PROJECT REPORT

(Project Term August-December 2021)

VIRTUAL ASSISTANT (ALEX)

Submitted by

Nishant Gupta   Reg number-11911240

Mohammad Shahnawaz Malik   Reg number-11911126

Course Code—INT 246

Under the Guidance of

Dr. Sagar Pande

School of Computer Science and Engineering

DECLARATION

We hereby declare that the project work entitled 'Virtual Assistant (Alex)' is an authentic record of our own work carried out as requirements of Project for the award of B.Tech degree in Computer Science And Engineering from Lovely Professional University, Phagwara, under the guidance of, Dr. Sagar Pande during August to December 2021. All the information furnished in this project report is based on our own intensive work and is genuine.

Name of Student 1: Mohammad Shahnawaz Malik

Registration Number: 11911126

Name of Student 2: Nishant Gupta

Registration Number: 11911240

Mohammad Shahnawaz
Date:20-11-2021

Date:20-11-2021

CERTIFICATE

This is to certify that the declaration statement made by this group of students is correct to the best of my knowledge and belief. They have completed this Project under my guidance and supervision. The present work is the result of their original investigation, effort and study. No part of the work has ever been submitted for any other degree at any University. The Project is fit for the submission and partial fulfillment of the conditions for the award of B.Tech degree in Computer Science And Engineering from Lovely Professional University, Phagwara.

Dr Sagar Pande

School of Computer Science and Engineering,
Lovely Professional University,
Phagwara, Punjab.

Date : 20-11-2021

ACKNOWLEDGEMENT

We, Nishant Gupta and Shahnawaz have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them.

We would like to express our special thanks of gratitude to our teacher Dr Sagar Pande who gave us the golden opportunity to do this project on the topic Virtual Assistant (Alex) .It helped us in doing a lot of Research and we came to know about a lot of things related to this topic.

We are highly indebted to Lovely Professional University for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

Finally, we would also like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame.

TABLE OF CONTENT

1. INTRODUCTION

Artificial Intelligence when used with machines, it shows us the capability of thinking like humans. In this, a computer system is designed in such a way that typically requires interaction from human. As we know Python is an emerging language so it becomes easy to write a script for Voice Assistant in Python. The instructions for the assistant can be handled as per the requirement of user. Speech recognition is the Alexa, Siri, etc. In Python there is an API called Speech Recognition which allows us to convert speech into text. It was an interesting task to make our own assistant. It became easier to open websites without typing any word, looking up any information on Wikipedia and performing many other daily tasks like playing music, opening applications in pc with the help of a single voice command. In the current scenario, advancement in technologies are such that they can perform any task with same effectiveness or can say more effectively than us. By making this project, we realized that the concept of AI in every field is decreasing human effort and saving time.

As the voice assistant is using Artificial Intelligence hence the result that it is providing are highly accurate and efficient. The assistant can help to reduce human effort and consumes time while performing any task, they removed the concept of typing completely and behave as another individual to whom we are talking and asking to perform task. The assistant is no less than a human assistant but we can say that this is more effective and efficient to perform any task. The libraries and packages used to make this assistant focuses on the time complexities and reduces time.

Functionalities of this project include:

1. It can tell current date and time.

2. It can tell about the CPU usage and battery percentage

3. It can take a screenshot and save it in a folder.

4. It can tell jokes.

5. It can play music.

6. It can do Wikipedia searches.

7. It can open websites like Google, WhatsApp Web, Stackoverflow etc., in a web browser.

8. It can give weather forecast.

9. It can remember what we ask it to remember.

10. It can give answer to basic questions.

11. It can open Codetantra portal and login automatically.

We have created the py file for this project in Visual Studio Code. Along with this we used following modules and libraries in our project. pyttsx3, speech_recognition, datetime, time, Wikipedia, webbrowser, os, pyjokes, pyautogui, psutil etc.

1.1. PRESENT SYSTEM

We are familiar with many existing voice assistants like Alexa, Siri, Google Assistant, Cortana which uses concept of language processing, and voice recognition. They listen the command given by the user as per their requirements and performs that specific function in a very efficient and effective manner.

