A

Mini Project Report on

**INTRODUCING FRIDAY**

Submitted for partial fulfilment of the requirements for the award of the degree of

**BACHELOR OF ENGINEERING**

in

**INFORMATION TECHNOLOGY**

by

**Shiva Manish Gajadi** (2451-21-737-009)

**Sai Vishal Meka (**2451-21-737-042)

**Tanish Pusthe** (2451-21-737-043)

Under the guidance of

**MS. N. Nithya Lakshmi**

Assistant Professor

Department of IT



**MATURI VENKATA SUBBA RAO (M.V.S.R) ENGINEERING COLLEGE**

**(An Autonomous Institute)**

Department of Information Technology

(Affiliated to Osmania University & Recognized by AICTE) Nadergul, SaroorNagar Mandal, Hyderabad – 501 510

AY: 2022-23



# *Certificate*

*This is to certify that the mini project work entitled “Introducing Friday” is a bonafide work carried out by Mr. Shiva Manish Gajadi (2451-21-737-009), Mr. Sai Vishal Meka (2451-21-737-042), Mr. Tanish Pusthe(2451-21-737-043) in partial fulfilment of the requirements for the award of degree of Bachelor of Engineering in Information Technology from Maturi Venkata Subba Rao (M.V.S.R.) Engineering College, an Autonomous Institution, affiliated to Osmania University Hyderabad, during the Academic Year 2022-23 under our guidance and supervision.*

*The results embodied in this report have not been submitted to any other university or institute for the award of any degree or diploma.*

**Internal Guide Project Co-ordinator**

**MS. N. Nithya Lakshmi Dr. H. Jayasree**

Assistant Professor Department of IT

Department of IT MVSREC

MVSREC

**HOD External Examiner**

**Dr. K. Venugopal Rao**

Dean – Academics, Professor & Head

Department of IT

MVSREC

# DECLARATION

This is to certify that the work reported in the present mini project entitled “**Introducing Friday**” is a record of bonafide work done by us in the Department of Information Technology, Maturi Venkata Subba Rao (M.V.S.R.) Engineering College, an Autonomous Institution, affiliated to Osmania University. The reports are based on the project work done entirely by us and not copied from any other source. The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of our knowledge and belief.

Shiva Manish Gajadi Sai Vishal Meka Tanish Pusthe

(2451-21-737-009) (2451-21-737-042) (2451-21-737-043)

# ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our Project Guide **Ms. N. Nithya Lakshmi** Assistant Professor for his valuable suggestions and interest throughout the course of this project.

We would like to express our sincere gratitude and indebtedness to our Project Co-ordinator **Dr. H. Jayasree**,Professor for her valued recommendations and attention throughout the course of this project.

We are also thankful to our Principal **Dr. G. Kanaka Durga**, **Professor** and **Dr. K.**

**Venugopal Rao,** Professor and Head - Department of Information Technology, DeanAcademics, Maturi Venkata Subba Rao Engineering College, for providing excellent infrastructure and a nice atmosphere for completing this project successfully as a part of our B.E. Degree (IT).

We convey our heartfelt thanks to the lab staff for allowing us to use the required equipment whenever needed.

Finally, we would like to take this opportunity to thank our family for their support through the work. We sincerely acknowledge and thank all those who gave directly or indirectly their support in completion of this work.

Shiva Manish Gajadi (2451-21-737-009)

Sai Vishal Meka (2451-21-737-042)

Tanish Pusthe (2451-21-737-043)

# VISION & MISSION,

**PROGRAM EDUCATIONAL OUTCOMES**

**Vision of the Department:**

To impart technical education producing competent and socially responsible engineering professionals in the field of Information Technology.

**Mission of the Department:**

M1. To make teaching learning process effective and stimulating.

M2. To provide adequate fundamental knowledge of sciences and Information Technology with positive attitude.

M3. To create an environment that enhances skills and technologies required for industry.

M4. To encourage creativity and innovation for solving real world problems.

M5. To cultivate professional ethics in students and inculcate a sense of responsibility towards society

**Program Educational Objectives**:

After 3 to 4 years of graduation, graduates of the Information Technology program will:

1. Apply knowledge of mathematics and Information Technology to analyze, design and implement solutions for real world problems in core or in multidisciplinary areas.
2. Communicate effectively, work in a team, practice professional ethics and apply knowledge of computing technologies for societal development.
3. Engage in Professional development or postgraduate education to be a life-long learner.

**PROGRAM OUTCOMES & PROGRAM SPECIFIC OUTCOMES**

**PROGRAM OUTCOMES (POs)**

**Engineering Graduates will be able to:**

* 1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineeringfundamentals, and an engineering specialization to the solution of complex engineering problems.
  2. **Problem analysis**: Identify, formulate, review research literature, and analyze complexengineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
  3. **Design/development of solutions**: Design solutions for complex engineering problems anddesign system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
  4. **Conduct investigations of complex problems**: Use research-based knowledge and researchmethods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
  5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modernengineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
  6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assesssocietal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
  7. **Environment and sustainability**: Understand the impact of the professional engineering solutionsin societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
  8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms ofthe engineering practice.
  9. **Individual and teamwork**: Function effectively as an individual, and as a member or leader indiverse teams, and in multidisciplinary settings.
  10. **Communication**: Communicate effectively on complex engineering activities with the engineeringcommunity and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
  11. **Project management and finance**: Demonstrate knowledge and understanding of theengineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
  12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage inindependent and life-long learning in the broadest context of technological change.

**PROGRAM SPECIFIC OUTCOMES (PSOs):**

PSO1: **Hardware design**: An ability to analyze, design, simulate and implement computer hardware / software and use basic analog/digital circuits, VLSI design for various computing and communication system applications.

PSO2: Software **design**: An ability to analyze a problem, design algorithm, identify and define the computing requirements appropriate to its solution and implement the same.

# COURSE OBJECTIVES & COURSE OUTCOMES

**COURSE OBJECTIVES**

* To familiarize phases of SDLC.
* To encourage students to work with innovative ideas to address simple societal issues.
* To motivate self-learning to explore new domains & frameworks.
* To make students to work in a team.

**COURSE OUTCOMES**

CO1: Identify a project topic of societal interest after surveying.

CO2: Analyse requirements and feasibility to implement the project.

CO3: Design the project in hardware and/or software.

CO4: Develop, implement, and test the project/ model / prototype.

CO5: Demonstrate effective written and oral communication skills.

# ABSTRACT

Now-a-days robots are playing a very important role in the industry level and also out of the industry. Dependency on robots is increasing for their fast and reliable working speed and accuracy. Considering that demand of the robot increasing every day. This research was conducted focusing on the necessity of robots in our daily life. This proposes a system where a robot can be controlled in different ways like voice, wireless and fully automatic mode. The prototype was built and tested. The robot prototype will be able to receive voice command from short distance.The high standards of living have encouraged automation to come on the van and be an integral part of home design. At the same time, the environmental relation has ensured that energy-efficient housing models and appliances are used. So, we proposed the system called FRIDAY, a robot which can do tasks .Our project aim is to develop a robot which can recognize the voice and perform tasks according to the command. It can be controlled manually remotely by android smartphone application or any internet enabled device through internet. So that it can be used by anyone and specially can be helpful for physically disable persons.

Now a day the high standards of living have encouraged automation to come

on the van and be an integral part of home design. At the same time, the

environmental relation has ensured that energy-efficient housing models and

appliances are used. So, we proposed the system called homey, a robot which

can do tasks on behalf of a person at home.

Our project aim is to develop a robot which can recognize the voice and

perform tasks according to the command. It can be controlled manually

remotely by android smartphone application or any internet enabled device

through internet. So that it can be used by anyone and specially can be helpful

for physically disable persons.

Now a day the high standards of living have encouraged automation to come

on the van and be an integral part of home design. At the same time, the

environmental relation has ensured that energy-efficient housing models and

appliances are used. So, we proposed the system called homey, a robot which

can do tasks on behalf of a person at home.

Our project aim is to develop a robot which can recognize the voice and

perform tasks according to the command. It can be controlled manually

remotely by android smartphone application or any internet enabled device

through internet. So that it can be used by anyone and specially can be helpful

for physically disable persons.

Now a day the high standards of living have encouraged automation to come

on the van and be an integral part of home design. At the same time, the

environmental relation has ensured that energy-efficient housing models and

appliances are used. So, we proposed the system called homey, a robot which

can do tasks on behalf of a person at home.

Our project aim is to develop a robot which can recognize the voice and

perform tasks according to the command. It can be controlled manually

remotely by android smartphone application or any internet enabled device

through internet. So that it can be used by anyone and specially can be helpful

for physically disable persons.

