SEER Optics

S. Fick, EE, D. Gabriel, EE, D. Shosho, CSE, and O. Terry, EE

***Abstract* — SEER Optics looks to aid civil servants by providing an advanced visual aid allowing teammates to be seen in real time. Missions conducted by civil servants like Firemen and Police Officers are often very hectic and confusing environments to be in. This can hinder missions and cause them to become more dangerous. Our design will allow everyone wearing this device to see the locations of each other even with obstructions in the way and allow these users to call out or "ping" objects that they are seeing. This design sends GPS locations over radios and uses the difference in GPS coordinates to find relative location. This location is then displayed on augmented reality optics as a green dot. This allows the user to see their environment with the enhancement of having these green dots displaying the locations of the others. Our MDR deliverable makes the use of a Google Pixel with the camera running real time to show this augmented reality.**

I. Introduction

Our product, SEER optics, will be designed to provide a visual aid that takes in teammates GPS data through radio and displays where they are. in addition, the product allows users to share a marked location using GPS data.

A. Significance

Civil servants (i.e. firefighters, policemen, search & rescue) suffer from a lack of situational awareness when conducting their missions. This creates a fog of war that leads to friendly fire, team members missing in action, delays in missions and other unnecessary issues. [1][2][5]

B. Context and Existing Products

Currently there are three main solutions that try to solve our problem but fall short in aspects that our product will succeed. These solutions are verbal communication, line-of-sight tracking via reflectors [3], and blue-force tracking [6]. Speech is a great ally for these civil servants during their missions, but in the advent of disaster you can't depend on mass infrastructure to keep teams coordinated [4]. Similarly, reflectors are great for short range missions with little obstructions, but this solution is specific to low-light/high-light (fire) situations and doesn't work well in situations with abnormal environments like collapsed buildings or forests. Finally, blue-force tracking is the closest competitor to our product, being that it offers a lot of the same benefits. The only issues with this solution being that some form of verbal communication is still required, and the team members depend on a central command operator that coordinates the team.

C. Societal Impacts

With our product, these civil servants can remedy the short comings of the other currently employed solutions. Our solution operates on personal, closed-system hardware that updates the individual with real-time whereabouts of every team member and their pinged locations. This combines the team coordination of verbal communication, the individual real-time visuals of reflectors, and the technology used in blue-force tracking. See Figure 1 below for a rudimentary depiction of the system.

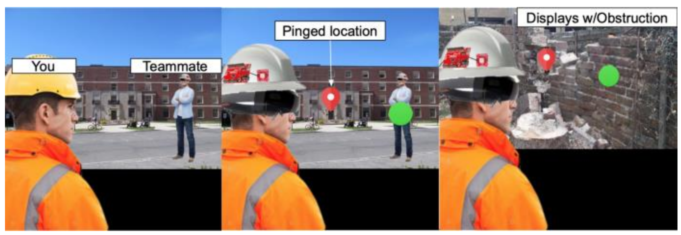

Figure 1: Depiction of the system in practice

D. Requirements Analysis and Specifications

As for the specifications of our product, seen in Table 1 below, we aimed to provide the most accurate and efficient system we could devise. For example, we chose to limit the range on our device to the specified due to inaccuracies of our GPS chip at the query timing we settled on for speed. When constrained to this range, we are able to achieve our desired display accuracy 95% of the time. To that end, we have plans to gather test data that shows we meet these specifications to the accuracy we claim. Additionally, the battery of the Google Pixel is rated at ~5 hours when using it constantly, this is the minimum battery life we want and aim to be at double or even triple that. And lastly, the Google Pixel is rated at IP53, which means it is protected against dust and should be unphased by a light rain. Our Node will at least match this specification if not supersede it.

S. Fick from Pittsfield, MA (e-mail: sfick@umass.edu).
D. Gabriel from Portsmouth, RI (e-mail: dgabriel@umass.edu).
D. Shosho from Revere, MA (e-mail: dshosho@umass.edu).
O. Terry from Amherst, MA (e-mail: oterry@umass.edu).

