Project Report

on

Virtual Assistant for PC

Submitted to

Sant Gadge Baba Amravati University

In partial Fulfillment of the Requirement

For the Degree of

Bachelor of Engineering in

Computer Science and Engineering

Submitted by:

1. Gaurav Ra. Dhale(2R-45)
2. Ankush D. Bhonde(2R-39)
3. Ishan Gawande(2R-49)
4. Gaurav V. Kaple(2R-47)

Under the Guidance of

Prof. C. M. Mankar

Department of Computer Science and Engineering

Shri Sant Gajanan Maharaj College of Engineering,

Shegaon – 444 203 (M.S.)

2022-23

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING,

SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that Mr. Gaurav Dhale, Mr. Gaurav Kaple, Mr. Ankush Bhonde, and Mr. Ishan Gawande, students of second year B.E. in the year 2022-23 of the Computer Science and Engineering Department of this institute have completed the project work entitled "**Virtual Assistant for PC**" based on the syllabus and has submitted a satisfactory account of his work in this report which is recommended for the partial fulfillment the of the degree of Bachelor of Engineering in Computer Science and Engineering.

Prof. C. M. Mankar
Project Guide

Dr. S. B. Patil
Head of Department

Dr. S. B. Somani
Principal

SHRI SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING,

SHEGAON – 444 203 (M.S.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled "**Virtual Assistant for PC**" Submitted by Mr. Gaurav Dhale, Mr. Gaurav Kaple, Mr. Ankush Bhonde, and Mr. Ishan Gawande, students in second year B.E. in the year 2022-23 of the Computer Science and Engineering Department of this institute, is a satisfactory account of his work based on the syllabus which is recommended for the partial fulfillment of the degree of Bachelor of Engineering in Computer Science and Engineering.

Internal Examiner **External Examiner**

Date: **Date:**

CONTENT

Content	Page No.
1. Problem Statement	1
2. Introduction	2
3. Hardware/Software Required	3
4. Theory	4
5. Program	5,6
6. Output	7
7. Conclusion	9

Problem Statement

Development of a Python-based Virtual Assistant for PC

Introduction:

In today's digital era, individuals often find themselves overwhelmed with tasks and information while using their computers. To simplify and streamline their computing experience, there is a growing need for a versatile and intelligent virtual assistant for PC. This virtual assistant, built using the power of Python programming language, will provide users with an intuitive and interactive interface to interact with their computers, perform various tasks, and access information seamlessly.

Problem Description:

The objective of this project is to develop a Python-based virtual assistant for PC that leverages the capabilities of natural language processing (NLP) and artificial intelligence (AI). The virtual assistant should be able to understand user queries and commands, respond with accurate and contextually appropriate information, and perform a wide range of tasks to assist users in their daily computing needs.

Introduction

In the modern era, computers have become an integral part of our lives, and we often find ourselves juggling multiple tasks and dealing with complex operations on our PCs. To enhance the user experience and simplify the management of computer-related activities, there is a need for a comprehensive AI assistant capable of operating and optimizing various aspects of the PC. This AI assistant will leverage advanced technologies, including artificial intelligence and machine learning, to provide users with a seamless control over their computers and assist them in tasks ranging from system management to internet speed testing.

Key Points - Features of Operating Whole PC:

System Monitoring and Optimization: The AI assistant will continuously monitor system performance, resource utilization, and temperature. It will identify bottlenecks, suggest optimizations, and provide real-time insights to enhance overall system efficiency.

Application and Process Management: The AI assistant will facilitate the management of applications and processes. It will allow users to launch, close, or switch between applications efficiently, manage startup processes, and control resource allocation to ensure smooth multitasking.

File and Folder Organization: The AI assistant will assist users in organizing and managing files and folders on their PCs. It will provide functionalities such as intelligent file categorization, automated file renaming, and smart search capabilities to streamline data management.

Internet Speed Testing: The AI assistant will have the ability to conduct internet speed tests to assess the performance of the user's internet connection. It will measure download and upload speeds, latency, and other relevant metrics to provide insights into the internet service quality and troubleshoot network issues.

Hardware/Software Requirement

Hardware Required:

1. Personal Computer (Windows 10/11)
2. Minimum Storage 18 Mb
3. Minimum Ram 2 Gb

Software Required:

1. OpenAI API key of any account
2. Python 3.10
3. Pycharm IDE

Theory

A virtual assistant for PC is an intelligent software application that leverages artificial intelligence (AI) and natural language processing (NLP) techniques to provide users with a seamless and interactive interface to interact with their computers. It assists users in various tasks, including system management, application control, internet speed testing, and more. This section provides an overview of the key concepts and technologies involved in building a virtual assistant for PC.

