A Project Report

On

Jarvis (Personal Assistant)

Submitted to

DELHI TECHNICAL CAMPUS

(Affiliated Guru Gobind Singh Indraprastha University New Delhi)
Greater Noida

Affiliated to GGSIP University, New Delhi
Approved by AICTE & Council of Architecture

in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology

by

Mayank Goyal
[35118002717]
B. tech (CSE), 4th Year

Department of Computer Science and Engineering
DELHI TECHNICAL CAMPUS
GREATER NOIDA (U.P.)

Declaration by the student

I/We Mayank Goyal student(s) of B. Tech (CSE branch) 4 year hereby declare that the project titled "**Jarvis (Personal Assistant)"** which is submitted by me/us to Department of Computer Science and engineering DELHI TECHNICAL CAMPUS, Noida, in partial fulfillment of requirement for the award of the degree of Bachelor of Technology in 7 semester 4 year, has not been previously formed the basis for the award of any degree, diploma or other similar title or recognition.

The Author attests that permission has been obtained for the use of any copy righted material appearing in the Dissertation / Project report other than brief excerpts requiring only proper acknowledgement in scholarly writing and all such use is acknowledged.

Name and Signature of Student(s): Mayank Goyal [35118002717]

Date: 05/01/2021

Certificate of Completion

ACKNOWLEDGEMENT

I had a great experience working on this project and we got to learn a plethora of new skills through this project. However, it would not have been possible without the kind support and help of many individuals. I would like to extend our sincere thanks to all of them. I am highly indebted to the Instructor of Udemy for their guidance and constant supervision as well as providing necessary information regarding the project and also for their support in completing the project.

I would like to express my gratitude towards my parents and friends for their cooperation and encouragement which help me in the completion of this project.

Name – Mayank Goyal [35118002717]

Date – 05/01/2021

ABSTRACT

The project aims to develop a personal-assistant Jarvis draws its inspiration from virtual assistants like Cortana for Windows, and Siri for iOS. It has been designed to provide a user-friendly interface for carrying out a variety of tasks by employing certain well-defined commands. Users can interact with the assistant either through voice commands.

As a personal assistant, Jarvis assists the end-user with day-to-day activities like general human conversation, searching queries in google, searching for videos on YouTube, it switches between the opening windows, we can extract the summary from wiki, opening and closing of notepad, CMD, web camera, finding IP Address, sending WhatsApp message and jokes.

Table of Figures

Table of Content

Chapter-1

Introduction to Python

- Python is a widely used general-purpose, high level programming language. It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.
- Python is a general purpose, dynamic, high level and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures.
- Python is easy to learn yet powerful and versatile scripting language which makes it attractive for Application Development.
- Python's syntax and dynamic typing with its interpreted nature, makes it an ideal language for scripting and rapid application development.
- Python supports multiple programming pattern, including object oriented, imperative and functional or procedural programming styles.

Features of python

- Easy to Learn and Use Python is easy to learn and use. It is developer-friendly and high-level programming language.
- Expressive Language Python language is more expressive means that it is more understandable and readable.
- Interpreted Language Python is an interpreted language i.e.; interpreter executes the code line by line at a time. This makes debugging easy and thus suitable for beginners.
- Cross-platform Language Python can run equally on different platforms such as Windows, Linux, Unix and Macintosh etc. So, we can say that Python is a portable language.
- Free and Open-Source Python language is freely available at official web address. The source-code is also available. Therefore, it is open source.
- Object-Oriented Language Python supports object-oriented language and concepts of classes and objects come into existence.
- Extensible It implies that other languages such as C/C++ can be used to compile the

code and thus it can be used further in our python code.

Python History

- Python laid its foundation in the late 1980s.
- The implementation of Python was started in December 1989 by **Guido Van Rossum** at CWI in Netherland.
- In 1994, Python 1.0 was released with new features like lambda, map, filter, and reduce.
- Python 2.0 added new features such as list comprehensions, garbage collection systems.
- On December 3, 2008, Python 3.0 (also called "Py3K") was released. It was designed to rectify the fundamental flaw of the language.