# TABLE OF CONTENTS

**CONTENTS**  **PAGE NO**

Certificate ii

Declaration iii

Acknowledgement iv

Vision & Mission, PEOs v

POs and PSOs vi

Course Objectives & Outcomes vii

Abstract viii

Table of Contents ix

List of Figures x

List of Tables xi

List of Abbreviations xi

**Chapters**

1. **Introduction** 1 - 4
   1. Problem statement
   2. Existing system
   3. Proposed system
   4. Scope of Study
2. **System Requirement Specifications** 5 - 6
   1. Software Requirements
   2. Hardware Requirements
   3. System Architecture / Block Diagram
3. **Design & Implementation** 7 - 12
   1. Design
   2. Environmental Setup
4. **Test & Results**  13 - 17
   1. Test cases
   2. Results
5. **Conclusion & Future Enhancements**  18

**References**  19

1. **Appendix A**: source/pseudo code 20-21

# List of Figures

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No** |
| 1.1 | Existing system | 3 |
| 2.1 | Flow chart of proposed system | 6 |
| 3.2.1 | Python installation | 9 |
| 3.2.2 | PyCharm download | 10 |
| 3.2.3 | PyCharm overview | 11 |
| 3.2.4 | Installation of modules | 11 |
| 3.2.5 | Streamlit application setup | 12 |

# List of Tables

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Table Name** | **Page No.** |
| 4.1 | System functional testing results | 14 |
| 4.2 | Emotions saved in proposed system | 15 |
| 4.3 | Testing results for neutral expression | 16 |
| 4.4 | Testing results for sad expression | 16 |
| 4.5 | Testing results for surprise expression | 17 |
| 4.6 | Testing results for happy expression | 17 |
| 4.7 | Summary of results tested | 18 |

**CHAPTER 1**

**INTRODUCTION**

Now-a-days robots are playing a very important role in the industry level and also out of the industry. Dependency on robots is increasing for their fast and reliable working speed and accuracy. Considering that demand of the robot increasing every day. This research was conducted focusing on the necessity of robots in our daily life. This proposes a system where a robot can be controlled in different ways like voice, wireless and fully automatic mode. The prototype was built and tested. The robot prototype will be able to receive voice command from short distance. In case of long-distance communication user will be able to connect through internet using IoT.

Now a day the high standards of living have encouraged automation to come on the van and be an integral part of home design. At the same time, the environmental relation has ensured that energy-efficient housing models and appliances are used. So, we proposed the system which is homey, a robot which can do tasks on behalf of a person at home. Our project aim is to develop a robot which can recognize the voice and perform tasks according to the command. It can be controlled by any internet enabled device through internet. So that it can be used by anyone and specially can be helpful for physically disable persons. Now a day the high standards of living have encouraged automation to come on the van and be an integral part of home design. At the same time, the environmental relation has ensured that energy-efficient housing models and appliances are used.

So, we proposed the system called homey, a robot which can do tasks on behalf of a person at home. Our project aim is to develop a robot which can recognize the voice and perform tasks according to the command. It can be controlled manually remotely by android smartphone application or any internet enabled device through internet. So that it can be used by anyone and specially can be helpful for physically disable persons. Now a day the high standards of living have encouraged automation to come on the van and be an integral part of home design. At the same time, the environmental relation has ensured that energy-efficient housing models and appliances are used. So, we proposed the system called homey, a robot which can do tasks on behalf of a person at home.

The high standards of living have encouraged automation to come on the van and be an integral part of home design. At the same time, the environmental relation has ensured that energy-efficient housing models and appliances are used. So, we proposed the system called FRIDAY, a robot which can do tasks on behalf of a person at home.

**1.1 PROBLEM STATEMENT**

There is a great difficulty in managing household activities which is mostly asynchronized. Taking in mind all the technological break through and development in IOT, how would you manage household activities in a better way?

**1.2 EXISTING SYSTEM**

Voice assistant is a software that carries out everyday tasks via versa command. It brings AI and ML together to recognise our voice. It can be found on all devices like smart speakers, phones, watches etc.

**1.3 PROPOSED SYSTEM**

This proposed system majorly focuses on introducing Friday, it can read and respond to physical world even having human like responses to commands. By its appearance it might look like a mechanical child’s toy but it is more than just a novelty.

**1.4 SCOPE OF STUDY**

Friday is AI powerful robot. It can be used to manage daily activities like setting up alarms, to automatically go to websites, YouTube or browse the internet just by altering simple commands. It can also on and off the applications.

**CHAPTER 2**

**SYSTEM REQUIREMENT SPECIFICATIONS**

**2.1 SOFTWARE REQUIREMENTS**

1. Programming language

Python idle 3.10.7 2022

* + - Main development software which includes interface design and coding is Vscode editor
    - To development of project and easy access to build a python project.
    - It is compatible with major Python libraries such as speech\_recognition, pyttsx3, pyfirmata, tkinter , ect

**2.2 Hardware requirements**

1. Laptop
2. Arduino nano
3. servo motor
4. relay module
5. 3D printed parts
6. Connecting wires

**2.3 System architecture**

**Flow chart**

+-------------+

| |

| Initiating |

| All Systems |

| |

+-+---------+--+

| |

+-----v-------+

| |

| Import Libraries|

| |

+----------------+

| |

+-----v-------+

| |

|Initialize Arduino|

| |

+----------------+

| |

+-----v-------+

| |

|Initialize Servos|

| |

+----------------+

| |

+-----v-------+

| |

|Initialize Voice|

| |

+----------------+

| |

+-----v-------+

| |

| Loop |

| |

+----------------+

| |

+-----v-------+

| |

| Listen |

| |

+----------------+

| |

+-----v-------+

| |

| Respond |

| |

+----------------+

Fig -2.1 - FLOW CHART OF PROPOSED SYSTEM

The code first imports the necessary libraries and initializes the Arduino board and servo motors. Then it initializes the voice recognition and synthesis. The program then enters a loop where it constantly listens for commands and responds accordingly. The specific commands that are recognized and the corresponding actions taken are defined in the functions such as "dance", "hi", and "searchGoogle".

**CHAPTER 3**

**Design & Implementation**

**3.1 Features**

**Speech recognition module**

Speech recognition module in Python is a library that enables the recognition of spoken words in a digital audio file or microphone input. This module can be used for a variety of tasks such as transcribing speech to text, performing voice commands, and more. There are various libraries available for speech recognition in Python such as CMU Sphinx, Google Speech Recognition, and others. These libraries are easy to use and can be integrated with other Python programs. They use complex algorithms to process the audio input and translate it into text. Some libraries also support multiple languages and can handle different accents and dialects.

**Implementation**

**3.2 Environmental setup of Visual studio code**

To download and install the Visual Studio Code (VSCode) editor, you can follow these steps:

1. Go to the official VSCode website: https://code.visualstudio.com/
2. Click on the "Download" button on the top right corner of the page.
3. Select the appropriate version of VSCode for your operating system (Windows, Mac, or Linux)
4. Follow the prompts to complete the installation process.
5. Once the installation is complete, you can open VSCode by searching for it in your start menu or applications and start using it for your development projects.

Note: you can also download the vs code from the official app store of your operating system.

It's also good to mention that VSCode is a lightweight, cross-platform code editor that is free and open-source. It's well-documented, customizable, and extensible.

**Visual Studio Code (VSCode) editor for Python development**

To set up the Visual Studio Code (VSCode) editor for Python development, you will need to do the following:

1. Install the Python extension for VSCode by searching for "Python" in the Extensions marketplace and clicking on the "Python" extension by Microsoft.
2. Install Python on your system if it is not already installed by downloading it from the official Python website and following the installation instructions.
3. Open the Command Palette in VSCode (Ctrl + Shift + P) and type "Python: Select Interpreter" to select the installed Python interpreter.
4. Configure the Python environment by creating a virtual environment or using an existing one. You can use the "Python: Create Virtual Environment" command to create a new virtual environment.
5. Create a new file with a .py extension and start writing Python code. You can run your code using the "Run Python File in Terminal" command.
6. Install any necessary modules or packages using pip by opening the terminal in VSCode and running pip install [package name].
7. To debug the python code, you can install the Python extension for Visual Studio Code in order to get a better debugging experience.

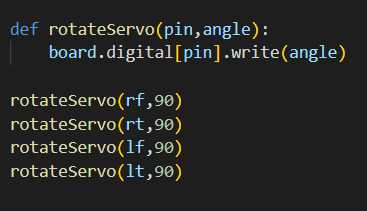
By following these steps, you will have a fully functional Python development environment set up in VSCode.

**CHAPTER 4**

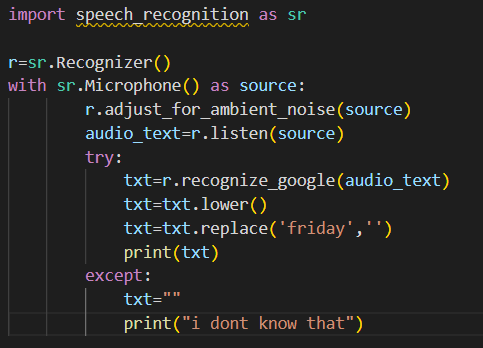
**TESTS AND RESULTS**

**4.1 Test cases**

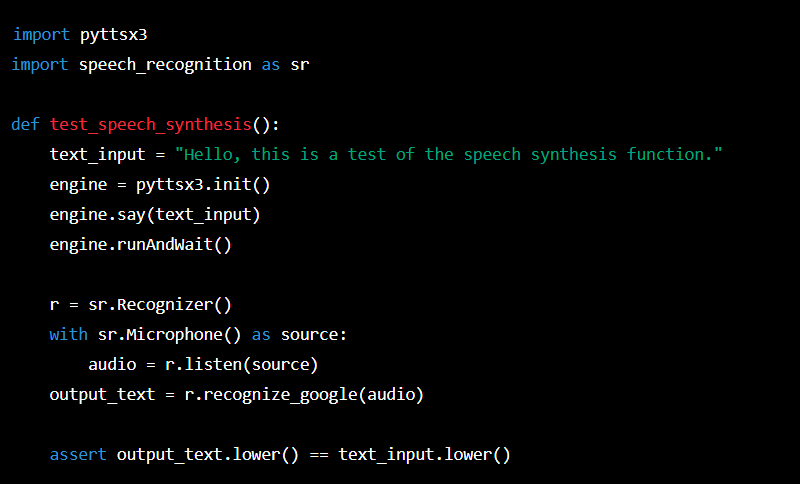
1. Test if the Servos are rotating as expected on the given angles: we first method to rotate the servo as per the given pin and angle



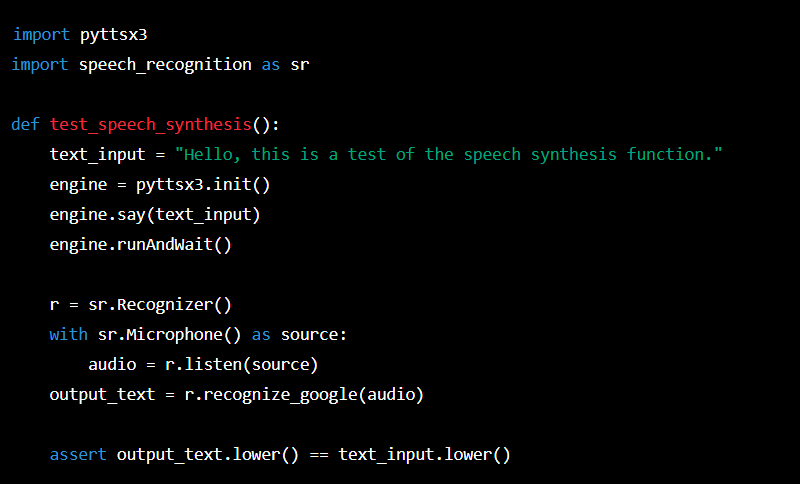
1. Test if the speech recognition is working as expected:we first create an instance of the Recognizer class, then use the microphone as the audio source. The script will prompt the user to speak, and it will then try to recognize the speech using the Google Speech Recognition API. If the speech recognition is successful, it will print the recognized text to the console and the assert statement will check if the recognized text is not none. If the speech recognition is unsuccessful, it will raise an exception and the assert statement will fail.



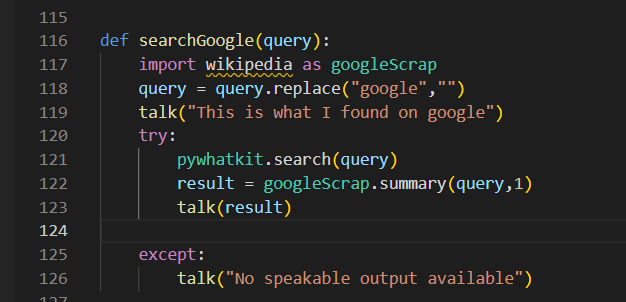
3)Test if the speech synthesis is working as expected: we first use pyttsx3 to synthesize speech from the text\_input, then use the speech\_recognition library to listen to the audio output using the microphone and try to recognize the speech and store it in output\_text variable. Finally, we compare the output\_text with the original text\_input

