**A Presentation Report for PYTHON LAB(22CS307PC)**

**On**

**DESKTOP ASSISTANT**

**Submitted**

**to**

**CMR Technical Campus, Hyderabad**

**In Partial fulfillment for the requirement of the Award of the Degree of**

#### BACHELOR OF TECHNOLOGY

##### in

#### COMPUTER SCIENCE & ENGINEERING

#### By

**MOHAMMAD SUBHAN**

**(227R1A05G3)**

**Under the esteemed guidance of**

## Dr.Nuthanakanti Bhaskar

## Associate Professor



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**CMR TECHNICAL CAMPUS**

***An UGC Autonomous Institute***

**Accredited by NBA & NAAC with A Grade**

**(Approved by AICTE, Affiliated to JNTU, Hyderabad)**

**Kandlakoya (V), Medchal (M), Hyderabad-501 401**

**(2023-2024)**



**CERTIFICATE**

This to certify that, the Presentation entitled is DESKTOP ASSISTANT submitted by MOHAMMAD SUBHAN bearing the Roll Number **227R1A05G3** of **B.Tech Computer Science and Engineering**, In Partial fulfillment for the requirement of the Presentation and for the award of the **Degree of Bachelor of Technology** during the academic year 2023-24.

**Subject Faculty**

Dr.Nuthanakanti Bhaskar

**Name of the student : MOHAMMAD SUBHAN**

**Roll No : 227R1A05G3**

**Year : B.Tech I/II/III/IV**

**Semester : I/II**

**Branch : computer science and engineering**

**Name of the Subject : PYTHON LAB**

**Name of the Topic : DESKTOP ASSISTANT**

**Date of Viva-voice :**

**Signature of the student:**

**Remarks or comments by faculty :**

**Name of the Faculty : Nuthanakanti Bhaskar**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Subject Viva-voice/ppt/presentation(pp)/case study(cs)** | | | | |
| **Name of the topic &concept** | **Presentation on Topic** | **Viva-voice on topic** | **Total**  **Marks** | **Final**  **Marks** |
| **15** | **15** | **10** | **40** | **5** |
|  |  |  |  |  |

**Signature of the Faculty :**

**Remarks or comments by faculty :**

**Name of the Faculty : Nuthanakanti Bhaskar**

**Signature of the Faculty :**

**ABSTRACT**

Early As we all know, how life is interlinked with the technology and the use of AI. AI-powered voice assistants have become an integral part of our lives, intertwining technology and daily tasks. A Personal Virtual Assistant allows a user to command or ask questions in the same manner that they would do with another human and are even capable of doing some basic tasks like opening apps, doing Wikipedia searches without opening a browser, playing music etc, with just a voice command. This project presents the development of a personal desktop assistant using Python, aiming to provide convenience, automation, and assistance to users in their computer-related activities. The assistant incorporates features such as voice recognition, natural language processing, and integration with external APIs to enhance its functionality and user experience.

Table of contents

1. INTRODUCTION
2. LITERATURE SURVEY
3. ANALYSIS AND DESIGN
4. IMPLEMENTATION
5. RESULT
6. SCOPE
7. CONCLUSION
8. REFERENCE

INTRODUCTION

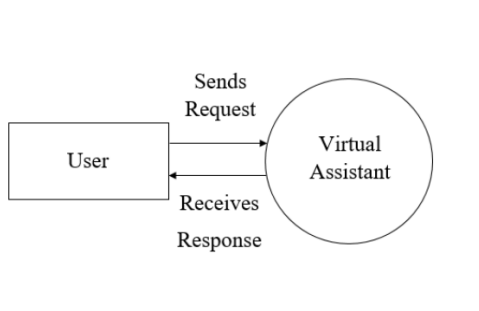
As the A virtual assistant is a type of software program designed to understand and execute voice commands given by users in natural language. This technology has become increasingly popular as it can perform a wide range of tasks on your computer, such as checking the date and time, searching the web, opening specific applications, and greeting you. These days, virtual assistant is being very useful to human beings as it helps us to work on or operate a laptop or a PC on voice commands only and we can do a lot of other computer-based things by the use of assistant. Virtual assistant helps us save our time. Virtual assistant provides us the flexibility for a user to modify us functionalities. For creating virtual assistant for your computer has to go from basics python to complex programming, accordingly. Virtual assistant has an ability to understand and perform requests. Overall, virtual assistants offer a useful tool to streamline and enhance computer-based activities, making it easier for individuals to interact with technology and achieve their daily goals in a more efficient and natural way.

