Bluetooth struggle

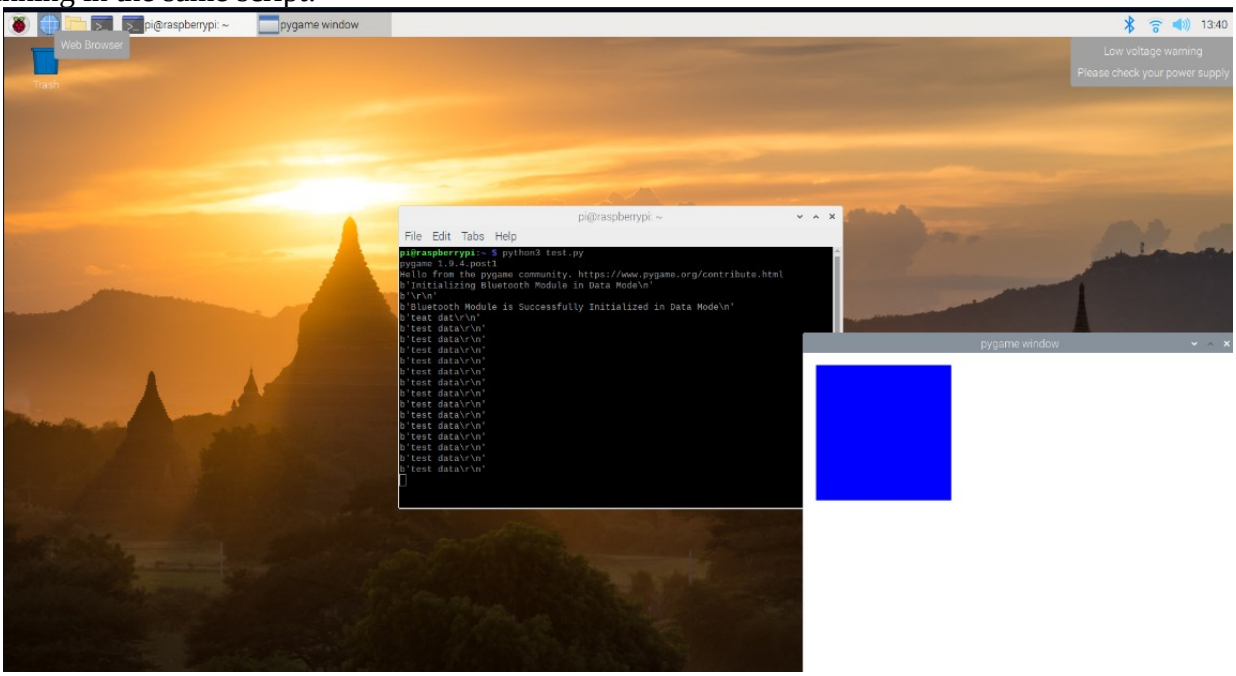
This document explains why we could not finalise the UI for our project.

As part of our project, we developed a python UI which allowed us to visualise they movements of the micromouse inside the maze. The UI was implemented with Pygame, and data from micromouse was transmitted using bluetooth. For this we used the BLE library, which worked, however the big issue was the large size of the BLE library, which took around 80% of space in our ESP32. This was not usable, as the code for micromouse could not fit in the remaining 20%. We thus went on a quest to find an alternative solution. Main idea was to use a lighter bluetooth library, notably the bluetoothSerial.h, but this presented the issue of receiving the data within the pygame code.

First approach was to try to pair the esp32 with a computer directly. This did not work on Windows, we did manage to setup a bluetooth serial terminal, but we did not manage to transfer the results into the python script. Next we tried on linux, setting up the connection manually with bluetoothctl, then using command line pipelines to get the data to the python, but this approach also failed as we could not setup the bluetooth communication well.

Next idea was to use a raspberry pi. This is because raspberry pi allows you to control the GPIO pins with a python script instead of a C code like in arduino or ESP, and so we planned to import pygame in that same script and use the UI that way. We tried this by hooking up a hc05 bluetooth sensor directly to the GPIO pins of raspberry pi, but soon discovered that since raspberry has bluetooth built in, like a computer, there was not any good support for external bluetooth module. Using the built in raspberry bluetooth would be equivalent to what we tried on linux machine, so we did not attempt that. Instead we jumped through yet another loop, connecting the hc05 to arduino, arduino with a cable to raspberry, where the python code would manage the serial connection to receive bluetooth data and the pygame UI.
Picture of connection:
HC05 is connected based on the wiring to arduino, with voltage divider on its rx pin as it only accepts 3.3v logic. USB cable connects the arduino to raspbery.

This finally worked. We were receiving bluetooth data sent from phone, and we had pygame running in the same script:



Code for the arduino and raspberry is also uploaded on github.

However, we ran into a last issue. Pairing the HC05 from arduino and ESP32 in micromouse ended up being more difficult than expected, and we did not manage to solve this in time to present UI in our final version of project. However we still spent a lot of time on this, and so I believe that having this document explaining the work is important.