Why the Name Python?

There is a fact behind choosing the name Python. **Guido van Rossum** was reading the script of a popular BBC comedy series "**Monty Python's Flying Circus**". It was late on-air 1970s. Van Rossum wanted to select a name which unique, sort, and little-bit mysterious. So, he decided to select naming Python after the **"Monty Python's Flying Circus"** for their newly created programming language.

Usage of Python

Python is a general purpose, open source, high-level programming language and also provides number of libraries and frameworks. Python has gained popularity because of its simplicity, easy syntax and user-friendly environment. The usage of Python as follows.

- Desktop Applications
- Web Applications
- Data Science
- Artificial Intelligence
- Machine Learning
- Robotics
- Internet of Things (IoT)
- Gaming
- Mobile Apps

Chapter-2

Introduction to Project

Have you ever wondered how cool it would be to have your own A.I. assistant? Imagine how easier it would be to doing Wikipedia searches without opening web browsers, and performing many other daily tasks like playing music on YouTube, search anything on google, switches between the windows or to listen jokes with the help of a single voice command.

What can this A.I. assistant do for you?

- It can send WhatsApp messages for you.
- It can play music for you on YouTube.
- It can do Wikipedia searches for you.
- It is capable of opening websites like Google, YouTube, etc., in a web browser.
- It is capable of opening your code editor or IDE with a single voice command.

HOW IT WORKS?

- Speak something (for e.g., “Play Songs”).
- Jarvis records the voice and match with available commands.
- If it is available then proper response is provided.
- And proper action is taken.

ADVANTAGES

- Easy to use.
- Can work with variety of commands.
- Custom commands.
- Secure.
- Helpful for disabled.
- Artificial intelligent.

DISADVANTAGES

- Limited language support.
- Costly.
- It cannot work in noisy environment.
- Limited Commands.

Chapter-3

Technologies used

Hardware Used:

- Microphone
- Web camera

Software Used:

- Python Environment with pip installation.
- Any IDE for writing code (I am Using **Visual Studio Code**)

Modules of Python Used:

- Date Time Module
- Pyttxs3 Module
- Speech_Recognition Module
- Wikipedia Module
- Webbrowser Module
- Pywhatkit Module
- Pyjokes Module
- Os Module
- Cv2 Module
- Requests Module
- Sys Module
- PyAutoGUI Module

Chapter-4

Development Environment

Installation of Python on Windows

- Select Version of Python to Install.
- Download Python Executable Installer.
- Run Executable Installer.
- Verify Python Was Installed on Windows.
- Verify Pip Was Installed.
- Add Python Path to Environment Variables.

Installation of Visual Studio Code on Windows

- Download the Visual Studio Code installer for Windows.
- Once it is downloaded, run the installer (VSCodeUserSetup-{version}.exe). This will only take a minute.
- Add the Python Interpreter to the Path of Visual Studio Code.

Installation of all the Modules

- Pyttxs3 Module (Pip install Pyttsx3).
- Speech_Recognition Module (Pip install Speech_Recognition).
- Wikipedia Module (Pip install Wikipedia).
- Webbrowser Module (Pip install Webbrowser).
- Pywhatkit Module (Pip install Pywhatkit).
- Pyjokes Module (Pip install Pyjokes).
- Cv2 Module (Pip install Cv2).
- PyAutoGUI Module (Pip install Pyautogui).

Modules used in this Project

Pyttxs3 Module

Pyttxs3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3.

Speech_Recognition Module

Speech recognition, as the name suggests, refers to automatic recognition of human speech. Speech recognition is one of the most important tasks in the domain of human computer interaction.

Wikipedia Module

Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia. Search Wikipedia, get article summaries, get data like links and images from a page, and more.

Webbrowser Module

The webbrowser module provides a high-level interface to allow displaying Web-based documents to users. Under most circumstances, simply calling the open() function from this module will do the right thing.

Pywhatkit Module

PyWhatKit is a Python library for Sending WhatsApp message at certain time, it has several other features too.

