N.K. Bagrodia Global School

Computer Science project report on

YARN

Session: 2021 - 22

Submitted to:

Mr. Himanshu Gupta PGT (Computer Science)

Submitted by:

Sujal Goel

CERTIFICATE

This is to certify that **Sujal Goel** of class **XII 'A'** of **N.K. Bagrodia Global School** has done his project on **Yarn – Voice Assistant** under my supervision. He has taken interest and has shown at most sincerity in completion of this project.

I certify this Project up to my expectation & as per guidelines issued by CBSE, NEW DELHI.

Internal Examiner ***External Examiner***

Principal

ACKNOWLEDGMENT

It is with pleasure that I acknowledge my sincere gratitude to our teacher, **Mr. Himanshu Gupta** who taught and undertook the responsibility of teaching the subject computer science. I have been greatly benefited from his classes.

I am especially indebted to our principal **Ms. Jaishree Nawani** who has always been a source of encouragement and support and without whose inspiration this project would not have been successful. I would like to place on record heartfelt thanks to her.

Finally, I would like to express my sincere appreciation for all the other students for my batch their friendship & the fine times that we all shared together.

Requirements

Hardware Requirements

- 4 GB RAM
- 1 GB free disk space
- x86 64-bit CPU (Intel / AMD architecture)

Software Requirements

- Modern Operating System:
  - Windows 7 or 10
  - Mac OS X 10.11 or higher, 64-bit
  - Linux: RHEL 6/7, 64-bit (almost all libraries also work in Ubuntu)
- Python 3.9.6 and IDE

Code Editor used:

- Microsoft Visual Studio:
  - Code
  - Insiders
- GitHub Atom text Editor

CODE

File Tree

- **Config**
  - Config.py
- **Functions**
  - **Miscellaneous**
    - AirQuality.py
    - CurrentTime.py
    - GetAdvice.py
    - GetQuote.py
    - Help.py
    - OpenGoogle.py
    - OpenYoutube.py
    - PasswordGen.py
    - Quit.py
    - SendWhatsappMsg.py
    - TellmeaJoke.py
    - WikipediaSearch.py
  - ChatBot.py
  - GTN.py
  - GTP.py
  - GoogleSearch.py
  - Greet.py
  - LieSwatter.py
  - NewsHeadlines.py
  - PlaySong.py
  - PlaySound.py
  - ProjectBase.py
  - RPS.py
  - SendEmail.py
  - SetAlarm.py
  - Trivia.py
  - Weather.py
  - YouTubePlay.py
  - YouTubeSearch.py

main.py

main.py

```
# Regular import
import os
import sys
import time

# Regular Functions
from Functions.RPS import RPS
from Functions.GTP import GTP
from Functions.GTN import GTN
from Functions.Greet import Greet
from Functions.Trivia import Trivia
from Functions.Weather import Weather
from Functions.ChatBot import ChatBot
from Functions.SetAlarm import SetAlarm
from Functions.PlaySong import PlaySong
from Functions.SendEmail import SendEmail
from Functions.LieSwatter import LieSwatter
from Functions.YouTubePlay import YouTubePlay
from Functions.ProjectBase import takeCommand
from Functions.GoogleSearch import GoogleSearch
from Functions.YouTubeSearch import YouTubeSearch
from Functions.NewsHeadlines import NewsHeadlines

# Miscellaneous Functions
from Functions.Miscellaneous.Quit import Quit
from Functions.Miscellaneous.Help import ShowHelp
from Functions.Miscellaneous.GetQuote import GetQuote
from Functions.Miscellaneous.AirQuality import GetAQI
from Functions.Miscellaneous.GetAdvice import GetAdvice
from Functions.Miscellaneous.OpenGoogle import OpenGoogle
from Functions.Miscellaneous.PasswordGen import PasswordGen
from Functions.Miscellaneous.OpenYoutube import OpenYoutube
from Functions.Miscellaneous.CurrentTime import CurrentTime
from Functions.Miscellaneous.TellmeaJoke import TellmeaJoke
from Functions.Miscellaneous.SendWhatsappMsg import SendWhatsappMsg
from Functions.Miscellaneous.WikipediaSearch import WikipediaSearch

# Greet Function
try:
    Greet()
    if not os.path.exists("Music"):
        os.makedirs("Music")
    elif not os.path.exists("Images"):
        os.makedirs("Images")

except KeyboardInterrupt:
```

```
    sys.exit()

# Main Code
while True:
    try:

        command = takeCommand()

        if command:
            if "open google" in command:
                try:
                    OpenGoogle()
                except KeyboardInterrupt:
                    sys.exit()

            elif "open youtube" in command:
                try:
                    OpenYoutube()
                except KeyboardInterrupt:
                    sys.exit()

            elif "time" in command in command:
                try:
                    CurrentTime()
                except KeyboardInterrupt:
                    sys.exit()

            elif "joke" in command:
                try:
                    TellmeaJoke()
                except KeyboardInterrupt:
                    sys.exit()

            elif "search on google" in command:
                try:
                    GoogleSearch()
                except KeyboardInterrupt:
                    sys.exit()

            elif "search on youtube" in command:
                try:
                    YouTubeSearch()
                except KeyboardInterrupt:
                    sys.exit()

            elif "play on youtube" in command:
                try:
                    YouTubePlay()
                except KeyboardInterrupt:
```

```
                sys.exit()

        elif "wikipedia" in command:
            try:
                WikipediaSearch()
            except KeyboardInterrupt:
                sys.exit()

        elif "news" in command:
            try:
                NewsHeadlines()
            except KeyboardInterrupt:
                sys.exit()

        elif "weather" in command:
            try:
                Weather()
            except KeyboardInterrupt:
                sys.exit()

        elif "air quality" in command:
            try:
                GetAQI()
            except KeyboardInterrupt:
                sys.exit()

        elif "play a song" in command:
            try:
                PlaySong()
            except KeyboardInterrupt:
                sys.exit()

        elif "advice" in command:
            try:
                GetAdvice()
            except KeyboardInterrupt:
                sys.exit()

        elif "quote" in command:
            try:
                GetQuote()
            except KeyboardInterrupt:
                sys.exit()

        elif "whatsapp" in command:
            try:
                SendWhatsappMsg()
            except KeyboardInterrupt:
                sys.exit()
```

```
        elif "email" in command:
            try:
                SendEmail()
            except KeyboardInterrupt:
                sys.exit()

        elif "password" in command:
            try:
                PasswordGen()
            except KeyboardInterrupt:
                sys.exit()

        elif "alarm" in command:
            try:
                SetAlarm()
                time.sleep(1.3)
            except KeyboardInterrupt:
                sys.exit()

        elif "rock paper and scissors" in command:
            try:
                RPS()
            except KeyboardInterrupt:
                sys.exit()

        elif "guess the pokemon" in command:
            try:
                GTP()
            except KeyboardInterrupt:
                sys.exit()

        elif "guess the number" in command:
            try:
                GTN()
            except KeyboardInterrupt:
                sys.exit()

        elif "lie catcher" in command:
            try:
                LieSwatter()
            except KeyboardInterrupt:
                sys.exit()

        elif "trivia" in command:
            try:
                Trivia()
            except KeyboardInterrupt:
                sys.exit()
```

```
            elif "chatbot" in command:
                try:
                    ChatBot()
                except KeyboardInterrupt:
                    sys.exit()

            elif "help" in command:
                try:
                    ShowHelp()
                except KeyboardInterrupt:
                    sys.exit()

            elif "shutdown" in command:
                try:
                    Quit()
                except KeyboardInterrupt:
                    sys.exit()

    except KeyboardInterrupt:
        sys.exit()
```