LITERATURE SURVEY

"On the track of Artificial Intelligence: Learning with Intelligent Personal Assistants" by Nil Goksel and Mehmet Emin Mutlu explores how intelligent personal assistants (IPAs) can revolutionize the way we learn and interact with information. They highlight the advanced computing technologies and natural language processing (NLP) capabilities of IPAs that enable personalized and collaborative learning experiences. The authors make a compelling case for the use of IPAs in education and training, emphasizing their potential to transform how we acquire and interact with information.

B.S. Atal and L.R. Rabiner have conducted research on speech analysis, which involves a pattern recognition technique for determining whether the voice input is voiced speech, unvoiced, or silence based on signal dimensions. However, the system has limitations, such as the need for the algorithm to be trained on the specific set of dimension selected and for the recording conditions to be consistent

ANALYSIS AND DESIGN

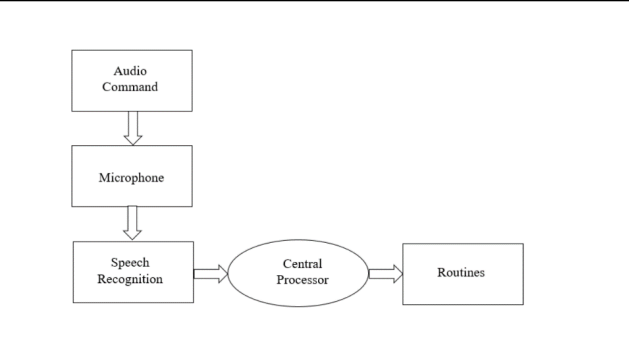


The system design includes:

-It will take input in the form of speech patterns using a microphone.

-It will recognize and convert the audio data into text format.

-It will compare the input with pre-defined commands.

-Finally, it will provide the desired output based on inputs 

IMPLEMENTATION

import pyttsx3

import datetime

import speech\_recognition as sr

import wikipedia

import webbrowser

import sys

import os

import random

sys.stdout.reconfigure(encoding='utf-8')

engine = pyttsx3.init('sapi5')

voices = engine.getProperty('voices')

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

def speak(audio):

engine.say(audio)

engine.runAndWait()

def wishme():

hour = int(datetime.datetime.now().hour)

if hour>=0 and hour <12:

speak("GOOD MORNING!")

elif hour >= 12 and hour <18:

speak("Good Afternoon!")

else:

speak("Good Evening!")

speak("Iam jenny.Please tell me how may i help you")

def takeCommand():

#it takes microphone input from the user and returns string output

r = sr.Recognizer()

with sr.Microphone() as source:

print("Listening...")

r.pause\_threshold = 1

audio = r.listen(source)

try:

print("Recognizing...")

query = r.recognize\_google(audio, language='en-in')

print(f"User said:{query}\n")

except Exception as e:

print("Say that again please...")

return "None"

return query

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

wishme()

while True:

#if True:

query = takeCommand().lower()

#logic for executing tasks based on query

if 'wikipedia' in query:

speak('Searching Wikipedia...')

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

results = wikipedia.summary(query, sentences=2)

speak("According to wikipedia")

print(results.encode(sys.stdout.encoding, errors='replace').decode(sys.stdout.encoding))

speak(results)

elif 'open youtube' in query:

webbrowser.open("youtube.com")

elif 'open google' in query:

webbrowser.open("google.com")

elif 'open instagram' in query:

webbrowser.open('instagram.com')

elif 'play music' in query:

music\_dir = 'C:\\Users\\mdsub\\Music'

songs = os.listdir(music\_dir)

print(songs)

random\_song = random.choice(songs)

os.startfile(os.path.join(music\_dir,random\_song))

elif 'the time' in query :

strTime = datetime.datetime.now().strftime("%H:%M:%S")

speak(f"The time is {strTime}")

elif 'play movie' in query:

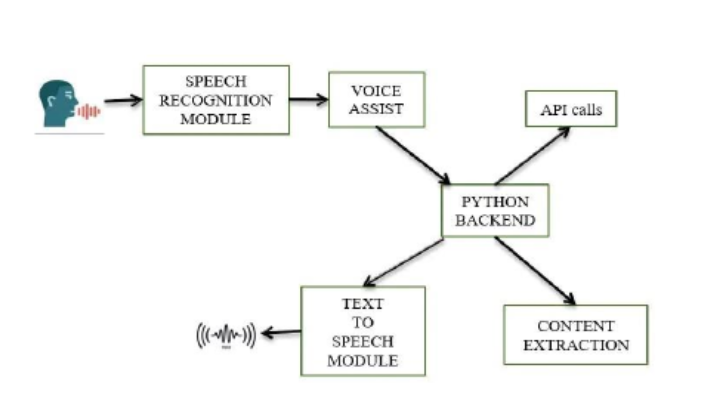
video\_dir = 'C:\\Users\\mdsub\\Videos'

movies = os.listdir(video\_dir)

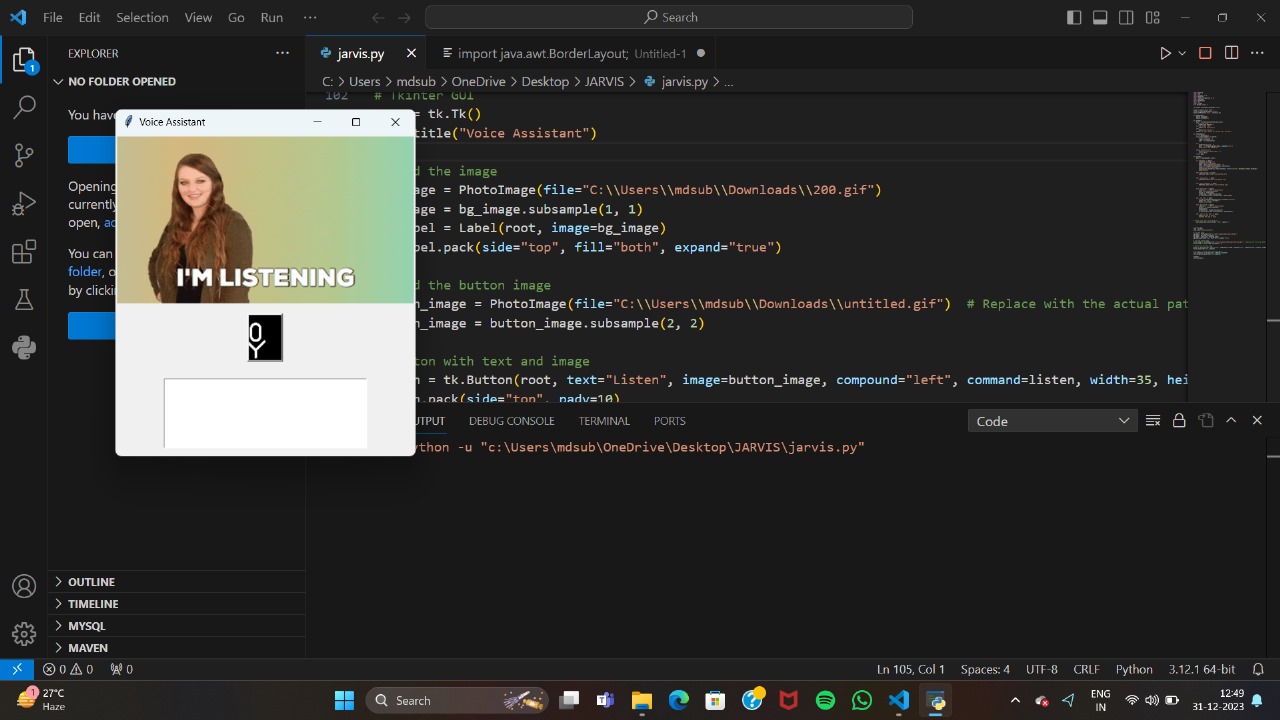
print(movies)