As these voice assistants are using Artificial Intelligence hence the result that they are providing are highly accurate and efficient. These assistants can help to reduce human effort and consumes time while performing any task, they removed the concept of typing completely and behave as another individual to whom we are talking and asking to perform task.

But for using these assistants one should have an account (like Google account for Google assistant, Microsoft account for Cortana) and can use it with internet connection only because these assistants are going to work with internet connectivity. They are integrated with many devices like, phones, laptops, and speakers etc.

1.2. PROPOSED SYSTEM

It was an interesting task to make our own assistant. It became easier to open websites without typing any word, looking up any information on Wikipedia and performing many other daily tasks like playing music, opening applications in pc with the help of a single voice command.Alex is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it does not require any internet connection while getting the instructions to perform any specific task.

The editor used in this project is Visual Studio Code. All the necessary packages were easily installable in VS Code. For this project following modules and libraries were used i.e. pyttsx3, SpeechRecognition, Datetime, Wikipedia, pyjokes, pyautogui etc.

With the advancements, Alex can perform any task with same effectiveness or can say more effectively than us. By making this project, we realized that the concept of AI in every field is decreasing human effort and saving time.

2. System Design

The data flow for ALEX is as follow:
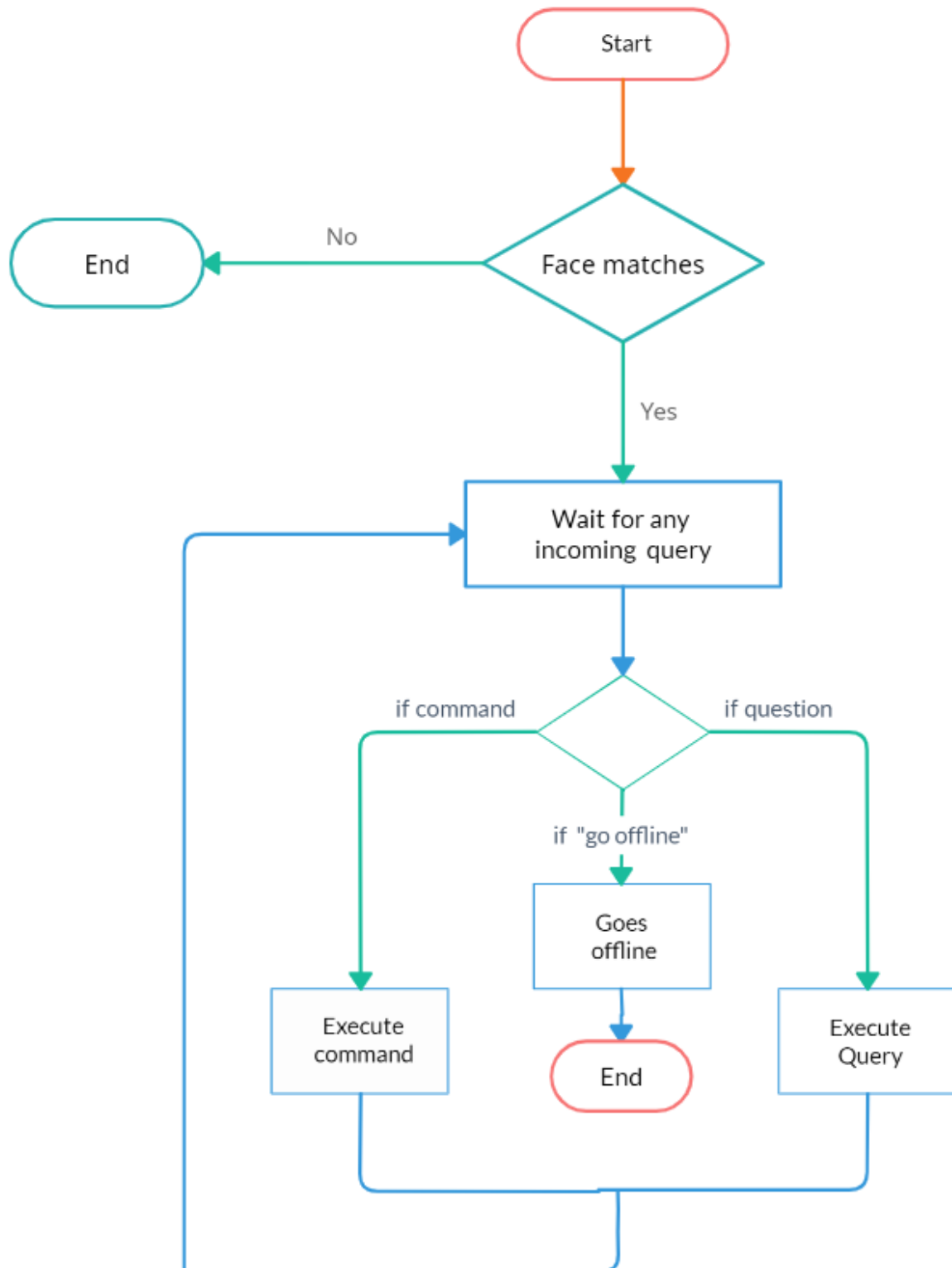
The system is designed using the concept of Artificial Intelligence and with the help of necessary packages of Python. Python provides many libraries and packages to perform the tasks, for example pyjokes can be used to get random online jokes. The details of these packages are mentioned in Chapter 4 of this report.