Pyjokes Module

One-line jokes for programmers (jokes as a service).

Cv2 Module

OpenCV (Open-Source Computer Vision) is a computer vision library that contains various functions to perform operations on Images or videos. OpenCV library can be used to perform multiple operations on videos.

PyAutoGUI Module

PyAutoGUI is a cross-platform GUI automation Python module for human beings. Used to programmatically control the mouse & keyboard.

Date Time Module

The datetime module has a class named date class that can contain information from both date and time objects.

OS Module

he OS module in python provides functions for interacting with the operating system. OS, comes under Python's standard utility modules.

Request Module

The requests module allows you to send HTTP requests using Python. The HTTP request returns a Response Object with all the response data (content, encoding, status, etc).

Sys Module

The python sys module provides functions and variables which are used to manipulate different parts of the Python Runtime Environment.

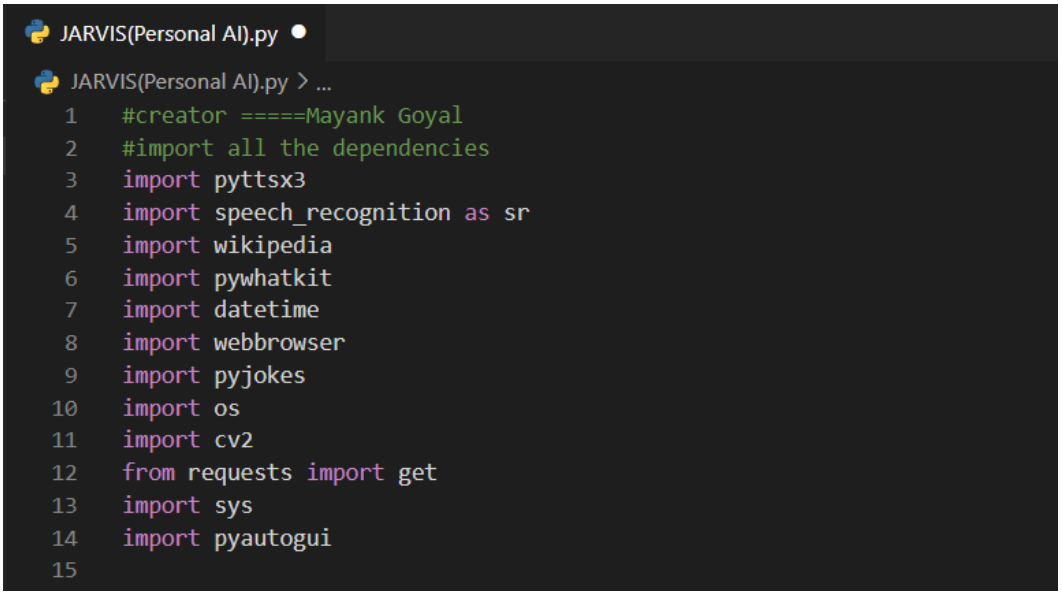

Fig. No. 1: Screenshot of all the dependencies

Chapter-5

Steps for making of the Project

- **Import all the Modules**

JARVIS(Personal AI).py

```python
#creator =====Mayank Goyal
#import all the dependencies
import pyttsx3
import speech_recognition as sr
import wikipedia
import pywhatkit
import datetime
import webbrowser
import pyjokes
import os
import cv2
from requests import get
import sys
import pyautogui

```

Fig. No. 2

- **Code for the pyttxs3 to start (Implementation of Pyttxs3)**

```python

#creator ====Mayank Goyal

#starting of jarvis
engine = pyttsx3.init('sapi5')
voices=engine.getProperty('voices')
print(voices[0].id)
voices=engine.setProperty('voice',voices[0].id)
engine.say("Test voice to check is Jarvis is running correctly... ")
engine.runAndWait()

```

Fig. No. 3

- **Writing our speak function (Convert text-to-speech function)**

```
26
27  #creator ====Mayank Goyal
28
29  #create text to speak function
30  def speak(audio):
31      engine.say(audio)
32      print(audio)
33      engine.runAndWait()
34
```

Fig. No. 4

- **Creates the Main Function**

```
71
72  #creator ====Mayank Goyal
73
74  #main function
75  if __name__ == "__main__":
76      # speak("Hello Mayank")
77      # talkcommand()
78          wishme()
79
```

Fig. No. 5

- **Create takecommand function (Jarvis take commands from this function)**

```
35  #creator ====Mayank Goyal
36
37  # jarvis take command to user
38  def talkcommand():
39      listen=sr.Recognizer()
40      with sr.Microphone() as source:
41          print("Listenning...")
42          listen.pause_threshold=1
43          audio=listen.listen(source,timeout=10,phrase_time_limit=5)
44
45      try:
46          print("Recognization.....")
47          query=listen.recognize_google(audio,language='en-in')
48          print(f"user said: {query}")
49
50      except Exception as e:
51          speak("Say that again please")
52          return None
53
54      return query
```

Fig. No. 6

- **Code Wishme function (Wishes whenever Jarvis starts)**

```
57  #creator ====Mayank Goyal
58
59  def wishme():
60      hour=int(datetime.datetime.now().hour)
61
62      if hour>0 and hour<12:
63          speak("Good Morning sir")
64
65      elif hour>12 and hour<18:
66          speak("Good afternoon sir")
67
68      else:
69          speak("Good Evening sir")
70
71      speak(" My name is jarvis Please tell me how may i help you")
72
```

Fig. No. 7

- **Build the Logic (Different outputs commands as per your choice)**

```
81
82  while True:
83          query=talkcommand().lower()
84
85          #Building logic
86
87          if 'notepad' in query:
88              npath="C:\\WINDOWS\\system32\\notepad.exe"
89              os.startfile(npath)
90
91          elif 'open command prompt' in query:
92              os.system("start cmd")
```

Fig. No. 8

- **Code closing command (As per your choice)**

```
165
166          elif 'no thanks' in query:
167              speak("Thanks for using me sir,have a good day")
168              sys.exit()
169
170
171
172          speak("sir do you have any other work")
173
174   #creator ====Mayank Goyal
175
176  ############################  END OF PROGRAM ##############################
```

Fig. No. 9

Output of project (Jarvis)

Example

Input Command: - “play see you again on YouTube”.