Requirement	Specification
Seer Node sees teammates with or without obstruction	Range of 25<x<300m
Node can ping locations relative to them	Range of 0-50m relative to originator
Seer Node sees pings with or without obstruction	Range of 25<x<300m
Displayed teammates and pings will be accurate	Within specified range at +/-5 degrees
Node will have long-lasting battery	Node will last as long as the Optics
Teammates GPS data is updated quickly between Nodes	Location is queried <500ms
Node will withstand moderate weather	Up to the standards of the Optics

Table 1: Requirements and Specifications

II. Design

A. Overview

Our solution involves allowing users to see their teammates locations in live time through augmented reality optics. To determine the relative location, we will use GPS location data and gyroscope technology. The user will also be able to ping objects that they would like to mark. To show this location we will use a Google Pixel showing what is seen from the camera plus dots overplayed showing locations of the nodes. We also have put a lot of consideration into using the Microsoft Hololens as our augmented reality optics but have made the design decision to go with the Google Pixel based on our design to interface with the AR Optics. The full layout of our design can be seen in Figure 2 below. To satisfy the above specs in Table 1: we needed to specify a range in which we limited the amount of error on the optics, this lead us to limitations based on the GPS chip not being accurate enough below 25m and the radios not being able to transmit and receive reliably past 300m; due to the range in the former spec, we believed it to be best to limit the range of the ping controller to 50m; the best accuracy and latency we could achieve with the hardware we had was why we chose to spec the accuracy and latency the way we did; finally, since our optics was a Google Pixel, we chose to conform the specs of the rest of the system to it such as water resistance, dust resistance, and battery life.

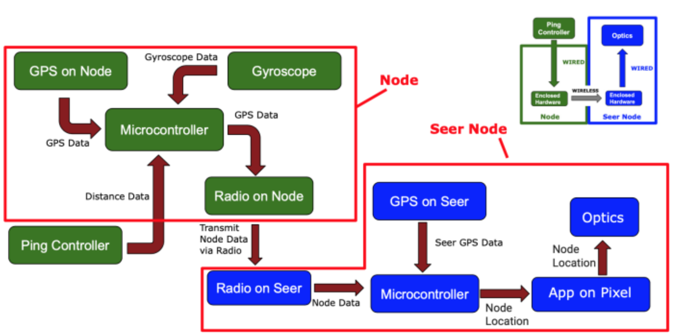

Figure 2: Block Diagram of the SEER Optics system.

B. Node

The Node will be the hardware that is carried by the person that is being seen by the SEER Node. This will include a GPS chip, microcontroller, Gyroscope, and a 915MHz radio transceiver. This Node is untethered and powered by a 9V battery. This allows the node to have a longer battery than the limiting factor of our project which is the Microsoft Pixel. This microcontroller will take inputs from the Ping controller which will be an analog input, The GPS chip, and the gyroscope. This info will be packaged by the microcontroller and sent over the radio where it will be received by the SEER Node.

C. SEER Node

The SEER Node will be the user that is able to see the Nodes location in real time on the Google Pixel. The data from the Node will be received over another 915Mhz radio receiver and will be processed by the microcontroller on the SEER Node. This SEER Node has its own GPS chip. The microcontroller compares the data from the Nodes coordinates and the SEER Nodes coordinates to calculate the location of the node relative to the SEER. This data will then serially be sent to the Google Pixel through a micro USB.

D. Google Pixel

This is the device that we have decided to use as our augmented reality optics, seen in Figure 2 below. We have developed an app for the Google Pixel that will take in the location and ping data from the SEER Node and display it as an overlay on live video that is being seen through the phone's camera. This app was developed using Unity software. This app will show relative location and distance of the node and the objects that the node pings.

