

**Annexure – IV**

**Micro Project Evaluation Sheet**

Name of Student: ……………………………………… Enrollment No: ………………

Name of Program: Computer Engineering Semester: Sixth

Course Title: Programming with Python Course Code:22616

**Title of the Micro-Project……………….**…………………………………………………………

**Cos addressed by Micro Project:** (Tick appropriate COs)

* Display message on screen using python script on IDE
* Develop python program to demonstrate use of Operators.
* Perform operations on data structure in Python
* Develop function for given problem
* Design classes for given problem
* Handle exception

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr. No** | **Characteristic to be assessed** | **Poor**  **( Marks1-3 )** | **Average**  **(Marks 4-5 )** | **Good**  **(Marks 6-8)** | **Excellent**  **( Marks9-10)** | **Sub Total** |
| 1. **Process and Product Assessment (Convert Above Total marks out of 6 Marks)** | | | | | | |
| 1 | Relevance to the course |  |  |  |  |  |
| 2 | Literature review/ Information Collection |  |  |  |  |
| 3 | Completion of the Target as per project proposal |  |  |  |  |
| 4 | Analysis of Data and representation |  |  |  |  |
| 5 | Quality of Prototype/Model |  |  |  |  |
| 6 | Report Preparation |  |  |  |  |
| 1. Individual Presentation / Viva (Convert above total marks out of 4 marks) | | | | | | |
| 7 | Presentation |  |  |  |  |  |
| 8 | Defense/Viva |  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Roll No.** | **Process and Product Assessment (6 Marks)** | Individual Presentation / Viva (4 Marks) | **Total Marks**  **10** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Name and designation of the Teacher**: Mrs.ShubhangiChintawar (Lecturer)

**Dated Signature**: