**Final Project Documentation**

“Voice Activated Mini-Vehicular Bot”

Project Lead

Luigi T. Francisco

Members

James Capalungan

Peter Escalicas

Christian Garcia

Kurt Torres

**Introduction**

Engineering is a field of coming up with innovative ways to solve something. The human has many muscles that we use to control things. Standard methods include maintaining a remote control to turn on a TV and pressing a button in an elevator to lift you to a certain level. Another significant facet of human management is the voice, like how we could order other humans to do specific work. We can also implement voice for machine movement, precisely what we will do.

**Objectives**

The team aims to create a program for vehicular control exclusively through voice. Incorporation of existing algorithms for Speech Recognition through Python and receiving instructions to be implemented through a microcontroller running the vehicle.

**Methodology**

The team has chosen CMUSphinx or PocketSphinx as a Python package for Speech Recognition. To minimize latency and achieve a more accurate result for the words recognized. The program segmented the phrase recorded and implemented and imported a list of keywords. Once a keyword is recognized the string is sent to the microcontroller of the vehicle through a common access point a WiFi/Bluetooth.

The bot would be programmed to receive a string from the computer and execute certain moves regarding movements and light signals to signify the status of vehicles.

October 6, 2023 Report

A stable configuration of the Speech Recognition has been achieved through the code

from pocketsphinx import LiveSpeech

print('Listening')

speech = LiveSpeech(lm=False,kws='keyword.list')

# an for in loop to iterate in speech+

for phrase in speech:

        # printing if the keyword is spoken with segments along side.

    word = (phrase.segments(detailed=False))

    print(word[0])

    transfer = word[0]

    match transfer:

        case "go ":

            print('Moving forward sir!')

        case "south ":

            print('Reversing Sir!')

        case "east ":

            print('Turning East Sir!')

        case "west ":

            print('Turning West Sir!')

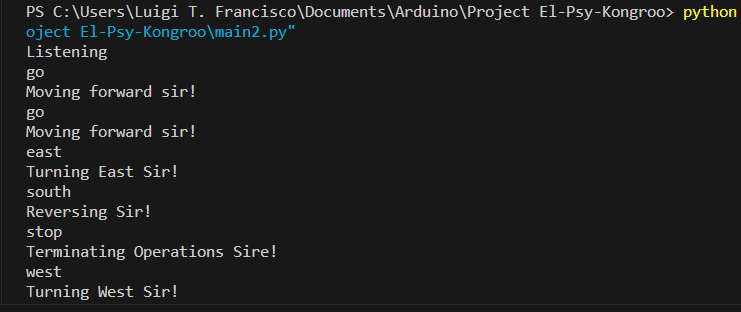
        case "stop ":

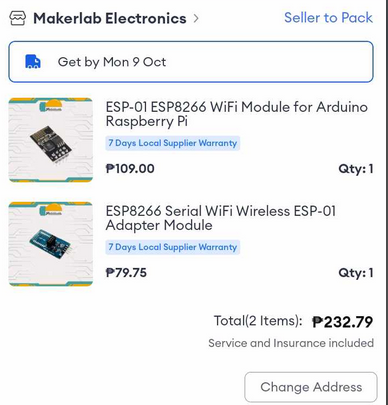
            print('Terminating Operations Sire!')

        case \_:

            print("No commands issued yet sire!")

Output



The team bought wifi-module esp8266 esp-01 and an adapter for Arduino Uno R3 for 232 php. Expected to arrive in Oct 9 the team awaits also prepare a Vehicle to be used either an old one or a new one.

**Oct 31 Report**

With the arrival of the following goods. I was struck by the sheer effort to establish a serial connection with an esp8266 through the adapter. Luckily, after six repositories in the git hub and numerous websites, I found gold that allowed me to confirm a relationship with it through the serial monitor using AT Command lines.

With that said, once it was able to establish connection. I faced challenges establishing a TCP server from esp 8266 that my computer could access. I had to use a certain command to make the Station IP of the ESP32 to be inline within the router ip address.

Once a good TCP server was established. I connected my computer through a Python program using the server in address and port number.

Once a good server was done, I incorporated the previous code for voice recognition to the connection with a few modifications, transferring the word and encoding it in a TCP packet to be read by the Arduino through the esp 8266 module.