The data in this project is nothing but user input, whatever the user says, the assistant performs the task accordingly. The user input is nothing specific but the list of tasks which a user wants to get performed in human language i.e. English.

3. Software Details

The editor used in this project is Visual Studio Code. All the necessary packages were easily installable in VS Code. For this project following modules and libraries were installed and used - pyttsx3, speech_recognition, datetime, time, Wikipedia, smtplib, webbrowser, os, pyjokes, pyautogui, psutil, wolframalpha.

3.1. VISUAL STUDIO CODE

Visual Studio Code is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

It can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python and C++. Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse. This allows it to operate as a language-agnostic code editor for any language. It supports a number of programming languages and a set of features that differs per language.
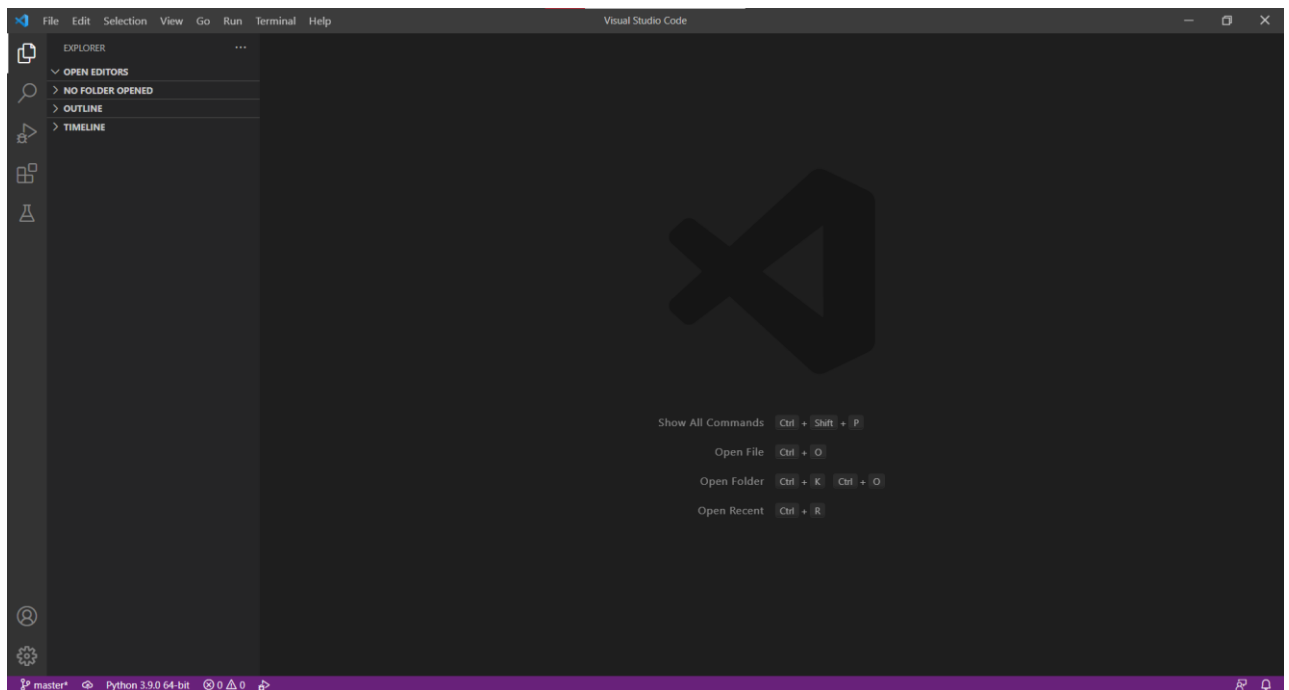
Figure 3.1: Visual Studio Code

