

A Project Report

on

Jarvis, Digital Life Assistant with Machine Learning, using Python

Submitted in partial fulfillment

requirement for the award of the degree of

Bachelor's of Technology in Computer Science Engineering



(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of

Dr.S.Annamalai

(Associate Professor)

Submitted By

Gaurav Gangwar

18SCSE1010419

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
DECEMBER, 2021**



**SCHOOL OF COMPUTING SCIENCE AND
ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA**

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **“Jarvis Digital Life Assistant using Deep learning”** in partial fulfillment of the requirements for the award of the Bachelors of Technology submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of month, Year to Month and Year, under the supervision of Name **“Dr.S.Annamalai”** Designation, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida.

The matter presented in the thesis/project/dissertation has not been submitted by us for the award of any other degree of this or any other places.

Gaurav Gangwar(18SCSE1010419)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr.S.Annamalai

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of
“**GAURAV GANGWAR (18SCSE1010419)** ” has been held on
_____ and their work is recommended for the award of
B.Tech(CSE).

Signature of Examiner(s)

Signature of Supervisor(s)

Signature of Project Coordinator

Signature of Dean

Date: December, 2021

Place: Greater Noida

ACKNOWLEDGEMENT

Apart from the efforts of our, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We would like to show my greatest appreciation to “**Dr.S.Annamalai**” and also our reviewer “**Mr.Vipin Rai**” and also our dean “**Dr. Munish Sabarwal**”. We can’t say thank you enough for his tremendous support and help. We feel motivated and encouraged every time, We attend his meeting. Without his encouragement and guidance this project would not have materialized.

The guidance and support received from all the members who contributed and who are contributing to this project, was vital for the success of the project. We are grateful for their constant support and help.

Abstract

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 process of converting speech into text. This is commonly used in voice assistants like Alexa, Siri, etc. In Python there is an API called SpeechRecognition which allows us to convert speech into text. It was an interesting task to make my own assistant. It became easier to send emails without typing any word, Searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE 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, I realized that the concept of AI in every field is decreasing human effort and saving time.

Functionalities of this project include:

1. It can send emails.
2. It can read PDF.
3. It can send text on WhatsApp.
4. It can open command prompt, your favorite IDE, notepad etc.
5. It can play music.

6. It can do Wikipedia searches for you.
7. It can open websites like Google, YouTube, etc., in a web browser.
8. It can give weather forecast.
9. It can give desktop reminders of your choice.
10. It can have some basic conversation.

Now the basic question arises in mind that how it is an AI? The virtual assistant that I have created is like if it is not an A.I, but it is the output of a bundle of the statement. But fundamentally, the main purpose of A.I machines is that it can perform human tasks with the same efficiency or even more efficiently than humans. It is a fact that my virtual assistant is not a very good example of A.I., but it is an A.I.

Table of Contents

Title	Content	Page No.
Declaration		I
Certificate		II
Acknowledge		III
Abstract		IV
List of Figures		V
Chapter 1	Introduction	
	1.1 Introduction	
	1.2 Formulation of Problem	
	1.2.1 Tool and Technology Used	
	1.3 Motivation	
	1.4 Objective	
	1.5 Project Features	
Chapter 2	Literature Survey and Project Design	
	2.1 Comparative study	
	2.2 Existing System	
	2.3 Proposed System	
	2.4 Architecture	

Chapter 3	System Requirement
Chapter 4	Project Design
Chapter 5	Implementation /Working of Project
Chapter 6	Results
Chapter 7	Future Scope
Chapter 8	Conclusion
Chapter 9	References

CHAPTER-1

Introduction

1.1 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 my own assistant. It became easier to send emails without typing any word, Searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE 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, I 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.

The functionalities include , It can send emails, It can read PDF, It can send text on WhatsApp, It can open command prompt, your favorite IDE, notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can give weather forecast, It can give desktop reminders of your choice. It can have some basic conversation.

Tools and technologies used are PyCharm IDE for making this project, and I created all py files in PyCharm. Along with this I used following modules and libraries in my project. pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib, pywhatkit, pyjokes, pyPDF2, pyautogui, PyQt etc. I have created a

live GUI for interacting with the JARVIS as it gives a design and interesting look while having the conversation.

1.2 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 listens 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. These assistants are no less than a human assistant but we can say that they are more effective and efficient to perform any task. The algorithm used to make these assistant focuses on the time complexities and reduces time.

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.

1.3 Proposed System:

It was an interesting task to make my own assistant. It became easier to send emails without typing any word, Searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE with the help of a single voice command. Jarvis 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 IDE used in this project is PyCharm. All the python files were created in PyCharm and all the necessary packages were easily installable in this IDE. For this project following modules and libraries were used i.e. pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib, pywhatkit, pyjokes, pyPDF2, pyautogui, PyQt etc. I have created a live GUI for interacting with the JARVIS as it gives a design and interesting look while having the conversation. With the advancement JARVIS can perform any task with same effectiveness or can say more effectively than us. By making this project, I realized that the concept of AI in every field is decreasing human effort and saving time.

Functionalities of this project include, It can send emails, It can read PDF, It can send text on WhatsApp, It can open command prompt, your favorite IDE, notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can give weather forecast, It can give desktop reminders of your choice. It can have some basic conversation.

1.2.1 Tools and Technology Used:

❖ SOFTWARE REQUIREMENT :

As the project is developed in python, we have used Anaconda for Python 3.6.5 and PyCharm.

1. Anaconda

It is a free and open source distribution of the Python and R programming languages for data science and machine learning related applications (large-scale data processing, predictive analytics, scientific computing), that aims to simplify package management and deployment. Package versions are managed by the package management system conda. The Anaconda distribution is used by over 6 million users, and it includes more than 250 popular data science packages suitable for Windows, Linux, and MacOS.