Config/Config.py

```
config = {
    "Password": "",
    "User-Agent": "",
    "Email": "",
    "WeatherAPI": "",
    "AQIAPI": "",
    "SujalAPI": "",
}
```

Functions/Miscellaneous/AirQuality.py

```
import requests

from Config.Config import config
from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def GetAQI():
```

```
    url = f"https://api.waqi.info/feed/here/?token={config['AQIAPI']}"
    response = requests.get(url).json()["data"]
    print(
        "\33[1m"
        + "\33[92m"
        + "The air quality index in your region is"
        + "\33[0m"
        + "\33[1m"
        + "\33[93m"
        + f" {response['aqi']}"
        + "\33[0m"
        + "\33[1m"
        + "\33[92m"
        + " and the dominent pollutant is "
        + "\33[0m"
        + "\33[1m"
        + "\33[93m"
        + f"{response['dominentpol']}"
        + "\33[0m"
        + "\33[1m"
        + "\33[92m"
        + "."
        + "\33[0m"
        + "\n"
    )
    speak(
        f"The air quality index in your region is {response['aqi']} and the dominent
pollutant is {response['dominentpol']}."
    )
    MidSound()
```

Functions/Miscellaneous/CurrentTime.py

```
import datetime

from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def CurrentTime():
    strTime = datetime.datetime.now().strftime("%H:%M:%S")
    print(
        "\33[93m"
        + "\33[1m"
        + "Sir, the current time is "
        + "\33[0m"
```

```
        + "\33[92m"
        + "\33[1m"
        + f"{strTime.split(':')[0]} hours, {strTime.split(':')[1]} minutes and
{strTime.split(':')[2]} {'seconds' if strTime.split(':')[2] > 1 else 'second'}."
        + "\33[0m"
        + "\n"
    )
    speak(f"Sir the current time is {strTime}")
    MidSound()
```

Functions/Miscellaneous/GetAdvice.py

```
import requests

from Config.Config import config
from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def GetAdvice():
    url = "https://api.sujalgoel.engineer/private/advice"
    advice = requests.get(url, headers={"User-agent": config["User-Agent"]}).json()[
        "advice"
    ]
    speak("Here is a piece of advice for you.")
    print("\33[1m" + "Advice: " + "\33[92m" + advice + "\33[0m" + "\n")
    speak(advice)
    MidSound()
```

Functions/Miscellaneous/GetQuote.py

```
import random
import requests

from Config.Config import config
from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def GetQuote():
    num = random.randint(1, 10)
    url = "https://api.sujalgoel.engineer/private/quote"
    response = requests.get(url, headers={"User-agent": config["User-Agent"]}).json()
```

```
    quote = response["quote"]
    author = response["author"]
    if num % 2 == 0:
        print(
            "\33[1m"
            + "\33[92m"
            + quote
            + "\33[0m"
            + " - "
            + "\33[1m"
            + "\33[93m"
            + author
            + "\33[0m"
            + "\n"
        )
        speak(f"According to {author}, {quote}")
        MidSound()
    else:
        print(
            "\33[1m"
            + "\33[92m"
            + quote
            + "\33[0m"
            + " - "
            + "\33[1m"
            + "\33[93m"
            + author
            + "\33[0m"
            + "\n"
        )
        speak(f"{author} has rightly said, {quote}")
        MidSound()
```