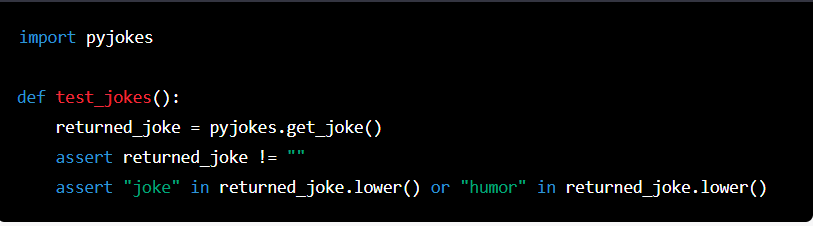


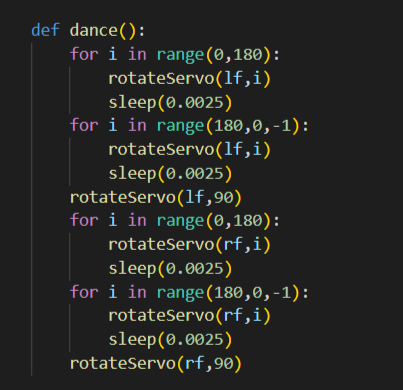
4)Test if the wikipedia search is working as expected: **wikipedia.summary(search\_term, sentences=2)** is a function from the wikipedia-api package which will search for the provided **search\_term** on wikipedia and return summary of that term in 2 sentences. Then we store the returned summary in the **result** variable and compare it with the ‘**expected\_summary’** variable to check if the returned summary is as expected or not.



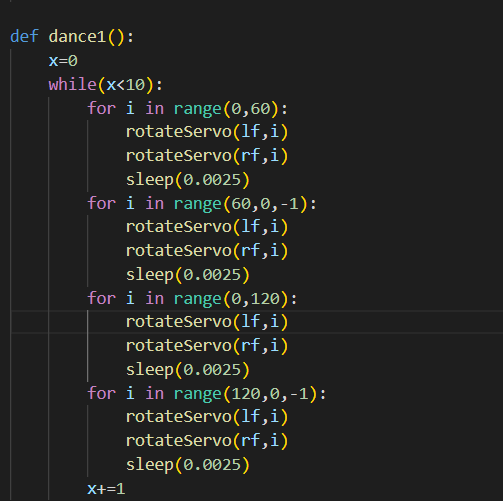
5)Test if the Google search is working as expected: the **search\_and\_verify** function takes in one argument, the **search\_query**. It uses the **pywhatkit.search(search\_query)** function to search for the **search\_query** on google. Then it uses **pywhatkit.get\_top\_result()** function to get the top result link from the search result. Then it uses requests library to get the HTML content of the top result link and parse it using BeautifulSoup.



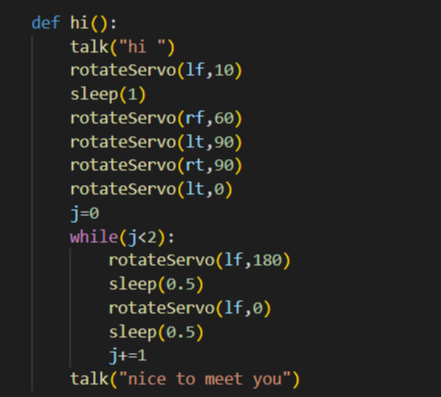
6)Test if the jokes function is working as expected:**pyjokes.get\_joke()** is a function from the pyjokes library that returns a random joke. The first assert statement checks if the returned joke is not empty and the second assert statement checks if the returned joke contains the word "joke" or "humor" to ensure that it is a valid joke.

7)Test if the dance function is working as expected: You can create a test function that calls the dance function, then use a servo tester or an oscilloscope to check if the servos are moving as expected.

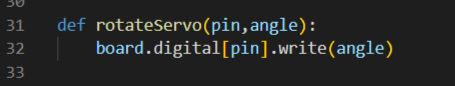
1. Test if the dance1 function is working as expected: You can create a test function that calls the dance1 function, then use a servo tester or an oscilloscope to check if the servos are moving as expected.



9)Test if the hi function is working as expected: You can create a test function that calls the hi function, then check if the output is as expected.



