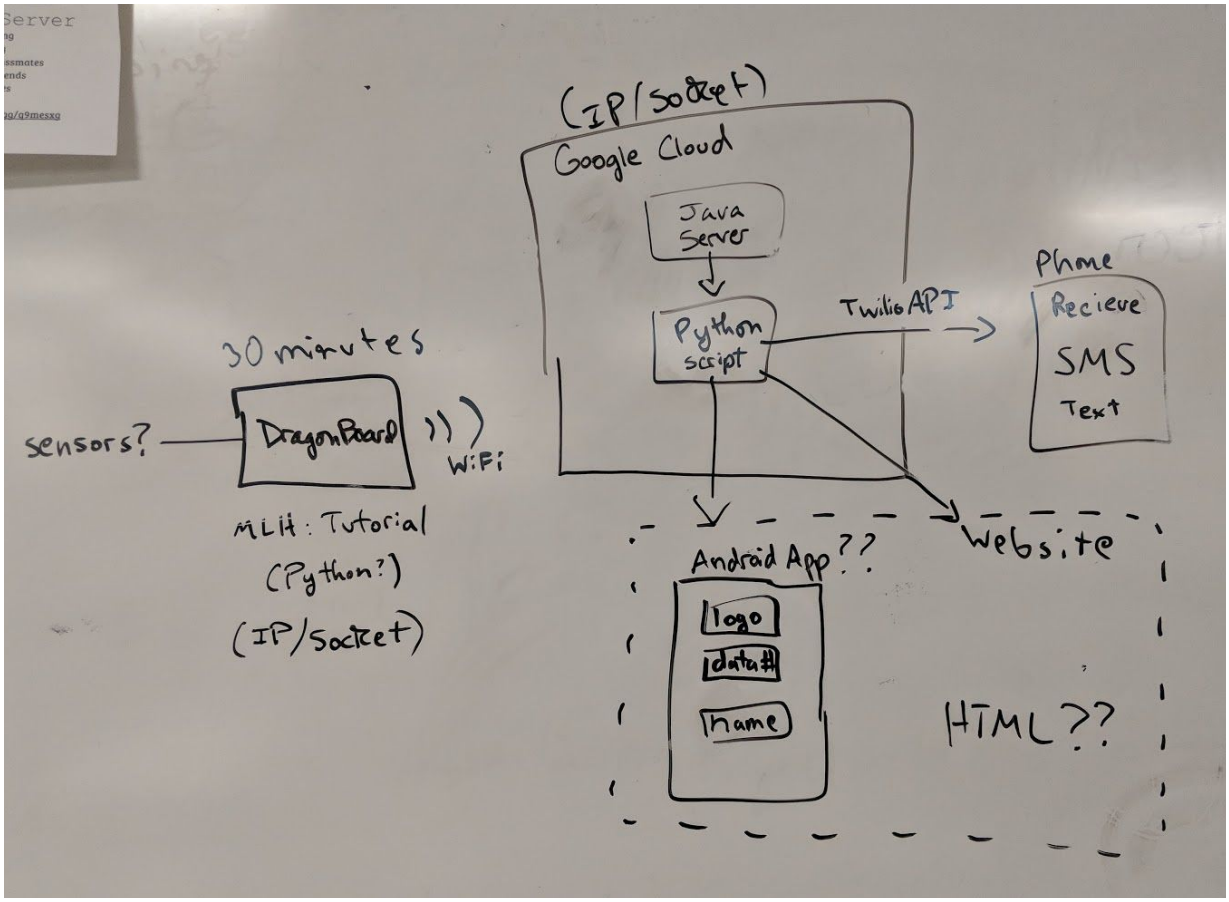
Step 1: Putting the Team Together

It happened on a fateful Friday night during Idea Jam 2: Electric boogaloo where Nigel, Jakub, and Zenry decided to form a team and make a boat able to detect when someone has peed in a pool.

And so the Hackathon begins and the gang makes a plan.