4. Implementation

ALEX, a desktop assistant is a voice assistant that can perform many daily tasks of desktop like playing music, opening our favorite site with the help of a single voice command. ALEX is different from other traditional voice assistants in terms that it is specific to desktop and user does not need to make account to use this, it does not require any internet connection while getting the instructions to perform any specific task.

4.1. REAL LIFE APPLICATION

4.1.1. Saves time: ALEX is a desktop voice assistant which works on the voice command offered to it, it can do voice searching, voice-activated device control and can let us complete a set of tasks.

4.1.2. Conversational interaction It makes it easier to complete any task as it automatically do it by using the essential module or libraries of Python, in a conversational interaction way. Hence any user when instruct any task to it, they feel like giving task to a human assistant because of the conversational interaction for giving input and getting the desired output in the form of task done.

4.1.3. Reactive nature: The desktop assistant is reactive which means it know human language very well and understand the context that is provided by the user and gives response in the same way, i.e. human understandable language, English. So user finds its reaction in an informed and smart way.

4.1.4. Multitasking: The main application of it can be its multitasking ability. It can ask for continuous instruction one after other until the user ask it to go "Offline".

4.1.5. No Trigger phase: It asks for the instruction and listen the response that is given by user without needing any trigger phase and then only executes the task.

4.2. DATA IMPLEMENTATION AND PROGRAM EXECUTION

As the first step, install all the necessary packages and libraries. The command used to install the libraries is "*pip install*" and then import it. The necessary packages included are as follows:

4.2.1. Python Libraries

4.2.1.1 pyttsx3 : It is a python library which converts text to speech.

4.2.1.2. **SpeechRecognition :** It is a python module which converts speech to text.

4.2.1.3. **Datetime :** This library provides us the actual date and time.

4.2.1.4. Wikipedia : It is a python module for searching anything on Wikipedia.

4.2.1.5. cv2 : It is a tool for image processing and performing computer vision tasks.

4.2.1.6. Pyjokes : It is a python libraries which contains lots of interesting jokes in it.

4.2.1.7. Webbrowser : It provides interface for displaying web-based documents to users.

4.2.1.8. Pyautogui : It is a python libraries for graphical user interface.

4.2.1.9. os : It represents Operating System related functionality.

4.2.1.10. sys : It allows operating on the interpreter as it provides access to the variables and functions that usually interact strongly with the interpreter.

4.2.1.11. psutil : It is a cross-platform library for retrieving information on running processes and system utilization (CPU, memory, disks, network, sensors) in Python

4.2.1.12. wolframalpha : It is a unique engine for computing answers and providing knowledge.

4.2.1.13 GoogleNews : This module searches and fetches latest news.

```
C: > Users > ASUS > Downloads > 🐍 Untitled12.py > ...
  1   import pyttsx3  # pip install pyttsx3
  2   import datetime
  3   import time
  4   import speech_recognition as sr  # pip install SpeechRecognition
  5   import wikipedia  # pip install wikipedia
  6   import smtplib
  7   import webbrowser as wb
  8   import os
  9   import pyautogui
 10   import psutil
 11   import pyjokes
 12   from wikipedia.wikipedia import search
 13   from bs4 import BeautifulSoup
 14   import requests
 15   import wolframalpha
 16
```

Figure 4.1: Imported Libraries

4.2.2. FUNCTIONS

4.2.2.1. taskexecution() : The function contains all the commands which can be used on the virtual assistant.

4.2.2.2. takeCommand() : The function is used to take the command as input through microphone of user and returns the output as string.

