Reflections Paper

John Wesley Hayhurst

December 2015

1 Introduction

The capstone project in my eyes was a great success. The final presentation was amazing; nothing failed on us,everyone seemed impressed, lots of people liked our poster, and the most important thing of all is that Paul Tribble tried out our project and was impressed. Our Project to sum up all the sum of the working parts was using a node.js server to communicate across various devices to display a NAO's vision onto an Oculus Rift and to translate accelerometer data into robotic head movements mimicking that of the users. The thing that I really took away from this project was an agile development cycle.

2 Tools and Resources

Like i said before i learned a lot about the inner workings of a project and how to start a project from the ground up and get it working. The first thing and most important thing that I learned was setting up a c++ SDK for use with the NAO robot. This was one of the most challenging components to this project. The NAO SDK was so poorly documented and the instructions were out of date it caused me to learn from scratch to how set up a computer environment so that the libraries are properly linked and able to be used on a global scope. The next thing I learned was how accelerometer data can be transformed into radians. The major problem is that the accelerometer data when gathered is not in radians it is in radians over time with respect to acceleration. When researching I learned that a lot of research and development has gone into translating this data into radians for normalized use. Instead of reinventing the wheel we used a node.js library called node-hmd which is a full oculus SDK implementation in node.js. I also learned more about node.js and just how versatile it is when it comes to writing scripts and servers. This is because node.js runs of google chrome's v8 engine and is a hybrid of c++ and java script. The last tool that i learned about was gstreamer which is a command line utility that streams image data gathered from any media mounted device. In this case the media mounted device was the robots built in camera mounted to /dev/video1. This is a powerful tool to stream data through either TCP or UDP and is pretty common to find in any package manager. This tool is so common for video streaming that it came included on the robot. This overall process has taught me that the agile development cycle is very powerful. It allowed us to get a baseline implementation of all the components required for the project and test them. Once we have that baseline to work off of we are then free to tweek and improve the components that we know are working allowing for a baseline to fall back in case of problems.

3 Conclusion

Overall this project was a rousing success and I learned a lot from completing it. I honestly dont know if I would change anything to make this project better. Looking back at the tools and how fast me and Juan were able to implement everything I wouldn't change a thing. Looking at the tools that are available to us now I would only really change one thing. A couple of weeks ago the company that makes the NAO robots released a JDK for the NAO. This would have made our project faster as a good portion of the time was working on understanding the C++ SDK and getting it to work for the NAO. The JDK would have also allowed us to use one central computer that would allow us to reduce latency more and better mimic human head movement. Given the tools at the time and how quickly me and Juan were able to tackle such a big project I would have not done it any other way. Me and Juan are extremely proud of our work and to call this project our capstone.