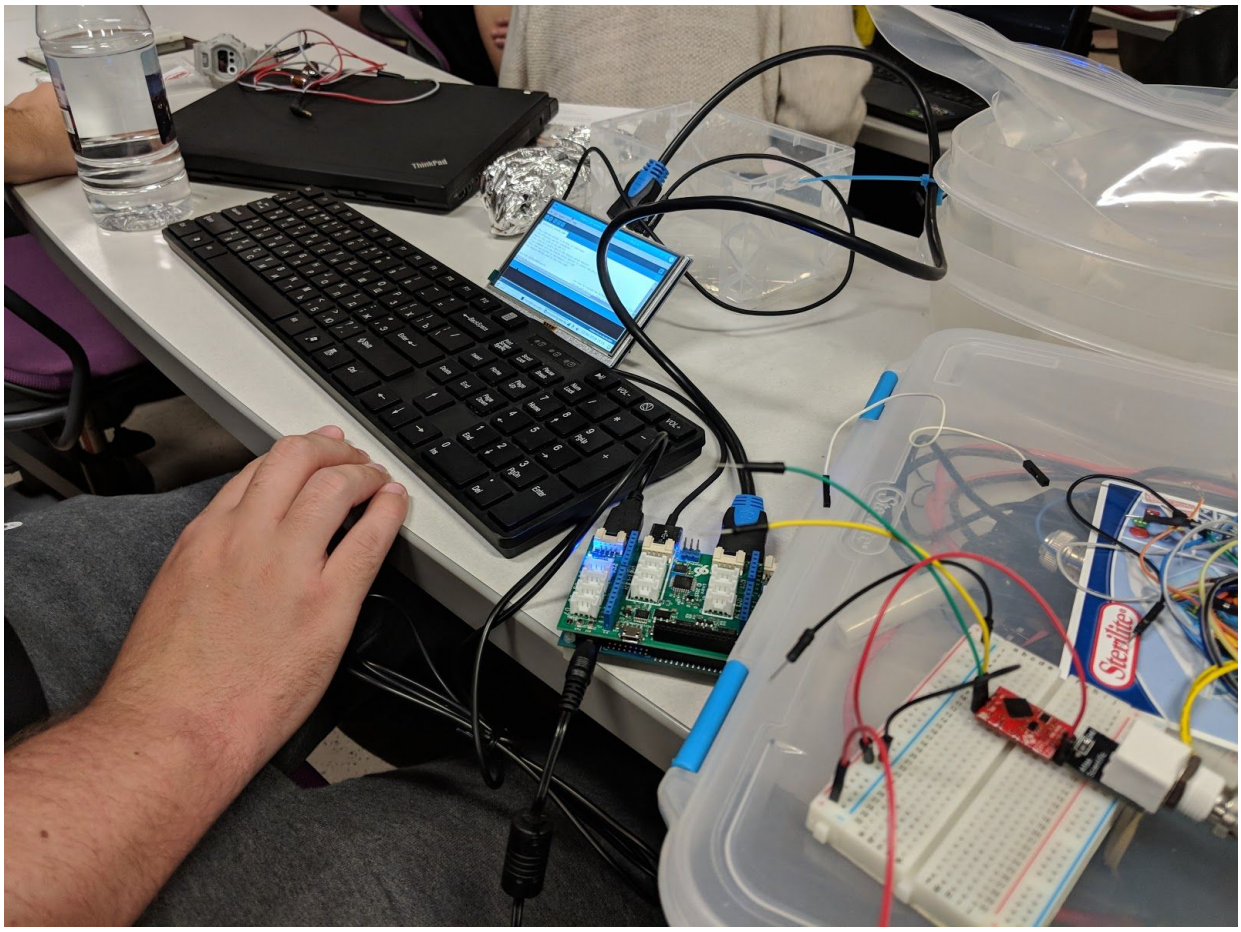
Step 2: Planning

The gang comes up with ideas on how to make this hack super cool and prioritizes what should be done as part of the essential hack. Their big hack idea is to make a pool boat that monitors the pH of the pool and sends the owner a text when the pH shift beyond what is considered normal (~7). Urine can be detected within the pool as it tends to be acidic, resulting in a shift in pH, which the owner can then be notified of.



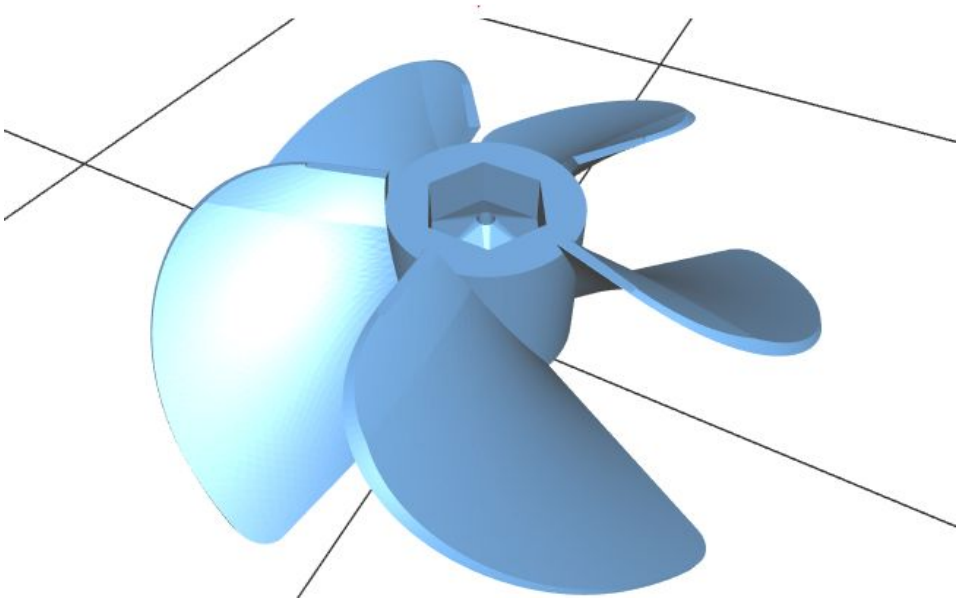
With an idea here and a helpful advice there, the gang slowly finishes up their brainstorming of how they should proceed and begin to focus on using the Dragonboard to manage the pH sensors and sending the data to their Google Could Server so that they can text the messages.

