

VSCADA Maintenance Manual  
ECE 492  
Spring 2018  
Prepared by Connor Nace, Geoff Watson, and Chen Xin

# Software

## Python Code

We reverted back to python for this which turned out to be a great decision. Python has great third-party libraries that allowed us to work with a variety of devices and software. Python also allowed us to develop quickly. Python is also an interpreted language meaning it does not need to be compiled before it runs. This allowed us to debug and test the program on the pi much faster. We highly recommend that python be used for future systems. If you have any problems or questions about the python-code feel free to reach out to Connor Nace (717-357-5116 or connornace13@gmail.com). I promise I will respond.

## C Code (serial communication)

Some C code was used for the serial communication with the dashboard display. Serial communication can be done in python (and we do it when reading button presses from the dashboard display), but the example code was written in C. We decided using the example code would be easiest, so we used the example code. This probably was a bad decision. This required used to turn the example code into C functions, factor those functions as python functions, and import them into getCANData.py. In the future, we would suggest removing all of the example code, and just communicate with the display through python. If you have any questions about this part of the code, feel-free to reach out to Geoff Watson (904-687-7585 or geoff.watson9595@gmail.com).

## SQLite Database

The SQLite Database is good, reliable, and fast. We were not able to implement a system for trimming the data, so this database will grow until the SD card is full. If data trimming is not implemented the database can always be deleted from the desktop. It will be recreated the next time the program runs, and the data should be stored in csv files anyway. If a mobile or web application is to be developed to see live data while the car is in motion the SQLite might need to be swapped for mySQL or something similar.

## PeeWee

PeeWee is just an ORM for python that makes creating and accessing the database much simpler. We could never use PeeWee to read data from the database, however. We could only create tables and write new entries. PeeWee is not a necessity, but it does make life much easier when dealing with the database.

## PyQt5 Graphic User Interface

PyQt5 fits closely to any Python programs as a GUI toolkit, because it is a set of Python bindings for v5 of the Qt application framework from the Qt Company. There are many handy built-in methods from Qt framework to help you build a prototype GUI in a very short period. The GUI

can be also previewed in open-sourced Qt Creator IDE, which makes it even more convenient to build the desired GUI. It is recommended to use Qt Creator or Qt Designer software to modify the GUI, so that you can select new widgets and make new fields in the GUI while looking at visual display of UI codes. Since PyQt5 is an efficient GUI tool, software can be tested from front-end as fast as possible.

## Hardware

### Raspberry Pi 3B

We kept the raspberry pi from last year and have had great success with it. It runs all of the peripherals we need. We experienced the pi slipping into low power mode like the team from 2017 did but we did not notice any changes in performance. This could be looked into but we deemed it low priority because the system seemed to be running fine.

### PiCAN 2

The PiCAN 2 is the name of the CAN interpreter we used. This thing is awesome. It has a very helpful user guide (find it on it google) which included tips on using it with python, as well as an example library which we use for CAN dumps. There is a CAN terminator soldered onto the board, which may need to be removed depending on how the network is expanded.

### USB Dashboard Display

The CrystalFontz dashboard display we have uses serial communication with the Pi to display vital car information to the driver. The display itself is awesome and the manufacturers have a great datasheet available for it. They also included an example library written C.