UNIVERSITY OF CENTRAL FLORIDA

Senior Design: Divide and Conquer 2.0

Group 6: BarkaRoo Pet Trainer Module

Department of Electrical Engineering and Computer Science

University of Central Florida

Dr. Samuel Richie

Dr. Lei Wei

Cody Khong, Computer Engineer, codykhong@knights.ucf.edu

Vincent Martinez, Electrical Engineer, vincem424@knights.ucf.edu

Allan Nevalga, Computer Engineer, allancnevalga@knights.ucf.edu

Jesse Ray, Electrical Engineer, jessrayee@Knights.ucf.edu

1.0 Executive Summary 6
2.0 Project Specifications....................7
2.1 Project Objectives....................8
2.2 Project Features....................9
2.3 House of Quality Analysis 10
3.0 Hardware Research....................11
Table 3.1 Accelerometer Options....................12
3.1.1 LIS3DH....................13
3.1.2 MPU-9250....................14
3.1.3 ICM-20789 15
3.1.4 IIS3DWB 16
3.1.5 LIS2DW12....................17
Table 3.2 SOC Options 19
3.2.1 ESP32 WROVER....................20
3.2.2 MT2502/MT2503....................21
3.2.3 RAK4600....................22
3.2.4 LO4 23
Table 3.3 Temperature Sensor Options....................25
3.3.1 TMP36....................25
3.3.2 TMP007....................26
3.3.3 PCT2202 26
3.3.4 AS621x 27
3.3.5 MAX6652 27
3.3.6 NXRT15XV10 28
Table 3.4 RGB LED Options....................30
Table 3.5 BLE Options 32
3.5.1 CC2541 32
3.5.2 CYW20721B1KUMLG 33
Table 3.6 Heart Rate Monitor Options 34
3.6.1 AD8233ACBZ-R7 34
3.6.2 XeThru X4 (UWB) 34

3.6.3 MAX30100 35
3.6.4 Pulse Sensor Amped 36
Table 3.7 Wi-Fi Options 37
3.7.1 FC9000 37
3.7.2 88W8977_SDS 38
3.7.3 WGM160PX22KGA..38
Table 3.8 Speaker Options..40
3.8.1 AR01232MS-SC12-WP-R..40
3.8.2 CMS-151125-076x-67 41
Table 3.9 Li-ion/Lipo Charging Chip Options 42
3.9.1 LM3658 43
3.9.2 ISL6292..44
3.9.3 BQ2407X 45
3.9.4 MP2615C..46
Table 3.11 LoRa Options 51
SX1276 51
SX1272 52
LLCC68..52
SX1280 53
4.0 Hardware Design and Manufacturing 55
4.1 Schematic and PCB design software..55
4.1.1 OrCAD (Cadence) 55
4.1.2 Altium 57
4.1.3 Eagle CAD (Autodesk) 58
4.2 Printing Board House Selected 60
4.2.1 pcbway..60
4.2.2 JLCPCB..61
4.2.3 OSHpark 61
4.3 Parts Shortage of 2021 63
4.4 Prototype Power Board 64
5.0 Realistic Design Constraints..67
5.0.1 Economic Constraints 68
5.0.2 Time Constraints..71

5.0.3 Environmental Constraints 72
5.0.4 Social and Political Constraints 73
5.0.5 Ethical Constraints 75
5.0.6 Health and Safety Constraints 76
5.0.7 Manufacturability Constraints 78
5.1 Related Design Standards 80
5.1.1 Weather 80
5.1.2 Wi-FI Technology 82
5.1.3 Bluetooth Technology 82
5.1.4 LoRaWAN 82
5.1.5 Microcontroller Data Communications 83
5.1.6 GPS(Global Positioning System) 84
5.1.7 Temperature Sensor 85
6.0 Firmware Programming 86
6.1 ESP-IDF 86
6.2 PlatformIO IDE 88
6.3 Arduino IDE 89
7.0 Software & Firmware Component Design 94
7.1 GSM/GPRS System 94
7.2 Ambient Temperature Sensor 96
7.3 Accelerometer Sensor 97
8.0 Software Research 98
8.1 Cloud Computing 98
8.1.1 Grafana 100
8.1.2 PubNub 101
8.1.3 ThingSpeak 102
8.2 Mobile Application Development 103
8.3 Database 106
8.4 Server 111
9.0 Initial Project Milestones 114
9.0.1 Design Document Project Milestones 114
9.0.2 Final Product Project Milestones 115
9.1 Estimated Budget and Financing 116

Table 9.1.2: Development Cost 117
Table 9.1.3: Cost per Production Unit 118
10.0 Testing 119
10.1 Prototyping 119
10.1.1 Hardware Testing 120
10.1.2 Software Testing 121
10.2 (Not) Animal Testing 122
10.3 Human Testing 123
10.3.1 Family Household Testing 124
10.3.2 Various Background Testing 125
11.0 Citations and References 126

1.0 Executive Summary

This section is the executive summary for our overall project. The motivation for the project will be discussed thoroughly. In addition, there will be a discussion of the project goals and objectives we achieved through research and development. Finally, there will be a discussion of the function of the project and how it will be applicable to the market we are entering. As the project reached maturity, additional references from customer input and marketing analysis were added to supplementally identify the project features.

While discussing project ideas a discussion about pet care began after a cat attempted to knock over one of our team members' monitors. We all discussed our love of furry companions and how sad it is to see people forgetting to take care of their animals after the initial excitement goes away. We noted that this was especially true for small children as they do not comprehend the responsibilities associated with owning a pet. As this was being discussed a member came up with the brilliant idea of making a collar /harness with automated reminders triggering LEDs to notify the child or forgetful owner of their dog needing their attention, whether this be the dog likely needing to; go to the bathroom, be fed, be walked, or be given affection. When looking up this idea we found there were plenty of activity trackers for pets (dogs especially); LINK AKC, PetPace, Whistle, and Fitbark to name a few of the biggest names in the market space at the time of our research. These competitive devices help to track the activity of your pet, alert you to when your dog might have got out and then help you to find them. They do this by monitoring your pet using a heart rate sensor and a GPS tracker. Although these products have great use cases, we believe that they are missing important features that do not allow them to be used for helping forgetful owners and teaching children the value of responsibility with their pets.

Our motivation with this project is that we believe that we can have most of the same features as the previously mentioned competitors at the same price point but with extra use cases and features. We want our product to be a lightweight, cost effective, low power, relatively fast to notify, and feature filled product that offers more for the same price as compared to the competitors mentioned earlier. This collar will effectively fill in the gaps of yearly mistreated dogs ending up in adoption centers or becoming sick. Our collar will implement the notable features from leading competitors such as GPS tracking, real-time notifications, customizable mobile application just to name a few. Together with these implementations a database will be kept from certified sources from general to specified dog hygiene to each of their own necessary diet. This in return will allow our LED to toggle for when it is the right time for feeding or going for potty.

2.0 Project Specifications

This table is our project specifications for our final product. The table includes necessary components that will be needed to put this device together. We came to agreement on these parts through extensive research discussed later in this paper. These are the core parts that will be listed on our final product's packaging to let the user know what hardware is used to compare with other devices.

Components	Specifications
Microcontroller	ESP32 WROVER
Multipurpose LED	WS2812B RGB LED
Collar	Generic Vinyl
GPS module	A9G GPS Module
Temp Sensor	NXRT15XV103FA1B040
Bluetooth module	Included in ESP32
Wi-Fi module	Included in ESP32
Power Supply	Lithium-Polymer

2.1 Project Objectives

This section is the project objectives for our final device. This list was the general outline and goals we hoped to attain while developing our product. The following bullet points are going to be a key point in validation to see how well our product performs in comparison with these set metrics. If the testing results did not meet these project objectives, then we will have a clear path to redesign or improve our project. There are many methods in validating and determining whether or not our device is successful. This section will serve as one of the first points of revisitation to evaluate our product.

- The cost of this device must be fairly cheap to obtain
- Able to tell if a dog has slipped his collar
- Able to see if a dog is left in a car (heat deaths)
- RTC should lose less than 10 seconds a day
- GPS accurate to under 10 meters
- LEDs should be visible around a corner at least 10 ft and 50 feet away if within eyesight
- Must allow the use of BLE (Bluetooth Low Energy) to utilize Geofence technology
- Last at least a week on a single charge

2.2 Project Features

This section is the project features of our product. This is a list of functionalities we aim to include in our final device. These features are the key selling point in our product if it ever reaches the market for mass production. We aimed to have features that were up to standard on something like a dog collar. In addition, we wanted to have enough features to differentiate ourselves from the competition and set ourselves apart from our competitors. While compiling this list, we kept the market we were entering in mind and attempted to make the collar as accessible as possible in a reasonable manner.

- Under a pound
- Small footprint for dog collar integration
- Waterproof / dustproof
- Update the heart rate at least every 5 seconds
- Mobile application must be able to send notification for:
    - Suggested feed time
    - Suggested potty times
    - Dog has left the designated Geofenced area within 5 seconds
- Software database populated by common AKC (American Kennel Club) potty times based on weight and age of dog
- A way to notify if the dog has been fed
- Have a medium distance option for tracking (medium = 1km < x < 10km)
- LEDs need to be color blind friendly
- Seamless connection switch from BLE to Wi-Fi

2.3 House of Quality Analysis

Depicted below is our House of Quality Analysis, this allowed us to get an overview of the whole projects decision making in terms of preparations for what we're willing to accomplish towards the end. Our team determined that range was a huge factor in order to compete with the leading modules out in the market. This is true as range allows for accessibility of features for BarkaRoo. Another big factor to take in effectiveness of our module is reliable data transmission allowing for seamless access to accurate data of BarkaRoos output. Our team took notice of things prevalent to success of our product such as power, dimensions and cost allowing consumers to have better access to our product.

Correlation Legends		Polarity Legends
++	Strong Positive	
+	Positive	↑ | Positive Polarity
-	Negative	↓ | Negative Polarity
--	Strong Negative	
	No Correlation	

	Engineering Requirements					
	Power	Cost	Time	Dimensions	Location Accuracy	Transmission Speed
	↑	↓	↑	↓	↑	↑
User Requirements						
Cost	-	--	--	-	+	+
Reliable Data Transmission	++	+	+		+	++
Connectivity	++	+	-	--		
Visual Indicators	-	-	+			
Range	++	+	++		++	+
Mobile Application	-	-		-	-	+
Target Engineering Requirements	1 week battery life	< $500	10 weeks	dog collar size	~ 200 meter radius	5-10second delay

3.0 Hardware Research

This section is the hardware research of our dog collar. Here, we compiled a list of potential parts that will need to be included in our device. We started by making a list of absolutely necessary components to make a working product. This list will undoubtedly change as time progresses. There will be parts that will need to be added. Also, there will be parts that we may deem unnecessary and remove down the line. Once necessary components were listed, a list for each component for different models that we will be using. Some components have various options while others have a limited amount of variations. Once we compiled a list of various models for each component, we narrowed down our options by features, costs, and other considerations. After narrowing down our options, a final model of the component was chosen for this project. With the list of specific components now available, PCB design, supply chain, and other various topics could now be looked into for consideration. The following sections will give an outlook on how and why we chose our components.

Table 3.1 Accelerometer Options

	Cost	Size	Availability	Minimum Current	Voltage requirement
LIS3DH	$1.39	3x3x1mm	Obsolete	2 µA	1.71V-3.6V
MPU-9250	$5.00	3x3x1mm	Obsolete	8µA	2.4V – 3.6V
ICM-20789	$7.60	4x4x1.37mm	Readily	20µA	1.71V-3.45V
IIS3DWB	$17.64	2.5x3x0.86mm	Readily	1.1mA	2.1V – 3.6V
LIS2DW12	$1.90	2X2X0.7mm	Readily	50 nA	1.62V -3.6V

The accelerometer is arguably the most important piece of hardware on our device and without question the most important sensor. With that being said our team found it crucial to find a good and well tested accelerometer. The most important features for our team with an accelerometer was for it to be: available, ultra-low power, and small. If it could be low cost and fit into the existing voltage rails, 5V or 3.3V, that is a huge bonus but not as big of a deal. Table 8.1 gives a brief overview of the most common five attributes our team monitored and used to make our decision.

