User Manual for Hom Lab GUI

Installation:

Software required: MongoDB, Python 3, Anaconda, PyQt5, PyQt5-tools, PyQtGraph, PyMongo, NumPy, PySerial

Hardware required: K-30 CO2 Nutrient Sensor and USB connection, Numato 8 USB Solid State Relay and USB connection

To install MongoDB proceed to their download center here: https://www.mongodb.com/download-center. On this page select the "Server" tab in the middle of the page between the "Cloud" and "Tools" tabs. Then proceed to fill out the appropriate operating system and choose MSI as the preferred package download method and then press download. When the MSI is finished installing, run it. As you get to the "Service Configuration" page, check the "Install MongoD as a Service" checkbox and select "Run Service as Network Service user." Then hit the "Next" button. Installing MongoDB-Compass is optional but highly suggested to easily track the contents of the database. The rest of the setup wizard should be straightforward, and no other changes are necessary.

To install Python 3 proceed to their downloads page here: https://www.python.org/downloads/. Installing the latest python version is recommended. Select your operating system and then click download. After it downloads, run the executable. Click the "Install Now" option. There should be no other changes needed to be made for the setup wizard.

To install Anaconda visit their download page here: https://www.anaconda.com/distribution/#download-section. Select the appropriate operating system and then select the Python 3.7 version of Anaconda to download. Once downloaded, run the setup wizard. The only needed change for the setup wizard is when you get to the "Advanced Options" page, make sure to select "Add Anaconda to my PATH environment variable." The rest should be default settings.

To download the rest of the required software, use the following commands in the Anaconda prompt:

"pip install PyQt5"

"pip install PyQt5-tools"

"pip install PyQtGraph"

"pip install PyMongo"

"pip install NumPy"

"pip install PySerial"

Use:

To run the GUI, you can compile the main GUI script by running this command in the Anaconda prompt:

"python filenamehere.py"

The layout for my GUI consists of two tabs for the user to interact with. The first is the Nutrient Sensor Data tab which allows the user to monitor the available nutrient sensor data being input into the database. The user is presented with both an interactive table and an interactive graph that both update with data in real time. The second tab is the Nutrient Depositing tab. This tab allows the user to interact with the power relay through checkboxes to open and close certain valves to release nutrients into the environments in precise measurements. Once the checkboxes are checked, the deposit button must be pressed to interact with the relay itself. There are also “Open All” and “Close All” buttons that allow the user to open all the relays or close all of them. There is also a functioning exit button in the top right of both tabs. A change is needed to the com port names in the script based on the computer used to compile the code. The correct communications port can easily be determined on Windows by opening the device manager and then expanding the “PORTS” section. The correct port should be listed there.

How To/FAQ:

The only way currently to turn off any relay is to either uncheck the box and re-click the deposit button next to the checkboxes or click the “Close All” button to turn all of the relays off.

Both the data graph and data table on the first tab are interactive with the mouse. On the graph, use the mouse wheel and or click and drag to change the proportions of the graph to get a bigger view. For the data table, you can double click a cell to expand the text inside if it is cut off. You can also click and drag the columns and rows to increase cell size.