2.PyCharm:-

It is an IDE i.e. Integrated Development Environment which has many features like it supports scientific tools(like matplotlib, numpy, scipy etc) web frameworks (example Django,web2py and Flask) refactoring in Python, integrated python debugger, code completion, code and project navigation etc. It also provides Data Science when used with Anaconda.

❖ Features include:

- editor with syntax highlighting and introspection for code completion
- support for multiple Python consoles (including IPython)
- the ability to explore and edit variables from a GUI

❖ Available plugins include:

- Static Code Analysis with Pylint
- Code Profiling
- Conda Package Manager with Conda

❖ Hardware Interfaces

1. Processor : Intel CORE i5 processor with minimum 2.9 GHz speed.
2. RAM : Minimum 4 GB.
3. Hard Disk : Minimum 500 GB

❖ Software Interfaces

1. Microsoft Word 2007
2. Database Storage : Microsoft Excel
3. Operating System : Windows10

1.4 Motivation:-

In previous time, JARVIS, a desktop assistant is a voice assistant that can perform many daily tasks of desktop like playing music, opening your favorite IDE with the help of a single voice command. Jarvis 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.

1.5 PROJECT FEATURES:-

The functionalities include , It can send emails, It can read PDF, It can send text on WhatsApp, It can open command prompt, your favorite IDE, notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can give weather forecast, It can give desktop reminders of your choice. It can have some basic conversation.

Tools and technologies used are PyCharm IDE for making this project, and I created all py files in PyCharm. Along with this I used following modules and libraries in my project. pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib, pywhatkit, pyjokes, pyPDF2, pyautogui, PyQt etc. I have created a live GUI for interacting with the JARVIS as it gives a design and interesting look while having the conversation.

CHAPTER-2

Literature Survey

2.1 Data Flow

DATA FLOW

The data flow for JARVIS is as follow:

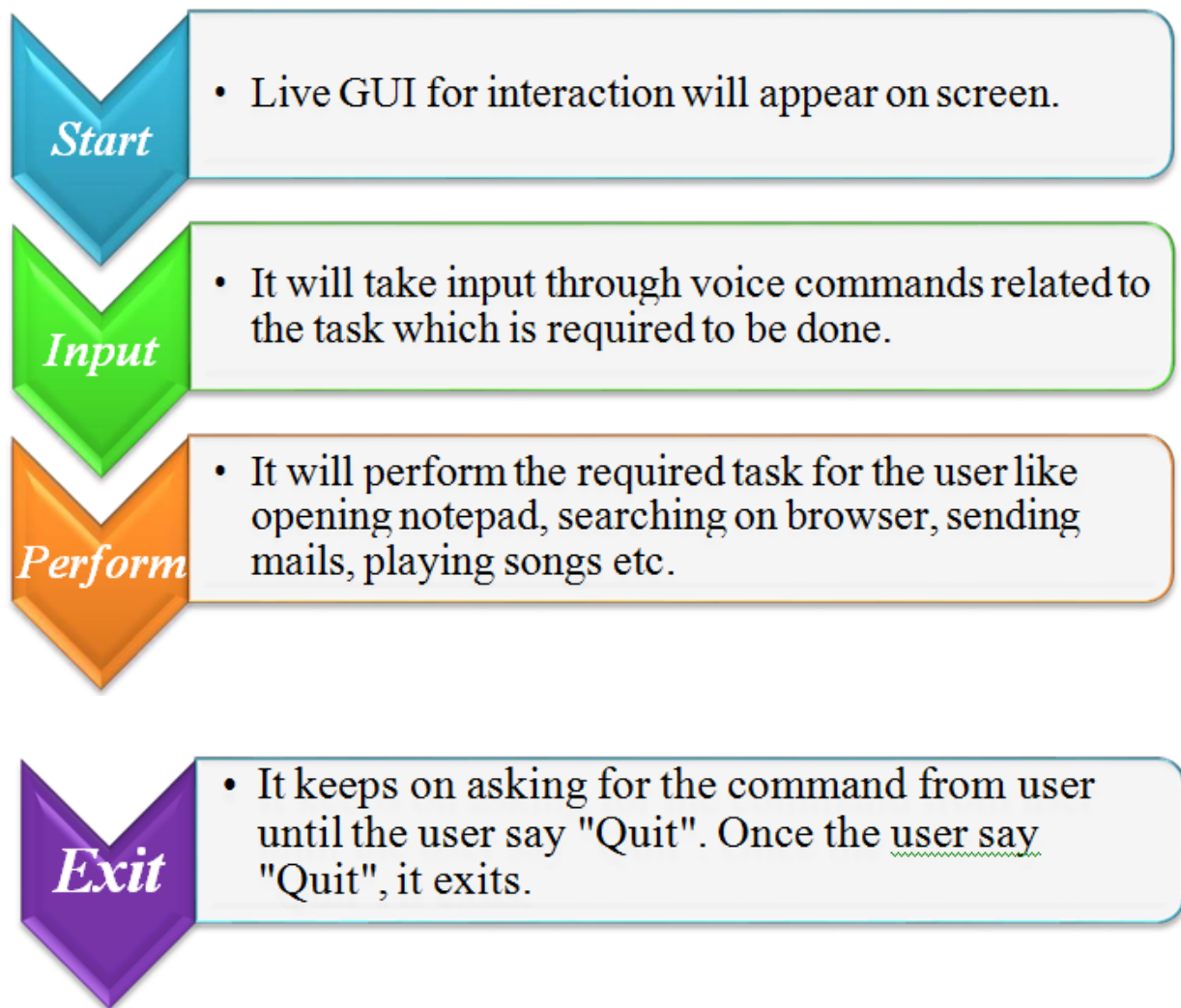


Figure Data flow for JARVIS

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 pyPDF2 can be used to read PDF. The details of these packages are mentioned in Chapter 3 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.

