A PROJECT REPORT

ON

FRADDY: The Personal Windows Assistant

Submitted in the partial fulfilment of award of

BACHELOR OF TECHNOLOGY

In

Computer Science Engineering

Submitted to:

Ms. Anushka Srivastava

Mr. Nikhil

Submitted by:

Rishabh Pandey (20/BCS/170)

Jalaj Pratap Singh(20/BCS/168)

Declaration

We do hereby declare that the report entitled "Fraddy-Personal-Assistant" submitted by us to USICT and Techno Cultural Club in participation of the requirement for the award of the Code Fellas

Place: Greater Noida

Date: 15/12/2021

Introduction:

In today's era almost, all tasks are digitalized. We have Smartphone in hands and it is nothing less than having world at your fingertips. These days we aren't even using fingers. We just speak of the task and it is done. There exist systems where we can say Text Dad, "I'll be late today." And the text is sent. That is the task of a Virtual Assistant. They are software programs that help you ease your day-to-day tasks, such as showing weather report, creating reminders, making shopping lists etc. They can take commands via text (online chat bots) or by voice. Voice based intelligent assistants need an invoking word or wake word to activate the listener, followed by the command. For our project the wake word is *theFraddy*. Call the wake word '*FRADDY*' followed by the command. And within seconds, it gets executed. Voice searches have dominated over text search.

What is theFraddy:

Fraddy is a Python Module which is able to perform task like Chatbot, Assistant etc. It provides the basic functionality for any assistant application. This *Fraddy* is built using Python, Pytorch, and other open-source libraries and frameworks.

This project is created for those, who want to make, heavy tasks easier in their day-to-day life. Generally, it took lots of time to write code from scratch to build Virtual Assistant. So, we have used pre-installed libraries, which gives you easy functionality to build your own Virtual Assistant.

Problem Statement:

In our daily busy schedule, we forget to call or drop a text to our favorite ones. Let's make their life easier by introducing our project called "*theFraddy*".

No matter how busy you're, just wake up *Fraddy,* tell him to perform a task for you and it will execute your command within a second. You'll not just able to make call or drop messages, but can also send email, ask to book your flight, or appointment, or just to entertain your mood by some *random jokes*.

Purpose-

This Software aims at developing a personal assistant for windows-based systems. The main purpose of the software is to perform the tasks of the user at certain commands, provided in either of the ways, speech or text. It will ease most of the work of the user as a complete task can be done on a single command. *Fraddy* draws its inspiration from Virtual assistants like Google Assistant for Android, Siri for iOS etc. Users can interact with the assistant either through voice commands or keyboard input.

Requirements-
  - Windows 8 or above
  - Processor: Intel i3 or Ryzen 3 and above
  - Python (> 3.0)

API Keys-

To run this program, you will require a bunch of API keys. Register your API key by clicking the following links

- OpenWeatherMap API
- Google Calendar API
- WolframAlpha API

Library used-
  - pyttsx3
  - requests
  - json
  - paramiko
  - webbrowser
  - smtplib

- random
- speechRecognition
- wikipedia
- datetime
- wolframalpha
- os
- sys
- googlesearch
- platform
- win32com
- pyautogui
- selenium.webdriver
- selenium.webdriver.support.ui
- selenium.webdriver.support
- selenium.webdriver.common.by
- selenium.webdriver.common.keys
- time
- subprocess
- paramiko

Technologies Used:

Language-

Python

Framework-

pyaudio

Sapi5 (it comes with windows10 OS)

[voice id: 0 (male)]

Project Description:

As a personal assistant, *Fraddy* assists the end-user with day-to-day activities like general human conversation, searching queries in various search engines like Google, Bing or Yahoo, searching for videos, retrieving images, live weather conditions, word meanings, searching for medicine details, and reminding the user about the scheduled events and tasks. The user statements/commands are analyzed with the help of machine learning to give an optimal solution.

Features-

It can do a lot of cool things, some of them being:

- Greet user
- Tell current time and date
- Launch applications/software's
- Open any website
- Tells about weather of any city
- Tells your current system status (RAM Usage, battery health, CPU usage)
- Tells about any person (via Wikipedia)
- Can search anything on Google
- Can play any song on YouTube
- Tells top headlines (via Times of India)
- Plays music (via Local Folder)
- Send email (with subject and content)
- Calculate any basic mathematical expression
- Answer any generic question
- Take important note in notepad
- Tells a random joke
- Tells your IP address
- Can switch off or reboot the system

- Can take screenshot and save it with custom filename
- Can hide all files in a folder and also make them visible again
- Command line user interface
- Get IP details
- Can search anything on YouTube
- Can take command (via Keyboard)
- Access Linux using remote connection

Scope:

Fraddy can be used in Railway stations, Airports, Government agencies, Research Organizations, Hospitals, Hotels, Colleges and the most importantly at Home by each member for different purposes.

Presently, *Fraddy* is being developed as an automation tool and virtual assistant. Among the Various roles played by *Fraddy* are:

1. Search Engine with voice interactions

2. Tells a random joke

3. Tells your IP address with detail

4. Plays music and send emails

5. Answer any question

There shall be proper Documentation available on its Official GitHub repository for making further development easy, we aim to release our virtual assistant as an Open-Source Software where modifications and contributions by the community are warmly welcomed.

https://github.com/jalaj-singh/The-Fraddy

Challenges Faced:

There are many virtual assistants present in the internet, most commonly used are, Google Assistant, Apple Siri etc. But we have tried something similar, but not same. Faced challenges to get idea, what else we can add new and unique.

Excess Libraries-

The project required extensive use of different libraries some of which were predefined and outsourced while some others had to be created by us, causing complexities.

Time Limitation-

We had a very small-time limit of a week, which made it challenging for us to prepare this whole project of more than 500 lines of code.

Code length and Complexity-

The length of the code became another hurdle for us as it was quite hard for us to filter out and correct the numerous syntactical errors that we encountered in this super lengthy program.

Conclusion:

Through this voice assistant, we have automated various services using a single line command. It eases most of the tasks of the user like searching the web, retrieving weather forecast details, vocabulary help and medical related queries. We aim to make this project a complete server assistant and make it smart enough to act as a replacement for a general server administration. The future plans include integrating *Fraddy* with mobile using React Native to provide a synchronized experience between the two connected devices. Further, in the long run, *Fraddy* is planned to feature auto deployment supporting elastic beanstalk, backup files, and all operations which a general Server Administrator does. The functionality would be seamless enough to replace the Server Administrator with *Fraddy*. We are planning to use *Fraddy* with Docker, so different people can have virtual space for *Fraddy*.

The upcoming updates of this assistant will have machine learning incorporated in the system which will result in better suggestions with IoT to control the nearby devices.