E. Ping Controller

The Ping controller is a piece of custom hardware that we have designed. This will be wired into the Node so that an analog voltage can be read off this device. The interface can be seen in Figure 3 below. This controller has four buttons and one dial. The buttons can be used to mark different types of objects, up to three different types. There is also a button to clear all of the pinged objects. The dial can be used to estimate the distance away from the user that the object is. This dial can be scrolled to show 0-50 Meters. When a button is pressed the Node will take the info about the distance from the dial and the type of object and send it over the radio to the SEER Node. The direction of the object will be known by the gyroscope on the Node so the Node must be facing the object when pressing the ping button.

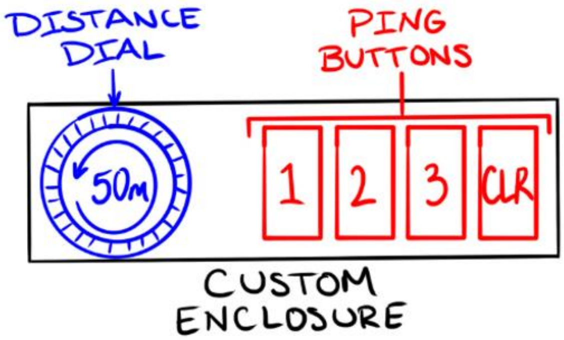

Figure 3: Ping Controller exterior design schematic.

III. The Product

A. Product Overview

Our final product consists of two nodes. One of these nodes has the ping controller and can ping objects, the other has a set of AR optics. This node with the optics can see the other node walking around and see the locations that are pinged. The optics are in the form of a Google Pixel running our app inside of a modified VR headset to fit our application. Our Node design can be seen in Figures 4 & 5 below. The final ping controller design is also seen in this figure.

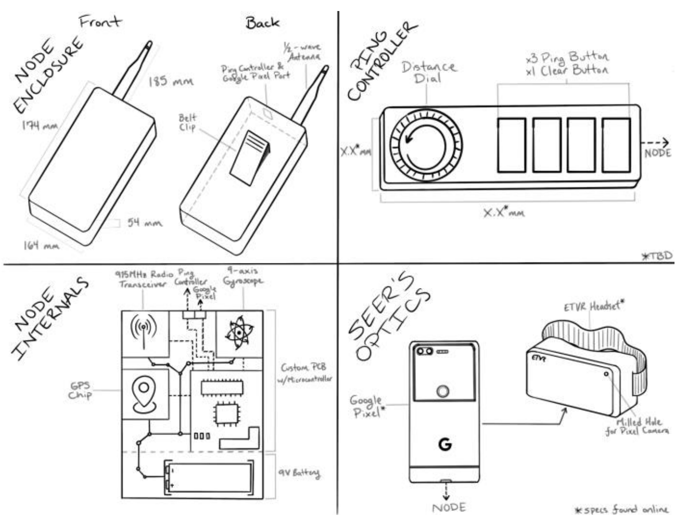

Figure 4: Visual Product Overview

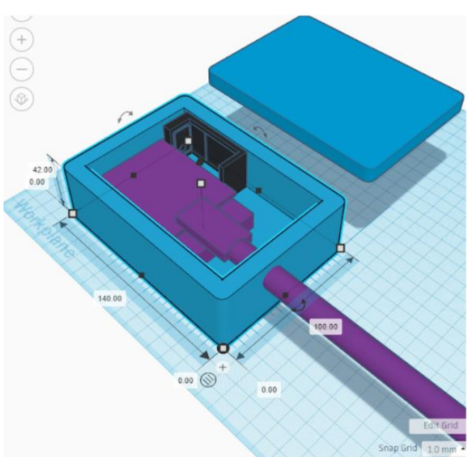

Figure 5: 3D model of Node

This ping controller is connected to the Pinging Node and the optics are connected to the SEER Node. Overall, this gives 4 pieces of hardware split between the two people using our product. The other main product that we have developed is our AR app for the Google Pixel. This app allows us to show our project through live use of the camera on the Google Pixel. This app shows the Pinging Nodes live location and any pings that have been made. A compass has been added to help orient the user and show where the objects are. Figure 6 gives and overview of what the user will see.