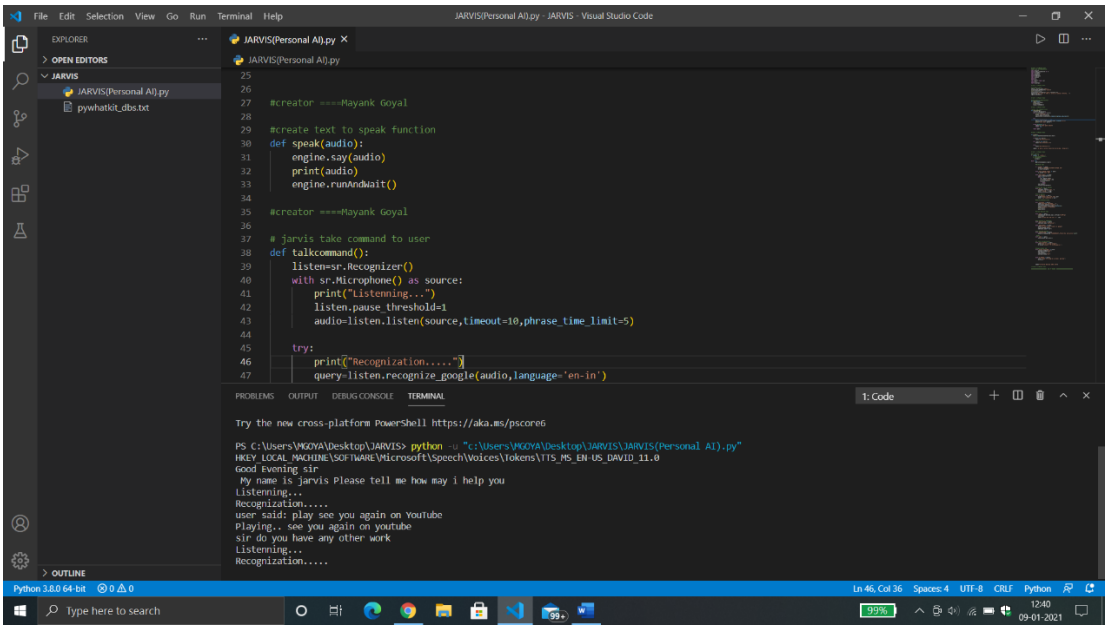

Fig. No. 10

Output Screen

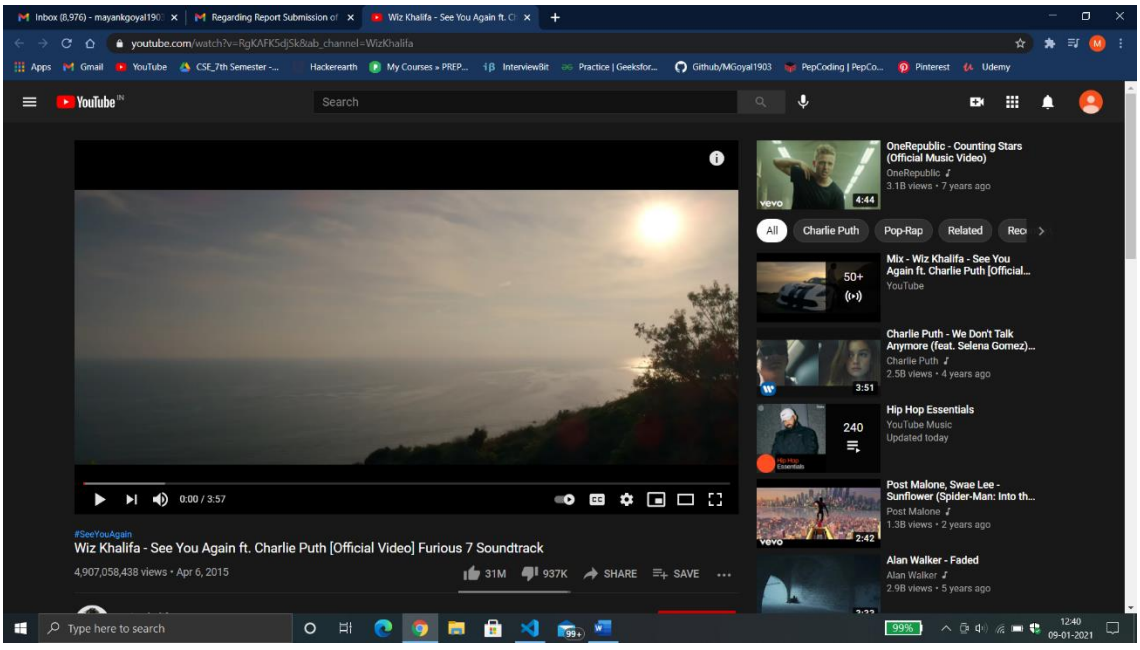

Fig. No. 11

CONCLUSION

Through this voice assistant, we have automated various services using a single line command. It eases most of the tasks of the user like searching the web, open any Application, Sending WhatsApp messages, Play music on YouTube etc. Jarvis is a digital and virtual assistant with an artificial intelligence. It's very flexible and useful technology. It provides a better interface to deal with it.

References

1. https://jarvis.readthedocs.io/en/latest/
2. https://www.ijert.org/research/jarvis-digital-life-assistant-IJERTV2IS1237.pdf
3. https://www.skyfilabs.com/project-ideas/jarvis-personal-assistant-using-python
4. https://applikeysolutions.com/blog/how-to-build-your-own-artificial-intelligence-assistant
5. https://docs.python.org/3/
6. https://www.tutorialspoint.com/python/index.htm
7. https://jarvis.readthedocs.io/en/latest/
8. https://www.udemy.com/course/software-development-in-python-a-practical-approach/learn/lecture/20547646#overview