4.2.2.3. wishme() : This function greets the user according to the time like Good Morning, Good Afternoon and Good Evening after saying some predefined initialization commands.

4.2.2.4. time1() : This function tells the current time of the system.

4.2.2.5. date() : This function tells the current date.

4.2.2.6. cpu() : This function tells the CPU usage and battery percentage of the system.

4.2.2.7. screenshot() : This function takes the screenshot of the screen and saves it in a folder.

4.2.2.8. jokes() : This function is used to tell a joke.

4.2.2.9. myclass() : This function opens myclass portal and login into it.

4.2.2.10. temperature() : This function takes the city as input and returns the weather in that city as otput. .

4.2.2.11. quit() : This function closes the program.

4.3. Working

User first trains the model which uses haarcascade classifier provided by OpenCV. Program 'Sample generator.py' takes the images of the user through system camera and stores them in a folder which is later trained by another program 'Model trainer.py'.

When 'Alex.py' is opened, it first verifies the user face through the face detection algorithm embedded in it and if the face matches, it allows the user to give the commands to voice assistant.

User can ask the system to tell time, date, cpu status, latest news, weather in a city or command it to take screenshot, open Chrome and some sites like Stackoverflow, Youtube etc.

5. Input/Output Screenshot

Figure 5.1: Input and output of program when time function is called

Figure 5.2: Input and output of program when date function is called

Figure 5.3: Input and output of program when CPU function is called

Figure 5.4: Input for the function to take a screenshot
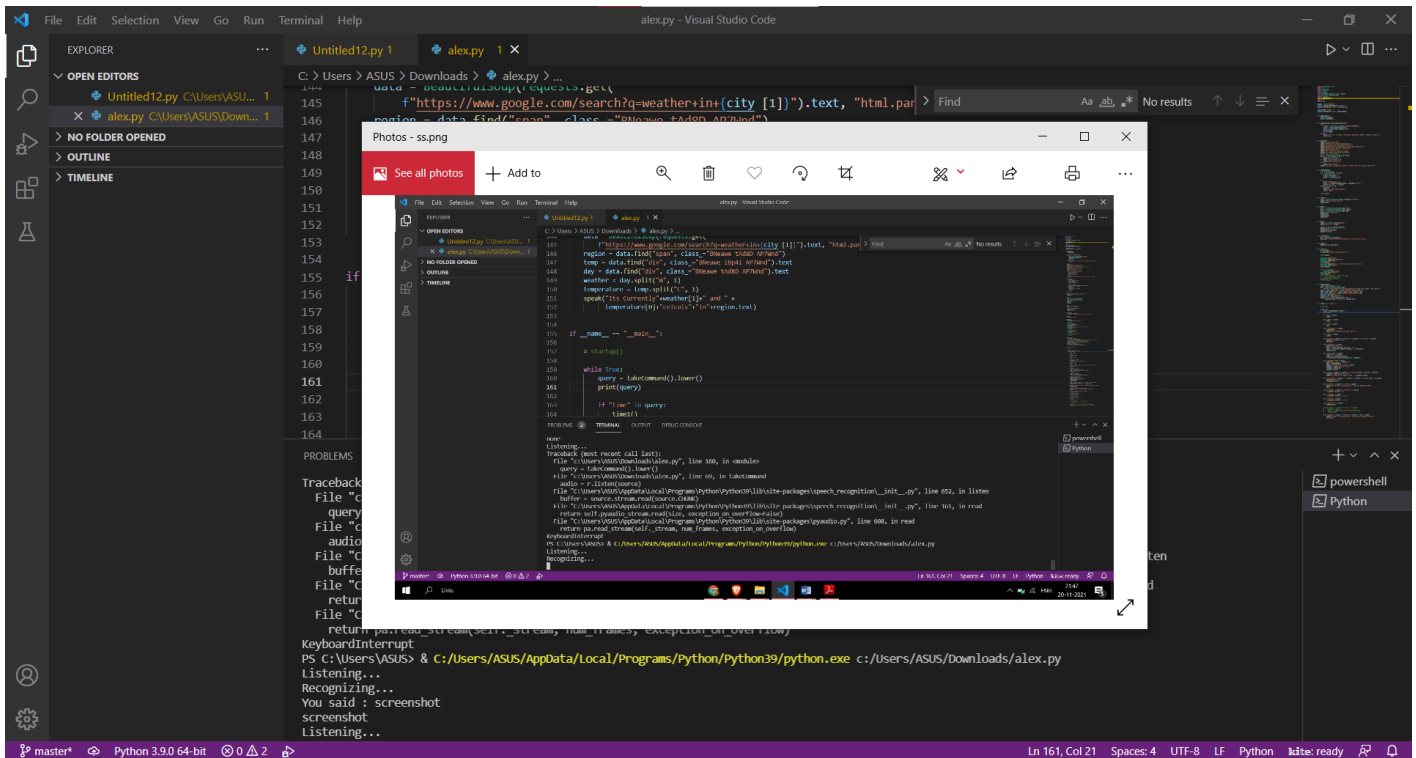
Figure 5.5: Output of screenshot function