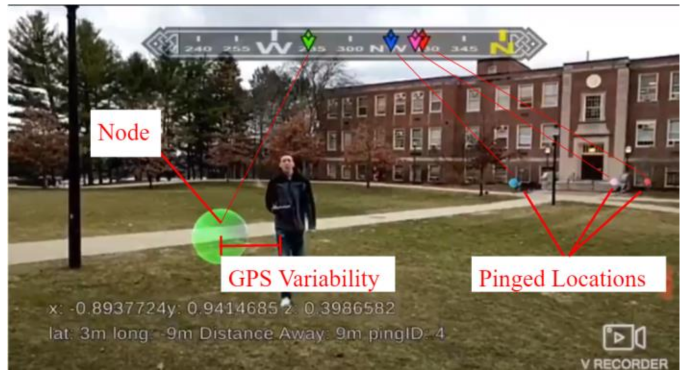

Figure 6: SEER Node View Through Optics

B. Electronic Hardware Component

For most of the semester we did our product testing with breadboards. We did not start working on our printed circuit board (PCB) until we had finished our base app design. Our PCB needed to be populated with a radio, gyroscope, logic converted, GPS, ping controller and microcontroller. Initially, we were going to make a 4-layer circuit board but had to resort to a 2-layer circuit board in which we would through-hole all the components because most parts were not available to the public. Due to the sudden transition to online classes we were not able to print the final version of the board but the design for the final version is shown in Figure 7.

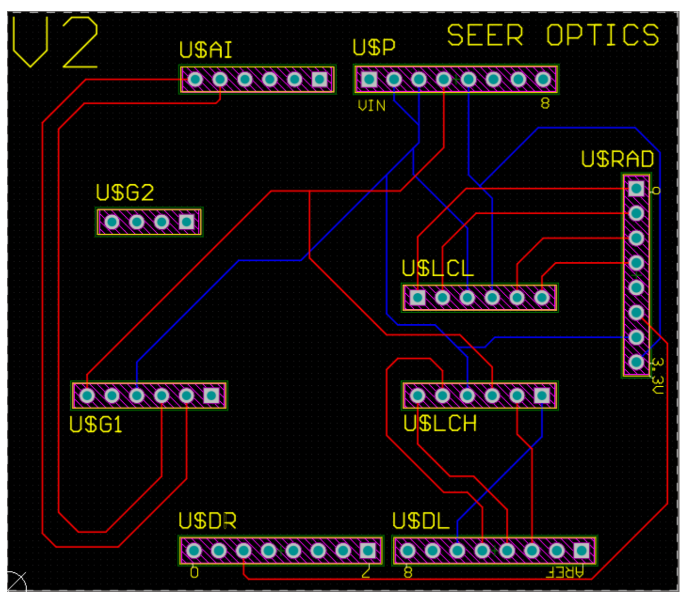

Figure 7: Final PCB design schematic

Another component of our hardware is the ping controller, which is a simple hand-soldered, protoboard design that will be connected to the PCB. For our design, it will only be connected to the Seer's PCB since we are limited to one set of Optics. Figure 8 shows our design for our ping controller. It is a simple design that inputs a voltage depending on which button is pressed. Both the dial and buttons were able to be incorporated in the final hardware design.