10)Test if the led1 is working a expected: You can create a test function that calls the led1 function, then check if the output is as expected.



**4.2 Result**

This is a Python script that utilizes various libraries to perform different tasks such as speech recognition, text-to-speech, and controlling an Arduino board.

The script uses the "speech\_recognition" library to recognize spoken words, "pyttsx3" for text-to-speech, "pywhatkit" for web scraping and searching the internet, "datetime" for working with date and time, "wikipedia" for getting information from Wikipedia, "pyjokes" for telling jokes, "threading" for multithreading, "pyfirmata" for communicating with an Arduino board and "tkinter" for creating graphical user interfaces.

It uses the arduino board to control the servo motors and digital pin to control the leds. The script also provides some predefined functions such as 'dance', 'dance1', 'hi' and 'searchGoogle'.

The script is using a while loop to keep listening to the audio input and perform the actions based on the command given.

**CHAPTER 5**

**Conclusion & Future Enhancements**

**5.1 CONCLUSION**

Voice-based robot assistants are becoming increasingly popular as they provide a convenient and efficient way for users to interact with technology. They use natural language processing (NLP) and speech recognition technology to understand and respond to spoken commands and requests. These assistants can be integrated into a variety of devices, such as smartphones, smart speakers, and home appliances, to perform a wide range of tasks, including setting reminders, playing music, and controlling smart home devices. Overall, voice-based robot assistants have the potential to greatly enhance the user experience by making it easier to interact with technology and access information. However, one of the challenge is to make it more natural, accurate, and secure.

**5.2 Future Enhancements**

The development of voice-based robot assistants has a lot of scope and potential in various industries. Here are a few examples:

1. Home Automation: Voice-based robot assistants can be used in homes to control appliances, lights, and other devices using voice commands. This can improve the convenience and accessibility of home automation for people with disabilities or mobility issues.
2. Healthcare: Voice-based robot assistants can be used in hospitals and care facilities to assist with tasks such as medication reminders, scheduling appointments, and providing information about a patient's condition.
3. Customer Service: Companies can use voice-based robot assistants to handle customer service inquiries and provide information about products or services. This can improve the efficiency and availability of customer service.
4. Education: Voice-based robot assistants can be used in classrooms to assist with teaching and learning. They can provide information, answer questions, and help with research.
5. Entertainment: Voice-based robot assistants can be used to play music, tell jokes, and provide other forms of entertainment.
6. Personal Assistance: Voice-based robot assistants can be used to help with scheduling, reminders, and other tasks that would typically be done by a personal assistant.
7. In industry: Voice based assistants can be used in industry to control machines, monitor the data, and other operations where human intervention is not possible or dangerous.

These are just a few examples of the many potential uses for voice-based robot assistants. As technology continues to improve, the possibilities for their use are likely to expand even further.

**REFERENCES**

**APPENDIX**

import speech\_recognition as sr import pyttsx3 import pywhatkit import datetime

import wikipedia import pyjokes import threading

from pyfirmata import Arduino,SERVO from time import sleep

from tkinter import \*

root =Tk()

port='COM10'

board=Arduino(port)

lf=6

lt=7

rf=8

rt=9

board.digital[lf].mode=SERVO

board.digital[lt].mode=SERVO

board.digital[rf].mode=SERVO

board.digital[rt].mode=SERVO

led1=board.get\_pin("d:4:o")

led1.write(1)

#path1= "C:\\Users\\HP\\OneDrive\\Desktop\\miniproject\\images\\normal\_2.jpeg"

def rotateServo(pin,angle):

board.digital[pin].write(angle)

rotateServo(rf,90)

rotateServo(rt,90)

rotateServo(lf,90)

rotateServo(lt,90)

r=sr.Recognizer()

text\_speech=pyttsx3.init()

engine = pyttsx3.init()

rate = engine.getProperty('rate') # getting details of current speaking rate

engine.setProperty('rate',180) # setting up new voice rate

voices = engine.getProperty('voices')

engine.setProperty('voice', voices[0].id)

def talk(text):

engine.say(text)

engine.runAndWait()

talk("Initiating all systems")

def dance():

for i in range(0,180):

rotateServo(lf,i)

sleep(0.0025)

for i in range(180,0,-1):

rotateServo(lf,i)

sleep(0.0025)

rotateServo(lf,90)

for i in range(0,180):

rotateServo(rf,i)

sleep(0.0025)

for i in range(180,0,-1):

rotateServo(rf,i)

