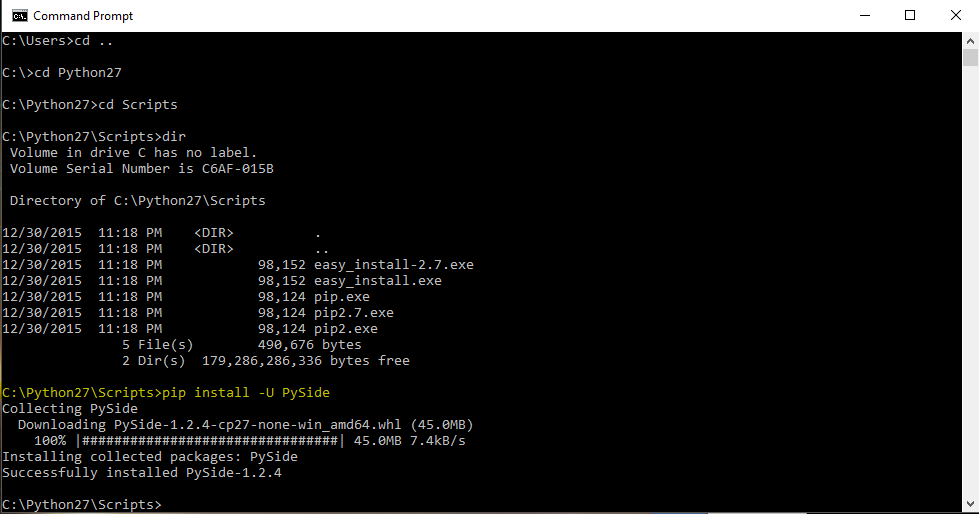
**Ground Station Software – Intro to PC software to run the Ground Station Antenna Array**

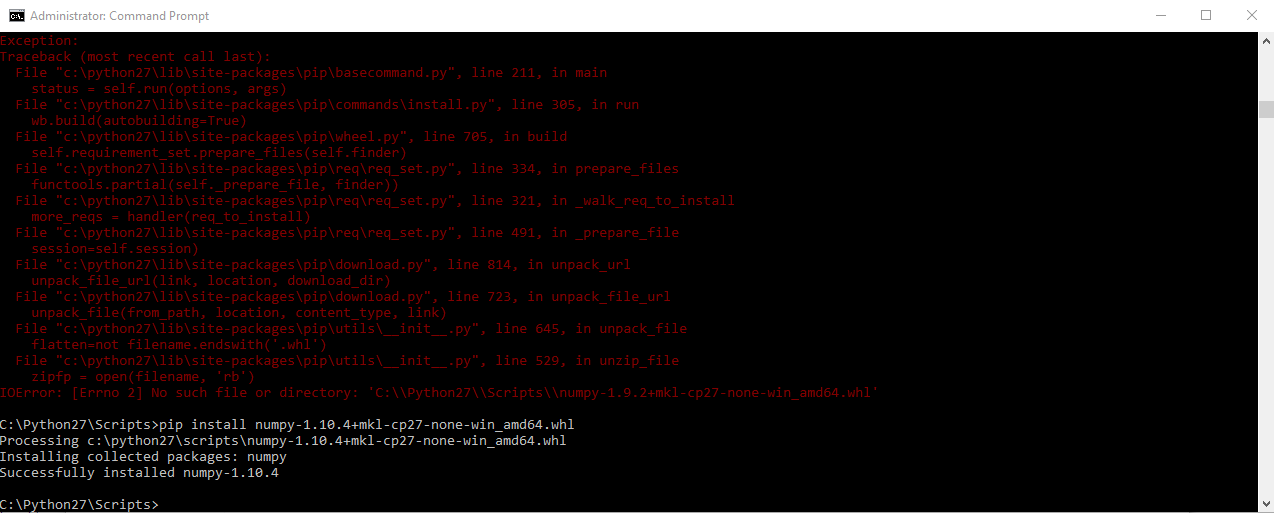
Setup & Base Software:

1. Install Qt Creator (v5.5.1)
   1. This is a windows software that will allow you to edit and build the GUI
   2. <http://www.qt.io/download-open-source/>
      1. This is the one I used but there are other versions (freeware vs open source)
   3. The file type we are editing is a .ui file. This must be compiled into python for us
2. Install Python 2.7
   1. Installing Python 2.7 will allow for editing/executing python via IDLE.
   2. Installing Python 2.7 also installs pip.py (in C:/Python27/Scripts)
3. Install PySide via “pip install –U PySide”
   1. Execute from command line:

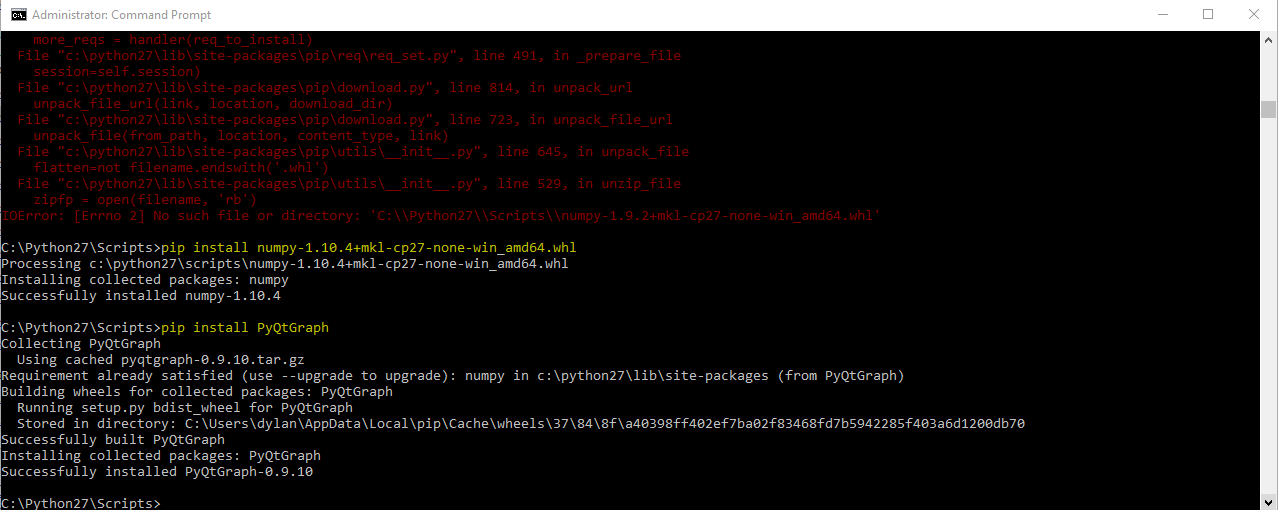


* 1. PySide is used to convert between .ui file type and python file type (.py).
     1. There are ways to run .ui file types directly through python but the code is provided is not currently written to work as such.
  2. Documentation - <https://wiki.qt.io/PySide_Binaries_Windows>
  3. Notice you must move to C:\Python27\Scripts to execute pip instruction

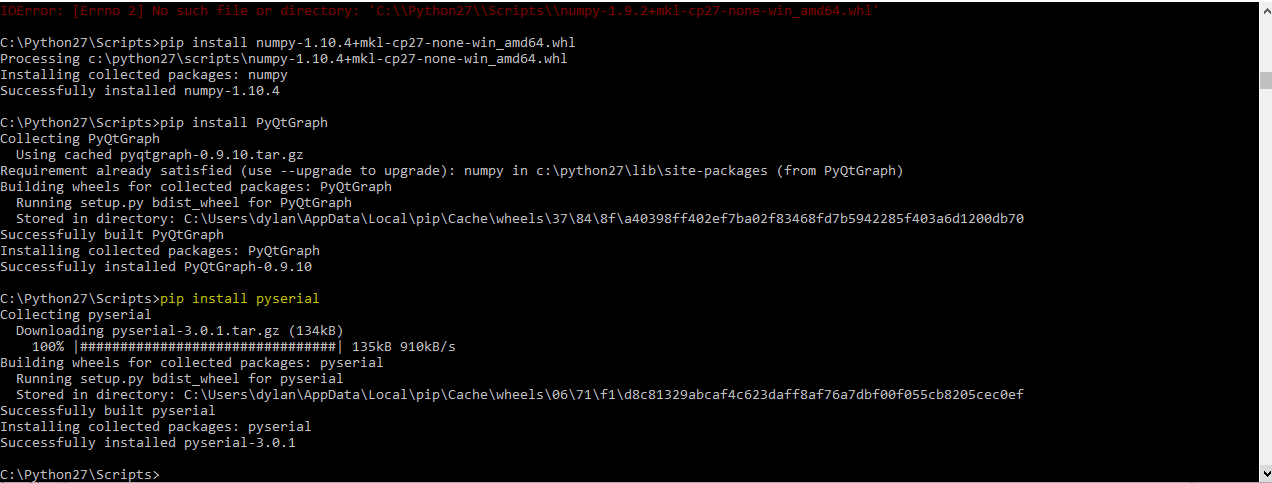
1. Install wheel via “pip install wheel”
2. Install NumPy “pip install numpy-1.10.4+mkl-cp27-none-win\_amd64.whl”
   1. The second pip install of numpy will have to match your architecture.
   2. <http://www.lfd.uci.edu/~gohlke/pythonlibs/#numpy>
   3. For my system “[numpy-1.10.4+mkl-cp27-none-win\_amd64.whl](javascript:;)”
   4. Download the .whl and place in C:\Python27\Scripts
   5. Then use “pip install numpy-1.10.4+mkl-cp27-none-win\_amd64.whl”
      1. The red is from where I did not have the .whl in the Scripts directory