Artificial Intelligence (AI):
AI is a branch of computer science that focuses on developing intelligent machines capable of performing tasks that typically require human intelligence. AI enables virtual assistants to understand and interpret user queries, learn from user interactions, and make informed decisions or recommendations. Machine learning algorithms play a vital role in training virtual assistants to improve their performance over time.

The pyttsx3 Module: The pyttsx3 module is a Python library that integrates the Text-to-Speech (TTS) engine with Python applications. It provides a simple and intuitive API to convert text into speech using different voices and settings. The pyttsx3 module is cross-platform and supports multiple operating systems, including Windows, macOS, and Linux.

Internet Speed Testing: Internet speed testing is the process of measuring the performance of an internet connection. Virtual assistants can incorporate internet speed testing capabilities to assess the quality and speed of the user's internet connection. This functionality involves measuring download and uploads speeds, latency, and other relevant metrics. Python libraries such as speedtest-cli or Ookla's Speedtest API can be utilized to perform internet speed tests programmatically.

Program

```
import datetime
import pyttsx3
from functions import Function
import AppOpener

class VirtualAssistant:
    def __init__(self):
        self.engine = pyttsx3.init('sapi5')
        self.list1 = ["wordpad", "whatsapp", "brave", "excel", "microsoft store", "microsoft
edge",
                      "microsoft office", "task manager", "file explorer", "word", "powerpoint",
"calculator",
                      'settings', "camera", "opera"]

    def tell(self, message):
        voices = self.engine.getProperty('voices')
        self.engine.setProperty('language', 'en')
        self.engine.setProperty('voice', voices[0].id)
        self.engine.setProperty('rate', 130)
        self.engine.say(message)
        self.engine.runAndWait()

if __name__ == '__main__':
    obj = VirtualAssistant()
    call = Function()

    def hear(text):
        if 'hi' in text:
            print('Hello')
            obj.tell('Hello')

        elif 'you' in text:
            print('I am your personal assistant')
            obj.tell('I am your personal assistant')

        elif 'am i' in text:
            print('You are Gaurav')
            obj.tell('You are Gaurav')

        elif 'exit' in text or 'stop' in text:
            print("Closing program")
```

```
            obj.tell('Okay, bye')
            raise SystemExit

        elif 'time' in text:
            call.time()

        elif 'date' in text:
            call.hdate()

        elif 'lock' in text:
            obj.tell("Locking your device")
            call.lock()

        elif 'shut' in text:
            call.shut()

        elif 'restart' in text:
            call.restart()

        elif 'speed' in text:
            call.speedtest()

        elif 'sleep' in text or 'rest' in text:
            obj.tell("Okay, call me anytime")
            call.sleep()

        elif 'mail' in text:
            call.email()

        elif 'my' in text and ('AI' in text or 'ai' in text):
            print("Connecting to your AI prompt...")
            obj.tell("Connecting to your AI prompt...")
            call.chatwithgpt()

        for app_name in obj.list1:
            if 'open' in text and app_name in text:
                AppOpener.open(app_name)

    while True:
        text_input = input("Enter Prompt ==> ")
        hear(text_input)
```

Output


Run:
index
C:\Users\gaura\AppData\Local\Programs\Python\Python311\Lib\site-packages\Scripts\python.exe H:\TEC-AI\index.py
Enter Prompt==>hi
hello
Enter Prompt==>show time
08:58
Enter Prompt==>todays dat
Enter Prompt==>today;s date
May 15 2023
Enter Prompt==>go to AI
connecting to your AI prompt...
successfully connected to your AI prompt....
Hey I am Rayan your AI prompt, tell me how can i help you?
Ask me anything: pm of india
The Prime Minister of India is Narendra Modi. He has been in office since 2014.
Ask me anything: take exit
disconnecting from your AI prompt
Enter Prompt==>open brave
OPENING BRAVE
Enter Prompt==>check my internet speed
your Internet download speed is 5.07 mb/s and upload speed is 3.12 mb/s
Enter Prompt==>ok stop executing
closing program
Process finished with exit code 0

Conclusion

The Virtual Assistant project is a Python-based implementation that utilizes various modules and functionalities to create a personalized assistant for your PC. The assistant is capable of performing a range of tasks, such as providing the time and date, locking and shutting down the device, opening specific applications, conducting internet speed tests, and even engaging in AI-powered conversations.

The project demonstrates the power of Python in creating interactive and useful applications. It leverages libraries like pyttsx3 for text-to-speech synthesis and AppOpener for launching applications. It also integrates with external functions from the functions module to perform specific actions like time retrieval, email automation, and interaction with an AI-powered chatbot.