Matthan Myers

Quadrotor Design - **Software**

**Overview**

For software, the most important things I will be doing will be the storage of data and the visualization of said data. I will be working most closely with Cameron because he will be working on the backscatter/visualization portion as well.

**General Design**

We have proposed using a NUC aboard the quadrotor, but, depending on the weight of the NUC compared to the quadrotor’s bearable load, we might use something smaller. Regardless of what we use, I want to minimize the processing power needed for storage (though it shouldn’t be a lot for simply storing the data), meaning this data will probably be pretty “raw”, but it will still be storable/sortable once I get the “base” going. The best way to transfer this data to the server/base is most likely via the docking station/plugging into the “home base” computer. It really is only a matter of preference. We previously decided that using cellular data to transfer the information would simply be too much dataflow wise and power wise.

At the “home base”, a simple MySQL relational database would most likely suffice, as we could store different data points based on a “grid” or coordinates of the location that data was taken, along with the dates and geological location. Think of a chess board that is located at a very specific GPS coordinate, from which we can assemble the data into visualized information. This leads into the last general design point, visualization. We would take that data we were given and turn it into a “terrain map”. Using a certain color, a lighter shade might be higher and a darker shade might be deeper.

**Steps in Rank of Importance**

The first step would be putting together the temporary storage using C++ which can later be placed into the database. If we can’t save the data onboard, our project will be useless.

The next step would be setting up a relational database to store the data in, most likely using an industry standard program like MySQL.

Next, I would set up the docking procedure so that the temporary data can be transferred from the quadrotor to MySQL such that it can be automatically stored and sorted based on each mission or grid.

Lastly, I would complete the visualization part. Though this is arguably the most important part in terms of results, it depends on every other part to work before it becomes relevant, hence why it would be the last step for me.

**Languages**

I will probably do most calculation/temporary storage in C++ since it’s my most confident language and is a solid low level programming language.

Working with the actual database will require using SQL, something that I have taken a class on before and worked with in the past as well.

The visualization part (graphics) is probably my area with the least experience. For this, I will probably use C# due to the fact that C++ is horrible for graphic design due to its lackluster libraries. C# will provide a much more stable solution. Java is also a (longshot) possibility, but I’d prefer C# due to it being a direct descendent of C++.