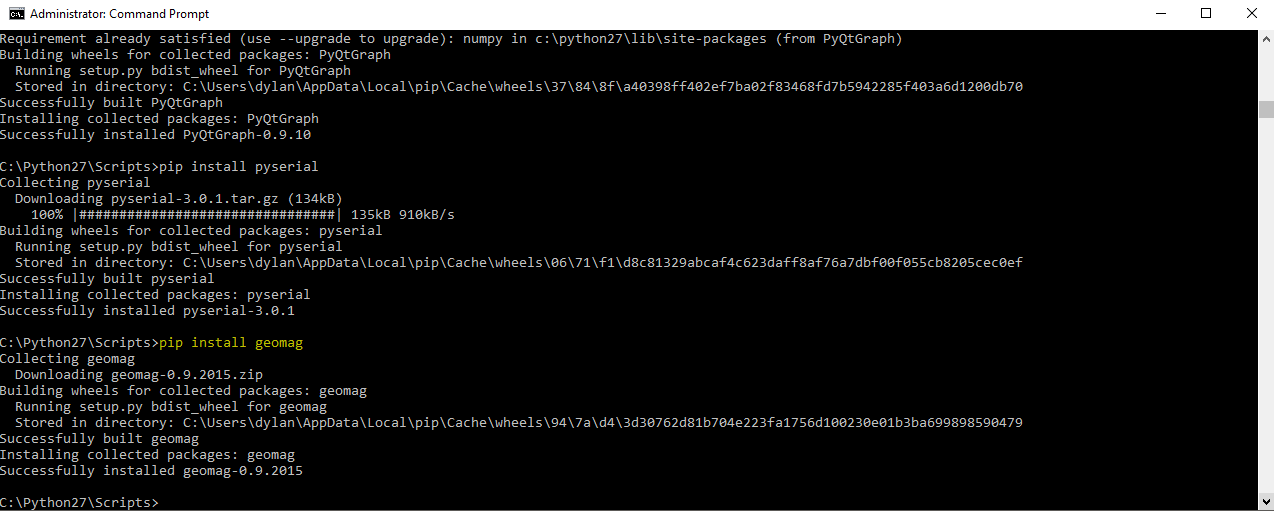
1. Install PyQtGraph via “pip install PyQtGraph”
   1. This package requires NumPy and will not install without it
   2. I highlight the lines yellow in screenshots, they’ll be white in case this bothered you