3.1.1 LIS3DH

When our team first started to search for an accelerometer, we came across the LIS3DH which had many features we wanted including: ultra-low power 2 μA, a small overall footprint 3X3X1mm, great input voltage ranges, and an excellent price. It also includes 2 programmable interrupts for drop detection, and an on-board temp sensor. Although this sensor seems perfect in every way it also turns out STMicroelectronics stopped making them and they cannot be purchased anywhere anymore nowadays.

List of features:

- **Embedded self-test**
- **Ultra-low power (2 μA)**
- **Small package/ footprint (3x3x1mm)**
- **16-bit output**
- **2 Programmable interrupts (fall detection/ motion detection for example)**
- **Embedded temp sensor**
- **6D/4D orientation detection**
- **Vibration monitoring and compensation**
- **Wide supply input (1.71V-3.6V)**

3.1.2 MPU-9250

The MPU-9250 is an incredibly feature rich accelerometer having a total of 9-axis it can measure and three ADCs on board to process the data from the 9-axis. This is done by putting a 3-axis accelerometer, 3-axis MEMS gyroscope, and 3-axis MEMS magnetometer all on the same chip while still having a package size of 3x3x1mm. It can also output this data through a completely integrated MotionFusion™ output. The outputs also have programmable digital filters and self-test capabilities for each output. Wake up interrupts for low power applications so that the accelerometer can sleep when no recording is needed and low the overall power of the unit. There are only 2 real downsides to this chip. The first is the current draw of the unit is 3.2mA for the onboard Gyroscope alone. Although the Gyroscope does account for 81% of the power required for the unit it is concerning none the less that if the Gyroscope does account for so much power, we may end up with a chip that we almost never use one of its defining features. The other major disadvantage of this chip is that it has been made obsolete by InvenSence and thus is not available to us.

List of features:

- **Self-test functions an all sensors on the chip**
- **Programable filtering on the Gyroscope output**
- **Ultra-low power sleep modes (8 µA for the gyroscope)**
- **3 Built in 16-bit ADCs for processing the data from the 3 sensors**
- **Integrated MotionFusion™ output**
- **Small package especially for the features (3x3x1mm)**
- **Wide supply input (2.4V-3.6V)**
- **User programmable interrupts**

3.1.3 ICM-20789

The ICM-20789 is a 7-axis accelerometer with a 3-axis Gyroscope, 3-axis accelerometer, and pressure sensor all included in the chip. It also comes with a temperature sensor in the pressure sensor and the 6-axis combination sensor (3-axis Gyroscope and 3-axis accelerometer). There are filters on the outputs for better signal to noise ratios. Ultra-low power with a 4Kb FIFO register for sending data less frequently and instead in large batches. The two major problems with this chip for our team is that it has the largest package size of all of our options albeit not by a terribly large amount. More concerning is that the only development board that exist is over $100 dollars it does make it very hard for a group of students to go out and prototype with it when there are other accelerometers that meet our criteria as well for much cheaper cost of both the IC and development boards.

List of features:

- **Self-test functions**
- **Filters on the output of each of the sensors**
- **User programable interrupts**
- **Ultra-low power (20μA)**
- **4Kb FIFO buffer for less transmissions but the same data**
- **2X integrated temperature sensors**
- **Integrated pressure sensor**
- **16-bit ADC for the Gyroscope and 24-bit ADC for pressure sensor**
- **Small package size (4x4x1.37mm)**
- **Wide supply input (1.71V-3.45V)**

3.1.4 IIS3DWB

The IIS3DWB is a low-noise accelerometer mainly used for vibration control but as it does have a programmable scale could be used in our application with high levels of accuracy. The package is small coming in at 2.5x3x0.83mm. It has an embedded temperature sensor and embedded self-test. The chip also comes with a 3Kb FIFO for more reliable data transmission (harder to ruin the data or lose it if there is a place to store it) and again to be able to send batches of measurements instead of continues single data measurements. It is also available which is very important as of the 2021 chip shortage and the fact that most of the chips our team has found thus far have become obsolete. The major negatives of this chip are the low power mode being less than impressive for our application at 1.1mA and although the accuracy is quite impressive, we have no need for it, therefore the extra cost of the unit is wasted on our project, and at $17.64 a chip that is a rather large waste.

List of features:

- **Self-test functions**
- **High stability to temperature and mechanical shocks**
- **Ultra-low noise density 75 µg/√Hz in 3-axis mode**
- **Extra temperature clearance going up to 105°C**
- **Embedded temperature sensor**
- **Wide supply input (1.71V-3.45V)**
- **Embedded FIFO 3Kb**
- **Small package size (2.5x3x0.83mm)**
- **Has interrupts for activity detection and inactivity detection**

3.1.5 LIS2DW12

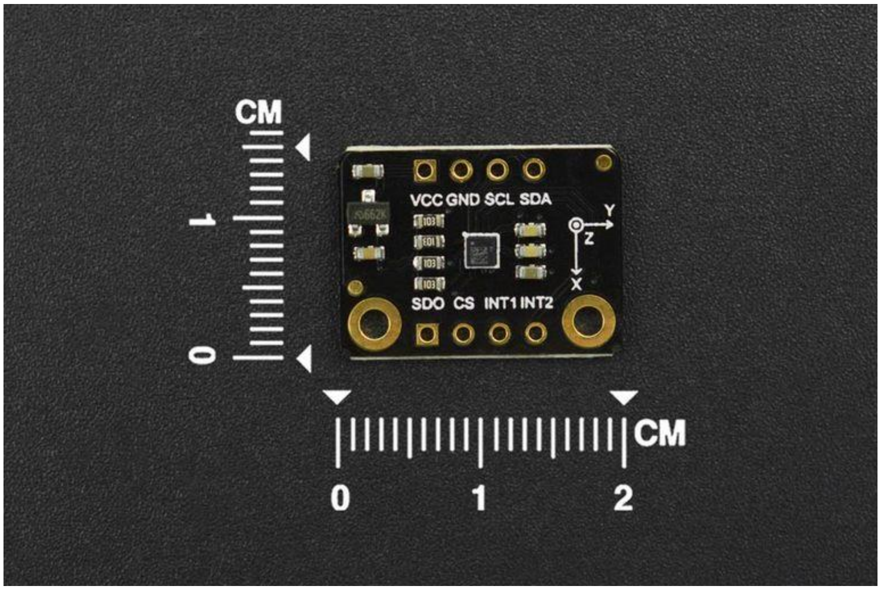
The LIS2DW12 is an ultra-low power (sub 1µA) 3-axis accelerometer. The chip has "very low noise: down to 1.3mg RMS in low power mode" (data sheet). This chip offers a 32 level FIFO for storing measurements and 16-bit outputs for high resolution data. It also comes with embedded temperature sensing and embedded self-test. Things such as gestures can also be recognized opening up the world to possibly being able to recognize when a dog might be eating or drinking. The package is also the smallest our team has seen at just 2x2x1mm. Lastly it is readily available which is again a major reason to use this chip. The cons of this chip is that is not as feature rich as some of the other units and does not give us as many options for programmable interrupts or other ways to modify the outputs nor does allow for the most sensing data to be collected.

List of features:

- **Self-test functions**
- **Embedded temperature sensor**
- **Embedded FIFO 32 level**
- **Small package size (2x2x1mm)**
- **Ultra-low power (less than 1µA)**
- **Gesture recognition**
- **Very low noise: down to 1.3 mg RMS in low-power mode**
- **High speed I2C and SPI**
- **Wide supply input (1.62V-3.6)**

Conclusion

In conclusion our teams' biggest issues with hardware was being able to find available parts with useful features for our project. Each accelerometer was incredible in its own right whether it was a 9-axis accelerometer with more features than we could ever power or a basic 3-axis that will draw almost no power, take up no room, and is available. In the end it was decided that sometimes basic is better and because it fits all of our major requirements for this category with the lowest power, smallest package size, and being the most readily available in both dev boards and chips we have chosen the LIS2DW12 for our project.



Courtesy of https://www.dfrobot.com/product-2337.html

Table 3.2 SOC Options

	Cost	Size	Availability	Minimum Current	Voltage requirement
ESP32 WROVER	**$2.13**	**18×31×3.3mm**	**Readily**	**550 μA**	**3.0V-3.6 V**
MT2502/MT2503	**$1.70**	**5.4×6.2mm**	**Obsolete**	**Not clearly defined**	**0.3V-4.4V**
RAK4600	**$12.11**	**15x23x2.5mm**	**Readily**	**2.0μA**	**2.0V-3.6V**
LO4	**$31.73**	**40x16x2.7mm**	**Readily**	**600μA**	**3.5V-5.5V**

Systems on a chip (SOC) were discussed by our team to get multiple features of our project done with one chip. Generally, this allows for a better price per feature and less space overall due to routing each chip as compared to the internal routing already being complete during the PCB design stage. Many SOCs and microcontrollers also have much better development communities allowing for easier development stages especially under a time crunch. The most important things to consider for our team while choosing if we wanted a SOC, and if so which of the many SOCs on the market would be the best for our project, were: cost versus performance and ease of implementation, availability, size, community built around the product (same thing as looking for good documentation), and making sure there was still some meat on the bone of the project for the Electrical Engineers of our group. Although the last consideration may seem trivial it was the most discussed idea within our team as giving the Software and Firmware teams the most amount of time could also mean that the board designs may end up over simplified. This was a highly debated issue because we all wanted maximized our growth in our prospective disciplines while still giving other members of the team the ability and time to do the same.

3.2.1 ESP32 WROVER

The Espressif ESP32 is a very common SOC for Bluetooth and Wi-Fi as it has an on-board processor and can be used as a micro controller to be programmed with Arduino C, C, or MicroPython. The community and documentation for the ESP32 is vast and explore every idea that someone could even think of. Up to 150Mbps speeds allowing for quick uploads and downloads of small sensor data used in our project. Has multiple kinds of sensor interfaces including ADCs for calculating values directly from an analog voltage reading such as for a Thermistor as well as I2C and SPI for other sensors such as our accelerometer and RGB LEDs. All of these features come with a built-in antenna and are readily available for less than $5. The major drawbacks of the ESP32 are the package size and the power draw of the complete unit.

List of features:

- **IEEE 802.11b/g/n 20MHz 2.4GHz**
- **Operating voltage of 3.0V-3.6V**
- **40MHz crystal oscillator**
- **SD card, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I 2S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, Two-Wire Automotive Interface (TWAI®, compatible with ISO11898-1)**
- **2X Embedded Xtensa® single-core 32-bit LX6 microprocessor**
- **Incredibly diverse and helpful documentation and community**
- **Embedded 448KB ROM, 520KB SRAM, and 16KB RTC SRAM**

3.2.2 MT2502/MT2503

The MT2502 is a very SOC with great power features and good Wi-Fi, Bluetooth, and GPIO support. This chip has the most features per mm^2 (5.4mm×6.2mm) of anything on the SOC list. It has Bluetooth, Wi-Fi, DACs for audio conversion, GSM/ GPRS (2G) radio, UART, and USB 1.1 all built within the chip. It also Includes power management systems such as an integrated Li-ion charging, LED drivers, and low quiescent current LDOs. Although this chip is incredibly feature rich it does not have any community support or useful documentation meaning that the time limit of Senior design would make implementing this chip very difficult. It also does not run as well as other SOCs on the list. It is overall more of a good at everything but great of nothing kind of SOC.

List of features

- **Bluetooth 4.0 Transceiver**
- **ARM7EJ-STM RISC processor**
- **UART, USB 1.1, and SDIO**
- **Keypad scanner**
- **SIM controller**
- **ADCs**
- **DACs**
- **GSM/GPRS (2G)**
- **Integrated Li_Ion Charger**
- **Embedded LDOs**
- **Small package (5.4mm×6.2mm)**

3.2.3 RAK4600

The RAK4600 is a Bluetooth and LoRa integrated SOC. It combines the super low power SX1276 LoRa chip with the nRF52832 Bluetooth chip to offer a very low power SOC (13.3 μA) in a form factor of 15 x 23 x 2.5 mm. It also includes support for UART, I2C, and GPIO. Although this chip is low power it also lacks features our projects needs such as Wi-Fi. This would mean having to add another chip which would likely get rid of the major advantages of low power and smaller form factor as we would need to add connection to another chip that would be at least 3 x 3mm.

