Interim Report:

<u>Completed</u>
1. <u>Drivers</u>

-Joe Adamczyk - wrote drivers for interfacing with the sensor. Since the sensor used a custom 1-wire interface that had no native support on the Pi, I made a custom driver to talk to the sensor, based on the protocol specified in the device's datasheet. This data, a 40-bit binary value, consisted of relative humidity, temperature and a checksum. I wrote algorithms to separate this bulk value into its constituent parts, and then converted them into integer values. These values are then stored in a text file as two values separated by a new line.

2. <u>Python Scripts</u>

-Jonathan Bing has registered the Pi's to the google API console giving access to the Service Account Key. With that key and json file wrote a python script that automatically logins into the spreadsheets and uploads static information to the sheet.

3. <u>User Interface (website)</u>

-Joseph Echeverria designed the web app using Angular 5 (HTML and CSS)

-implemented the web app to display collected data (line graph) and update the data in real time from Google Sheets using the Google Charts API and Javascript.

<u>To Be Completed</u>
-Need to update spreadsheet.py to upload the correct temperature data collected by the drivers.
-Implement plan to allow seamless integration of new Pi
-Design a method of adding new charts and devices on the web app