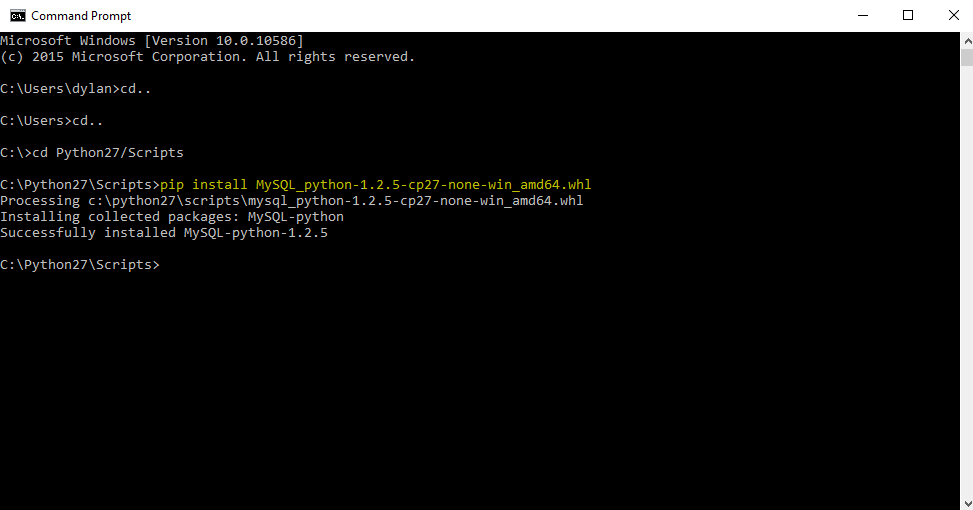
1. Install pySerial via “pip install pyserial”



1. Install Geomag via “pip install geomag”

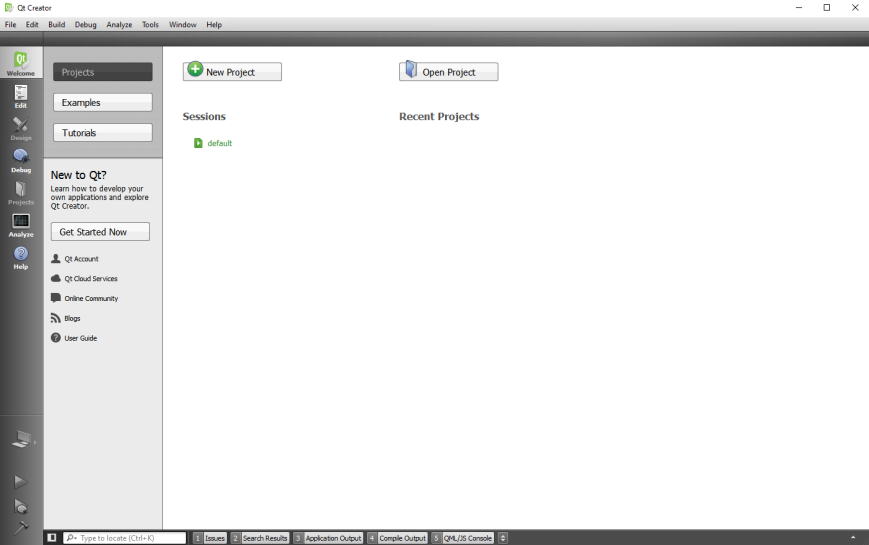


1. Install MySQL-python through a .whl file like the numpy install
   1. <http://www.lfd.uci.edu/~gohlke/pythonlibs/#mysql-python>
   2. “pip install MySQL\_python-1.2.5-cp27-none-win\_amd64.whl”