2.2 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 listens 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. These assistants are no less than a human assistant but we can say that they are more effective and efficient to perform any task. The algorithm used to make these assistant focuses on the time complexities and reduces time.

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.

2.3 PROPOSED SYSTEM:-

It was an interesting task to make my own assistant. It became easier to send emails without typing any word, Searching on Google without opening the browser, and performing many other daily tasks like playing music, opening your favorite IDE with the help of a single voice command. Jarvis 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 IDE used in this project is PyCharm. All the python files were created in PyCharm and all the necessary packages were easily installable in this IDE. For this project following modules and libraries were used i.e. pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib, pywhatkit, pyjokes, pyPDF2, pyautogui, PyQt etc. I have created a live GUI for interacting with the JARVIS as it gives a design and interesting look while having the conversation.

With the advancement JARVIS can perform any task with same effectiveness or can say more effectively than us. By making this project, I realized that the concept of AI in every field is decreasing human effort and saving time. Functionalities of this project include, It can send emails, It can read PDF, It can send text on WhatsApp, It can open command prompt, your favorite IDE, notepad etc., It can play music, It can do Wikipedia searches for you, It can open websites like Google, YouTube, etc., in a web browser, It can give weather forecast, It can give desktop reminders of your choice. It can have some basic conversation.

CHAPTER-3

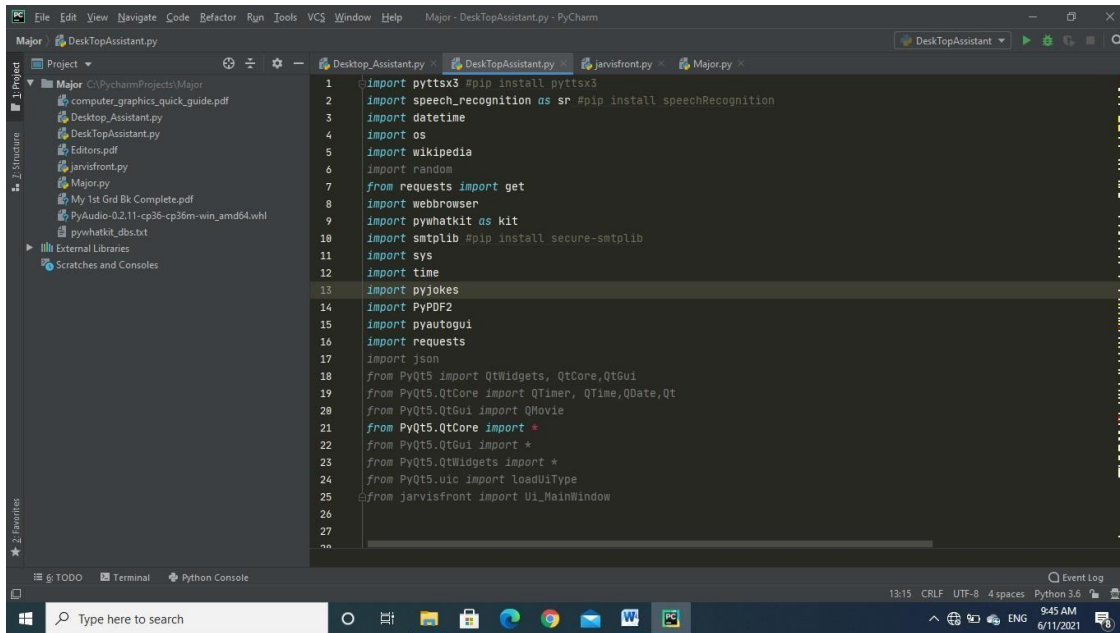
System Requirement

3.1 Introduction:-

The IDE used in this project is PyCharm. All the python files were created in PyCharm and all the necessary packages were easily installable in this IDE. For this project following modules and libraries were used i.e. pyttsx3, SpeechRecognition, Datetime, Wikipedia, Smtplib, pywhatkit, pyjokes, pyPDF2, pyautogui, PyQt etc. I have created a live GUI for interacting with the JARVIS as it gives a design and interesting look while having the conversation

3.1.1 PYCHARM:-

It is an IDE i.e. Integrated Development Environment which has many features like it supports scientific tools (like matplotlib, numpy, scipy etc) web frameworks (example Django, web2py and Flask) refactoring in Python, integrated python debugger, code completion, code and project navigation etc. It also provides Data Science when used with Anaconda.