Figure 5.6: Input and Output when basic questions are asked

Figure 5.7: Input for myclass function
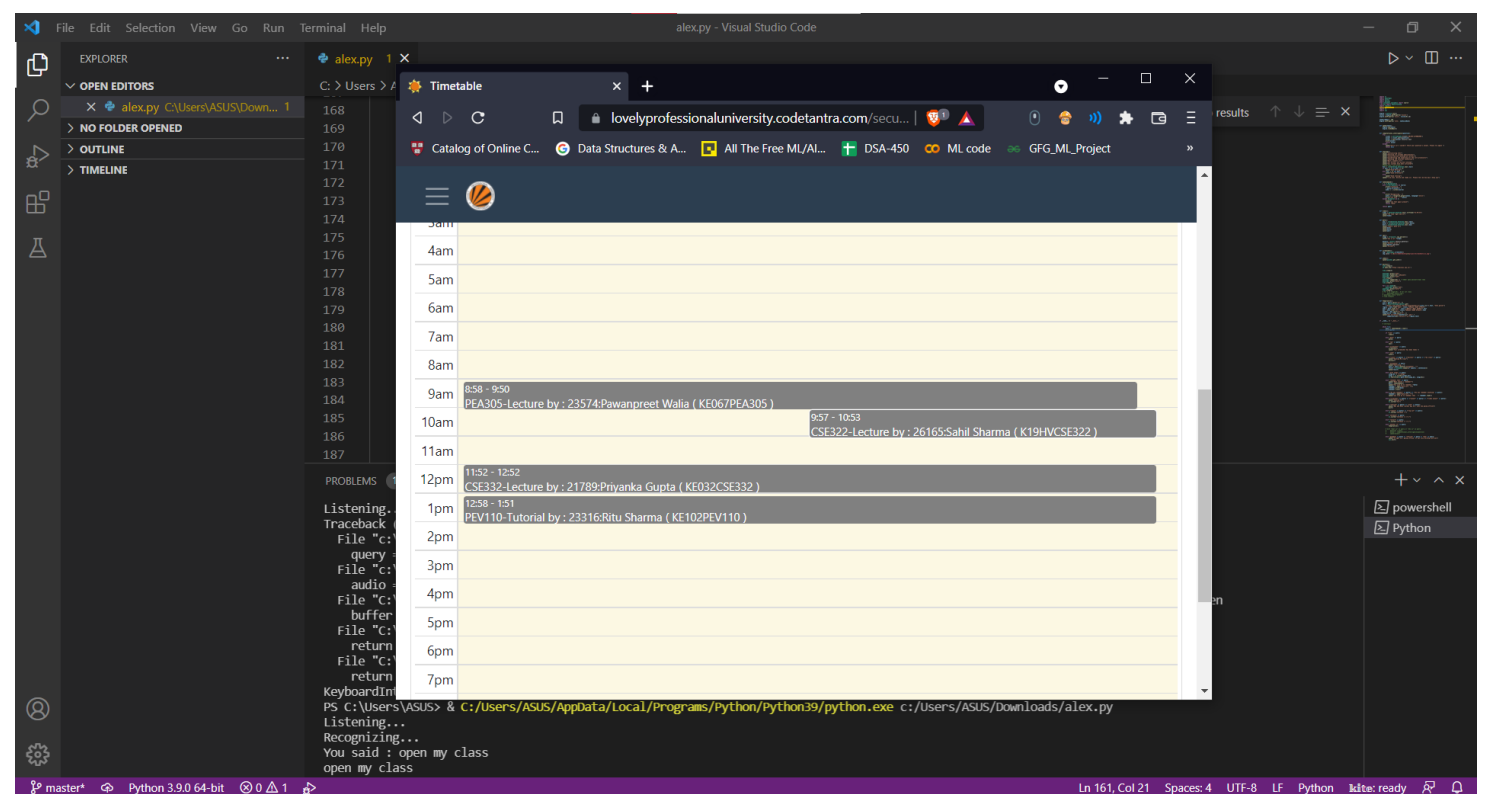
Figure 5.8: Output for myclass function which opens myclass portal