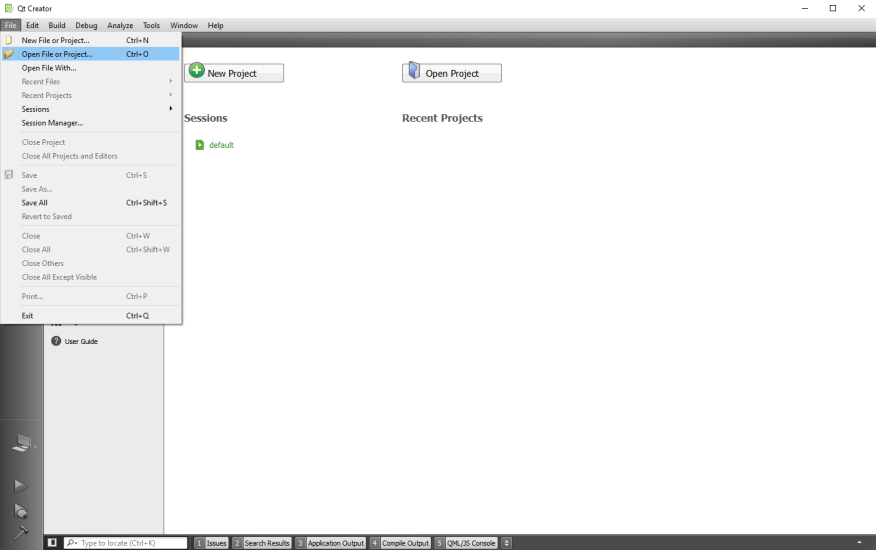


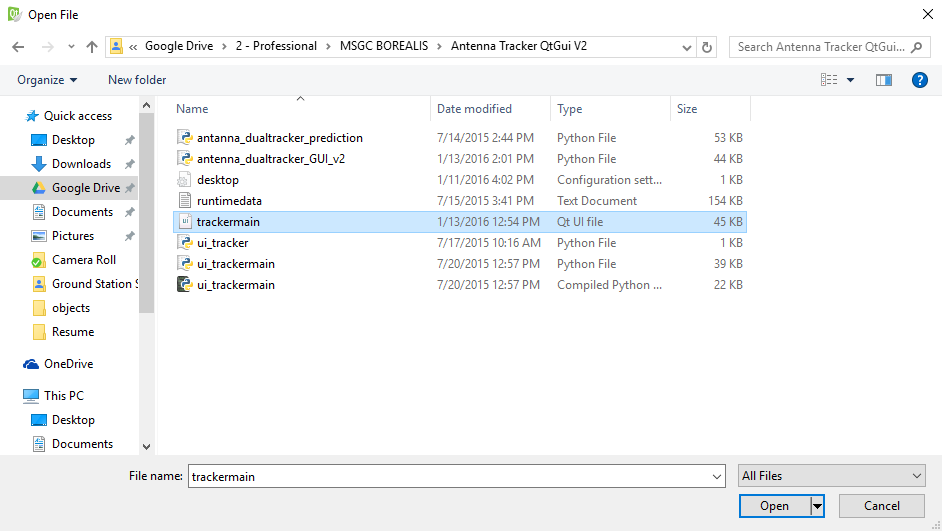
Using Qt Creator

When launching Qt Creator, you’ll be greeted with a screen like this:

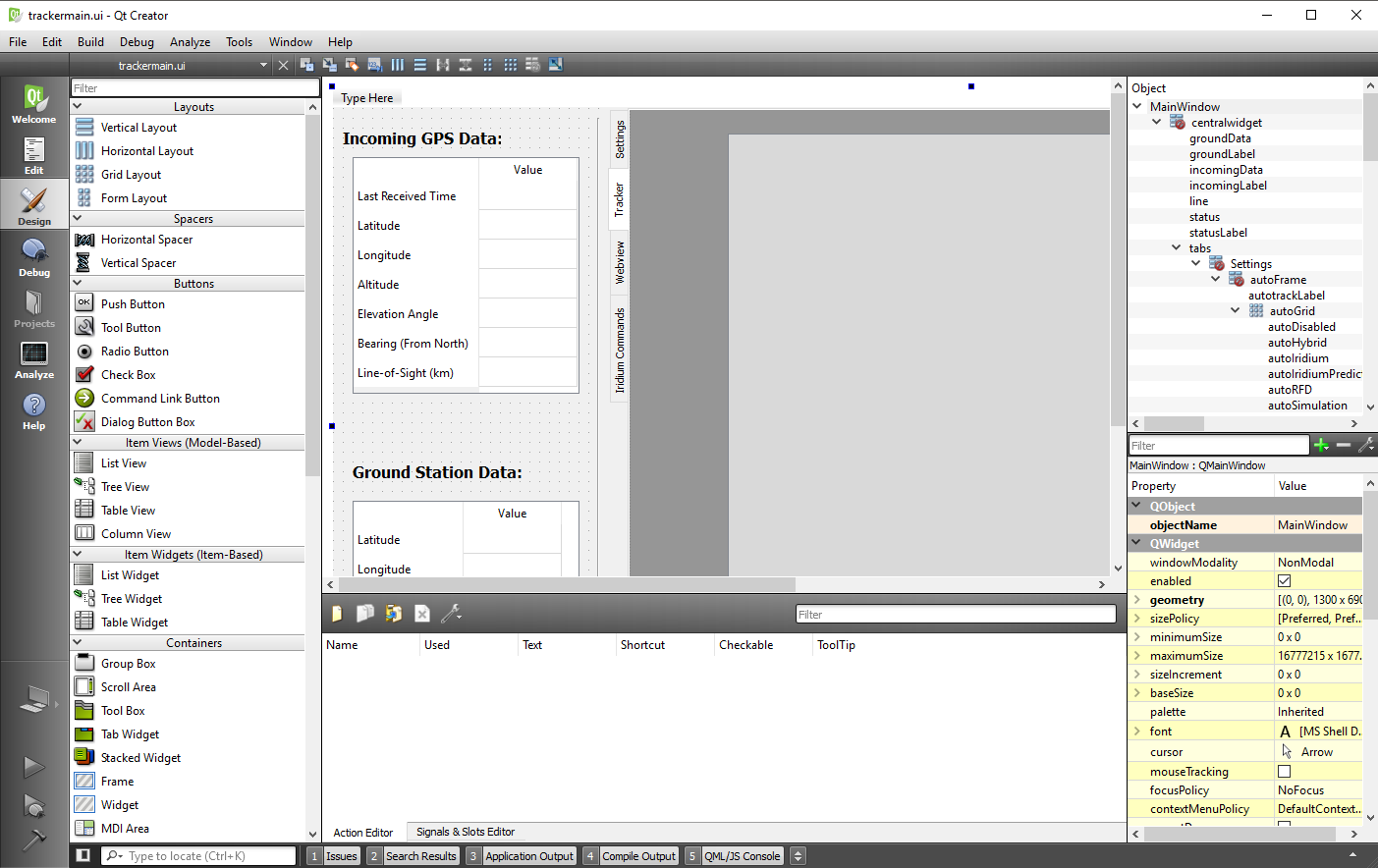


To edit the existing GUI, do this:





And here is your main editor window:



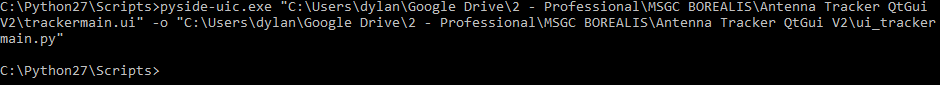
There are a lot of options here; I recommend working through some of the online tutorial resources.

<http://doc.qt.io/qt-4.8/designer-manual.html>

Once you have edited trackermain.ui, save the file as trackermain.ui. Changing the file name will require changes in the antenna tracker code.

Once saved, compile the .ui file into trackermain.py (<http://stackoverflow.com/questions/4442286/python-code-genration-with-pyside-uic>)