List of features:

- **LoRa Transceiver**
- **Bluetooth 5.0 Transceiver**
- **Wide supply range (2.0V - 3.6V)**
- **Ultra-low power (13.3 μA)**
- **Small package (15 x 23 x 2.5 mm)**
- **Wide frequency range (7.8KHz - 500KHz)**
- **All components linked and designed together already**
- **Great documentation and Community support**

3.2.4 LO4

The LO4 by Pycom is a simple to use Wi-Fi, Bluetooth, LoRa, and Sigfox integrated SOC. This SOC has large community support almost matching the ESP32 and great documentation for future implementations. This SOC combines the best features of the ESP32 and RAK4600 as it literally combines them into an easy-to-use preassembled PCB for quick prototyping. The major issues with this SOC are that the input voltage is not a voltage our team had planned to provide as a rail meaning that we would need to add another rail to our PCB and it has a large overall footprint and price per unit. The price of this SOC per unit is almost $40 before shipping and the footprint is 40 x 16 x 2.7 mm. The only other major issue our team has is that we felt it over simplified out product for our Electrical engineering team as the overall PCB would be made much simpler with this SOC.

List of features:

- **Bluetooth 5.0, Wi-Fi, LoRa, and Sigfox integrated**
- **Easy to design around**
- **Excellent documentation and Community support**
- **Every communication protocol we could ever want enabled**

Conclusion

Both the LO4 and MT2502 came out as obvious favorites in the beginning do to the massive number of features and the extra time it game to the firmware and software development teams. The major issues with the LO4 were its overall size, the fact that if chosen much of the design for the Electrical Engineers would be simplified so much so that any real design work would be taken away from them, and its high upfront cost of $40 a board while still needing auxiliary boards to program it. For these reasons it was decided that this would not be a worthwhile choice for our project. The MT2502 is again very feature rich but it struggles with the most common problem of the project, availability. It is very hard to source the MT2502 and when we could fine them it was going to take at least three weeks to arrive from overseas. We were also unable to ever find any sort of community or decent documentation built around this chip. This would mean that for all the features it brought to the table it would likely take us as long to access them as to design and purchase them separately. Thus, we had to cross this SOC off the list as well. Once our two favorite SOCs were off the list we had to dig deeper and thus we found the RAK4600 and ESP32 WROVER. The RAK4600 is a super-efficient SOC with good features built in but we felt that the features it was the biggest on were also the features we were the most certain on as far as what we were sure our project was going to keep and move forward with. The ESP32 WROVER although not as power efficient did have features where it counted most, in all the areas we knew we would need no matter what direction further research brought us. It also has a community unmatched by almost any other chip and therefore it checks of the box for ease of development while still giving our Electrical Engineers enough freedom to design around and with it. After much discussion the ESP32 WROVER was chosen for our project.

Courtesy of

https://www.mouser.com/new/espressif/espressif-esp32-wrover-b-module/

Table 3.3 Temperature Sensor Options

	Cost	Size	Availability	Current	Voltage
TMP36	$1.50	3.5x4.6x19mm	Readily	50uA	3.3V
TMP007	$14.95	1.9X1.9X 0.6mm	Obsolete	270uA	1.7V
PCT2202	$0.37	0.69x1.09x0.38mm	Readily	30uA	1.8V
AS621x	$1.04	1.5x1mm	Readily	6uA	3.6V
MAX6652	$2.06	3.05x5.05x1.10mm	Readily	10uA	3.3V
NXRT15XV103FA1B040	$0.56	1x0.5mm	Readily	-	-

The temperature sensor for our product will be used to test and measure out the ambient temperature so we can talk to the customer/trainer to beware of heat exhaustion on the dog on hot days. This will allow our software to tell the customer/trainer to bring the dog inside and give them some water and/or cut time on their walks outside. To make sure our product has great battery life is to have a temperature that is low power and have a shutdown feature that allows us to save power as we do not need to check temperature consistently.

3.3.1 TMP36

This part allows us to monitor the ambient temperature. This IC is low power that provides a voltage that corelates to the ambient temperature, because of this we do not need to calibrate this device making it easier to test and implement on our product. This device also has a shutdown feature when not in heavy use. Also has a very solid accuracy at around 1 to 2C.

List of features:

- **Low power**
- **Shutdown**
- **Auto calibrated**
- **-40 C -125 C**

3.3.2 TMP007

This IC is an inferred temperature sensor that is a 14-bit resolution which allows it to be very accurate, it's very low power only needing 1.7V and is very small 1.9X1.9mm and has a shutdown feature, this will be a nice fit for our compact and small design we want to have. Lastly it is compatible with I2C and SMBus.

List of features:

- **Low power**
- **Shutdown**
- **Inferred temperature**
- **-40C – 125C**

3.3.3 PCT2202

This ultra-low temperature sensor only needs about 1.8V to run with a resolution of 12 bits and an accuracy of 0.5C between 0 to 85C, this is a very small range but it offers more accuracy and for being low power this is a great choose. This part not only is low power it also has a shutdown when not in heavy use, we can make this device take temps every so often using a logic signal to turn it on and back off. Lastly this device offers high speed bus interface, but we will not be using this feature as we only want it to tell the temp to the application on the phone.

List of features:

- **Low power**
- **Shutdown**
- **High speed bus interface**
- **I2C and SMBus compatible**
- **0 C – 85C**

3.3.4 AS621x

This device is an ultra-low power sensor only needing 1.71V to 3.6V which 3.6V is what we plan to use for our product. Also, this IC comes pre calibrated with easy to set up serial buses with a shutdown function which allows us to save battery life which what we are looking for in an IC. This IC also very accurate of 0.2C between –40C to 125C. Although we do not need the IC to be very accurate but it is nice to have.

List of features:

- **Low power**
- **Shutdown**
- **I2C compatible**
- **High accuracy 0.2C**
- **-40C – 125C**

3.3.5 MAX6652

This sensor is low voltage it can operate at 3.3V which is what we want it is not very accurate having an accuracy of +2C but again we are not picky with accuracy. It's also compatible with I2C and SMBus which allows it to have an alert interrupt which is nice sense we can program the interrupt temperature to what we want.

List of features:

- **Low power**
- **I2C and SMBus compatible**
- **-40C – 125C**
- **Programable interrupt**

3.3.6 NXRT15XV10

This part is a through hole Thermistors that will allow us to measure ambient temperature although it will not be very accurate but we don't need to be very point accurate when taking the temperature of the area. Also, this part is very low power about 7.5 mW and can read temperatures –40C to 125C and because this part is just a thermistor it is very small 1.00x0.5mm.

List of features:

- **Low power**
- **-40C – 125C**
- **Very small (1x0.5mm)**

Conclusion

We pick out a lot of parts for this because we wanted to find the best fit part that was readily available and was well priced and fit the light weight and size requirements we have set up for our product. We found many IC chips that are low power and contactless with shutdown/interrupt features, but these chips are hard to find and could cause more issues if say a chip went bad and we need a replacement. Seeing how little time we have and having an IC chip would be nice and could allow more flexibility better accuracy and a way to communicate an interrupt but some of the chips require calibration and making sure that they can communicate with already existing parts like the processor. So, we made a choose to go with the thermistor NXRT15XV10 its very light accuracy is fair with a +1C but this product does not need to be extremely accurate to read the ambient temperature unless we want to add another temperature sensor that can read the dogs temperature which might be added on as another feature but is to be determined. Going back to the part it of course does not need to be program and can read temperature change based on resistor change, but we would need a voltage reader to see the change which will not be hard to add and will not take away too much space. We want this product to be waterproof so the case needs to be insulated and because of this some of the IC will not work properly and so our idea is to have the sensor be in a sperate area away from the rest of the components so we can preserve the components from water and have a temperature sensor read the ambient temperature.

Courtesy of https://www.digikey.com/en/products/detail/murata-electronics/NXRT15XV103FA1B030/4422450

Table 3.4 RGB LED Options

	Cost	Size	Availability	Voltage	Current
CHINLY 3.3ft 60leds WS2812B	$8.90	Modular	Readily	5V	60mA (with all 3 colors on per LED)
RGB LED weatherproof flexi-strip - 30 LED/m	$16.00	Modular	Readily	12V	60mA (with all 3 colors on per LED)

When deciding on the RGB LEDs we wanted a simple yet modular solution. The LEDs, of course, must be waterproof at least up to IP67. This will mean that we can put it on the exterior of a dog collar without issues such as degradation of the circuitry due to the humidity, rain, and possible submersion if your animal wants to go for a swim. It is also important to make sure the LEDs are a voltage that is already designed to be a rail in our power design. The last major thing is the overall cost of the number of LEDs we get with one meter. This is of course important because no matter whether we wanted 10 LEDs or two LEDs per collar we would want each LED to be a smaller strip and the most inexpensive for the most LEDs.

CHINLY 3.3ft 60leds WS2812B

The CHINLY RGB LEDs are waterproof and the correct voltage for our power rails as they are currently designed leading to less overall board redesign. It is also the cheapest option in both the price per meter and the price per LED category. The only real concern with these LEDs is that there is led protection circuitry built around each LED and the quality control is not the best. It should also be noted that the power draw per LED is also quite high.

List of features:

- **Cuttable LED strips for modularity**
- **Low cost per LED ($0.15/LED)**
- **5V input voltage**
- **Waterproof (IP67)**

Adafruit: RGB LED weatherproof flexi-strip - 30 LED/m

The Adafruit LEDs are also modular but add some extra protections like current limiting resistors in an effort to protect the LEDs for the hobbyist. They are flexible and waterproof though a rating for the waterproofing is not offered. Sadly, these LEDs have more disadvantages than advantages but none the less our group did consider them just in case quality of the CHINLY was a major issue. The major disadvantages are the added components on the LED strip which decreases the modularity because thew cutting positions are further apart, the 12V requirement, and the high price per meter and per LED as compared to the CHINLY ($0.53 vs the $0.15 of the CHINLY).

List of features:

- Good build quality
- Weatherproof
- Modular

Conclusion

As was evident in the previous section, we went with the CHINLY RGB LEDs because they are better in every way with the exception of possibly build quality. It does not make sense for this project to more than triple the cost of the LEDs for a possible build quality issue that has not surfaced. It also is a terrible decision if we do not have to add another rail to our power system as we are trying to minimize power and another rail for 12VC is just a waste of power and parts.

Courtesy of https://www.amazon.com/CHINLY-Individually-Addressable-Waterproof-waterproof/dp/B01LSF4Q00

Table 3.5 BLE Options

	Cost	Size	Availability	Voltage	Current
CC2541	$2.20	6x6mm	Readily	3.6V	20mA
CYW20721 B1KUMLG	$8.76	5x5mm	Readily	3.6V	65mA

Our low energy Bluetooth BLE will be used to communicate information such as if the dog has the collar on and what it is doing and other activities we wish to track and convey to the user though their phones and this is easily done thought Bluetooth as its common in every device and is easy to set up. The main things we hope to fine is a Bluetooth that is low energy as we feel that battery life is crucial, we also want a Bluetooth that is easy to set up that has a debugger and dev-board to make testing easy. One thing to note is that if we find a product that already has Bluetooth in it for example a WIFI/BLE we will go with that as we want to make our product as small and light as possible.

3.5.1 CC2541

This is a 2.4GHz that is a low power Bluetooth IC with three different power saving modes that also has great long-range communication. We want a Bluetooth IC that is not only small but is also low power that has good range as the product needs to communicate to the user phone. We also want easy accessibility; we choose this one as it has a debugger I2C compatibility software for coding.

List of features:

- **Low power**
- **Long range signal**
- **I2C compatible**
- **3 low power modes**
- **Debugger**
- **Dev board**
- **Flash memory 256KB**

3.5.2 CYW20721B1KUMLG

This IC is a low power Bluetooth with the latest version 5, this IC has 5 different power modes having these option will help decide what is best for the product and be more flexible. This one has a dev board which makes accessibility go up and is compatible with I2C and with add security although we do not need this as we do not use any sensitive information.

List of features:

- **Low power**
- **I2C compatible**
- **5low power modes**
- **Debugger**
- **Dev board**
- **Flash memory 1MB**

Conclusion

We did not choose any of these BLEs as we found out that yes, they do have debuggers and dev boards the issue is that they charge too much to use and we would have to order directly from the manufacturer meaning we could be waiting a long time and have to learn their system which could be time consuming. Although they have a great small size they don’t offer as much as a SOC and we would have to add in Wi-Fi along with this which will take up more space and power so we want a small IC that has both to save on space.