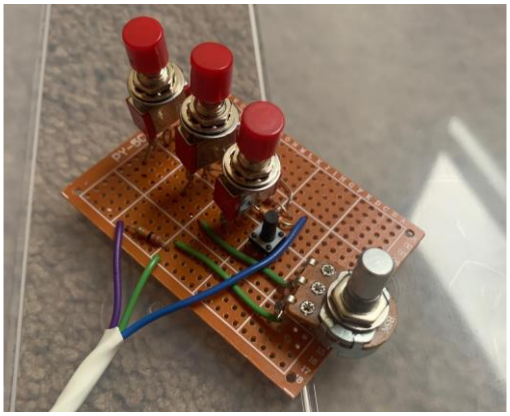

Figure 8: Ping Controller on protoboard

C. Product Functionality

At the time of CDR we were able to demonstrate full functionality of parts shown in our block diagram. At MDR we did not have gyroscopes installed on either of the nodes so the SEER Node had to stay stationary and the Pinging Node could only place a ping marker directly on themselves. In our CDR demo we showed that the user with the Pinging Node could mark locations up to 50 meters away from them in the direction that they attended. These pings successfully stayed in place as both nodes continued to move around. The SEER Node with the optics was able to walk around and turn their head to the location of their choice. What the evaluator saw is shown above in figure 5 from part A. We had full functionality at the time of CDR but needed to incorporate our enclosure and PCB for our product to be finalized.

D. Product Performance

The only performance checks we completed were to get functionality and calibration of our hardware components. Due to the COVID-19 outbreak, we were unable to test and polish our product to meet the specs in Table 1, as this is what we planned to be doing once we returned from spring break until Demo Day.

IV. Conclusion

Given the complexity and difficulty of this project, the result of all of this progress throughout the fall semester has directly come from the perseverance, passion, and dedication of every group member. From struggling with interfaces like the Microsoft Hololens and the radios, learning new software like Unity, and making difficult but crucial design decisions, our team has successfully put ourselves in good standing for the spring semester to clean up and refine our entire project.

All in all, our team is in good standing for next semester. Although we have a lot of different and difficult challenges to face in the spring, we have a solid and robust plan that will help us succeed. We greatly look forward to cleaning up, finalizing, and eventually providing a demo of our project to evaluators as well as others once we showcase our final design.

Acknowledgment

We would like to thank fellow SDP20 Team 1 and more specifically colleagues Ibrahim Tahoun and Cormac Kennedy for letting us borrow their android phone for development while we did not have our Google Pixel. This support from them was much appreciated during the debugging phase of our prototype as we had an actual unit to test the application that was developed for SEER Optics using Unity.

We would also like to thank our advisor, Professor Goeckel, and evaluators, Professors Gao and Aksamija, for thoughtful insight, critique, and advice throughout the development of our project.

References

[1] A. Watkins and A. Southall, “N.Y.P.D. Officer Was Killed by 'Friendly Fire' During a Struggle With Suspect,” *The New York Times*, 30-Sep-2019. [Online]. Available: https://www.nytimes.com/2019/09/30/nyregion/bronx-officer-brian-mulkeen-nypd.html. [Accessed: 24-Sep-2019].

[2] D. Wallace, “Coast Guard suspends search for firefighters lost at sea after Florida fishing trip,” *Fox News*, 23-Aug-2019. [Online]. Available: https://www.foxnews.com/us/coast-guard-florida-suspends-search-two-firefighters-missing-fishing-trip-justin-walker-brian-mccluney. [Accessed: 24-Sep-2019].

[3] Fischer, C. and Hazas, M. (2019). *Firefighter - an overview | ScienceDirect Topics*. [online] Sciencedirect.com. Available at: https://www.sciencedirect.com/topics/engineering/firefighter [Accessed 24 Sep. 2019].

[4] Frank, T. (2019). *Cell Phone Service Must Be Restored Quicker after Hurricanes*. [online] Scientific American. Available at: https://www.scientificamerican.com/article/cell-phone-service-must-be-restored-quicker-after-hurricanes/ [Accessed 24 Sep. 2019].