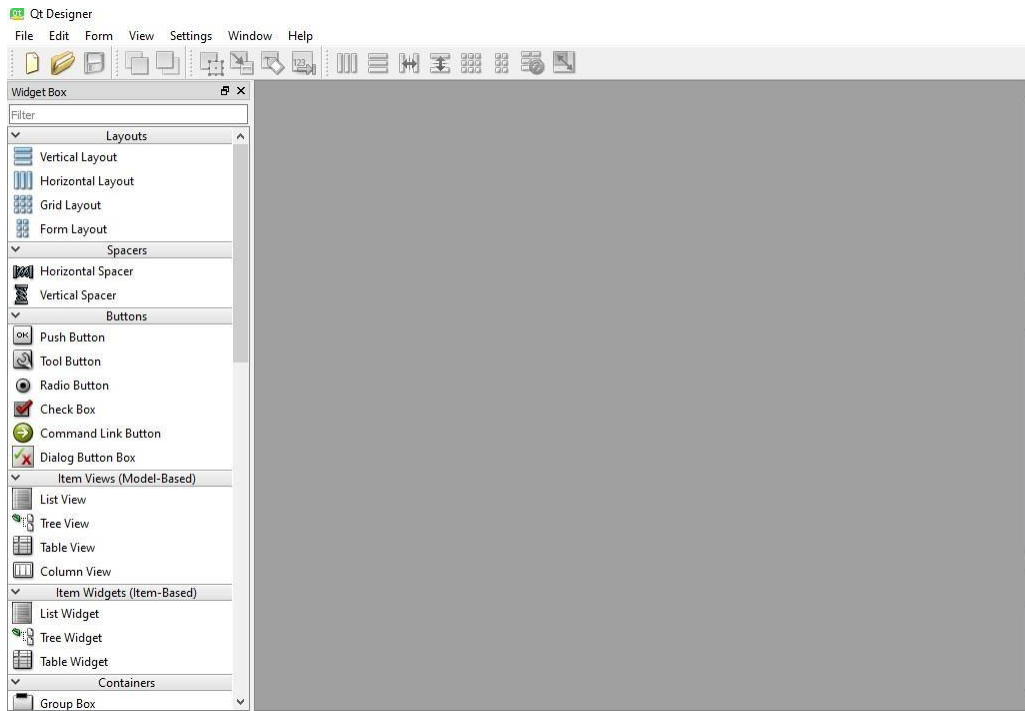


```
1 import pyttsx3 #pip install pyttsx3
2 import speech_recognition as sr #pip install speechRecognition
3 import datetime
4 import os
5 import wikipedia
6 import random
7 from requests import get
8 import webbrowser
9 import pywhatkit as kit
10 import smtplib #pip install secure-smtplib
11 import sys
12 import time
13 import pyjokes
14 import PyPDF2
15 import pyautogui
16 import requests
17 import json
18 from PyQt5 import QtWidgets, QtCore, QtGui
19 from PyQt5.QtCore import QTimer, QTime, QDate, Qt
20 from PyQt5.QtGui import QMovie
21 from PyQt5.QtCore import *
22 from PyQt5.QtGui import *
23 from PyQt5.QtWidgets import *
24 from PyQt5.uic import loadUiType
25 from jarvisfront import Ui_MainWindow
26
27
28
```

3.1.2 PYQT5 FOR LIVE GUI:-

PyQt5 is the most important python binding. It contains set of GUI widgets. PyQt5 has some important python modules like QtWidgets, QtCore, QtGui, and QtDesigner etc.

PyQt5 is the most important python binding. It contains set of GUI widgets. PyQt5 has some important python modules like QtWidgets, QtCore, QtGui, and QtDesigner etc.



PYTHON LIBRARIES:-

In JARVIS following python libraries were used:

- **Pyttsx3:** It is a python library which converts text to speech.
- **SpeechRecognition:** It is a python module which converts speech to text.
- **Pywhatkit:** It is python library to send WhatsApp message at a particular time with some additional features.
- **Wikipedia:** It is a python module for searching anything on Wikipedia.

- **Smtplib:** Simple mail transfer protocol that allows us to send mails and to route mails between mail servers..
- **PyPDF2:** It is a python module which can read, split, merge any PDF.
- **Pyjokes:** It is a python libraries which contains lots of interesting jokes in it.
- **Web Browser:** It provides interface for displaying web-based documents to users.
- **Pyautogui:** It is a python libraries for graphical user interface.
- **OS:** It represents Operating System related functionality.
- **System:** It allows operating on the interpreter as it provides access to the variables and functions that usually interact strongly with the interpreter.


```

1  import pyttsx3 #pip install pyttsx3
2  import speech_recognition as sr #pip install speechRecognition
3  import datetime
4  import os
5  import wikipedia
6  import random
7  from requests import get
8  import webbrowser
9  import pywhatkit as kit
10 import smtplib #pip install secure-smtplib
11 import sys
12 import time
13 import pyjokes|
14 import PyPDF2
15 import pyautogui
16 import requests
17 import json
18 from PyQt5 import QtWidgets, QtCore,QtGui
19 from PyQt5.QtCore import QTimer, QTime,QDate,Qt
20 from PyQt5.QtGui import QMovie
21 from PyQt5.QtCore import *
22 from PyQt5.QtGui import *
23 from PyQt5.QtWidgets import *
24 from PyQt5.uic import loadUiType
25 from jarvisfront import Ui_MainWindow

```

Imported Modules

CHAPTER-5

Implementation /Working of Project

JARVIS, a desktop assistant is a voice assistant that can perform many daily tasks of desktop like playing music, opening your favorite IDE with the help of a single voice command. Jarvis 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.

→ REAL LIFE APPLICATION :

Saves time: JARVIS 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.

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.

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.

Multitasking: The main application of it can be its multitasking ability. It can ask for continuous instruction one after other until the user “QUIT” it.

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.

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

LIBRARIES AND PACKAGES

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

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

pywhatkit: It is python library to send WhatsApp message at a particular time with some additional features.

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

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

Smtplib: Simple mail transfer protocol that allows us to send mails and to route mails between mail servers.

pyPDF2: It is a python module which can read, split, merge any PDF.

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

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

Pyautogui: It is a python library for graphical user interface.

OS: It represents Operating System related functionality.

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

FUNCTIONS

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

wishMe(): This function greets the user according to the time like Good Morning, Good Afternoon and Good Evening.

taskExecution(): This is the function which contains all the necessary task execution definition like sendEmail(), pdf_reader(), news() and many conditions in if condition like “open google”, “open notepad”, “search on Wikipedia”, “play music” and “open command prompt” etc.