Table 3.6 Heart Rate Monitor Options

	Cost	Size	Availability	Current	Voltage
AD8233ACBZ-R7	$4.55	2x1.7mm	Out of stock	50uA	8.5uV
XeThru X4 (UWB)	$300	-	readily	-	-
MAX30100	$10.49	5.6x2.8mm	readily	1.2mA	3.3v
Pulse Sensor Amped	$24.99	15.8mm(D)	readily	4mA	5.5v

We want to have a heart rate monitor so we can give the user the ability to check on his/her dogs health as the day goes on, we want a heart rate monitor that can track the heart rate of a dog with minimal noise and can easily do so thought the dog's hair with reasonable accuracy and most importantly consumes very low power as battery life is important to maintain here.

3.6.1 AD8233ACBZ-R7

This device will read out the dog's heart rate when it wears the product though we quickly found out that this IC needs to have good direct contact with the skin in order to have any kind of reading, so we cannot use this. Which is unfortunate as this IC has very good noise filtering with rail-to-rail output.

List of features:

- **Low power**
- **I2C compatible**
- **Good noise filtering**

3.6.2 XeThru X4 (UWB)

This device is a sonar we want to the heart rare though the dog's hair so this was one of the ideas to try sonar, but the cost is way too high and it will be time consuming as we will need to code the IC to look for and count the pluses. So, this part will not be use.

List of features:

- **Low power**
- **Able to see though hair**

3.6.3 MAX30100

This is an optical heart rate monitor that uses two LEDs and a photodetector to read heart rate. With great low power options only needing 1.8 to 3.3v and as low as 7uA this device can be very efficient with fast sample rates and noise canceling and motion detection this is a great choose for the product the down side is that we still have no way of seeing though the dog's hair and thus will mess up the readings.

List of features:

- **Low power**
- **Low noise**
- **Optical sensor**
- **Power saving modes**

3.6.4 Pulse Sensor Amped

This device is an optical heart rate sensor, this is cheap and easy to use, although this device is very big and requires a lot of power to use and would require us to put this on a dog's ear and that is not what we want for the product.

List of features:

- **Low cost**
- **Easy to use**

Conclusion

We wanted to add the heart rate sensor as we thought that this was a good way to give the user information about the dog's health, but as it turns out having a reliable cheap and low power heart rate sensor is not possible as we do not have any way to reliably read a dog's heart rate though all the hair. We also decided that the product is more aimed towards training the user to take care of the dog on a daily basis rather than managing the health of the pet.

Table 3.7 Wi-Fi Options

	Cost	Size	Availability	Current	Voltage
FC9000	$10.99	7x7x0.4mm	Readily	-	3.3V
88W8977_SDS	$11.71	8x8x0.85mm	Readily	50uA	3.3V
WGM160PX22KGA	$11.80	23.8x14.2x2.3mm	Readily	147uA	3.6V

We want to add a Wi-Fi IC to the product in the case that one of other communication IC (Bluetooth, LoRa etc.) do not included it not only that the main reason why we want this chip is in the case that the dog is left at home and we do not have Bluetooth connection we have a way to communicate if the dog is in the geofence area. So, we want a chip that is low power, small and efficient for our need.

3.7.1 FC9000

This Wi-Fi chip is low power using only 3.3V and has an ultra-low power mode this is good in case we do not need the Wi-Fi chip to run we can save power. This chip runs on standard Wi-Fi IEEE 802.11b/g/n running only at 2.4GHz which is fine as we do not need high speed connection, we just need a ping to acknowledge the dog's location.

List of features:

- **Low power 3.3V**
- **IEEE 802.11b/g/n 20MHz 2.4GHz**
- **Three sleep modes**

3.7.2 88W8977_SDS

This chip is low power needing only 2.2V-3.3V to operate it has three power saving modes this chip runs on standard Wi-Fi 802.11a/b/g/n with 2.4 and 5GHz with low power Bluetooth but this might cross over with another chip but we will still look at this one as we might go with a GPS/LoRa chip that do not have Wi-Fi or Bluetooth. This Bluetooth is also low power and operates with 5.2.

List of features:

- **Low power 2.2V-3.3V**
- **IEEE 802.11a/b/g/n 20MHz 2.4GHz and 5GHz**
- **Three sleep modes**
- **BLE 5.2**

3.7.3 WGM160PX22KGA

This chip is low power needing only about 3-3.6V, in case we wanted we can add an antenna to the chip to help with heavy noise and works off the standard Wi-Fi 802.11b/g/n with 2.4GHz. although this chip is on the bigger side in case we run into where the chip has to handle a lot of noise this one can handle it. Plus, this one offer two sleep modes to save power.

List of features:

- **Low power 3V-3.6V**
- **IEEE 802.11b/g/n 20MHz 2.4GHz**
- **Two sleep modes**

Conclusion

We may not pick parts from this list but it is a good idea to have backups. I believe the best choose from this list is the FC9000 as this chip is ultra-low power and runs of just the right amount of needed speed with out to many unneeded features and it is very small for a Wi-Fi chip. In case we need something that can handle noise because we do not know as of writing this if there will be heavy noise when the dog is wearer a collar and so having a chip that can handle such noise might be a good idea the WGM160PX22KGA can do that, thought this chip is large we might need to do more research to ensure that we can find a better size chip that can handle noise as well.

Courtesy of https://www.everythingrf.com/products/wireless-soc-s/dialog-semiconductor/787-970-fc9000

Stuck with the ESP32 and it worked well.

Table 3.8 Speaker Options

	Cost	Size	Availability	Wattage	Impedance
AR01232MS-SC12-WP-R	$1.61	12.00x 6.00mm	Readily	30mW	32Ω
CMS-151125-076x-67	$1.87	15.00mmx 11.00mm	Readily	1W	6Ω

We wanted to add in a speaker to the collar to help find and locate the owner in case the dog is lost i.e. is outside the geo-fence. The collar will have a message on repeat that the owner will record telling the person their name the address and phone number. This will hopefully improve the return rate of lost dogs. For the product we want a speaker that is small has very little energy use and has good sound quality so that the person can understand what the message is saying and can hear musical sounds if we go that route.

3.8.1 AR01232MS-SC12-WP-R

This speaker is waterproof rated at IP67 which can last for a while submersed in water. With a very nice small size of 12x6mm and a dB rating of 116 this little speaker has very good sound quality, although it as an impedance of 32Ω which is quite high for a speaker and only a frequency range of 300 to 7kHz. Although these might look like short comings, they are fine for what we want out of this speaker, as we want the speaker to play an audio clip of the owner saying a phone number/ address when the dog is considered lost.

List of features:

- **Low wattage 30mW**
- **Small size**
- **Waterproof IP67**
- **300Hz-7kHz**
- **116 dB average**

3.8.2 CMS-151125-076x-67

This speaker is waterproof rated at IP67 which is excellent for handling large amounts of water. Thought this speaker is a little bigger at a size of 15x11mm but it has a lower impedance of 6Ω and a good range of frequencies from 100Hz to 20kHz but has a power usage of 1W and an average dB rating of 92.5. this speaker may not seem as good as the previous one but this one allows for more ranges of sounds.

List of features:

- **Wattage**
- **Waterproof IP67**
- **200Hz-20kHz**
- **92.5dB average**

Conclusion

We choose the AR01232MS-SC12-WP-R as we feel that having the small size and low wattage is a best fit for the product. We want to make sure we are using the less amount of power as to increase the life span of our products battery life. The CMS is also a really good choose giving us more sound range but we don't really need this as we only want the product to speak a language which we only need 200-5kHz to make a human voice. If we ever wanted to make little sounds and music we can switch to the other speaker if need to.

Courtesy of https://www.digikey.com/en/products/detail/pui-audio-inc/AR01232MS-SC12-WP-R/8021810

Table 3.9 Li-ion/Lipo Charging Chip Options

	Cost	Size	Availability	Max Current	Max Voltage
LM3658	$1.40	3x3mm	Readily	1.0A	4.2V
ISL6292	$2.63	3x3mm	None	2.0A	6.5V
BQ2407X	$2.30	3x3mm	Readily	1.5A	4.2V
MP2615C	$2.85	3x3mm	Readily	2.1A	8.4V

For our team we felt the most important features for our battery protection circuit are: small overall package size, ease of implementation, needs to be able to charge our battery with at least 1A, feature rich for the price, and needs to draw a small amount of power when the battery is not being charged (While sleeping or in standby). The reason we have found these to be the most important lies in the fact that our battery will have a Battery Management System (BMS) already installed so many of the basic safety features are already covered and have no need to be doubled. Some of these features of BMS' are over voltage protection, under voltage protection, reverse polarity protection, over current protection, and over temperature. With the major safety concerns taken case of we decided that overall size should come first as a priority as one of the main feature of our project is the small size we are aiming for and the only way to keep a small form factor is to start with small form factor parts. Next, we only really have four months to complete this project so we cannot afford to use a chip that is brand new to the market and no one knows how to use. For this reason, we found it important that there is good information out about the chip we wanted to use such as excellent datasheets and possibly other projects implementing the same chip. After this the list becomes very simple to explain as we want our project to have a low price point if we ever wanted to turn it into a project as well as we want it to be able to last a relatively long time one a single battery charge and so we must watch out for things that will drain our battery.

3.9.1 LM3658

The LM3658 comes with safety features such as: multiple charging time monitoring systems, over-current and over temperature monitoring, and pre-conditioning for our battery if it is ever overly depleted. Other great features for our project include a 1A charging current if using a wall adapter and a USB limiting feature for when this product might be connected to a USB port such as one located on your laptop. The chip also comes with a sleep mode, low-quiescent current, and a package size of 3x3mm. The most useful feature of this charging chip is that it is designed for the purpose we intent to use it in (USB power interfaces) so it can communicate through USB to any device or just take its max allowed current. The only real drawback of this chip is we are limited to 1 cell batteries if we ever wanted to change the battery voltage, we would also need to change this charging IC, but it comes at the benefit of a much cheaper price for the features included.

List of features:

- **Charges through AC USB wall adapter or USB female port of any device**
- **Preconditioning for depleted batteries**
- **Built in safety timers**
- **1A charge current**
- **Sleep mode**
- **Low-quiescent current**
- **Over-current and over-temperature monitoring**
- **Small package size (3x3mm)**

3.9.2 ISL6292

The ISL6292 is a single cell Li-ion charger chip that integrates good thermally efficient safety precautions with a high output charger. The chip offers low heat dissipation and along with a 2A charging current matching the highest output current on the list. This chip can also simplify the charging circuit in use with it as it has a diode included meaning that there is no need for a blocking diode. Included as well are integrated current sensors and an impressive 1% voltage accuracy on the charging side. Of course, it also communicates through USB to change its output current dependent on the device it plugs into. There are three major disadvantages of this chip. First is that the features that it really touts our project has no real need for and second is that those extra features our group would have to play with have a premium attached to them. Lastly, as always, no matter how good the chip if it isn't available, we cannot pick it.

List of features:

- **2A charge current**
- **Charges with USB**
- **No external diode needed**
- **Integrated pass element and current sensor**
- **Very low thermal dissipation**
- **Over-current and over-temperature monitoring**
- **Small package size (3x3mm)**

3.9.3 BQ2407X

The BQ2407X is a one cell linear battery charger that has great safety features as well as industry leading charging features to lengthen the life of the battery it is charging. First the safety features include reverse current protection, short-circuit protection, and thermal protections for the battery. It also has integrated pre-charge and fast-charge timers, programmable termination current, and limits start up inrush current. The current limit of 1.5A also places it solidly over the 1A threshold that was required of the charging circuit. Lastly, is the added feature "Integrated dynamic power path management (DPPM) function simultaneously and independently powers the system and charges the battery" (datasheet). The major disadvantage of this product is the price.