Figure 5.9: Input and output of program when weather of certain place is asked

16

```
PS C:\Users\ASUS> & C:/Users/ASUS/AppData/Local/Programs/Python/Python39/python.exe c:/Users/ASUS/Downloads/alex.py
Listening...
Recognizing...
You said : Wikipedia what is python
wikipedia what is python
Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of signific
ant indentation.
Listening...
```

Figure 5.10: Input and output of program for 'Wikipedia' command

```
Listening...
Recognizing...
You said : headlines
headlines
News updates from Hindustan Times: Nawab Malik launches fresh attack on Sameer Wankhede, and all the latest news,News updates from HT: Supreme Court to hear Trinamool's plea ove
r Tripura violence today and all the latest news,South Africa: Today's latest news and headlines, Thursday 2 December 2021,WATCH
```

Figure 5.11: Input and output of program for 'headlines' command

```
PS C:\Users\ASUS> & C:/Users/ASUS/AppData/Local/Programs/Python/Python39/python.exe
Listening...
Recognizing...
You said : go offline
go offline
Bye and Have a great Day Sir, alex now going offline
PS C:\Users\ASUS>
```

Figure 5.12: Input and output of program for 'offline' command

6. Testing

The system testing is done on fully integrated system to check whether the requirements are matching or not. The system testing for ALEX desktop assistant focuses on the following four parameters:

6.1. FUNCTIONALITY

In this we check the functionality of the system whether the system performs the task which it was intended to do. To check the functionality each function was checked and run, if it is able to execute the required task correctly then the system passes in that particular functionality test. For example to check whether ALEX can search on Google or not, as we can see in the figure 6.1, user said "Open Google", then ALEX asked, "What should I search on Google?" then user said, "What is Python", ALEX opens Google and searched for the required input.

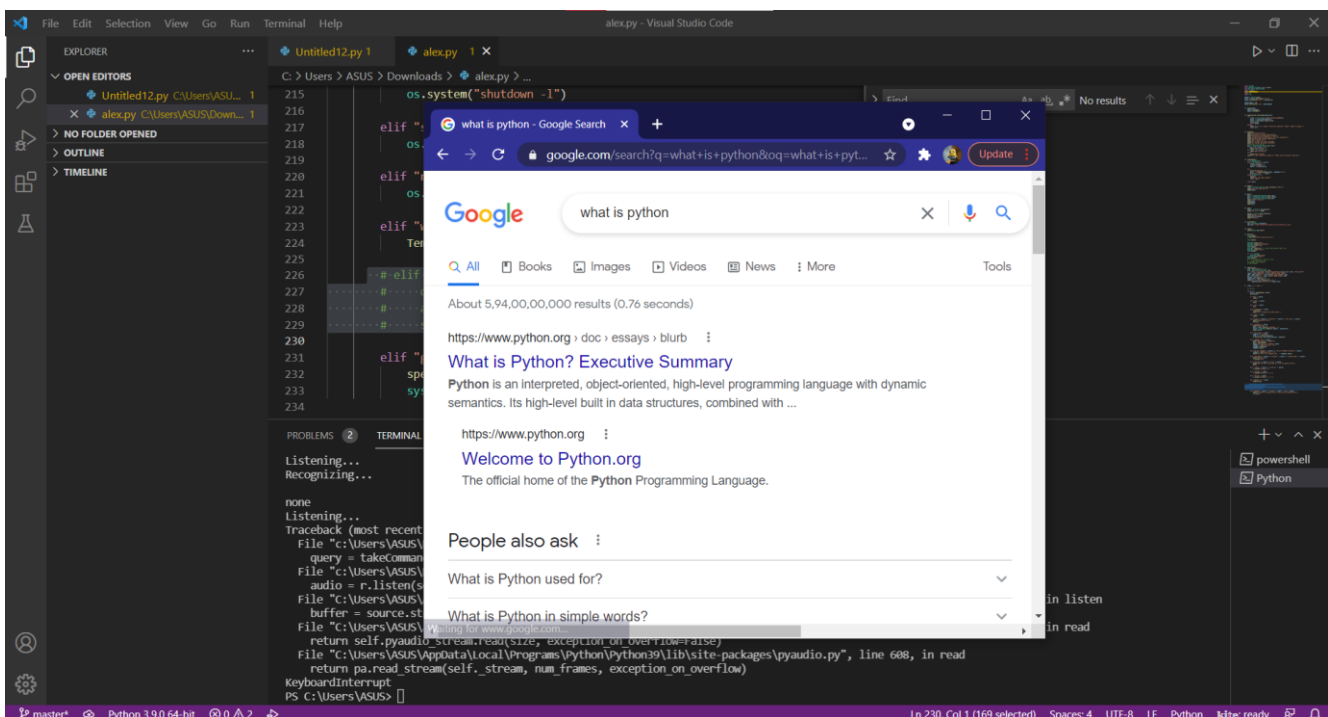**Figure 6.1: Input through voice commands**

