Longtai Zhong

20 Jan 2022

ENGR180

Lab 2 Report

1. Program a working MQTT subscriber and publisher to have two-way communications. If you have something relevant to your project, do it. If not, an easy idea is to ping pong between you and a partner a counter that increments each time you receive a message (with a delay). Screenshot the terminal and don’t forget to mention your partner. In your group of 4, try doing a simple 4-waycommunications task. Notice the increase in complexity with each additional member. Also note any possible communications lag that you may be having.

Consider how to use MQTT for the project. Based on your experiences, what is made possible using MQTT? What seems fairly difficult using MQTT? If you were to use MQTT, what would be areasonable communications lag time be? Would you prefer to use a different method of transmitting data?

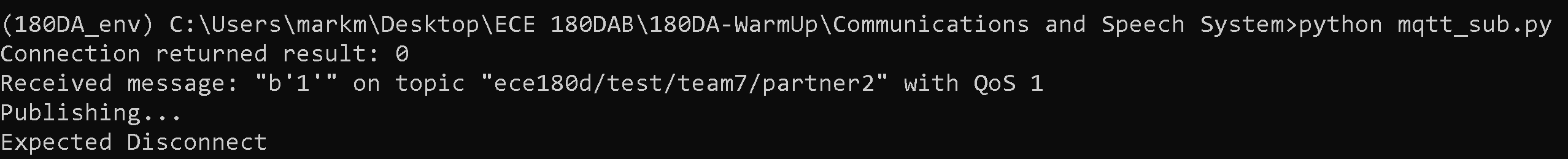
Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated



- For our project, we are thinking about a dart game, so this information communication can be used to implement a multiplayer system, where data such as score and name are transmitted to each other.

2. Choose some speech program, set it up, and work on an easy task. For instance, choose two fairly different words (e.g. “cat” and “dog”). See how accurate your program can work for that. If you can get that working, try limit testing your speech program. Try more similar words. Try two different phrases and see if it continues to work with longer phrases. Play some music in the background. Does it work under noise?

* For my implementation, I used CMU sphinx that uses pyaudio. And I tested audio transcribe capability. I tested this with both short (few words) to long (10+ words) sentences, and they worked well. I also tested with some background noise and the program was able to transcribe audio to word as well.
* From this program, we are aiming to detect simple words such as “start the game” and “end the game” for our project.

Graphical user interface, text, application, email

Description automatically generated