Functions/Miscellaneous/Help.py

```
import webbrowser

from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def ShowHelp():
    webbrowser.open("https://github.com/sujalgoel/Yarn#help-menu-")
    print("\33[92m" + "\33[1m" + "Opening help menu." + "\33[0m" + "\n")
    speak("Opening help menu.")
    MidSound()
```

Functions/Miscellaneous/OpenGoogle.py

```
import webbrowser

from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def OpenGoogle():
    webbrowser.open("https://google.com/")
    print("\33[92m" + "\33[1m" + "Opening Google." + "\33[0m" + "\n")
    speak("Opening Google.")
    MidSound()
```

Functions/Miscellaneous/OpenYouTube.py

```
import webbrowser

from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def OpenYoutube():
    webbrowser.open("https://youtube.com/")
    print("\33[92m" + "\33[1m" + "Opening YouTube." + "\33[0m" + "\n")
    speak("Opening YouTube.")
    MidSound()
```

Functions/Miscellaneous/PasswordGen.py

```
import random
import pyperclip
from word2number import w2n

from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def PasswordGen():
    try:
        SpecialChar = list("@#$%=:?./|~>*()<")
        AlphaNum = list(
            "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz"
```

```
        )

        Allowed = AlphaNum
        print(
            "\33[92m"
            + "\33[1m"
            + "What should be the password length?"
            + "\33[0m"
            + "\n"
        )
        speak("What should be the password length?")
        length = takeCommand()
        if length:
            print(
                "\33[33m"
                + "\33[1m"
                + "Do you want special characters in your password?"
                + "\33[0m"
                + "\n"
            )
            speak("Do you want special characters in your password?")
            special = takeCommand()
            if special:
                if "yes" in special.lower():
                    Allowed += SpecialChar
                Password = ""
                for _ in range(w2n.word_to_num(length)):
                    Password = Password + random.choice(Allowed)
                print("\33[1m" + "Password: " + "\33[92m" + Password + "\33[0m" +
"\n")
                speak("Your password has been generated and copied to the clipboard.")
                MidSound()
                pyperclip.copy(Password)
            else:
                Password = ""
                for _ in range(w2n.word_to_num(length)):
                    Password = Password + random.choice(Allowed)
                print("\33[1m" + "Password: " + "\33[92m" + Password + "\33[0m" +
"\n")
                speak("Your password has been generated and copied to the clipboard.")
                MidSound()
                pyperclip.copy(Password)
        else:
            PasswordGen()
    except ValueError:
        PasswordGen()
```

Functions/Miscellaneous/Quit.py

```
import sys

from Functions.ProjectBase import speak
from Functions.PlaySound import EndingSound


def Quit():
    print("\33[92m" + "\33[1m" + "Shutting Down." + "\33[0m")
    speak("Shutting Down.")
    EndingSound()
    sys.exit()
```

Functions/Miscellaneous/SendWhatsappMsg.py

```
import webbrowser
import urllib.parse

from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def SendWhatsappMsg():
    print("\33[93m" + "\33[1m" + "What is the message?" + "\33[0m" + "\n")
    speak("What is the message?")
    message = takeCommand()
    if message:
        url = f"https://web.whatsapp.com/send?&text={urllib.parse.quote(message)}"
        webbrowser.open(url)
        speak("Sending message on whatsapp.")
        MidSound()
    else:
        SendWhatsappMsg()
```

Functions/Miscellaneous/TellmeaJoke.py

```
import time
import random
import pyjokes
import requests

from Config.Config import config
```

```
from Functions.ProjectBase import speak
from Functions.PlaySound import LaughSound, MidSound


def TellmeaJoke():
    num = random.randint(1, 10)
    if num % 2 == 0:
        joke = pyjokes.get_joke()
        print("\33[1m" + "Joke: " + "\33[92m" + joke + "\33[0m" + "\n")
        speak(joke)
        LaughSound()
        MidSound()
    else:
        url = "https://api.sujalgoel.engineer/private/joke"
        response = requests.get(
            url, headers={"User-agent": config["User-Agent"]}
        ).json()
        joke = response["joke"]
        punchline = response["punchline"]
        print("\33[1m" + "Joke: " + "\33[92m" + joke + "\33[0m")
        speak(joke)
        time.sleep(0.3)
        print("\33[1m" + "Punchline: " + "\33[92m" + punchline + "\33[0m" + "\n")
        speak(punchline)
        LaughSound()
        MidSound()
```

Functions/Miscellaneous/WikipediaSearch.py

```
import wikipedia

from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def WikipediaSearch():
    print("\33[93m" + "\33[1m" + "What should I search on Wikipedia?" + "\33[0m" +
"\n")
    speak("What should I search on Wikipedia?")
    term = takeCommand()
    if term:
        try:
            result = wikipedia.summary(term)
            print(
                "\33[1m" + "Wikipedia: " + "\33[92m" + result.strip() + "\33[0m" +
"\n"
```

```
            )
            speak(result.strip())
        except wikipedia.exceptions.PageError:
            speak(
                f"Sorry, but I couldn't find anything related to {term} on wikipedia."
            )
        MidSound()
    else:
        WikipediaSearch()
```

