**Memo**

To: Professor Pisano, Professor Hirsch, Professor Osama

From: Ben Livney, Maxine Loebs, Sergio Pareja, Emanuel Perez, Noah Spahn

Team: 27 - Plants are Neat

Date: 4/9/21

Subject: Final Test Report

**1.0 Required Software and Hardware Components**

1.1 Hardware2 Adafruit Feather M0 microcontrollers

* 1 custom PCB
* 1 enclosure with sensor probes and peripherals
* 1 USB - micro USB cable

1.2 Software

* Arduino IDE
  + ServerClient
  + MotherNode
* RadioHead.h
* RH\_RF95.h
* MatLab IoT ThingSpeak
* Node JS
  + SerialTest.js
* Batch Files
  + PANinstall
  + runMESH

**2.0 Completed Tests**

2.1 Network Tests

The test had three microcontrollers communicating with each other (depending on their individual role). The mother node sent a request to each node for sensor data. Each outer node received the mother node request, collected data from sensors, and sent the data back to the mother node. The mother sent the collected data to the computer via serial port.

2.2 Network Conclusions

The test demonstrated communication between the nodes successfully. Collected data from the outer nodes were accurate, and it was sent successfully to the mother node without any package lost. The mother node uploaded the data via serial to the computer. The computer received the data without any corruption (the same string as the string sent by the outer node).

2.3 Console Tests

The test here is composed of three different files: 1 nodejs file titled SerialTest and 2 batch files labeled runMESH and PANinstall. The SerialTest file is for reading from the COM ports to interact with the mothernode to send information to the IOT thingspeak channel. The batch files are for the users convenience, PANinstall installs all NodeJS dependencies they need and the runMESH runs the SerialTest.js code.

2.5 Console Conclusions

The test demonstrated proper acquisition of data from the mother node and correct updating of online elements. The serial reading works properly and the information is stored within the computer almost instantaneously. The batch files also properly run and install everything, same as if using the command line to install all NodeJS dependencies but faster and easier.

2.6 Power

Unfortunately, logistics and planning did not allow for us to gather all of the necessary materials to perform a battery-operated networking test. We could, however, show that the power path management electronics work well while using only solar power. We were able to show in the second prototyping test successful power path management with solar and battery power, without the networking component. Powering the featherboards allows any sketch to run on the processor, and we have run networking sketches on the featherboards while being powered from battery power via the power electronics, but have not updated data to the console using this setup. We anticipate that it will easily be possible to add the battery power to the networking, as we have demonstrated all the necessary components before.

2.7 Website/Notifications Test

The test had a single channel with five fields that were reading from the things peak API. This channel was cleared of all data so new data could be read into the track when the test was initialized. Data was demonstrated over time graphically with point-slope graphs and real-time gauges displaying the most recent data point. Notifications are handled with a Matlab script checking fields whenever variables are uploaded. This is similar to the notification function as it triggers a Matlab script when no data has been read into the channel after 30min of no activity. These visualizations received data from the medium and would update when information was successfully uploaded to the track.

2.8 Website/Notifications Conclusion

During the test, data was uploaded to the channel and demonstrated live updating on the track. When we pushed the fields outside of the thresholds, we got related communication in an email notifying the datapoint breached and the node it has been breached on. Similarly, when we remove a node for some time, we see a broken node notification trigger after the channel hasn't received data for over-the-set testing.