**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: 4/5/21

Subject: Final Test Plan

1. **Abstract**

For our test, we have decided to show our progress in a few key areas of our project, namely node communication, power, and data display for the user. We will show that our nodes are able to communicate with one another and transmit data between LoRa modules to the gateway node, which can upload the data to be accumulated, parsed, and displayed by the website.

**2.0 Required Software and Hardware Components**

2.1 Hardware

* 2 Adafruit Feather M0 microcontrollers
* 1 custom PCB
* 1 enclosure with sensor probes and peripherals
* 1 USB - micro USB cable
  1. Software
* Arduino IDE
* MatLab IoT ThingSpeak
* RadioHead.h
* RH\_RF95.h
* Node JS
* Batch Files

**3.0 Communication + Collect Data + Visualization**

3.1 Setup

* Upload all relevant code to each microcontroller, and make sure other website/node.js code is running.
* Insert sensor node 1 into the custom PCB.
* Plug in a 5V wall adapter to the PCB to simulate a solar panel.
* Attach the battery pack to the battery leads on the PCB.
* Initialize the Matlab ThingSpeak API via Node.js and open the website in a browser.
* Plug the gateway node into a laptop via a micro-USB cable

3.2 Network Procedure

* Mother node requests data from sensor nodes
* Sensor node receives request, collects current data from sensors, and send data to mother node
* Mother node receives data and sends it to computer
* Node JS server parses data and uploads to remote website
* Remote website displays the data in graphs

3.3 Console procedure

* Initialize the Node.js script that uploads data points to the ThingSpeak API.
* Open relevant gauges and graphs to verify data is being collected properly.

3.4 Network Criteria

* Poll is successfully sent to other nodes
* Outer node successfully sends uncorrupted sensor data back
* Mother node successfully sends to the mother computer

3.5 Console criteria

* Graphs should update with the data corresponding to the data points that are printed in the node.js script periodically.
* Data should reflect changing conditions (i.e. shining a light on the light sensor should increase the lux value).
* Website should display an error message when it hasn’t heard from a particular sensor node after 1 minute (adjustable).

3.6 Power criteria

* Test should be repeatable on “solar” power and on battery power and data streaming should not be interrupted.

**4.0 Conclusion**

After the conclusion of our tests, we determined that we are able to successfully transmit data between sensor nodes, implement a sleep wake system that bounces data between nodes, operate all electronics on battery and solar power, upload data and generate visualisation with collected sensor data. Our next task for our group is to finalize the optional features of our product and prepare it for customer installation and ECE day.