CHAPTER-6
Result or Output

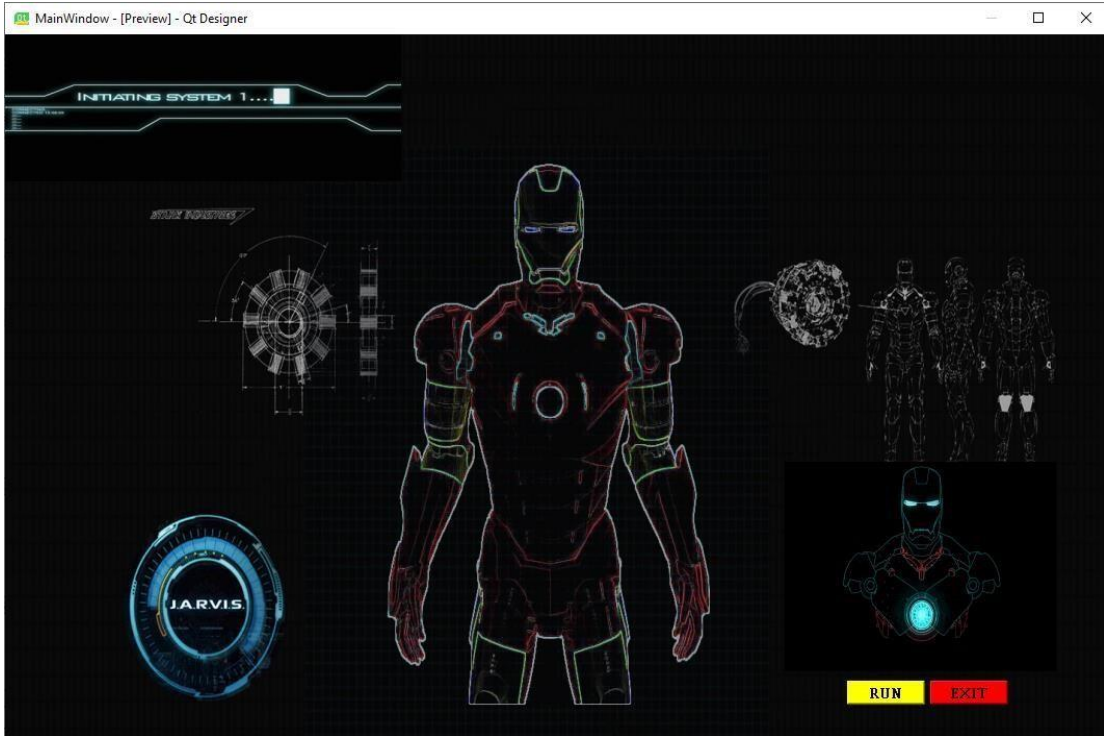


Figure 6.1 Live GUI of JARVIS

```
Run: Major x
C:\Python36\python.exe C:/PycharmProjects/Major/Major.py
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS_MS_EN-US_DAVID_11.0
Good Morning!
Hello!, I am Zira. Please tell me how may I help you
Listening...
Recognizing...
User said: open Google

Yamini, what should I search on google?
Listening...
Recognizing...
User said: what is python
```