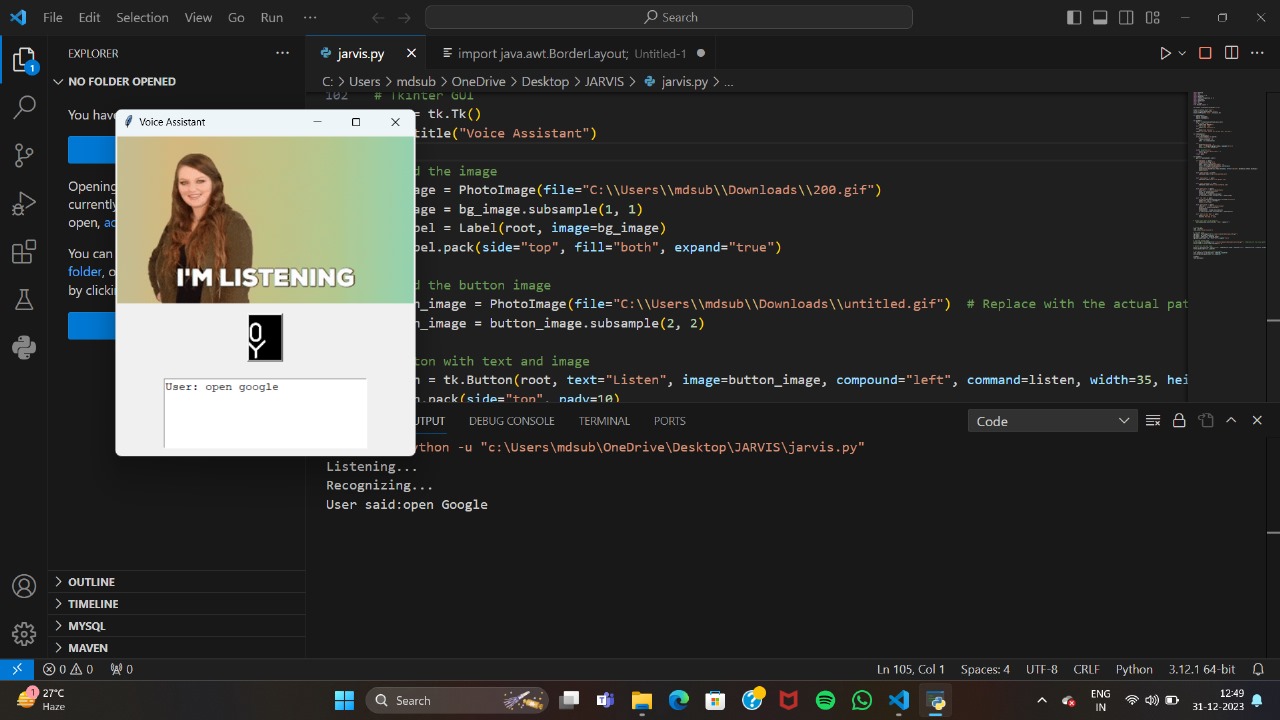
random\_movie = random.choice(movies)