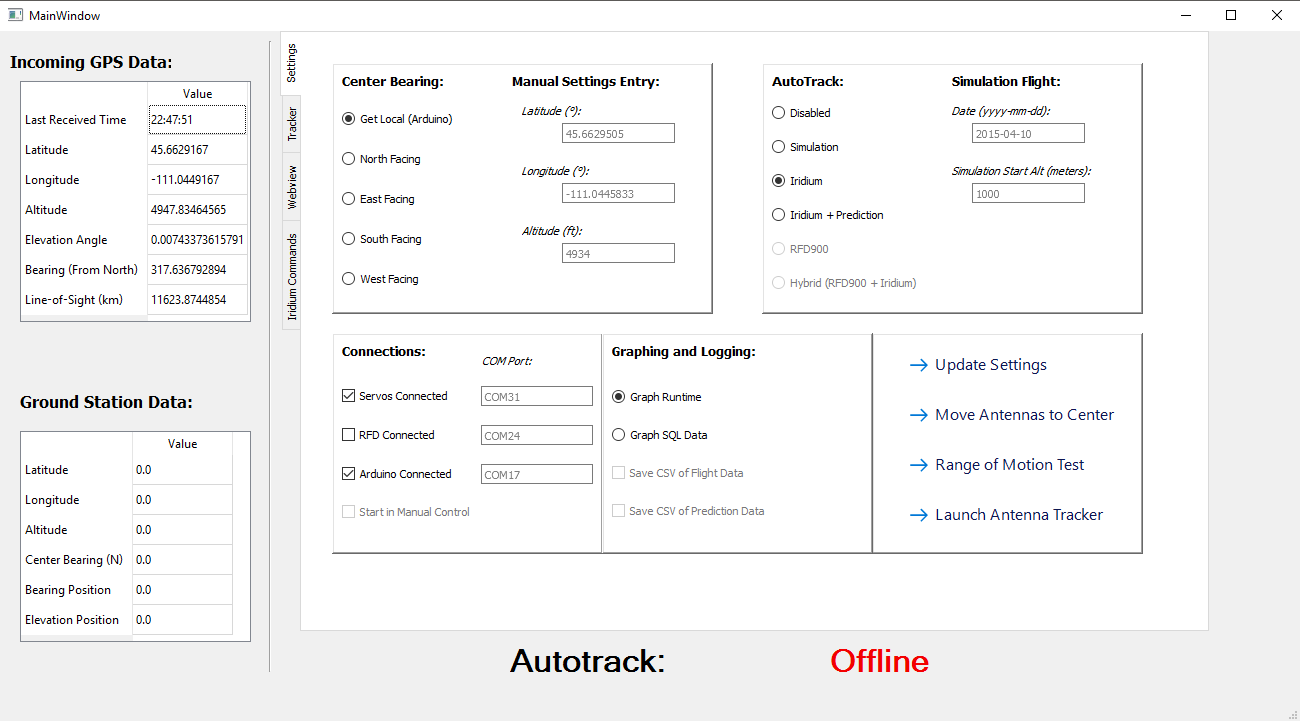
“pyside-uic.exe trackermain.ui –o ui\_trackermain.py”

It will probably be needed for you to state the entire path: 

It is a huge pain to type the entire path. Often when I am working, I will put the tracker code on my desktop to make the path smaller. You can also avoid this by setting up a PATH for the pyside-uic.exe. This is not terribly difficult and can save you some annoyance in the future.

Running the System

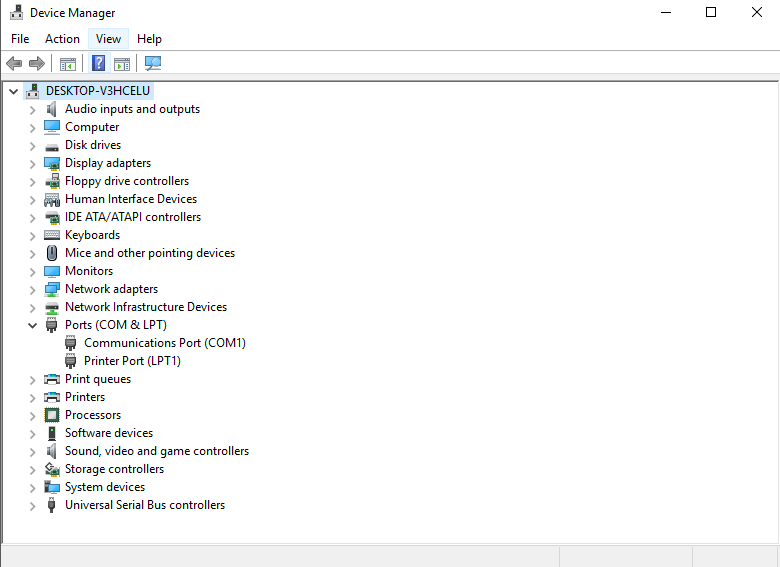
You can launch the system by simply double clicking on the script or running after opening the code in IDLE (Python2.7).