Figure 6.2 Input for Google search

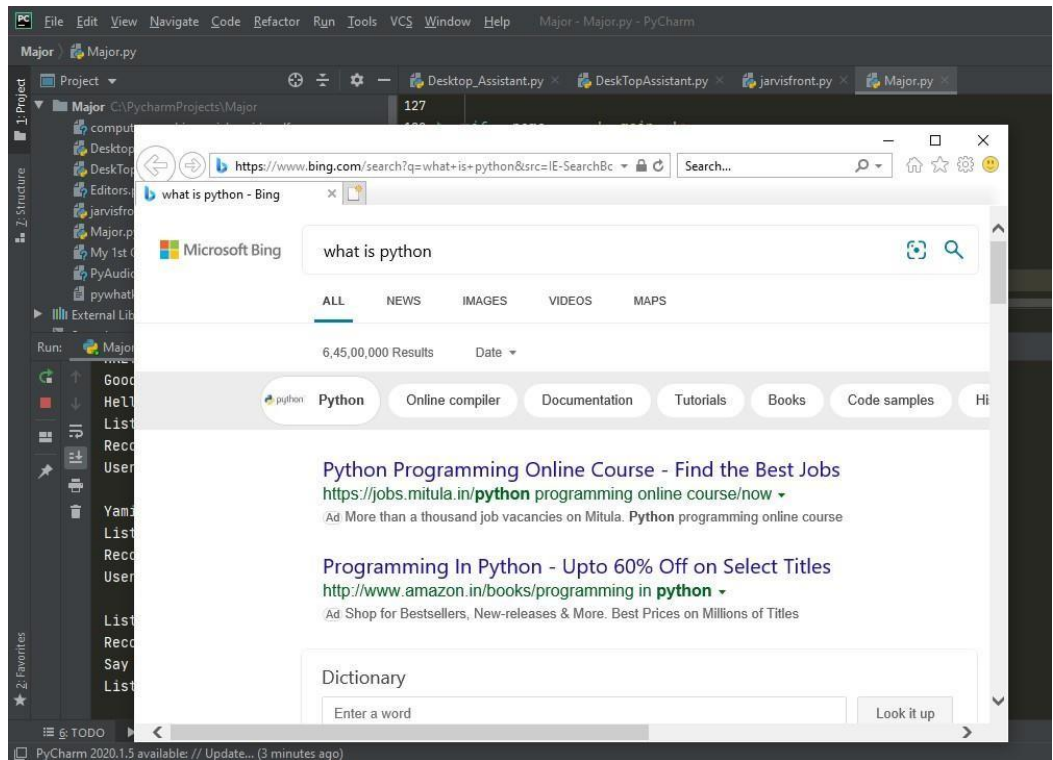


Figure 6.3 Output for Google search

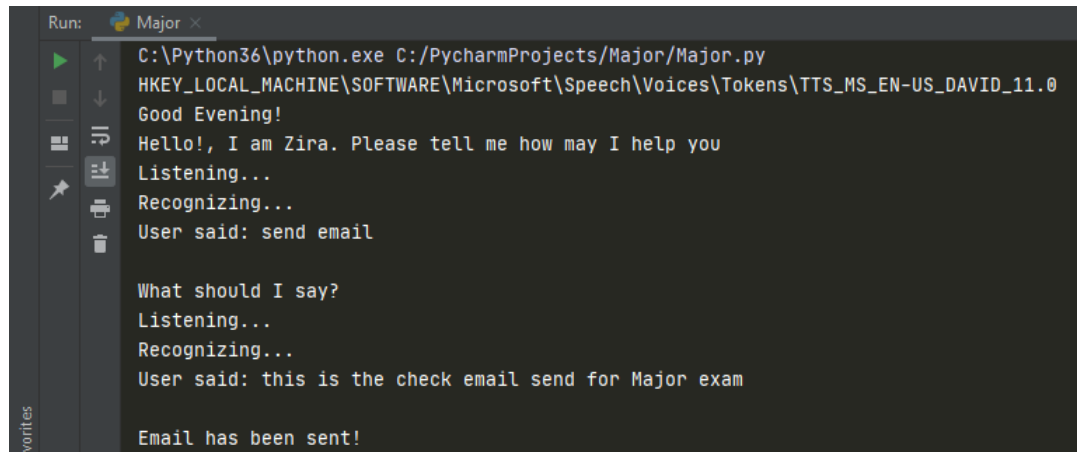


Figure 6.4 Input to send Email

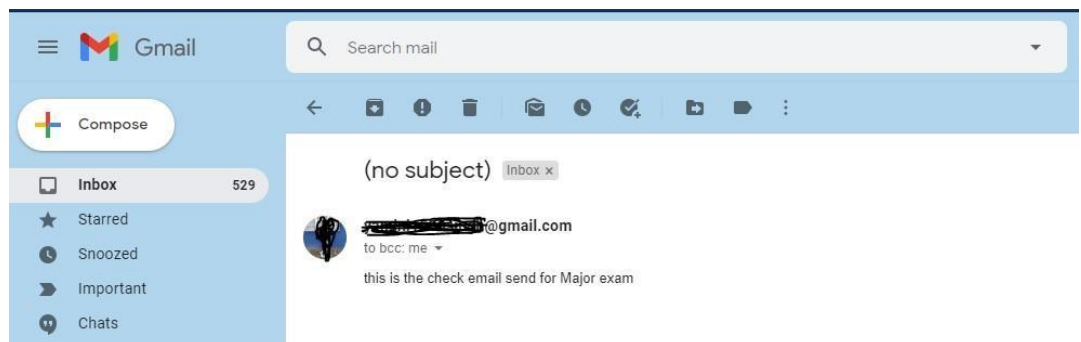


Figure 6.5 Output to send Email

```
Run: Major x
C:\Python36\python.exe C:/PycharmProjects/Major/Major.py
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS_MS_EN-US_DAVID_11.0
Good Evening!
Hello!, I am Zira. Please tell me how may I help you
Listening...
Recognizing...
User said: open YouTube

Yamini, what should I search on YouTube?
Listening...
Recognizing...
User said: calma song
```

Figure 6.6 Input for YouTube search

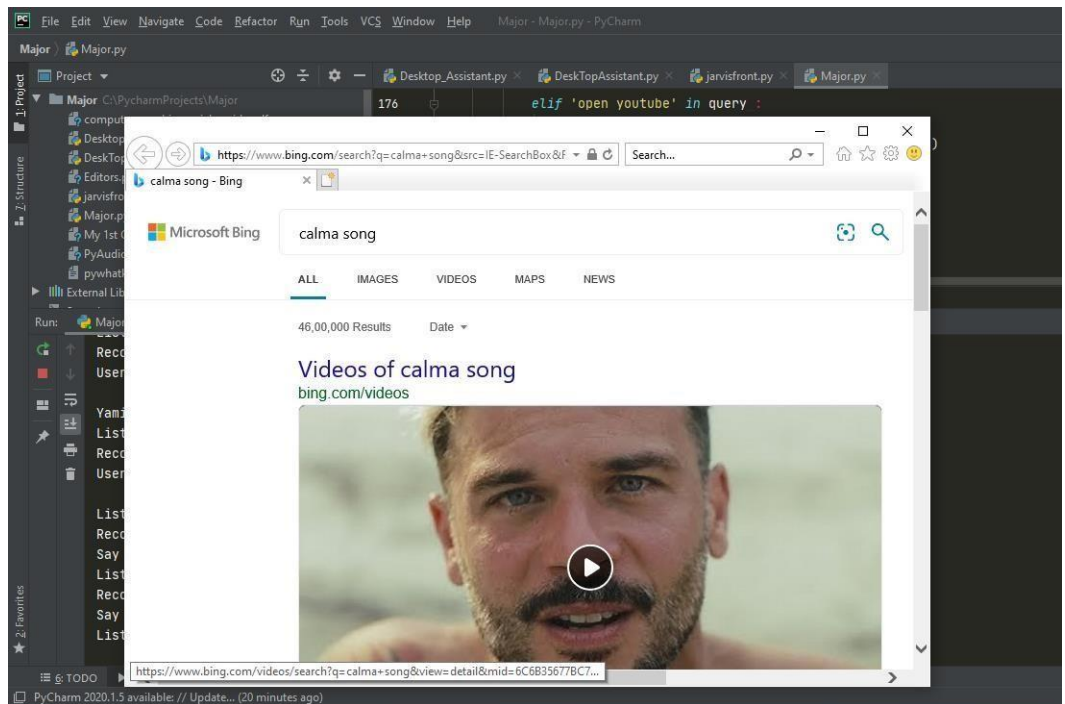


Figure 6.7 Output for YouTube search

```
Run: Major x
C:\Python36\python.exe C:/PycharmProjects/Major/Major.py
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS_MS_EN-US_DAVID_11.0
Good Evening!
Hello!, I am Zira. Please tell me how may I help you
Listening...
Recognizing...
User said: play music
```