[5] NBC, “NYPD Releases Bodycam Footage from Fatal Shooting of Det. Brian Mulkeen,” *NBC New York*, 14-Dec-2019. [Online]. Available: https://www.nbcnewyork.com/news/local/nypd-releases-bodycam-footage-from-fatal-shooting-of-det-brian-mulkeen/2241734/. [Accessed: 24-Sep-2019].

[6] U.S. Army, Timeandnavigation.si.edu. (2019). *The Blue Force Tracker System | Time and Navigation*. [online] Available at: https://timeandnavigation.si.edu/multimedia-asset/the-blue-force-tracker-system [Accessed 24 Sep. 2019].

[7] “Developer Guides: Android Developers,” *Android Developers*. [Online]. Available: https://developer.android.com/guide/index.html. [Accessed: 1-Dec-2019].

[8] *SIK Experiment Guide for Arduino - V3.3*. [Online]. Available: https://learn.sparkfun.com/tutorials/sik-experiment-guide-for-arduino---v33. [Accessed: 3-Nov-2019].

Appendix

A. Design Alternatives

As mentioned in our overview and conclusion, we have put a lot of effort in trying to interface with the Microsoft Hololens. After extensive research, we decided to move to our main alternative, the Google Pixel. The reason we moved away from the Microsoft Hololens was because the developer documentation revolving around the product was lackluster when compared to other augmented reality capable products. Furthermore, the lack of documentation sprouted to other problem when we attempted to interface with the product. For example, while Microsoft computers are capable of serial port communication, the Microsoft Hololens is not because of the operating system that it runs on.

We defined our criteria for an alternative to be any device that was capable of running a Unity program and could be used as an augmented reality device. The alternatives that met our criteria were smart phones and other augmented reality devices similar to the Microsoft Hololens such as Google Glasses. Due to the vast amount of developer documentation and cost efficiency we decided on the Google Pixel [7]. Creating alternatives gave us the intuition to know when we should try to interface with another product or to find a new interfacing method.

B. *Technical Standards*

In order to strengthen the technical reliability and soundness of our product, we used many standards that are utilized worldwide in both our hardware and our software when we were designing Seer Optics.

One of these standards that we used was the USB Protocol and Standard when transferring information from the microcontroller on the Seer Node to the Google Pixel and the Optics. This standard is widely used in the transfer of data and is one of the most popular and efficient methods of data transfer through wire. This is because it consists of several layers of protocols where the host must initiate all transactions. Several packets are exchanged in a handshake process to then determine more information regarding the transfer. This USB Standard very closely relates to the IEEE 802.6 standard regarding information exchange between systems. This standard supports 150 Mbit/s transfer rates which is along the same lines as the transfer rates as the USB standard.

One other standard that we used when designing Seer Optics was the IEEE 1855 standard for Fuzzy Markup Language. This is a technical standard that allows for modeling of simplistic systems and diagrams that make designs easily understandable all around. Throughout the process of designing Seer Optics, we adhered to this standard with both our simplistic block diagram and our schematic of the actual design. This standard was very important to follow because it greatly helped with describing concepts and design decisions to our evaluators and even the general public.

Another standard that we used in design of our product was the FCC standard regarding radios and radio frequencies. This standard was used when we wanted to communicate with the nodes in our design. To do this, we utilized two radios which each node in the design consisted of a radio with a transmitter and a receiver to communicate efficiently. Our radio had a frequency of 915MHz which was within the range of frequencies that we were not allowed to be in (1KHz through 100 GHz). Despite this, our design is running at such low power that it complies with the FCC standard as it does not disrupt more powerful signals. This standard adheres to IEEE standard C95.7-2005 on recommended practiced with radios and radio frequencies. This standard explains that RF signals should be relatively low power, or as much possible. This is so that it can comply with the FCC standard as well as to avoid potential safety hazards.