Data set for ground station via either manual input or Arduino

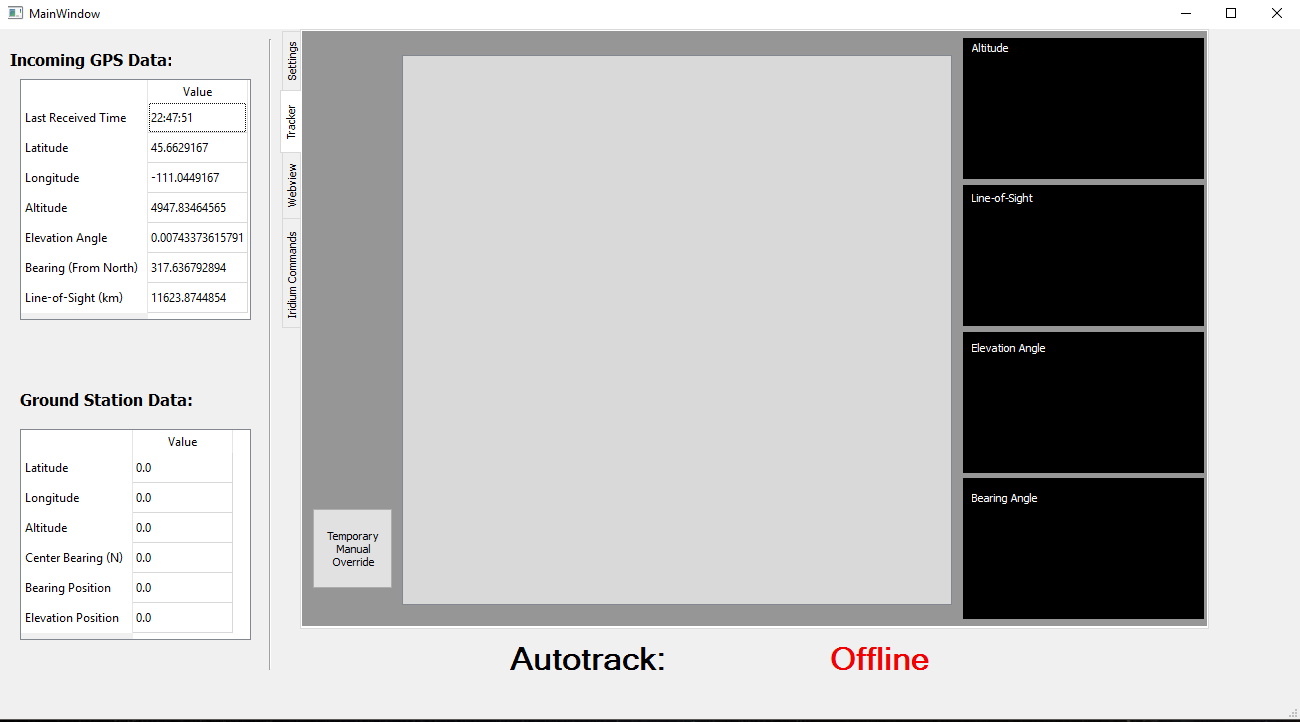
Data coming from a query on the SQL database

The basic setup has two options. You can either use the Arduino UNO and BNO055 IMU or Manual Input to determine the antenna direction and GPS position. Once you have selected either “Get Local” to use the Arduino, or a facing to do manual input, you need to set up the USB connections. The Connections defaults to turning on the Servos and the Arduino but you need to enter the COM port that the devices are operating on. This can be found by going to your device manager on Windows. The connection will show as Arduino/Genuino on newer OS’s I believe. When in doubt, watch device manager as you plug in each connection. Once you learn the COM ports for a certain device, as long as you do not change computers or the hardware, the COM ports will remain the same.



Once you’re confident of your settings, try hitting “Update Settings” on the lower right hand corner. If you selected to use the Arduino to get the ground station settings, the antenna array will begin moving to calibrate the IMU. Once calibrated, the dish will center and read the settings. This process can be watched through the shell that launches behind the GUI window. Looking at the ground station settings, if all the values look reasonable, hit “Launch Antenna Tracker”. If the settings look off, simply hit “Update Settings” again and give the GPS and IMU another chance to get the correct settings.

After the Antenna Tracker is launched, you will end up on this page (ran from home, disconnected).



Data will be plotted here while the tracker is online.

This will be green and say Online

Click inside this area and use the arrow keys to adjust the antenna’s ground station values. This allows minor adjustments for error correction of the IMU.

Manual Override will stop the iridium modem updates from controlling the antenna. I recommend avoiding using this mode without good reason.

Once the system kicks off, there should not be a reason to do more than watch the system run.

6/4/2019: **Upgrade/port to Python 3.7.3**

\_\_UNOFFICIAL\_\_ Additional Installs (pip):

* SIP
* matplotlib
* email
* PyQtWebEngine
* MySQL
* serial