Figure 6.8 Input to play music

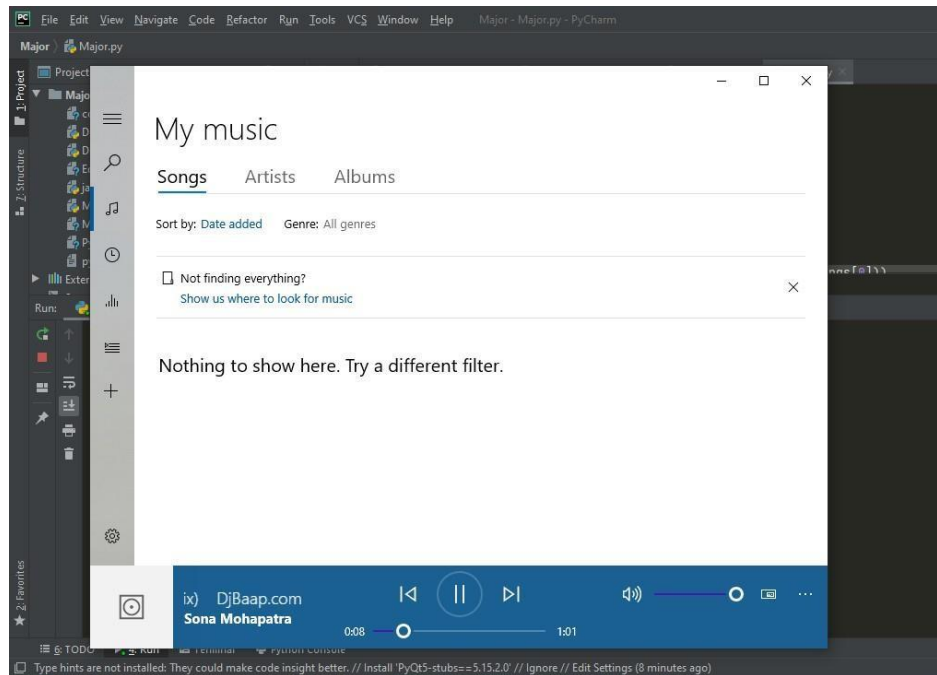


Figure 6.9 Output to play music

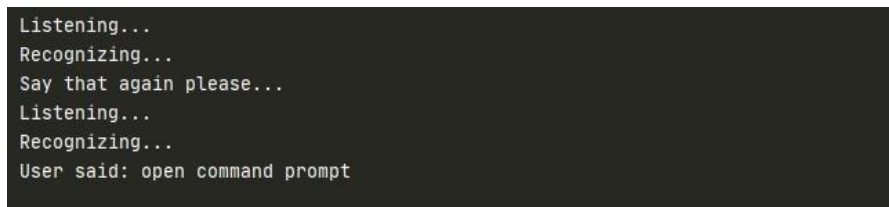


Figure 6.10 Input to open cmd

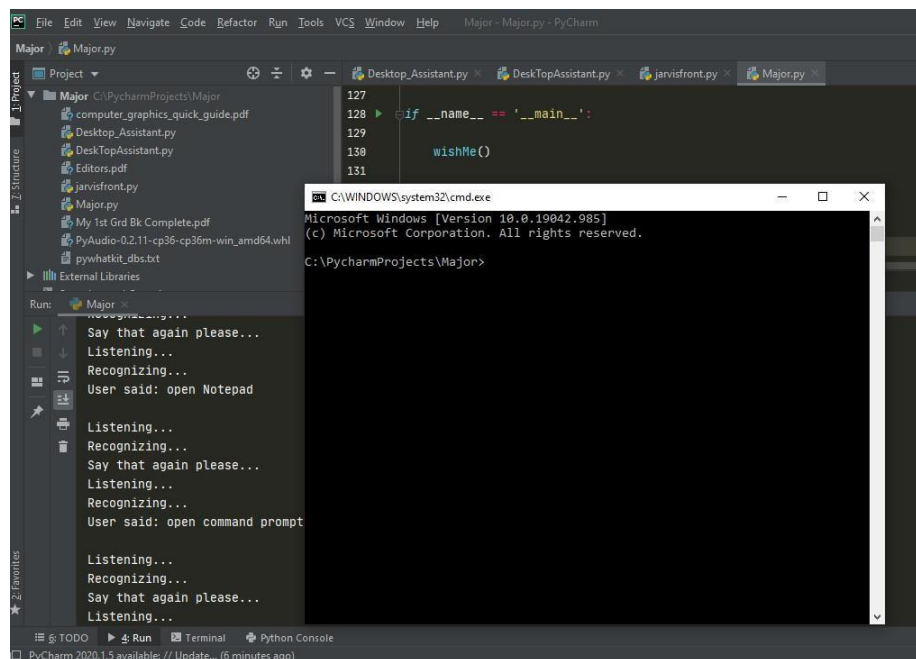


Figure 6.11 Output to open cmd

```

Run: Major x
C:\Python36\python.exe C:/PycharmProjects/Major/Major.py
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS_MS_EN-US_DAVID_11.0
Good Evening!
Hello!, I am Zira. Please tell me how may I help you
Listening...
Recognizing...
User said: search Python on Wikipedia

Searching Wikipedia...
Yamini, According to Wikipedia....
Python is an interpreted high-level general-purpose programming language. Python's design philosophy

```

Figure 6.12 Input and output for Wikipedia search

```

Run: Major x
C:\Python36\python.exe C:/PycharmProjects/Major/Major.py
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Speech\Voices\Tokens\TTS_
Good Evening!
Hello!, I am Zira. Please tell me how may I help you
Listening...
Recognizing...
User said: open Microsoft Office

