IOT Project 2020 – Progress Report

NOTE: *This page is just a text based progress report, and glosses over a lot of details. If you want more details about any problems I encountered, consult the Issues and Mitigation document, and if you like images take a look at the gallery.*

6th March: Initial setup and testing of weather station equipment. Software was already installed on the pi from a similar project, but I followed the updated guide and reinstalled most of the software. Equipment worked as expected, readings were showing up on the Pi when the equipment was plugged in and manipulated, but the SQL software and MariaDB that was previously installed on the Pi was interfering with my fresh install, causing an error later on that I could not work around. A clean install of Raspbian or Debian would be needed.

29th March: Got a memory card reader and preformed a fresh reinstall of Raspbian, without the software pre-packaged (There was an option to get a version with all the necessary software pre-installed and organised, but I decided against that to get a bit more experience working with Linux and packages). I started to install and test the software and hardware as I went along, following the instructions outlined on the Raspberry Pi weather station guide: https://projects.raspberrypi.org/en/projects/build-your-own-weather-station and other similar guides for clarity when minor issues were encountered. The equipment is working expectedly, and I managed to bypass the issue with the MariaDB and SQL from the previous attempt. Some more software work needs to be done, and some minor issues need to be fixed here and there, such as the real time clock (RTC) but so far most of the software and hardware is working.

7th April: I decided to try fix the issue with the RTC. I spent a while attempting to troubleshoot and fix it, but to no avail. I eventually decided to switch from Raspbian Buster to Raspbian Stretch in the hopes that it would solve my issue due to it being the version used for the tutorial. After many attempts at stuff, such as reinstalling the Distro, trying different versions, installing extra packages, I just could not get the RTC up and running, so I called it a day.

13th April: Found a solution to my issue with the RTC (This and any other issues are outlined with my solutions in the issues and mitigation document.) I implemented this solution and continued to work along with the rest of the software, eventually ending on the Automating updating of the database.

22nd April: While working on the Automating updating of the database the micro SD card ended up becoming corrupted and so had to do a reinstall and redo the software I had already set up. Luckily I had scribbled down a handful of small notes and knew the procedure, but the corruption really killed my motivation for working on the project for the day and with other exams fast approaching I had to call it off for a while.

1st May: After finishing up other business I sat down and got to reworking the software. Soon I was back to where I finished off last time. I decided at this point It would be a brilliant idea to make a copy of the micro SD just in case of another catastrophic failure. I used Win32DiskImager to create a copy of the SD card on my PC. This process took me about 15

minutes. I then ran into an issue with the log_all_sensors.py script. MySQLdb couldn't be imported. I tried using PyMySQL as an alternative, and while It did get me a tiny bit further into the project, It did not work, and I didn't fancy rewriting an entire script for a new package, so I kept looking and trying to install other packages or trying other solutions in the hopes that something would work. After a while I decided to finish for the day.

7th May: After some more exams I got back to work. While trying to solve my previous days problem, I found a solution. It was a simple package install which worked and I was then well on my way to concluding the project. I decided at this point It would be a good idea to sign up to the Oracle DB to log my data, It said it would take 24 hours to get my authentication code. I soon ran into an issue with the scripts and couldn't figure out what was going on. I decided I could sort it out tomorrow when I get the code from the Oracle DB.

10th May: Checking in on the emails awaiting the authentication code I found nothing yet, so I decided to register again in case there was an error. I also tried getting my head around what was wrong, and eventually discovered it was a simple syntax error since some of the module keywords had changed. After kicking myself over this I fixed the issue and went about testing the sensor logging. The sensors all logged perfectly fine, so I called it a day.

12th May: I checked my emails again, and still nothing back about an authentication code. I decided to register for a third and final time and in the meantime try and use a different service. I would have to write scripts in python to connect to this new service, but I was confident I could do it easily enough. I chose Wia and set to work, following school lab notes and online tutorials, but I couldn't get very far.

15-17th May: After a final exam It was time to finally finish up the project. I spent some time trying to get a better grasp at python and understanding how to upload the data to Wia. The task seemed simple, and I consulted many guides, slides and even the scripts for uploading to Oracle DB to see what was done in there. I of course checked my emails to see if I received that code and to my surprise found nothing. The manipulation of data from databases was a lot more complicated than I had first anticipated. I had to wrap my head around both python (I knew the basics, but not much else) and databases like MariaDB (and the modules like the MariaDB connector and lots of SQL stuff). In the end I could not get the Project online in time, however the device does work and collects data, and its only a matter of putting a little extra work in to get it up and running!

Ill make sure to update this Repo with any developments!