Step 3: Hacking!



In order to manage the DragonBoard 410c and Arduino Uno, the gang at first used a tiny screen and a mouse-less keyboard, relying on binding the numpad to the movements of the mouse, but added on a USB hub to use a wireless mouse (cause keyboard commands were a pain in the butt). Once focus on building the software was in motion, the gang decides that we can't all use the keyboard and tiny screen so we begin work on the propeller aspect of the hack, specifically by finding and making some 3D models of propellers, boats, and motor fittings.

After some realizations that we wouldn't be able to fully print out a boat model, we improvised and decided to only print out the propeller.



We needed a lot of water (for drinking).

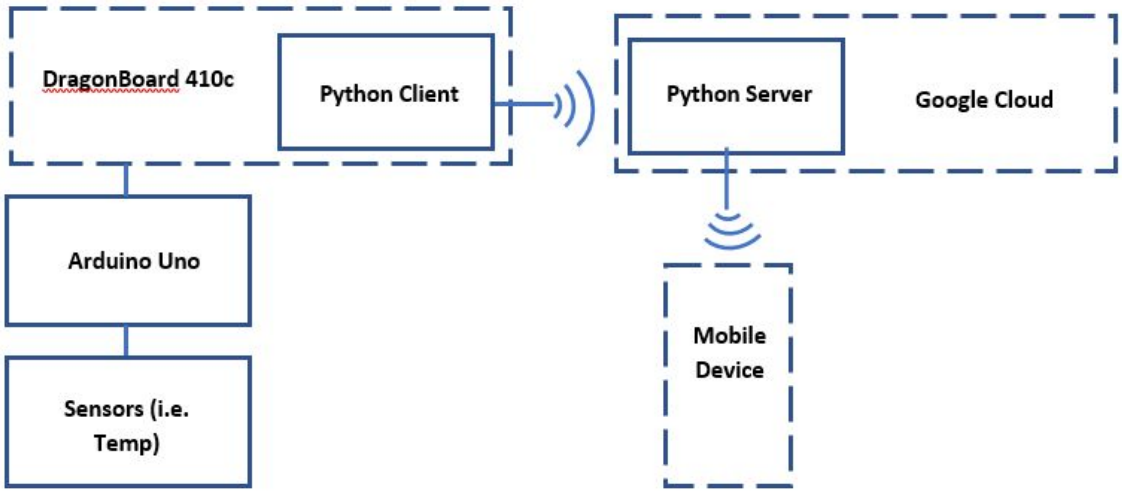
So after a lot of twists and curves thrown when dealing with Arduino Uno, we eventually decided to make a complicated hardware design that would be able to automatically send data to the Google Cloud! Due to a faulty pH meter, we were not able to pursue the pH sensor technology so used a temperature sensor instead. Eventually we finished programming our Arduino and setup communication with the Debian Linux RTOS on the DragonBoard and after a few Python hiccups we were able to relay temperature readings from the Arduino through the DragonBoard and right into our Google Cloud based Python server. To make life easier, we set up a cronjob (a system command that is run periodically) to automatically connect to the Cloud server whenever its available (~1 minute intervals). In our cloud Python server we used a Twilio API integration to

send an SMS text to a person’s phone when the water reaches a desired temperature. Having finished with software, we continued on by completing the boat.

This of course had to be implemented by a foam pool noodle as a form of buoyancy. We 3D printed the boat propeller a few times due to the motor shaft not fitting properly. We constructed the boat out of popsicle sticks and hot glue. We also didn’t have a knife so we had to cut the hot noodle with a piece of 3D print plastic (innovation at its finest).

Step 4: $$$

And so you have reached the end and are now witnessing our results! Watch and be amazed as we tell you the temperature of our custom made hot tub through our amazing floating IoT WiFi smart pool device! See the next page for our final block diagram and prototype!


DragonBoard 410c
Python Client
Python Server
Google Cloud
Arduino Uno
Sensors (i.e. Temp)
Mobile Device