List of features:

- **Integrated dynamic power path management (DPPM)**
- **1.5A charge current**
- **Programmable termination current**
- **Proprietary start up inrush current limited**
- **28V input rating with over voltage protection**
- **Output voltage tracking**
- **Short-circuit, reverse current, and over-temperature monitoring**
- **Small package size (3x3mm)**

3.9.4 MP2615C

The MP2615C can charge both single cell and 2 cell batteries with up to 2.1A output current. Configurable safety timers are also included. It also provides incredible accuracy with 0.75% battery voltage accuracy. This chip is super-efficient while still having a duty-cycle up to 99%. With the integrated internal control loop and lack of need for an external reverse current diode. The two issues with this chip are that dependent on the current rating of our chosen battery the 2.1A current maybe a bit too much and the price is more than double our cheapest IC for few features we will use.

List of features:

- **Thermal shutdown protection**
- **2.1A charge current**
- **Fully integrated power switches**
- **Internal loop compensation No external diode required**
- **Cycle-by-cycle current protection**
- **Configurable safety timers**
- **Small package size (3x3mm)**

Conclusion

In conclusion this section was a rather easy option. All the sizes were the same so any choices dependent on that were thrown out of the window, and the cheapest option meets all specifications for the part our team wants and has the simplest implantation circuit. With this in in mind we have chosen the LM3658.

Courtesy of https://www.ti.com/product/LM3658?utm_source=google&utm_medium=cpc&utm_campaign=app-null-null-GPN_EN-cpc-pf-google-wwe&utm_content=LM3658&ds_k=LM3658&DCM=yes&gclid=CjwKCAjwgISIBhBfEiwALE19SUQrt96qP41O03YGUyPmviOEY8dWMQ1qS21mmklMDpnjLiW6gwtDDxoCwr4QAvD_BwE&gclsrc=aw.ds

Table 3.10 Li-ion/Lipo Battery Power analysis

Part:	Max current (A):	Min Current (A):
Accelorometer	0.00009	0.000003
LIPO charger IC	0.0006	0.000005
Boost	0.000109	0.000005
Buck	0.000948	0.0000006
ESP32	1.1	0.0008
Speaker	0.01	0.006
RGB LEDs	0.1	0.02
Thermistor	0.001	0.001
GPRS/ GSM	0.5	0.0007
# of Hours In a week:	168	168
	Total worst case current draw:	Total best case current draw:
	1.712657	0.0285136
Amount of time a 1AH could run it for (Hours):	0.583888076	35.07098367
Amount of time a 1.5AH could run it for (Hours):	0.875832113	52.60647551
Amount of time a 2AH could run it for (Hours):	1.167776151	70.14196734
Amount of time a 3AH could run it for (Hours):	1.751664227	105.212951
Amount of time a 4AH could run it for (Hours):	2.335552303	140.2839347
Amount of time a 5AH could run it for (Hours):	2.919440378	175.3549184

From Table 3.10 it is clear to see that the only real option for our groups project if we want to try and achieve one week of use for a week is a 5000mAH battery (5AH). In searching for batteries of this capacity and in a form factor that could keep with our small footprint design the major question remaining was should we do two batteries in parallel (2 x 2500mAH)or 1 larger battery (1 x 5000mAH).

3.10.1 5000mAH Battery

5000mAH batteries come in all shapes and tons of different sizes and are readily available which is a nice change of pace as compared to what our team has seem with every other device on the market. The major disadvantages of 5000mAH are that they are more then double the size of 2500mAH batteries and generally do not come with a on cell BMS for protection against being over drained or over charged.

List of features:

- 5000mAH total capacitance
- Max discharge rate of 3C
- Normal discharge rate of 0.3C
- Small size (133 x 101 x 4.7mm)
- Lithium Polymer Chemistry
- Lifetime of 500 Charge cycles

3.10.2 2 x 2500mAH Batteries

2500mAH batteries are just as abundant as the 5000mAH variants. Again, it is important to emphasize being available is the hardest thing to come by in part right now and both of our possible batteries are available. 2500mAH cells come with a built-in battery management system that can protect them from things such as over discharge, over charge, and short circuit protections. The major advantages to using two 2500mAH cells in parallel is that we could if needed draw more current as the limit of let's say 1C is per cell so if we have two cells that are each rated 1C we can get a total draw of 2C as well as charge them at double the rate. Also important is the size being less than half of the 5000mAH batteries leading to an overall much smaller footprint for our design.

List of features:

- 2 x 2500mAH total capacitance
- Max discharge rate of 3C
- Normal discharge rate of 0.5C
- Small size (60 x 50 x 7.8mm)
- Lithium Polymer Chemistry
- Lifetime of 500 Charge cycles
- BMS included with the cells
- Connector / Leads also included so we can easily attach the cells to our PCB and each other without heating up the cells as much.

Conclusion

In conclusion it makes more sense for our team to go with the 2 x 2500mAH solution as it provides more modularity a smaller overall size and better protection. The only real use for a 5000mAH battery would be if you really wanted or needed a single cell and no parallel was allowed. Luckily for us that is not the case, and we are able to parallel our batteries to our hearts content. The hope will be that we can place one cell above and one cell below the PCB to spread the heat as well as minimize the overall width and length. The depth within realistic limits is not much of an issues as our team is prepared to use almost all surface mount components.

Courtesy of https://www.welectron.com/LiPo-Pouch-Battery-785060-37V-2500mAh

Table 3.11 LoRa Options

	Cost	Size	Availability	Max Current	Max Voltage
SX1276	$7.68	6x6mm	Readily	120mA	3.7V
SX1272	$7.20	6x6mm	Out of stock	125mA	3.7V
LLCC68	$30.00	6x6mm	Limited	118mA	3.7V
SX1280	$6.60	4x4mm	Out of stock	24mA	3.7V
SX1262	$7.18	4x4mm	Out of stock	60mA	3.7V

We wanted to use LoRa as a GPS tracker without having to pay for a subscription like other GPS trackers. This LoRa tracker will pinpoint the location of the dog in hopes of the dog getting lost the owner can easily find them with the collar still active. As a bonus feature, we want to add a travel tracker in which lora can track how far you walked with your pet. By using a similar triangle positioning tracking method much like other GPS expect we will not be needing satellites. In terms of what we want is a nice strong signal LoRa chip that functions at low energy to preserve the battery life.

SX1276

For this chip we wanted to have low power with great low noise signals and LoRa can give us that with this part we can have that and the ability to have a wide range of bandwidth with frequencies going from 137MHz to 1.02GHz this can cover a lot giving us more flexibility in dealing with noise, with a built-in temperature sensor and low battery detection this will make monitoring for power and making sure we don’t over load our chip. On a side note, we must be care of where we get these parts from as we found out these chips run on different frequencies depending on what country as each country uses different frequencies, but this is fine for this one as we can see this one is the international version and has access to a large range of frequencies.

List of features:

- **Thermal shutdown protection**
- **Low battery detection**
- **Low power**
- **Long range**
- **Large bandwidth**
- **Low noise immunity**

SX1272

This chip is very similar to the SX1276 but this one is the north American version, meaning we lose on a little bit of the frequency range but we don't really need it for now as we only want to test the product here in the United States and it will save on money. Although we still have a very large bandwidth 860MHz to 1.02GHz. this chip is also low power and has a noise filter.

List of features:

- **Thermal shutdown protection**
- **Low battery detection**
- **Low power**
- **Long range**
- **Large bandwidth**
- **Low noise immunity**

LLCC68

This chip was recommended as a good LoRa this a standard LoRa chip that has enough features to make it valuable. Which is fine for us as we don't need to many features and it will lower the cost but sadly this part is limited and offers nothing unique to the table and has a very high cost so we will be skipping this chip.

List of features:

- **Low power**
- **Sleep modes**

SX1280

This chip has a very long range of communication as it uses 2.4GHz band, with that it can utilize some Bluetooth tech as a way of communication giving us more space to use on the board or a faster way to communicate the location as we need the Bluetooth for our other features on the board. This board allows use to control the bit rate as needed the allows some flexibility in case of noise issues that might arise from testing.

List of features:

- **Low power**
- **BLE compatible**
- **2.4GHz**
- **Long range**
- **Noise filters**
- **Programmable bit rate**
- **Small size**

SX1262

This chip works under every signal in terms of sub-GHz between 150-960MHz meaning this chip is an international one thus we can take this chip and it can communicate with other LoRa towers in different countries this chip is also low power and has a very small size fitting greatly in our product.

List of features:

- **Low power**
- **Long range**
- **Noise filters**
- **Small size**
- **Large bandwidth**

Conclusion

In conclusion LoRa is a very good solution to our need of wanting to locate your pet, track your pets travel distance and making sure your dog does not go pass your “fence”. The ability to have a non-subscription “GPS” system and have been low powered long range and low noise sounds like a great idea, but as we found out that LoRa does have long range communication and a greater data bit transfer rate the downsides to LoRa is that it doesn’t do good going through walls and other objects as it operates in low frequencies so as we know with low frequencies, we need to have a larger receiver so that the receiver can see said signal. Also, the receiver needs to be in a high location in most cases but with our product being on a dog it will be very low to the ground and may cause issues. Another downside is that LoRa is still a new technology that has only just started to grow as in the whole state of Florida there is only one located in Orlando, this would be nice for the company that made LoRa as they can get more traffic but not for the consumer and most certainly not for us as we will be testing it and many things could go wrong if the only tower where to go down. So ultimately, we will not be using LoRa for our product as it will be inconsistent with its data receiving and it is also very expensive to get some of the chip selected her and also the dev boards that we want to use to program and test the chips and there just is not enough support for this technology to make it easy enough to set up with its lack of parts and tower support so we will not be using this and will be using gps/gprs in its place.

4.0 Hardware Design and Manufacturing

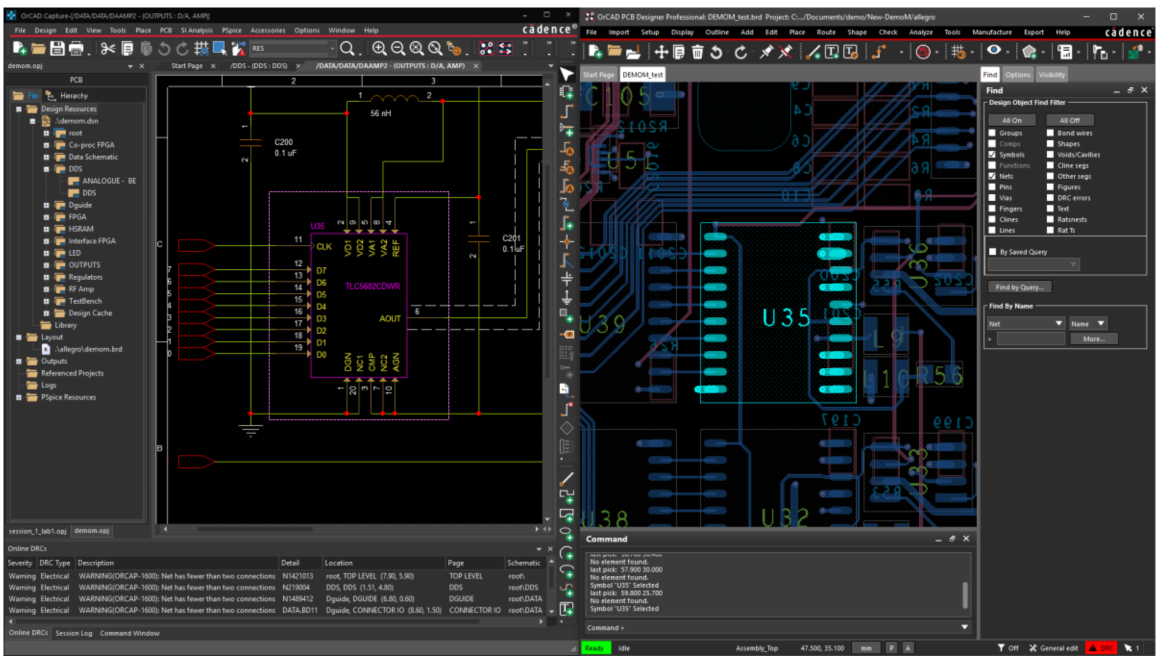
When beginning our hardware design, we knew we wanted a separate power test board as our power circuit was going to be custom not a simple TI WEBENCH model. This decision to not use the TI WEBENCH was made for the added design difficulty and because the TI parts that WEBENCH would generally recommend circuits for are all out of stock as of the Summer of 2021 and are slated to be gone well into 2022. This shortage of parts will be discussed in a later section. In this section we will talk about our team's choice of schematic and PCB design software's, what board house we used for printing our circuit boards, the parts shortage that led us to design our own prototype power board, the prototype power board, and the schematic for the complete board.