Functions/ChatBot.py

```
import requests
import urllib.parse

from Config.Config import config
from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand

Flag = True


def ChatBot():
    global Flag
    if Flag:
        print(
            "\33[93m"
            + "\33[1m"
            + "Entering chatbot mode. Say something to activate it."
            + "\33[0m"
            + "\n"
        )
        speak("Entering chatbot mode. Please say something to activate it!")
        Flag = False
    while True:
        message = takeCommand()
        if message:
            if "exit" in message:
                print("\33[93m" + "\33[1m" + "Exiting chatbot mode." + "\33[0m" +
"\n")
                speak("Exiting chatbot mode.")
                Flag = True
                MidSound()
                break
```

```
            url =
f"https://api.sujalgoel.engineer/private/chatbot?message={urllib.parse.quote(message)}
"
            response = requests.get(
                url, headers={"User-agent": config["User-Agent"]}
            ).json()["reply"]
            print("\33[1m" + "Bot: " + "\33[92m" + response + "\33[0m" + "\n")
            speak(response)
        else:
            ChatBot()
```

Functions/GoogleSearch.py

```
import webbrowser

from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def GoogleSearch():
    print("\33[93m" + "\33[1m" + "What should I search on Google?" + "\33[0m" + "\n")
    speak("What should I search on Google?")
    term = takeCommand()
    if term:
        webbrowser.open("https://www.google.com/search?q=" + term)
        print(
            "\33[93m"
            + "\33[1m"
            + "Searching "
            + "\33[0m"
            + "\33[92m"
            + "\33[1m"
            + term
            + "\33[0m"
            + "\33[93m"
            + "\33[1m"
            + " on Google."
            + "\33[0m"
            + "\n"
        )
        speak(f"Searching {term} on Google.")
        MidSound()
    else:
        GoogleSearch()
```

Functions/Greet.py

```
import datetime

from Functions.ProjectBase import speak
from Functions.PlaySound import OpeningSound, MidSound


def Greet():
    greetings = ""
    hour = datetime.datetime.now().hour

    OpeningSound()

    if 5 >= hour <= 12:
        greetings = "Good morning sir!"

    elif 12 >= hour <= 18:
        greetings = "Good afternoon sir!"

    else:
        greetings = "Good evening sir!"

    greetings += "\nI am Yarn 🧶, your virtual assistant! How may I help you?"

    print("\33[92m" + "\33[1m" + greetings + "\33[0m" + "\n")
    speak(greetings)
    MidSound()
```

Functions/GTN.py

```
import random

from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def GTN():
    live = 6
    win = False
    guesses = 0
    num = random.randint(0, 1000)
    print(
        "\33[93m"
        + "\33[1m"
```

```
        + f"The number is chosen and you only have {live} chances to guess that
number."
        + "\33[0m"
        + "\n"
    )
    speak(
        f"The number is chosen and you only have {live} chances to guess that number."
    )
    while guesses < live:
        guesses += 1
        try:
            speak("Enter your guess.")
            print("\33[93m" + "\33[1m" + "Enter your guess: " + "\33[0m", end="")
            guess = int(input())
        except ValueError:
            speak("Enter your guess.")
            print("\33[93m" + "\33[1m" + "Enter your guess: " + "\33[0m", end="")
            guess = int(input())
        if guess == num:
            print(
                "\33[92m"
                + "\33[1m"
                + "\nCongratulations, you have won the game in "
                + "\33[0m"
                + "\33[91m"
                + "\33[1m"
                + str(guesses)
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + f" {'guesses' if guesses > 1 else 'guess'}."
                + "\33[0m"
                + "\n"
            )
            win = True
            speak(
                f"Congratulations, you have won the game in {guesses} {'guesses' if
guesses > 1 else 'guess'}."
            )
            MidSound()
            break

        if guess < num:
            if (live - guesses) != 0:
                print(
                    "\33[91m"
                    + "\33[1m"
                    + "\nThe number is greater than "
                    + "\33[0m"
```

```
                    + "\33[92m"
                    + "\33[1m"
                    + str(guess)
                    + "."
                    + "\33[0m"
                    + "\33[91m"
                    + "\33[1m"
                    + " You have "
                    + "\33[0m"
                    + "\33[92m"
                    + "\33[1m"
                    + str(live - guesses)
                    + "\33[0m"
                    + "\33[91m"
                    + "\33[1m"
                    + f" {'chances' if (live - guesses) > 1 else 'chance'} left."
                    + "\33[0m"
                    + "\n"
                )
                speak(
                    f"The number is greater than {guess}. You have {live - guesses}
{'chances' if (live - guesses) > 1 else 'chance'} left."
                )
        elif guess > num and (live - guesses) != 0:
            print(
                "\33[91m"
                + "\33[1m"
                + "\nThe number is smaller than "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + str(guess)
                + "."
                + "\33[0m"
                + "\33[91m"
                + "\33[1m"
                + " You have "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + str(live - guesses)
                + "\33[0m"
                + "\33[91m"
                + "\33[1m"
                + f" {'chances' if (live - guesses) > 1 else 'chance'} left."
                + "\33[0m"
                + "\n"
            )
            speak(
```

```
                f"The number is smaller than {guess}. You have {live - guesses}
{'chances' if (live - guesses) > 1 else 'chance'} left."
            )
    if guesses >= live and win is False:
        print(
            "\33[93m"
            + "\33[1m"
            + "\nIt seems that you ran out of chances. The number which I guessed was
"
            + "\33[0m"
            + "\33[1m"
            + str(num)
            + "."
            + "\33[0m"
            + "\n"
        )
        speak(
            f"It seems that you ran out of chances. The number which I guessed was
{num}."
        )
        MidSound()
```