Figure 6.2 : Output

18

6.2. USABILITY

Usability of a system is checked by measuring the easiness of the software and how user friendly it is for the user to use, how it responses to each query that is being asked by the user.

It makes it easier to complete any task as it automatically do it by using the essential module or libraries of Python, in a conversational interaction way. Hence any user when instruct any task to it, they feel like giving task to a human assistant because of the conversational interaction for giving input and getting the desired output in the form of task done.

The desktop assistant is reactive which means it know human language very well and understand the context that is provided by the user and gives response in the same way, i.e. human understandable language, English. So user finds its reaction in an informed and smart way.

The main application of it can be its multitasking ability. It can ask for continuous instruction one after other until the user goes "offline". It asks for the instruction and listen the response that is given by user without needing any trigger phase and then only executes the task.

6.3. SECURITY

The security testing mainly focuses on vulnerabilities and risks. As ALEX is a local desktop application, hence there is no risk of data breaching through remote access. The software is dedicated to a specific system so when the user logs in, it will be activated.

6.4. STABILITY

Stability of a system depends upon the output of the system, if the output is bounded and specific to the bounded input then the system is said to be stable. If the system works on all the poles of functionality then it is stable.

7. Conclusion

ALEX is a very helpful voice assistant without any doubt as it saves time of the user by conversational interactions, its effectiveness and efficiency. But while working on this project, there were some limitations encountered and also realized some scope of enhancement in the future which are mentioned below:

7.1. LIMITATIONS

7.1.1. Security is somewhere an issue, there is no voice command encryption in this project.

7.1.2. Background voice can interfere

7.1.3. Misinterpretation because of accents and may cause inaccurate results.

7.1.4. ALEX cannot be called externally anytime like other traditional assistants like Google Assistant can be called just by saying, "Ok Google!"

7.2. SCOPE FOR FUTURE WORK

7.2.1. Make ALEX to learn more on its own and develop a new skill in it.

7.2.2. ALEX android app can also be developed.

7.2.3. Make more Alex voice terminals.

7.2.4. Voice commands can be encrypted to maintain security.

8. User Manual

1. Open and run the python program named 'Sample generator.py' which can be found on the github link provided . But before running change the path of 'haarcascade_frontalface_default.xml' (which is available in the github folder) to the project folder in your system.

2. Open the python program named 'Model Trainer.py' and make changes to the path of image data and the folder to save trained file according to your system.

3. Open and run 'Alex.py', the program will verify your face and once verified you are ready to use your own virtual assistant.

9. Github Link for source code

https://github.com/Nishant9508/mlproject/tree/main/ML_assigment_INT243