4.1 Schematic and PCB design software

For PCB design there are a few industries level tools. The major names are ORCAD, Altium, and Eagle CAD by Autodesk. In the following section we will describe the major differences between the three and why our team chose to go with the one we chose. Our groups criteria when deciding on the software was that we wanted something that was well respected in the PCB design industry and would further our groups Electrical Engineers career opportunities. The software should have good tutorials/ learning material and a community backing it so that any questions could be answered relatively quickly and thoroughly enough for a student to understand. It also is important that the software is easy to use and brings efficiency to the user. If it takes more actions from the user's side to do a single action on one software as compared to another it is a waste of time so the question is who has optimized their systems to make sure that the things you do the most in PCB design can be done quickly and not waste the user's time. Last but most important was the fact that they would need to have a student version available as our team was not willing to spend in excess of $1000 for a software that could be substituted with a free software. Credit also needs to be given to San Francisco Circuits are their comparison was used for much of this decision making due to the thorough nature of the comparison and the fact that our group had no ability to use and learn all three software's just in an attempt to decide which software we might want to use.

4.1.1 OrCAD (Cadence)

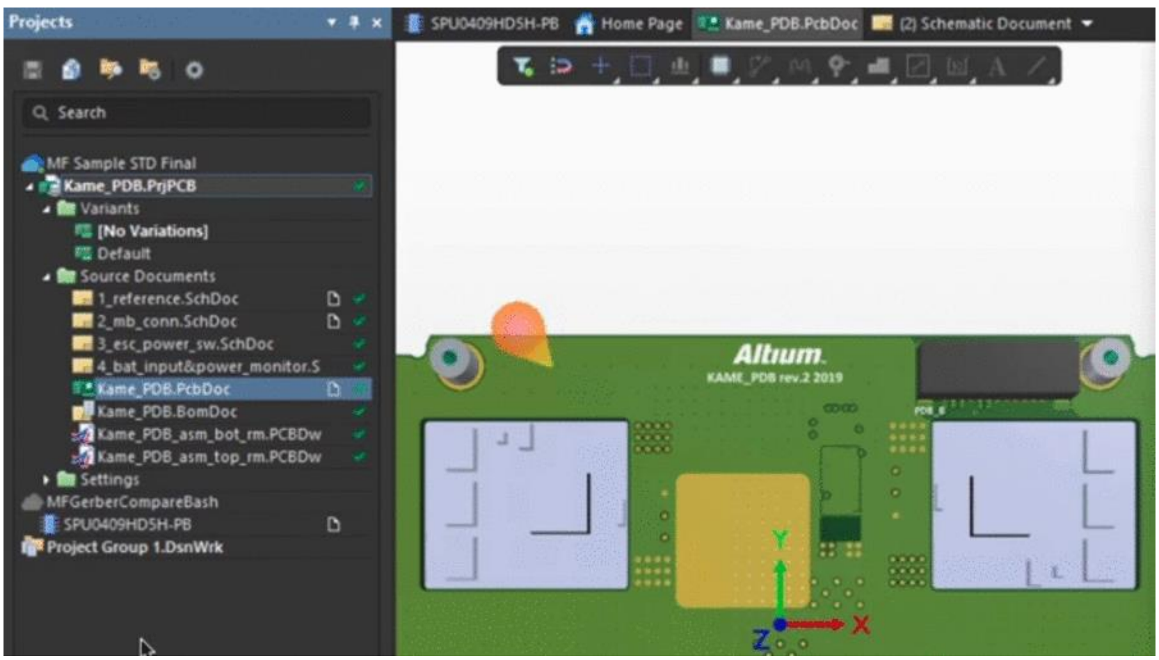
OrCAD was a very serios option as of one our Electrical Engineers has an internship were they regularly use OrCAD and therefore we had experience with the software already and of course with familiarity comes confidence. OrCAD did not come up in requirements for most of the jobs our Electrical Engineers had seen in their search for jobs as they prepare to enter the work force and therefor lost a point or two but again as it was used in one of our member's internship, we do know it has widespread use in industry just not as much in the industries our Electrical Engineers may envision their careers going. OrCAD has a decent amount of help tutorials but overall has the worst/ least learning material as compared to our three selections. The community was not considered as much for this as we already had connections to power users of the software so we figured we would most likely just ask question to the power users of the software if our team needed help. The ease of this software ranks in the middle of our three selections as it has a steep learning curve but once you understand it the PCB editor and basic functions are super nice to use. The schematic editor is a little lacking but overall, quite good as we would expect.



Courtesy of https://www.orcad.com/products/customize-pcb-design-experience

4.1.2 Altium

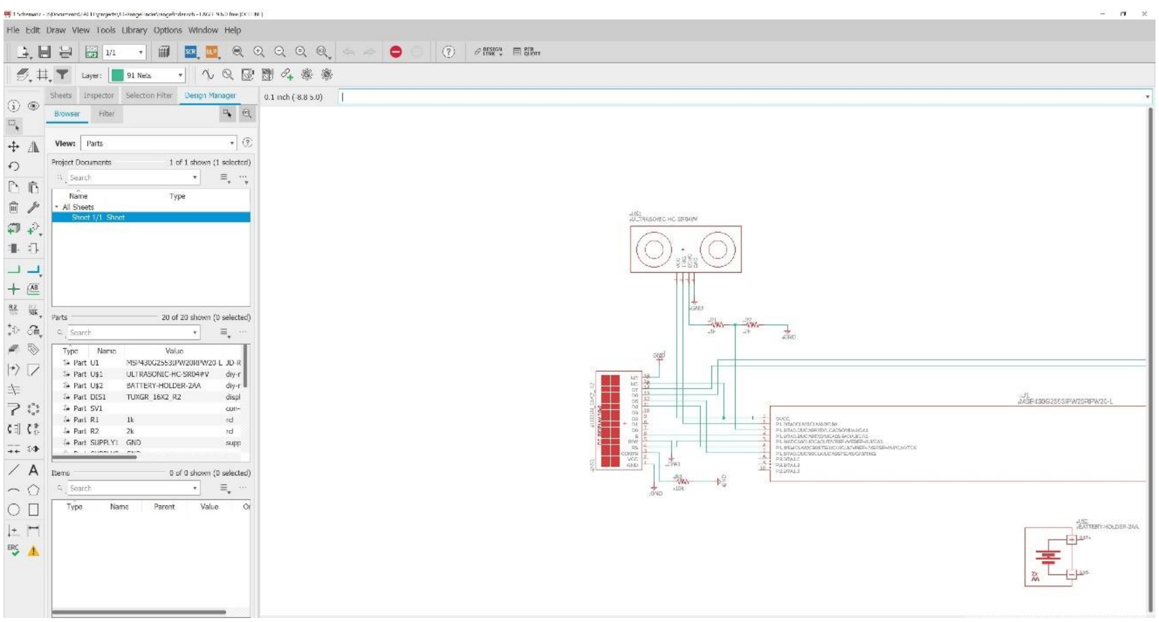
Altium also had made it into the list as our team has small amounts of experience with it as well as knew some power users of the software. Altium did come up on many jobs post as a requirement for application or a preference for hiring. This matters a lot considering the major purpose of this class / project is to make sure you understand not only the book knowledge of the classes but also the application of those classes to better the student's ability to get a job. The educational material for Altium at all levels is also excellent as long as you are trying to find something about a current version. From person experience our team understands that previous versions of Altium are essentially unsupported. The community around Altium is also incredibly active and included everything from interns and students just learning the basics of schematic and PCB design all the way up to the Senior engineers with thirty-five plus years work experience. The overall ease of this software ranked very well on San Francisco Circuits comparison with the best ranking in schematic editing, schematic library editing, and PCB library editing. The learning curve is not quite as back but it still could be better. Overall a great schematic and PCB editor as we would expect.



Courtesy of https://www.altium.com/altium-365

4.1.3 Eagle CAD (Autodesk)

Eagle is used by all Electrical Engineers and Computer Engineers at The University of Central Florida as we are strongly recommended to use it for Junior Design. Due to this, our Electrical Engineering team has experience with this software which could lead to more comfort on this software making it an easy choice. As for jobs our Electrical Engineers found more jobs of interest that required or wanted the applicant to be proficient with Eagle but not as many as Altium. To our team this meant that it passed this test and was at the very least a software that had permeated the barrier that generally separates educational content from the software with industrial use. Eagle has without a doubt the most educational content available of all three of these with Altium being in second. Anything you could ever want to do in eagle has been done and documented. The community around Eagle is also vast as it is connected to the Autodesk systems for free users with limited features and thus has lots of hobbyist with large amounts of knowledge all over the forums as well as students just beginning their education into the software and even the occasional Senior Engineer with a career filled with experience. Sadly, this software drops the ball when it comes to the ease of use ranking last of our three choices in four of the five categories tested in San Francisco Circuits comparison. Also, through personal experience our Electrical Engineers found it unintuitive and clunky.



Screenshot from Team Member, Cody Khong

Conclusion

In conclusion we found Altium to be the best choice. This is due to it having the most job opportunities in the career fields our Electrical Engineers were interested in and being well respected as the industry norm. It is also important to note that the community around Altium is committed to high level education of the PCB design community and Altium has and continues to put out very high-level content and features that the community has asked for. Being the most common easiest to use and having arguable the second-best tutorials Altium was an easier selection. The only major downfalls of Altium are the lack of notes for previous versions and the steep learning curve as compared to a software such as Eagle.

4.2 Printing Board House Selected

Once our team had decided on our schematic and PCB design software, we must decide were we want to print our boards. The major PCB printing houses are pcbway, JLCPCB, and OSH park. Each of the choices listed has their own disadvantages and advantages. Some have better quality some of just faster and getting your board built and shipped. Nevertheless, we must choose what happens to be the most important aspects of the board house we want. For our team, the most important things will be the quality of the PCBs manufactured, the tolerances, and then the time it takes to get printed and then mailed to us.

4.2.1 Pcbway

The board house pcbway is the most common and most well-known board house in industry. They are known to have a large number of selectable options and very fast order to deliver time. Some of the major features that pcbway has for selection is the ability to do up to ten-layer boards, both FR-4 and aluminum PCBs as standard with options to do FLEX and many others, board thickness anywhere between 0.2-2.4mm and traces as small as 4mils. They also have an excellent error correction department for fixing/ catching small PCB design mistakes visible in the gerbers. Generally, your boards will be reviews within the same day to the next day and completed within the next three days. Meaning that you can get your PCB back within six to ten days after ordering making them excellent for prototyping. The only disadvantage of pcbway is the price. There are much cheaper options so at times it may not make sense to spend as much per board.

4.2.2 JLCPCB

JLCPCB is also a well know PCB manufacturing house as they are generally pretty fast and cheap with a decent number of options. As a comparison JLC can do up to 6-layer boards, only FR-4 PCBs, anywhere between 0.4-2.0mm board thickness, and their smallest allowed trace width 5mil. JLCPCB is generally the cheapest solution when you need PCBs printed and they are relatively fast. Their PCB review engineers generally get back with a decision or correction within two to three days, then they build the board in another two days, and generally the shipping takes the same three to seven days leading to a minimum of eight days to get your PCB back and twelve as a general high side of the time you might wait. The only major issue with JLC is the quality of the PCBs they manufacture. The quality is generally subpar with member of our team have delt with open vias and other PCB manufacturing defects. This is not always a problem but when you are prototyping a PCB board it can be extremely difficult to find any issues with your design when there is a. issue with the PCB that is not your fault.

4.2.3 OSHpark

OSHpark is a United States based PCB manufacturing facility with decent prices great quality but subpar building and shipping times. The options available with OSHpark are very limited. They are limited to up to six-layer boards like JLCPCB, again only FR-4, a board thickness of 1.6mm, and the smallest allowed trace width being 6mils. OSHpark generally has decent pricing being not as expensive as pcbway but much more expensive than JLCPCB. Their engineering team in charge of reviewing boards are also a bit slower than the options. The turnaround time for OSHpark is also not very good as they take at least five days to get your PCB printed and then you have to wait the general three to seven days for the PCB to ship. This means that the best time for OSHpark is eight days and the slowest general time would be for their fastest option would be twelve days.