Functions/GTP.py

```
import os
import time
import requests
import pyautogui

from Config.Config import config
from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound

MainDir = os.getcwd()


def GTP():
    response = requests.get(
        "https://api.sujalgoel.engineer/fun/pokemon",
        headers={"Authorization": f"Sujal {config['SujalAPI']}"},
    ).json()

    HiddenImage = requests.get(response["data"]["HiddenImage"]).content
    with open(f"{MainDir}\\Images\\hidden-image.png", "wb") as HiddenImageFile:
        HiddenImageFile.write(HiddenImage)
```

```
    os.startfile(f"{MainDir}\\Images\\hidden-image.png")
    print(
        "\33[93m"
        + "\33[1m"
        + "Opening the hidden image of the pokémon. You got 10 seconds!"
        + "\33[0m"
        + "\n"
    )
    speak("Opening the hidden image of the pokémon. You got 10 seconds!")

    t = 10
    while t > 0:
        speak(t)
        t -= 1
        time.sleep(1)

    pyautogui.hotkey("alt", "f4")

    speak("Now enter the pokemon name!")

    PokemonName = response["data"]["name"]
    print("\33[93m" + "\33[1m" + "Enter the pokemon name: " + "\33[0m", end="")
    UserInput = input()
    if UserInput:
        if UserInput.lower() == PokemonName.lower():
            print(
                "\33[92m"
                + "\33[1m"
                + f"\nIt was a {PokemonName}. You got it correct!"
                + "\33[0m"
                + "\n"
            )
            speak(f"It was a {PokemonName}. You got it correct!")
            ShowImage = requests.get(response["data"]["ShowImage"]).content
            with open(f"{MainDir}\\Images\\show-image.png", "wb") as ShowImageFile:
                ShowImageFile.write(ShowImage)
            os.startfile(f"{MainDir}\\Images\\show-image.png")
            time.sleep(3)
            pyautogui.hotkey("alt", "f4")
            MidSound()
        else:
            print(
                "\33[91m"
                + "\33[1m"
                + f"\nThe pokémon isn't {UserInput} but it was a {PokemonName}. You
got it wrong!"
                + "\33[0m"
                + "\n"
            )
```

```
            speak(
                f"The pokémon isn't {UserInput} but it was a {PokemonName}. You got it
wrong!"
            )
            ShowImage = requests.get(response["data"]["ShowImage"]).content
            with open(f"{MainDir}\\Images\\show-image.png", "wb") as ShowImageFile:
                ShowImageFile.write(ShowImage)
            os.startfile(f"{MainDir}\\Images\\show-image.png")
            time.sleep(3)
            pyautogui.hotkey("alt", "f4")
            MidSound()
    else:
        print(
            "\33[91m"
            + "\33[1m"
            + f"\nThe pokémon which I chose was a {PokemonName}."
            + "\33[0m"
            + "\n"
        )
        speak(f"The pokémon which I chose was a {PokemonName}.")
        ShowImage = requests.get(response["data"]["ShowImage"]).content
        with open(f"{MainDir}\\Images\\show-image.png", "wb") as ShowImageFile:
            ShowImageFile.write(ShowImage)
        os.startfile(f"{MainDir}\\Images\\show-image.png")
        time.sleep(3)
        pyautogui.hotkey("alt", "f4")
        MidSound()
```

Functions/ LieSwatter.py

```
import time
import requests

from Config.Config import config
from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def LieSwatter():
    response = requests.get(
        "https://api.sujalgoel.engineer/fun/lieswatter",
        headers={"Authorization": f"Sujal {config['SujalAPI']}"},
    ).json()["data"]
    answer = ""
    Truth = False
    question = response["question"]
```

```
    if response["answer"] == "True":
        answer = "no"
        Truth = True
    else:
        answer = "yes"
        Truth = False
    print(
        "\33[93m"
        + "\33[1m"
        + '"'
        + "\33[0m"
        + "\33[1m"
        + question
        + "\33[0m"
        + "\33[93m"
        + "\33[1m"
        + '"'
        + "\33[0m"
        + "\33[1m"
        + " - "
        + "\33[0m"
        + "\33[93m"
        + "\33[1m"
        + "is it a lie? (yes/no)"
        + "\33[0m"
        + "\n"
    )
    speak(question)
    time.sleep(0.5)
    speak("is it a lie?")
    userinput = takeCommand()
    if userinput:
        if answer == userinput.lower():
            print(
                "\33[92m"
                + "\33[1m"
                + f"You are correct! It was a {'Truth' if Truth else 'Lie'}."
                + "\33[0m"
                + "\n"
            )
            speak(f"You are correct! It was a {'Truth' if Truth else 'Lie'}.")
            MidSound()
        else:
            print(
                "\33[91m"
                + "\33[1m"
                + f"You are wrong! It was a {'Truth' if Truth else 'Lie'}."
                + "\33[0m"
                + "\n"
```

```
            )
            speak(f"You are wrong! It was a {'Truth' if Truth else 'Lie'}.")
            MidSound()
    else:
        LieSwatter()
```

Functions/NewsHeadlines.py

```
import urllib.request

from word2number import w2n
from bs4 import BeautifulSoup

from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def NewsHeadlines():
    try:
        url = "https://news.google.com/news/rss"
        if url.lower().startswith("http"):
            news = urllib.request.urlopen(url).read()
            headlines = BeautifulSoup(news, "xml").findAll("item")
            news_headlines = []
            for new in headlines:
                news_headlines.append(new.title.text)
            print(
                "\33[93m"
                + "\33[1m"
                + "How many headlines you want me to read?"
                + "\33[0m"
                + "\n"
            )
            speak("How many headlines you want me to read?")
            number = takeCommand()
            if number:
                num = w2n.word_to_num(number)
                if num > len(news_headlines):
                    num = len(news_headlines)
                for i in range(num):
                    if i + 1 == num:
                        print(
                            "\33[1m"
                            + f"{i + 1}) "
                            + "\33[0m"
                            + "\33[92m"
```

```
                            + news_headlines[i]
                            + "\33[0m"
                            + "\n"
                        )
                        speak(news_headlines[i])
                        MidSound()
                    else:
                        print(
                            "\33[1m"
                            + f"{i + 1}) "
                            + "\33[0m"
                            + "\33[92m"
                            + news_headlines[i]
                            + "\33[0m"
                        )
                        speak(news_headlines[i])
        else:
            NewsHeadlines()
    except ValueError:
        NewsHeadlines()
```

