**Team: NoiseHub (Team #8)**

**Team Leader: Allen Zou**

**Next Team Leader: Ibrahim Chand**

**Goals:**

* Establish fully working scripts for our sensor suite
  + Lidar Tripwire
  + Audio Sensing
* Refine our product to feel more “finalized”
* Set up a more permanent home testing platform for our system
* Second Prototype Testing

**Roles:**

* Benjamin Brewer - Sensor Scripting
  + Lidar Tripwire
  + Setup at home testing environment to simulate real use case
  + Raspberry Pi casing research
* Ibrahim Chand - Sensor Scripting
  + Audio Sensing
  + Soldering sensors to the Pi
  + Study space search functionality
  + Mobile application implementation
* Alex Prior - Sensor Scripting
  + Lidar Tripwire
  + Audio Sensing
  + Setup at home testing environment to simulate real use case
* Allen Zou - Sensor Scripting
  + Lidar Tripwire
  + Soldering sensors to the Pi
  + Enabled system compatibility on campus

**Progress:**

At the beginning of my time as team lead, our team had our 2 lidar sensors, audio sensor, and thermistor connected and pulling data repeatedly to the Pi. A significant portion of the work we did was dedicated solely to taking those sensors and creating the algorithms to make the data useful and human readable.

Our first step after verifying that sensors were functioning as expected was to create a more “real life” testbed at home. Up until this point, we had only been testing the lidar and audio sensors on top of our minifridge with poorly connected jumper wires, just to ensure we could attach and pull data from the devices. To set up our system, we soldered both lidar sensor and thermistors to all the corresponding pin outs. We used long, flexible wires and heat shrunk the connections for an overall more polished feel and secure, stable connection. On the other side of things, we went to the lab to measure the average door sizes and deduced that the most realistic setup was with the two sensors 3’ ½” apart and around chest height to pick up the most accurate body signal. They then attached them to our hallway at home for testing.

For the dual Lidar system, the tripwire algorithm took up a lot of time to accurately design. I will discuss the troubleshooting that we did in the next section but our final algorithm uses three states: Idle, Entering, Exiting. The system starts in an Idle state and samples at a rate of 25ms. Depending on which sensor measures a distance less than our threshold (around 85% of full measurement), we would know if the person is exiting or entering. Then, the system will switch to the corresponding state, increment or decrement the headcount, and sleep for 0.75s. This timeout was determined after a lot of testing and calibration to account for a variety of walking speeds and repeated entrances. After the system sleeps, it returns back to the Idle state and repeats the above algorithm. This gave us a pretty solid headcount accuracy and we will continue to verify its validity.

For the audio sensing, we are using three states: Low, Medium, and High to denote the volume of a room. The current script starts at a low state and defines an explicit threshold for RMS values. It detects how many times the sensor exceeds the threshold in a set time interval and will increase or decrease the state accordingly. The microphone will be placed on a separate Pi that is closer to the middle of a study space to maximize accuracy for what a student will experience.

For the prototype testing, we will be presenting the lidar test and audio sensing scripts and use the senior design lab setup that we have created.

**Issues:**

The main issues during this team lead section was, as I stated above, the sensor algorithms for Lidar and audio.

For the Lidar tripwire, our issues mostly revolved around inaccurate sensing of bodies. When we began writing the tripwire, we were not sensing fast enough to accurately determine which sensor was being tripped first. We also tried many algorithm layouts, one of which was where we would trip a state and then wait for an untripped second sensor to change headcount. This complicated algorithm caused more issues than not and it could have been because it was too computational intensive for the fast sample rate. As a result, we decided to opt for a much simpler algorithm that relies only on the first sensor tripped and uses a timeout instead of further checking sensors for the completion of an exit or entrance.

For the microphones, we had many issues due to the hardware being very limited. The sensor was not very sensitive at all and had extremely noisy data even when it was in a silent room. The threshold separating low and medium volume was very small. This could be just due to the microphone being very cheap but it will work for our system. By defining the three states of Low, Medium, High and calibrating the thresholds, we are able to at least come relatively close to accurate readings.

**Progressive Measurement Methodology:**

We have continued using the Trello board to keep track of progress. It has also been especially useful for referencing how we set up a system from weeks ago. We have had to reflash our Pi and having the resources labeled on Trello made the process a lot easier since we had it documented.

**Work for Next Period:**

* Refining how we send headcount, audio levels, and temperature to timestream and the user app.
* Fully linking the front end to the database
* Calibrating our sensor systems and ensuring it is accurate in the lab setting as well

**Personal Assessment:**

Our team has still been able to work very efficiently. No one has to be told exactly what to do and we just pick up tasks or work together on specific action items during our allotted senior design periods. The only thing I really needed to keep track of as team lead was deadlines for IDRs and making the presentation.