Conclusion

When making this decision we weighed the pros and cons of each with the importance of each of the part and decided to go with pcbway. This decision was made because we do not have time to deal with possible board house manufacturing issues while prototyping and testing boards are there are bound be things we misunderstand and build incorrectly and then later have to change. With JLCPCB it just did not make sense for us to wait possibly well over a week when for OSHpark when pcbway would only be a slight bit more expensive. Also, the added optionality plays a large role in our choice as the options allow for changes that may need to be made down the road whether it being adding possible FLEX PCBs to attach our LEDs or going down to 4mil signal traces.

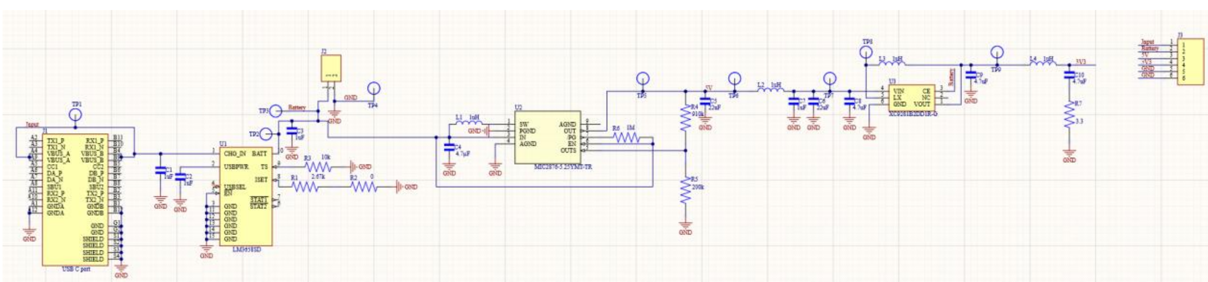
4.3 Parts Shortage of 2021

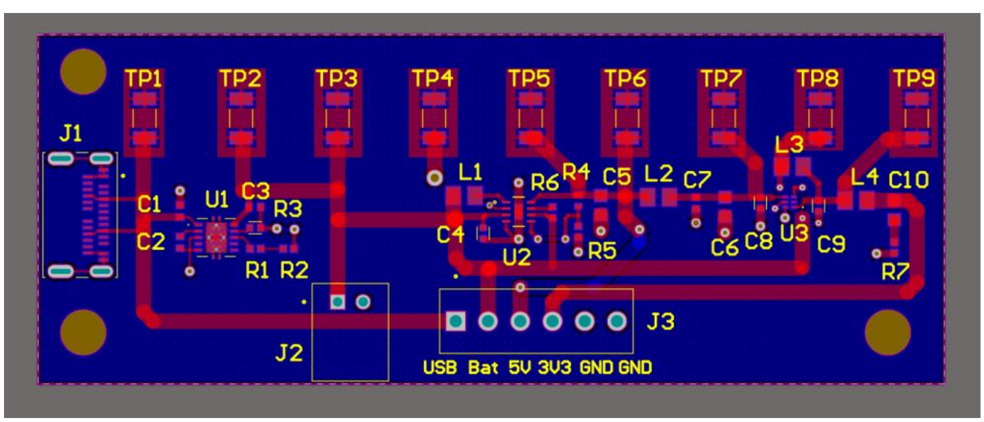
During the covid pandemic most industries slowed or shut down and the IC and auto industries are no different. This is especially true when you remember that most the IC manufacturing facilities are located in parts of the world that had the strictest restrictions about lockdowns. Due to this the auto industry when opening back up emptied up the coffers of spare ICs while all other industries kept their normal order qualities and consumers have spent more money on electronics as compared to outside of the lockdown times. With this all happening there are many parts with over fifty-two week leads times that have not gone down for the last fifty-two weeks. With parts being so hard to aquire and the lead times being so unreliable our group had to look outside of the most common parts to find any availability. This made resources such as WEbench absolutely useless because all of the parts that the software would recommend are all bought up and have lead times over six months. Thus, our team decided we needed to do some extra levels of prototyping and designing to make sure each sub system would work. The first subsystem to be affected was our power circuitry as most of the common buck and boost ICs are bought out especially all of the fixed Li-ion application specific ones, which also happens to be the section of the IC market we are checking through. Our team decided to look for parts outside of the norm due to slight deficiencies or added design complexity such as using an adjustable boost converter as compared to a fixed boost converter because there is some availability in that market. Generally, the adjustable output boost is avoided because of the added voltage divider needed to select an output voltage. Our group decided we would much rather use a $0.60 adjustable boost with two added 0402 resistors as compared to a $3.00 fixed boost with no added voltage divider.

4.4 Prototype Power Board

Due to the parts shortage discussed above the entirety of our power circuitry was designed from scratch using the datasheets from parts found off of DIGIKEY and MOUSER. Luckily for us the parts we chose were built with the for the application we were using them for. This definitely added complexity to our design and required that we test our power circuitry separately before building a protoboard with all sensors in an effort to not destroy any of our sensors while testing the power circuitry and also being able to check the stability of our three-point-three volt and five-volt rails with no load all the way to full load of all sensors, processors, and eventually even RGB LEDs.

When designing this board our group knew we would need a Li-ion/ LiPo charger/management IC as well as power ICs that could convert the variable voltage of the battery to three-point-three volts and five volts. To do this we decided to use a boost converter to get the voltage of the battery to a constant five volts and then from that five-volt rail make our three-point-three-volt rail using a buck converter. This design requires that we have a boost converter that can handle both the five-volt rail demand as well as the three-point-three-volt rail demand. This meant that the max current draw of the boost regulator should at least be one-point-five-amps. About one amp for the RGB LEDs and then another half an amp for the ESP32 and sensors.





4.5 Schematic for Complete board

We worked with Altium to do the complete board to do this with minimum headaches we also did this in parts and test them one by one although this will increase the price of our design budget but this will ensured that no parts will break, will save time on troubleshooting, and will save time if we miss something on one of the prototype boards. As we learned to use Altium we found it to be a very nice, straight forward program that does not have any weird program issues like eagle. We used websites such as SNAPEDA to find the footprints thankfully unlike eagle which our team had many issues with before, trying to download and upload footmarks in to eagle.

To make sure both Electrical Engineers are involved we will work on different designs and pick out which one is the most reliable and easy to manufacture, as they say no two-engineer design are the same, Both Electrical Engineers wish to learn to use Altium, and want to have experience looking over other designs and checking for manufacturability. It is important to note that we did wish to try and make the final product a multilayer board with more than 4 layers this is so we can improve on the overall size of the product as we wish to not have a giant block sticking out of the dog's neck. We know young dogs can be very rough and could hurt itself or damage the product, so we want to make sure the product is very comfortable for the dog to wear both in terms of weight and shape.

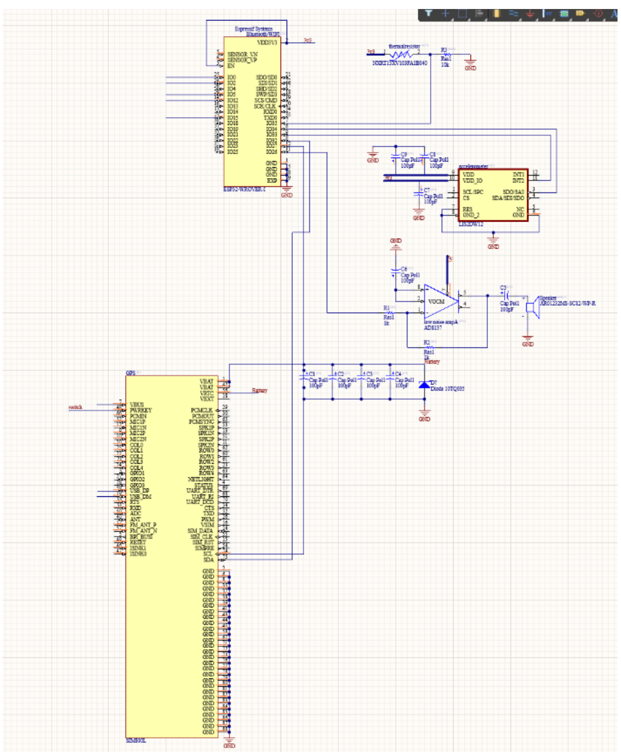
As was mentioned above we want to work on the board in parts as to not have all the issues at once and then try to figure which one is causing the issues and having to spend days of troubleshooting and testing to find the issues. So, we started with the power board as seen above this board is the most important part of our board as its name says it will be powering our device and charging our battery to continue giving power to out device, we talk about having a buck/boost design to maintain that 3.3volts and also boost the power to give to our LEDs which would need 5 volts. Now we have this set up we wanted to place a port on the board to test to see if the component is work but also to test if the layout works in person as sometimes simulation is not always true so we place nodes at the start and end of every chip.

After testing this board, the voltage ripple was seen to be below 10% with the only major weakness of this design being a severe dip to near 4 volts on the 5V line when loaded. Luckily this did not affect the 3.3V rail which was the most sensitive as 4V still had plenty of clearance for the 3.3V rail to function as expected.

Next, we tested out the SOC as this will be the brain of the operation it will be receiving and transmitting all the information from all the different component and will be sending the signal to the user's phone. We put all of the sensing components on here such as the thermistor and accelerometer, and were able to have it read out the heat next, and number of steps. Next we tested the Wi-Fi on the chip from here to see if it can send the signal as well as the BLE for the same functionality, Lastly the speaker and amplifier were tested to make sure we could effectively store and then product the sound from a file within the SOC.

After testing our sensors we kept on adding parts to the SOC to see if it could handle all the parts giving all kind of different data and we gave the GPS/GPRS a test run here as well to see if we can manage to get the SOC to communicate the GPS and then communicate that information to the phone of the user. This feature took a good amount of time to implement but makes our product able to complete in the pet active wear space which we believe is crucial.

Finally, the very last thing we put it all together and tested to see if every part can communicate to one another see if there are any noise issues with any of the parts.



5.0 Realistic Design Constraints

The following few sections discuss the varying constraints to consider while designing and producing our device. These constraints will be a major factor in how we were able to test and build our project. They are useful to keep in mind since many of these constraints will also serve a secondary purpose of being guidelines for us to follow. By discussing and exploring the constraints of various topics, a clearer picture is made of what is possible and what should be avoided while conducting our research and development.

Coming up with the idea of helping new and current dog owners preserve a lifetime responsibility with their dogs introduced several challenges throughout our research and development from day one up until to this day. We will examine closely these constraints in the next few sections as a means of providing our own opinion, considerations, and solutions. In realistic design terms from day one, we were able to sketch and come up plenty of features, use case and application. The discussions for ideas had also plenty of challenges. Mainly, we were concerned about the amount of time we were able to work with to introduce a legitimate prototype and detailed engineering design. When timing becomes a challenge, it also becomes a factor of parts availability as we know that during this time chip shortage has become inevitable. We quickly realize that researching for hardware components was going to become prominent for the next few weeks as we continue to research the most appropriate parts for our development.

As hardware research was ongoing, we set another focus on safety and comfort of the housing design. We thought that this was a primary concern as we want to make BarkaRoo collar be able to handle the most common weather scenarios and handling of the device while attached to a dog. Thus, the challenge was to create a stand-alone collar or a collar attachment. As per group discussions, it was concluded that because of time constraints and economically appropriation that a collar attachment is the best route for this project.

The few constraints from above were just an example of challenges we had to go through. As a group of future professional engineers, we wanted to design this BarkaRoo collar in accordance with standard regulations for safety and design. Economic, time, environmental, ethical, manufacturability, health and safety, and social and political constraints always kept us on our toes during research and development of product lifecycle. This in return will give us confidence with the development of our collar.

5.0.1 Economic Constraints

In this section, the economics constraints are discussed. This is an in-depth discussion about the market our product is attempting to enter and its impact on the economy. This constraint will help us keep our overall spending in mind and to be wary of approaching an undesirable final product price. Since price and demand is always varying for any product, this factor must be kept in mind while coming up with a parts list for our device. At the end of the day, money is a huge driving force behind the success or failure of a device. The economic constraints that will be encountered should be acknowledged before they are encountered. If a device is too cheap and offers too little, it will fail. If a device is too expensive and has unnecessary components or features, it will fail.