C. *Testing Methods*

Our product, Seer optics, requires a plethora of interfaces. Every interface we designed went through an excessive number of tests. Firstly, grabbing GPS data. This interface was between the Arduino Uno and the GPS chip, see Figure 2. Our test to make sure this interface worked was to first print out the GPS data to the serial port where we could view it on a serial port monitor. To further test our system, we walk around outside to see how reliable the GPS chip was. We concluded that the device is reliable outside but completely cuts out when in doors. We tested the GPS chip outside of the lab, but further tests need to be conducted in areas such as forests or under certain weather conditions in order for us to deem the reliability of this product.

Secondly, sending GPS data over radios. This interface was between the Seer Node and the Node, see Figure 2. Our test to make sure this interface worked was to send a phrase of the radios to make sure the other end could receive data. After we were able to send a message over the radios, we tested their accuracy with distance and inside of buildings. We found that the frequency was effective for the distance that we specified in the specifications regardless of the obstacles that were in the way. Our future tests with this interface will revolve around testing which baud rates increase the speed at which we can send the data.

Lastly, serial port communication with the Google Pixel. This interface was between the Arduino Uno on the Seer Node and the Google Pixel, see Figure 2. Our test to make sure this interface worked was to write code to pull data from the serial port and display it onto the screen to make sure it was what the Arduino Uno was sending to the serial port. After we were able to get the data in the correct format, we attempted to try send data that would vary in length to see if the serial port buffer would cut off any of the data being sent. We found that none of the data being sent was corrupted. After our testing of the serial port communication we deemed that the interface met our goals and required no more testing. All of these interfaces were referenced with SparkFun’s guide [8].

D. *Team Organization*

We as a team communicated about everything throughout the semester. The team had a shared idea on what the product would look like and to verify that it was shared vision we would communicate how we thought the design would operate whenever we made any changes to our original idea. When designing the interfaces all team members were present and were able to explain, face-to-face, how the interface would work and what the format would be of the output. This method of communication was extremely effective in our project because when a teammate would test the interface another teammate made both of them would be present and be able to fix any errors that occurred. The team was split into a hardware and software group since the work was even on both ends of the hardware-software spectrum. The software team, Dhimiter and O’siris, were able to test different input to their program before the hardware team finished their design. This was helpful when connecting the hardware and software at the end. The hardware team, Daniel and Samuel, communicated their design to the software team and the output of their system with extreme detail which was pivotal in the progression of the team.

The team never broke down and never sway from the idea that we had to all be there together until the product was finished. There were still road bumps that we had to get passed. The main one was having to switch from the Microsoft Hololens to the Google Pixel three days before MDR. This meant that the majority of progress that the software team made had to be scrapped because of the change in platforms. The software team spent their time before MDR leaving the lab no later than 3 a.m. During this time the hardware team had finished majority of their work but still stayed by the software teams' side until the product was finished. This proved to be very helpful for morale and efficiency as some aspects of the hardware needed to be altered in order to interface with the Google Pixel properly.

E. Beyond the Classroom

Most of the team shares the same problems. During presentations Dhimiter has made leaps and bounds on being more concise with his words in order to communicate his points better, but there is still improvement to be made for next semester. The other team members also have room for improvement for not only presenting but for also considering changes to the product they are developing. O'siris has improved with accepting new ideas when they come about but still has a lot more room for improvement when it comes to patience. For next semester both him and Dhimiter's goals are to take a deep breath before they speak/listen and for both to orient themselves to the information they are receiving or the information they are about to give out. Samuel has improved greatly in his patience with teammates this semester. Samuels goal for next semester is to adapt to people's ideas and moods in order to bring out the best in them. That way in times that morale falls low he will be there to set the team back on track. Daniel contributes a lot to the team when a new idea is proposed because he uses his critical thinking skills and provides the team with foresight on how this addition could hurt or help the team. Daniels goal for next semester is to provide this constructive criticism in a way that improves upon the idea. Each member provides skills that are pivotal to the teams' effectiveness and chemistry.