Functions/PlaySong.py

```
import os
import time
import requests
import youtube_dl
import urllib.parse

from pydub import AudioSegment
from pydub.playback import play


from Config.Config import config
from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


# Example from https://github.com/ytdl-org/youtube-dl#embedding-youtube-dl
def YTDL(url, Title):
    class MyLogger:
        def debug(self, msg):
            pass  # To hide the debug info

        def warning(self, msg):
            pass  # To hide the warning info
```

```
        def error(self, msg):
            pass  # To hide the error info

    def hook(d):
        if d["status"] == "finished":
            print(
                "\33[93m"
                + "\33[1m"
                + "Playing "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + Title
                + "\33[0m"
                + "\n"
            )
            speak(f"Playing {Title}")

    ydl_opts = {
        "format": "bestaudio/best",
        "postprocessors": [
            {
                "key": "FFmpegExtractAudio",
                "preferredcodec": "mp3",
                "preferredquality": "192",
            }
        ],
        "outtmpl": "/Music/%(title)s.%(ext)s",
        "logger": MyLogger(),
        "progress_hooks": [hook],
    }

    with youtube_dl.YoutubeDL(ydl_opts) as ydl:
        ydl.download([url])


def PlaySong():
    print("\33[93m" + "\33[1m" + "What song do you want me to play?" + "\33[0m" +
"\n")
    speak("What song do you want me to play?")
    song = takeCommand()
    if song:
        url =
f"https://api.sujalgoel.engineer/private/ytsr?query={urllib.parse.quote(song)}"
        response = requests.get(
            url, headers={"User-agent": config["User-Agent"]}
        ).json()
        url = response["url"]
```

```
        Title = response["title"]
        Title = (
            Title.replace("/", "_")
            .replace("\\", "_")
            .replace(":", "_")
            .replace("*", "_")
            .replace("?", "_")
            .replace('"', "_")
            .replace("<", "_")
            .replace(">", "_")
            .replace("|", "_")
        )
        FileExists = os.path.isfile("./Music/" + Title + ".mp3")
        FileName = f"{os.getcwd()}\\Music\\{Title}.mp3"
        if FileExists:
            print(
                "\33[93m"
                + "\33[1m"
                + "Playing "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + Title
                + "\33[0m"
                + "\n"
            )
            speak(f"Playing {Title}")
            play(AudioSegment.from_mp3(FileName))
            time.sleep(0.5)
            MidSound()
        else:
            print(
                "\33[93m"
                + "\33[1m"
                + "Downloading "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + Title
                + "\33[0m"
                + "\n"
            )
            speak(f"Downloading {Title}")
            YTDL(url, Title)
            play(AudioSegment.from_mp3(FileName))
            time.sleep(0.5)
            MidSound()
    else:
        PlaySong()
```

Functions/PlaySound.py

```
import os
import time

from pydub import AudioSegment
from pydub.playback import play

MainDir = os.getcwd()


def OpeningSound():
    play(AudioSegment.from_mp3(f"{MainDir}\\Sounds\\OpeningSound.mp3"))
    time.sleep(0.2)


def MidSound():
    play(AudioSegment.from_mp3(f"{MainDir}\\Sounds\\MidSound.mp3"))
    time.sleep(0.2)


def EndingSound():
    play(AudioSegment.from_mp3(f"{MainDir}\\Sounds\\EndingSound.mp3"))
    time.sleep(0.2)


def LaughSound():
    play(AudioSegment.from_mp3(f"{MainDir}\\Sounds\\LaughSound.mp3"))
    time.sleep(0.6)


def AlarmSound():
    play(AudioSegment.from_mp3(f"{MainDir}\\Sounds\\AlarmSound.mp3"))
    time.sleep(1)
```

Functions/ProjectBase.py

```
import pyttsx3
import speech_recognition

from Functions.PlaySound import OpeningSound

engine = pyttsx3.init("sapi5")
voices = engine.getProperty("voices")
engine.setProperty("voice", voices[0].id)
```

```
def speak(text):
    engine.say(text)
    engine.runAndWait()


def takeCommand():
    speech = speech_recognition.Recognizer()
    speech.pause_threshold = 0.5
    with speech_recognition.Microphone(device_index=1) as AudioSource:
        OpeningSound()
        print("\33[31m" + "\33[1m" + "Listening..." + "\33[0m" + "\n")
        speech.adjust_for_ambient_noise(AudioSource)
        audio = speech.listen(AudioSource)
    try:
        query = speech.recognize_google(audio).lower()
        print("\33[92m" + "\33[1m" + "You said: " + "\33[0m" + query + "\n")
    except speech_recognition.UnknownValueError:
        query = None
    return query
```