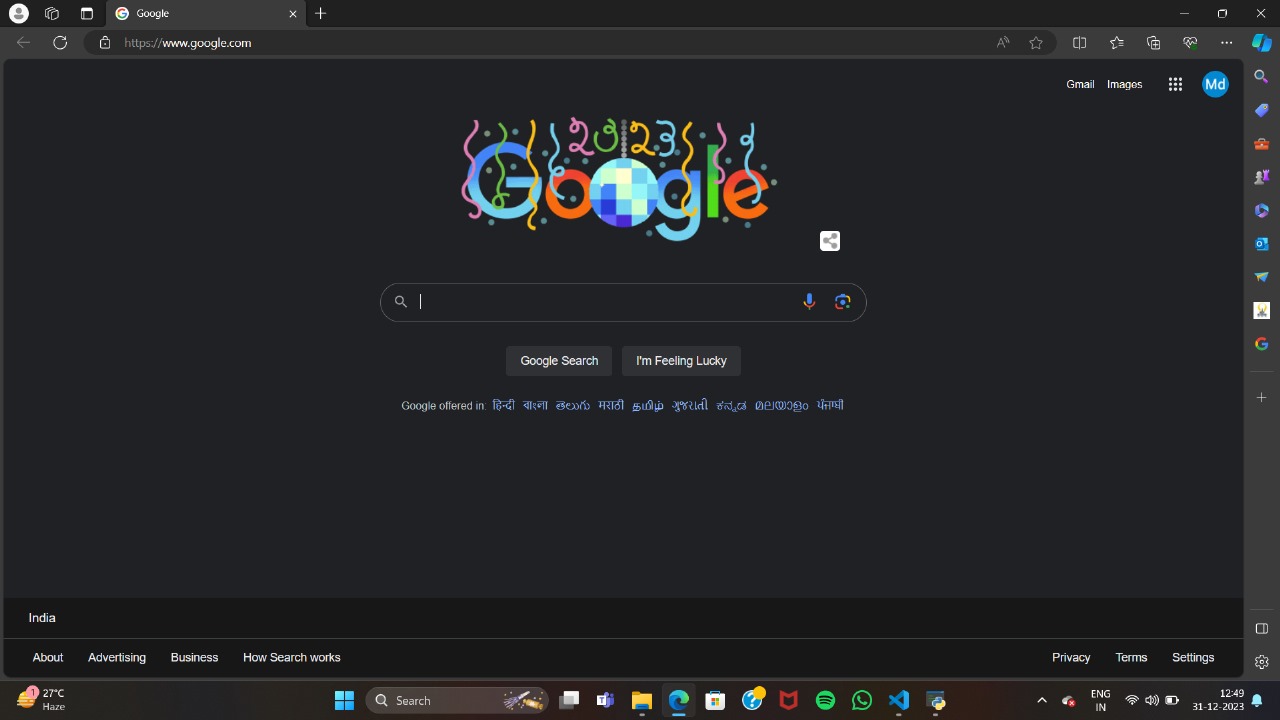
os.startfile(os.path.join(video\_dir, random\_movie))



RESULT







SCOPE

The goal of our project is to create a voice assistant that can help users efficiently perform various

tasks on their personal computers. The assistant will operate through voice commands, minimizing the need for physical hardware. It will be able to open applications and websites, play media, tell the time and date, and even greet users based on the current time. We are also working on integrating AI technology to make the assistant more interactive and engaging for users. With a wide range of possible tasks that can be programmed, the potential uses for the assistant are virtually limitless. As we continue to develop and improve the system, we hope to make it a valuable tool for users to streamline their computer use and maximize productivity

CONCLUSION

Our virtual assistant is designed to fulfill a wide range of user commands, from opening specific

files on the system to browsing the web and gathering information. We have taken a straightforward approach to solving this problem by utilizing Python. The assistant supports various user tasks, including web searches, accessing YouTube videos, sending voicemails, and more. Moving forward, our aim is to enhance the project by integrating artificial intelligence technologies such as machine learning and neural networks. Additionally,

we plan to explore the possibilities offered by the Internet of Things to further enhance the capabilities of our voice assistant. By incorporating these advancements, we will be able to introduce new and exciting features

to our assistant.

**REFERENCE**

T. R. M, Vinoth Kumar V, Lim S-J (2023) UsCoTc: Improved Collaborative Filtering (CFL)

recommendation methodology using user confidence, time context with impact factors for

performance enhancement. PLoS ONE 18(3): e0282904.

https://doi.org/10.1371/journal.pone.0282904

[2] Ramakrishna, M.T.; Venkatesan, V.K.; Bhardwaj, R.; Bhatia, S.; Rahmani, M.K.I.; Lashari, S.A.;

Alabdali, A.M. HCoF: Hybrid Collaborative Filtering Using Social and Semantic Suggestions for

Friend Recommendation. Electronics 2023, 12, 1365. https://doi.org/10.3390/electronics12061365

[3] K. Gunasekaran, V. Vinoth Kumar, A. C. Kaladevi, T. R. Mahesh, C. Rohith Bhat and K.

Venkatesan, "Smart Decision-making and Communication Strategy in Industrial Internet of Things,"

in IEEE Access, doi:10.1109/ACCESS.2023.3258407.

[4] Venkatesan, V.K.; Ramakrishna, M.T.; Izonin, I.; Tkachenko, R.; Havryliuk, M. Efficient Data

Preprocessing with Ensemble Machine Learning Technique for the Early Detection of Chronic

Kidney Disease. Appl. Sci. 2023, 13, 2885. https://doi.org/10.3390/app13052885