sleep(0.0025)

rotateServo(rf,90)

def dance1():

x=0

while(x<10):

for i in range(0,60):

rotateServo(lf,i)

rotateServo(rf,i)

sleep(0.0025)

for i in range(60,0,-1):

rotateServo(lf,i)

rotateServo(rf,i)

sleep(0.0025)

for i in range(0,120):

rotateServo(lf,i)

rotateServo(rf,i)

sleep(0.0025)

for i in range(120,0,-1):

rotateServo(lf,i)

rotateServo(rf,i)

sleep(0.0025)

x+=1

def hi():

talk("hi ")

rotateServo(lf,10)

sleep(1)

rotateServo(rf,60)

rotateServo(lt,90)

rotateServo(rt,90)

rotateServo(lt,0)

j=0

while(j<2):

rotateServo(lf,180)

sleep(0.5)

rotateServo(lf,0)

sleep(0.5)

j+=1

talk("nice to meet you")

def searchGoogle(query):

import wikipedia as googleScrap

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

talk("This is what I found on google")

try:

pywhatkit.search(query)

result = googleScrap.summary(query,1)

talk(result)

except:

talk("No speakable output available")

info="my name is friday .I am an AI robot assistant. I can do many thing like searching the internet ,play songs,say jokes and many more all on your command. I can even control your aplliacnce just by saying simple commands.I was created by Shiva manish,vishal and tanish. "

def take\_command():

with sr.Microphone() as source:

print("Talk")

rotateServo(rf,90)

rotateServo(rt,90)

rotateServo(lf,90)

rotateServo(lt,90)

talk("hi, how may i help you")

r.adjust\_for\_ambient\_noise(source)

audio\_text=r.listen(source)

talk('I am processing your command')

print("Wait!!!")

try:

txt=r.recognize\_google(audio\_text)

txt=txt.lower()

txt=txt.replace('friday','')

print(txt)

except:

txt=""

print("i dont know that")

return(txt)

def run\_friday():

print("in run friday")

command=take\_command()

if 'play'in command:

song=command.replace('play','')

talk('playing'+song)

pywhatkit.playonyt(song)

sleep(5)

dance1()

exit()

elif'google'in command:

searchGoogle(command)

elif'on'in command:

led1.write(0)

elif'off'in command:

led1.write(1)

elif 'time'in command:

time=datetime.datetime.now().strftime('%I:%M %p')

print(time)

talk('current time is'+time)

elif 'about you'in command:

talk(info)

elif'say'in command:

say=command.replace('say','')

talk(say)

elif'search'in command:

search=command.replace('search','')

wiki\_info=wikipedia.summary(search,1)

talk(wiki\_info)

elif'joke'in command:

joke=pyjokes.get\_joke()

print(joke)

talk(joke)

elif 'dance'in command:

dance()

else:

talk("I dont know that")

run()

print("out off while")

def run():

rotateServo(rf,90)

rotateServo(rt,90)

rotateServo(lf,90)

rotateServo(lt,90)

with sr.Microphone() as source:

print("listening......")

r.adjust\_for\_ambient\_noise(source)

t=r.listen(source)

t=r.recognize\_google(t)

t=t.lower()

print(t)

if'friday'in t:

run\_friday()

if'hai'or 'hi'in t :

hi()

if'sleep'in t:

for i in range(90,0,-1):

rotateServo(rf,i)

sleep(0.005)

for i in range(90,180):

rotateServo(lf,i)

sleep(0.005)

exit()

else:

run()

sleep(1)

for i in range(0,90):

rotateServo(lf,i)

sleep(0.005)

for i in range(180,90,-1):

rotateServo(rf,i)

sleep(0.005)

def stop():

exit()

talk("All systems are online")

while(True):

root.title("friday")

root.configure(bg="#141414")

root.geometry("400x400")

button1=Button(root,text="stop",height=2,width=15,command=threading.Thread(target=stop).start).place(x=250,y=320)

button2=Button(root,text="run",height=2,width=15,command=threading.Thread(target=run).start).place(x=50,y=320)

# img1= ImageTk.PhotoImage(Image.open(path1))

from PIL import ImageTk ,Image

img1 = ImageTk.PhotoImage(Image.open("C:\\Users\\HP\\OneDrive\\Desktop\\miniproject\\images\\normal\_2.jpeg") )

panel = Label(root, image = img1)

panel.pack(side = "bottom", expand = "yes")

root.mainloop()