Functions/RPS.py

```
from random import randint

from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def RPS():
    options = ["Rock", "Paper", "Scissors"]
    computer = options[randint(0, 2)]
    print(
        "\33[93m"
        + "\33[1m"
        + "Choose an option from rock, paper and scissors."
        + "\33[0m"
        + "\n"
    )
    speak("Choose an option from rock, paper and scissors.")
    term = takeCommand()
    if term.lower() == computer.lower():
        print(
            "\33[93m"
            + "\33[1m"
            + f"Both of us chose {computer.lower()}. So, it's a tie!"
            + "\33[0m"
```

```
            + "\n"
        )
        speak(f"Both of us choose {computer.lower()}. So, it's a tie!")
        MidSound()
    elif term.lower() == "rock":
        if computer == "Paper":
            print(
                "\33[91m"
                + "\33[1m"
                + "You chose rock and I chose paper. So, I won!"
                + "\33[0m"
                + "\n"
            )
            speak("You chose rock and I chose paper. So, I won!")
            MidSound()
        else:
            print(
                "\33[92m"
                + "\33[1m"
                + f"You chose rock and I chose {computer.lower()}. So, You won!"
                + "\33[0m"
                + "\n"
            )
            speak(f"You chose rock and I chose {computer.lower()}. So, You won!")
            MidSound()
    elif term.lower() == "paper":
        if computer == "Scissors":
            print(
                "\33[91m"
                + "\33[1m"
                + "You chose paper and I chose scissors. So, I won!"
                + "\33[0m"
                + "\n"
            )
            speak("You chose paper and I chose scissors. So, I won!")
            MidSound()
        else:
            print(
                "\33[92m"
                + "\33[1m"
                + f"You chose paper and I chose {computer.lower()}. So, You won!"
                + "\33[0m"
                + "\n"
            )
            speak(f"You chose paper and I chose {computer.lower()}. So, You won!")
            MidSound()
    elif term.lower() == "scissors":
        if computer == "Rock":
            print(
```

```
                "\33[91m"
                + "\33[1m"
                + "You chose scissors and I chose rock. So, I won!"
                + "\33[0m"
                + "\n"
            )
            speak("You chose scissors and I chose rock. So, I won!")
            MidSound()
        else:
            print(
                "\33[92m"
                + "\33[1m"
                + f"You chose scissors and I chose {computer.lower()}. So, You won!"
                + "\33[0m"
                + "\n"
            )
            speak(f"You chose scissors and I chose {computer.lower()}. So, You won!")
            MidSound()
    else:
        RPS()
```

Functions/SendEmail.py

```
import smtplib

from Config.Config import config
from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def SendEmail():
    print("\33[93m" + "\33[1m" + "What is the message?" + "\33[0m" + "\n")
    speak("What is the message?")
    message = takeCommand()
    if message:
        speak("Would you like to change the message?")
        print(
            "\33[93m"
            + "\33[1m"
            + "Would you like to change the message?"
            + "\33[0m"
            + "\n"
        )
        correctMessage = takeCommand()
        if correctMessage and correctMessage == "no":
            speak("Please enter the receiver's email address.")
```

```
            print("\33[93m" + "\33[1m" + "Receiver's Email: " + "\33[0m", end="")
            Email_to = input()
            smtp = smtplib.SMTP("smtp.gmail.com", 587)
            smtp.starttls()
            smtp.login(str(config["Email"]), str(config["Password"]))
            smtp.sendmail(str(config["Email"]), Email_to, message)
            smtp.quit()
            print(
                "\33[92m"
                + "\33[1m"
                + "\nSuccessfully sent a mail to "
                + "\33[0m"
                + "\33[93m"
                + "\33[1m"
                + Email_to
                + "\33[0m"
                + "\n"
            )
            speak(f"Successfully sent a mail to {Email_to}")
            MidSound()
        else:
            SendEmail()
```

Functions/SetAlarm.py

```
import time
import datetime
import threading

from Functions.ProjectBase import speak
from Functions.PlaySound import AlarmSound, MidSound


def SetAlarm():
    print(
        "\33[93m"
        + "\33[1m"
        + 'The exact hour and minute seperated with ":" (colon) -> Ex: "12:30" \nAlarm
time: '
        + "\33[0m",
        end="",
    )
    speak(
        "Please type the time you want to set the alarm to. Make sure to follow the
format."
    )
```

```
    message = input()
    if message:
        alarmHour = int(message.split(":")[0])
        alarmMinute = int(message.split(":")[1])
        if alarmHour > 23 or alarmMinute > 60:
            print(
                "\33[91m" + "\33[1m" + "\nPlease enter a valid time." + "\33[0m" +
"\n"
            )
            speak("Please enter a valid time.")
            SetAlarm()
        elif alarmHour < 0 or alarmMinute < 0:
            print(
                "\33[91m" + "\33[1m" + "\nPlease enter a valid time." + "\33[0m" +
"\n"
            )
            speak("Please enter a valid time.")
            SetAlarm()
        AlarmThread = threading.Thread(
            target=ThreadAlarm, args=(alarmHour, alarmMinute)
        )
        AlarmThread.start()


def ThreadAlarm(alarmHour, alarmMinute):
    alarm_time = datetime.datetime.combine(
        datetime.datetime.now(), datetime.time(alarmHour, alarmMinute)
    )
    waiting_time = alarm_time - datetime.datetime.now()

    if waiting_time < datetime.timedelta(0):
        print(
            "\33[91m"
            + "\33[1m"
            + "\nThe alarm time has passed. Please set the alarm again."
            + "\33[0m"
            + "\n"
        )
        speak("The alarm time has passed. Please set the alarm again.")
        SetAlarm()

    else:
        print("\33[92m" + "\33[1m" + "Done setting the alarm." + "\33[0m" + "\n")
        speak("Done setting the alarm.")
        MidSound()
        time.sleep((alarm_time - datetime.datetime.now()).total_seconds())
        AlarmSound()
```