```

Figure 6.13 Input to open Microsoft Office

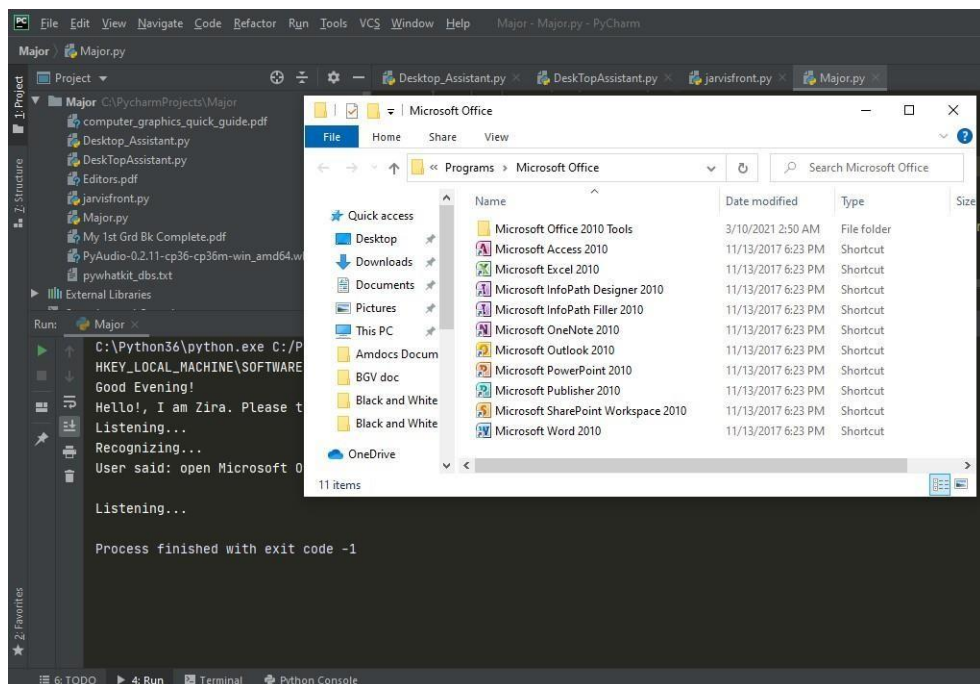


Figure 6.14 Output to open Microsoft Office

CHAPTER-7

Future Scope

The extraction of the best parametric representation of acoustic signals is an important task to produce a better recognition performance. The efficiency of this phase is important for the next phase since it affects its behavior. MFCC is based on human hearing perceptions which cannot perceive frequencies over 1Khz. In other words, in MFCC is based on known variation of the human ear's critical bandwidth with frequency [8-10]. MFCC has two types of filter which are spaced linearly at low frequency below 1000 Hz and logarithmic spacing above 1000Hz. A subjective pitch is present on Mel Frequency Scale to capture important characteristic of phonetic in speech.

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 JARVIS can search on Google or not, as we can see in the figure 7.1, user said "Open Google", then Jarvis asked, "What

should I search on Google?” then user said, “What is Python”, Jarvis open Google and searched for the required input.number.

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 “QUIT” it. 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.

SECURITY

The security testing mainly focuses on vulnerabilities and risks. As JARVIS 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.

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.

Chapter 8

Conclusion

JARVIS 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.

LIMITATIONS

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

Background voice can interfere

Misinterpretation because of accents and may cause inaccurate results.

JARVIS cannot be called externally anytime like other traditional assistants like Google Assistant can be called just by saying, “Ok Google!”

SCOPE FOR FUTURE WORK

Make JARVIS to learn more on its own and develop a new skill in it. JARVIS android app can also be developed.

Make more Jarvis voice terminals.

Voice commands can be encrypted to maintain security.

CHAPTER-9

REFERENCES

- [1] P. Pal, A. N. Iyer and R. E. Yantorno, "Emotion Detection From Infant Facial Expressions And Cries," *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, Toulouse, 2006,pp. II-II.
- [2] M. Dahmane and J. Meunier, "Emotion recognition using dynamic grid-based HoG features," *Face and Gesture 2011*, Santa Barbara, CA, 2011, pp.884-888.
- [3] K. S. Nathan, M. Arun and M. S. Kannan, "EMOSIC — An emotion based music player for Android," *2017 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Bilbao, 2017,pp. 371-276.
- [4] M. S. Hossain and G. Muhammad, "An Emotion Recognition System for Mobile Applications," in *IEEE Access*, vol. 5, pp.2281-2287, 2017.
- [5] K. Kulkarni *et al.*, "Automatic Recognition of Facial Displays of Unfelt Emotions," in *IEEE Transactions of Affective Computing*.

- [6] Y. Li, J. Tao, B. Schuller, S. Shan, D. Jiang and J. Jia, "MEC 2017: Multimodal Emotion Recognition Challenge," *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, Beijing, 2018, pp.1-5
- [7] Y. Seanglidet, B. S. Lee and C. K. Yeo, "Mood prediction from facial video with music “therapy” on a smartphone," *2016 Wireless Telecommunications Symposium (WTS)*, London, 2016, pp. 1-5.
- [8] A. V. Iyer, V. Pasad, S. R. Sankhe and K. Prajapati, "Emotion based mood enhancing music recommendation," *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, Bangalore, 2017, pp. 1573-1577.
- [9] T. Moriyama, K. Abdelaziz and N. Shimomura, "Face analysis of aggressive moods in automobile driving using mutual subspace method," *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, Tsukuba, 2012, pp. 2898-2901
- [10] M. Pantic and L. J. M. Rothkrantz, "Automatic analysis of facial expressions: the state of the art," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424-1445, Dec. 2000.

→ **URL:**

- [http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=895976&isn
umber=19391](http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=895976&isn
umber=19391)
- <https://ieeexplore.ieee.org/Xplore/home.jsp>
- https://en.wikipedia.org/wiki/Main_Page
- [https://www.google.com.bd/?gws_rd=cr&ei=sK0XWLPAEcvmvgT12o
7AAg](https://www.google.com.bd/?gws_rd=cr&ei=sK0XWLPAEcvmvgT12o
7AAg)
- Etc.....