Functions/Trivia.py

```
import requests

from Config.Config import config
from Functions.ProjectBase import speak
from Functions.PlaySound import MidSound


def Trivia():
    response = requests.get(
        "https://api.sujalgoel.engineer/fun/trivia",
        headers={"Authorization": f"Sujal {config['SujalAPI']}"},
    ).json()["data"]

    answer = response["answer"]
    options = response["options"]
    question = response["question"]

    print("\33[93m" + "\33[1m" + question + "\33[0m")
    speak(question)
    for i in range(4):
        if i == 3:
            print(
                "\33[91m"
                + "\33[1m"
                + f"{str(i + 1)}) "
                + "\33[0m"
                + "\33[1m"
                + options[3]
                + "\33[0m"
                + "\n"
            )
            speak(f"or {options[3]}")
        else:
            print(
                "\33[91m"
                + "\33[1m"
                + f"{str(i + 1)}) "
                + "\33[0m"
                + "\33[1m"
                + options[i]
                + "\33[0m"
            )
            speak(options[i])
    print("\33[93m" + "\33[1m" + "Enter the option (1/2/3/4): " + "\33[0m", end="")
    speak("Enter the option")
    userinput = int(input())
    if userinput:
```

```
        if options.index(answer) == userinput - 1:
            print(
                "\n"
                + "\33[92m"
                + "\33[1m"
                + f"You are correct! It was {answer}."
                + "\33[0m"
                + "\n"
            )
            speak(f"You are correct! It was {answer}.")
            MidSound()
        else:
            print(
                "\n"
                + "\33[91m"
                + "\33[1m"
                + f"You are wrong! It was {answer}."
                + "\33[0m"
                + "\n"
            )
            speak(f"You are wrong! It was {answer}.")
            MidSound()
    else:
        Trivia()
```

Functions/Weather.py

```
import requests
import urllib.parse

from Config.Config import config
from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def Weather():
    print(
        "\33[93m" + "\33[1m" + "What city do you want to know about?" + "\33[0m" +
"\n"
    )
    speak("What city do you want to know about?")
    city = takeCommand()
    if city:
        url =
f"https://api.openweathermap.org/data/2.5/weather?appid={config['WeatherAPI']}&q={urll
ib.parse.quote(city)}"
```

```
        response = requests.get(url).json()
        if response["cod"] != "404":
            print(
                "\33[93m"
                + "\33[1m"
                + f"Current weather in {response['name']}"
                + "\33[0m"
                + "\33[93m"
                + "\33[1m"
                + " is "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + response["weather"][0]["main"]
                + "\33[0m"
                + "\33[93m"
                + "\33[1m"
                + " and temperature is "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + str(
                    format(
                        response["main"]["temp"] - 273.15,
                        ".2f",
                    )
                )
                + "°C."
                + "\33[0m"
                + "\33[93m"
                + "\33[1m"
                + " Humidity is "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + str(response["main"]["humidity"])
                + "%."
                + "\33[0m"
                + "\n"
            )
            weather = (
                "Current weather in "
                + response["name"]
                + " is "
                + response["weather"][0]["main"]
                + " and temperature is "
                + str(
                    format(
                        response["main"]["temp"] - 273.15,
```

```
                        ".2f",
                    )
                )
                + "°C. Humidity is "
                + str(response["main"]["humidity"])
                + "%."
            )
            speak(weather)
            MidSound()
        else:
            print(
                "\33[91m"
                + "\33[1m"
                + "Sorry, I couldn't find that city."
                + "\33[0m"
                + "\n"
            )
            speak("Sorry, I couldn't find that city.")
    else:
        Weather()
```

Functions/YouTubePlay.py

```
import requests
import webbrowser
import urllib.parse

from Config.Config import config
from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def YouTubePlay():
    print("\33[93m" + "\33[1m" + "What should I play on YouTube?" + "\33[0m" + "\n")
    speak("What should I play on YouTube?")
    song = takeCommand()
    if song:
        url =
f"https://api.sujalgoel.engineer/private/ytsr?query={urllib.parse.quote(song)}"
        response = requests.get(
            url, headers={"User-agent": config["User-Agent"]}
        ).json()
        if response:
            webbrowser.open(response["url"])
            title = response["title"]
            print(
```

```
                "\33[93m"
                + "\33[1m"
                + "Playing "
                + "\33[0m"
                + "\33[92m"
                + "\33[1m"
                + response["title"]
                + "\33[0m"
                + "\33[93m"
                + "\33[1m"
                + " on YouTube."
                + "\33[0m"
                + "\n"
            )
            speak(f"Playing {title} on YouTube.")
            MidSound()
    else:
        YouTubePlay()
```

Functions/YouTubeSearch.py

```
import webbrowser

from Functions.PlaySound import MidSound
from Functions.ProjectBase import speak, takeCommand


def YouTubeSearch():
    print("\33[93m" + "\33[1m" + "What should I search on YouTube?" + "\33[0m" + "\n")
    speak("What should I search on YouTube?")
    term = takeCommand()
    if term:
        webbrowser.open(f"https://www.youtube.com/results?search_query={term}")
        print(f"Searching for {term} on YouTube")
        speak(f"Searching {term} on YouTube.")
        MidSound()
    else:
        YouTubeSearch()
```

Reference and Bibliography

Websites referred

- www.github.com
- www.google.co.in
- www.stackoverflow.com
- www.pythonprogramming.net
- www.tutorialspoint.com

Discord Servers

- Python
- Programmer’s Hangout