As of right now (7/6/2021), the predicted cost of our smart collar would be around 135 dollars. This predicted price includes the microcontroller, various sensors, modules and other miscellaneous components (wires, resistors, capacitors, etc.). Our initial thought process for this project was to think of all the features that are currently offered by competitors and include extra features into our device to differentiate ourselves. As a result of some of these ideas, different modules and sensors were included in our preliminary cost distribution to accommodate for these features. As we test these different features and see what works and doesn’t, we will adjust the final cost the accurately reflect our finished product.

In addition to wanting to differentiate ourselves from existing products, we also had to consider the how much each member was willing to spend on the developing the product. The price range that everyone was comfortable on spending was around 100 dollars a person (subject to change as time goes on). With a set limit now in mind, the goal now was to find reasonably priced hardware that must be able to handle the functionality we wanted. When researching into the parts for this project, the price and the features was taken into heavy consideration to ensure a functional, cost-effective product.

After taking into consideration what we wanted and/or needed in our product, we had to consider the market space we were attempting to enter. While researching competing products, we found some of the following prices:

- **Animo - $99.90** - https://www.surepetcare.com/en-nz/animo
- **FitBark 2 - $49.95** https://www.amazon.com/dp/B077MDJYKQ?tag=technobark-20&linkCode=ogi&th=1&psc=1
- **Whistle FIT - $79.95** - https://www.whistle.com/products/whistle-fit-pet-health-tracker-activity-monitor?irclickid=TDmWEwV%3AYxyLUpiRdEQQzzdZUkB3L8XRRxpq0k0&irgwc=1&utm_source=impact&utm_medium=affiliate&utm_content=707410&utm_campaign=Technobark

As seen from the above prices, the space we are trying to enter is extremely competitive in terms of pricing. However, we took into consideration that our ideal product, as of right now, will have included features that are not found in the competitors'. For example, we plan on including scheduled LEDs, location tracking, and database-backed information and profiles for each dog. Those are only some of the features that we were able to see from the surface that separates us from the competition. In addition, our estimated cost being currently reported is for producing exactly one, finished product at the time. Since various components in our device like the Wi-Fi module, Bluetooth module, and temperature sensor can be mass ordered for a lower cost per unit, we should ideally be able to dramatically lower the cost of the finished product using the same quality parts. With this in mind, we predict the finished product cost to be in the range of 100-140 dollars. This price range in addition with our extra features offer us a padding for being a bit higher cost than the competing products.

Taking into account our mass produced, high featured device, this would likely lead to our competitors releasing an updated version of their product that would include the features that were previously missing. As a result, this would raise the price of their newest iteration of the dog collar. Furthermore, the increase cost of production and final product release would lead to adjustment of selling price since the profit margins have to be taken into account as well. Our product would encourage competition and innovation in this market space.

As it stands now, our expected cost range for one device would be from 100 to 140 dollars. We expect the pricing to change constantly as we develop and perfect prototypes. In a competitively-priced market, we aim to have finished product priced a bit higher than others but have more built-in features. Although similar products exist, we hope to standout with more features, better ease-of-use, and no paid subscriptions.

Since our products aim is to be low cost, we can confidently say that the maintenance of our product will also be low of cost. The main concern of degradation will lie within our battery system that will require a replacement within 2 years. This maintenance will also alleviate the cost to our consumers and making sure consumers have ease of access at the comfort of their home. This cut the cost for logistics of sending the device to repair centers and back for regular battery replacement.

Courtesy of

https://www.whistle.com/products/whistle-fit-pet-health-tracker-activity-monitor?gclsrc=aw.ds&&utm_source=google&utm_medium=cpc&utm_campaign=Google_Search_Brand_Beta&utm_term=whistle%20fit&utm_content=521314117841&gclid=CjwKCAjwo4mIBhBsEiwAKgzXOMHRZCBvzI1uJL_YPbSmoWRLTF_5pzLiBk2-EC_ANfyJ8pTJPRPn9hoCosIQAvD_BwE&gclsrc=aw.ds

5.0.2 Time Constraints

This is the time constraint section where time is discussed as an aspect of our project to properly be properly aware of. Time is a huge factor in any project since this will give us a general idea of how much time we can allocate to certain aspects of the project. This constraint will help us form a general timeframe for research and development. With a timeframe to build this dog collar, milestones can be drawn up to ensure the best use of time is being made. These milestones will help us keep track of overall progress. Additionally, they will help with ensuring that project members are getting tasks done to contribute to the finished product. By keeping time in mind, procrastination could be kept to a minimum and rushing a product will also be avoided.

Time is one of the most important aspects of our project. Our team consisted of three engineers with an internship and one with a part-time job. It was necessary for us to be able to come up with an engineering design that is prevalent to how much time we are able to commit on our project. Academics and job assignments fulfilment every week most definitely sets up a lot of constraints on the overall design of our project. Summer semester timelines is a great challenge to tackle in time management and engineering development.

Fast prototyping was essential during this time of the year for senior design due to its short week terms. This itself was a huge setback on being able to acquire materials for testing and the ability to find excellent component integrations. Hardware research timeline was much shorter than anticipated during the first few weeks. Much of that timeline spent was because shortage of popular chips and dev board kits made available to us. This would have been very beneficial for our team to be able to acquire a variety of MCU’s, development kits, and other component subsystems for testing.

Furthermore, weekly team meetings indicates that time constraints on software developments will be applicable for our project as we see a growing trend for the need of web application capabilities associated with BarkaRoo’s features. The software team decided to compile a short list of necessary features that needs to be integrated towards the software development. Rather than filling in the gaps of features between the lower end and higher end collar solutions due to underlying time constraints.

5.0.3 Environmental Constraints

Another constraint will be the environmental constraints. Most products in the United States require a certain level of environmental friendliness. Considerations were needed to ensure that the dog collar is not deadly or harmful to the environment. Extensive research will be required for each physical part for the device since there could be potential to be not eco-friendly. Should there be an encounter with potentially harmful parts, then appropriate steps to avoid this part will be taken and accompanying parts could be adjusted as well. This constraint provides an extra level of care and consideration when determining the overall environmental impact of our device.

Moving the world forward to a greener future strongly discouraged our development from using materials that could be toxic, non-recyclable and inefficient. Though we planned to use a 3D printer to produce our housing prototype we looked closely at the material that is commonly used with 3D printers, PLA or known as Polylactic acid. “PLA is partially biodegradable made from cornstarch that breaks down easier than other synthetic filaments.”(all3dp.com) Although PLA is partially biodegradable, specialized recycling facilities known to handle this material are able to break it down into a useful filament. As for the power source of our developments, lithium-ion are recyclable and again specialized recycling facilities are the only ones able to handle this electronic waste. We can assure our consumers that our product is 100% environment friendly adhering to local and state regulations for safety disposal and recycling of our product.

5.0.4 Social and Political Constraints

This section will be discussing the social and political constraints involved with our dog collar. The political constraints will explore the laws and regulations our device will be residing under. Since the device is going to be made in the United States and aim to be released in the United States, the device will have to adhere to regulatory bodies and government’s laws and rules. This constraint is especially important since the device's success will rely on adhering to laws and regulations. If the device does not follow the regulations placed by the government and regulatory bodies, then the device will never reach the market. Therefore, this and all constraints should be held with utmost importance to ensure the success of our dog collar.

Additionally, the social constraint will be discussed in this section as well. The device we are intending to produce will rely heavily on human interaction. Since this device will be primarily used by humans, human error and other behaviors have to be taken into consideration. The possibility for error coming from someone not us will require careful considerations and forward thinking. Most erroneous interaction with our device must be taken into account. If we can take this into account, then the proper steps can be taken to ensure that we are not liable to possible repercussions.

The goal of our device is to improve the quality of life for both dogs and dog owners alike. Our original goal was to make a product that encouraged good petkeeping by having most of the features revolve around the pet owner’s behavior in addition to monitoring the dog’s health and activity rather than adjusting the dog’s behavior. In essence, we would like to provide a tool that help form better habits for people to increase the quality of life for the dog. Furthermore, this device could also be useful for dog owners with children since this device would provide visual indicators using the LEDs to let the kids know when it is time for certain activities (feeding, bathroom, playtime, etc.). A social constraint for us would be a child’s receptibility to these LED indicators. Although our goal is to have a product that easily understandable by various age groups, we recognize that the younger age group will be the most difficult to reach. As such, we plan on fine tuning our user interaction with both our mobile application and hardware to be readily understood and used with general ease.

Furthermore, another social constraint we thought of while discussing our product and researching other products would be the placement of the collar by whoever decides to purchase it. Our device will have tracking hardware and software on both the collar and application. If this collar is not used as intended on pets and by pet owners, then a situation may occur where someone is using the collar to track someone or something that is not a dog.

In terms of political constraints, we see very few issues our product might encounter with the proper documentation and included, advised discretion. Although we intend on creating a product very similar to those that already exists, we will design our device completely by us using tools, products, and devices that are intended to be used to make separate, more intricate product. The hardware will be assembled and tested completely by our team. The software will also be programmed completely our team. The tracking hardware and software we intend on using and creating for this project does not break any United States laws. We are planning on including a warning on our product as well that state that this device is intended only for use on dogs and nothing else.

5.0.5 Ethical Constraints

The ethical constraint section will discuss the ethics that will be expected while researching and developing our dog collar. This section helps us conduct our development and testing in a way that is not harmful to any participants. Our main concerns are going to be with the dog since that is where the device will be residing for the majority of its product lifetime. Even before the collar makes it onto the dog, ample testing must occur to ensure that no harm can be caused by placing the collar on the dog. Testing will be limited since animal testing will clearly be off the table to ensure no laws or regulations will be infringed. The device will also be placed in or around a person’s home. Therefore, necessary precautions must be taken when asking various testing groups to handle the device. With a device relying on human interaction with a living pet, proper ethics must be placed to ensure no one is harmed in any capacity.

Since this product is intended to be worn by dogs and used by humans, one of our ethical concerns would be how we would test the product. Ideally, we would like to be able to see how this product works in the real world. Seeing the interaction between humans and dogs would help us see the true effectiveness of our product. Due to United States’ law, animal rights groups’ regulation and other ethical concerns, we decided it was best to not practice any animal testing especially with prototypes potentially being a risk factor. Instead, we will be physically simulating a dog in our experiments to test the GPS, Wi-Fi, Bluetooth, indicators, and other features on our device. Our main concern would be testing the various health related sensors on a simulated, inorganic dog. However, we will be able to test different methods in achieving an accurate result and corresponding readings from our sensors. Regardless, this methodology would work well for our device since this collar would be for the adjustment of human behavior rather pet behavior. Therefore, a simulation of the pet would provide minor obstacle in testing our product since this would negate the potential of any animals being harmed in the process.

Looking deeper into the motivation behind our project, we will also need to consider the human side of our product interactions. The goal of our product is to help make pet owners’ and/or their children form better behavior for their pets’ quality of life. Taking this into consideration, we will potentially have different test groups with different family situations to test out our prototypes as they are released. Of course, we will have a list of items we would like the test groups to critique as well as any potential malfunctions we did not see before. This survey will be fine-tuned and specified as we are closer to a physical product. From there, we will take this feedback from our test groups and revise the iteration of our device. This is our plan for human testing as of right now. However, this does raise the ethical concern of making sure that device we use for testing is fully safe for human testing and does not harm anyone in anyway. Once we can ensure to the best of our abilities that device is safe for use by others, we will then give it to our test groups for the feedback with information on the device and how to properly use it and monitor it without getting hurt.

5.0.6 Health and Safety Constraints

In this section, the health and safety constraints will be discussed for our device. This dog collar will be placed on dogs and used by humans. Taking these two groups into consideration, necessary precautions and planning must take place to ensure that no one gets injured or harmed in any capacity. The device will be primarily around the dog. Therefore, we must ensure that no components or features will harm or disturb the dog. Additionally, the device will be used by humans. Any testing or use by humans must be carefully planned and taken into account to make sure that no one can be injured or harmed by the device. If the device is clearly not safe or healthy to use, then the final product will either not be released or even approved for release.

While developing and testing the product, we will ensure to the best of our abilities that our device does not trigger any health and safety concerns. Since this device will predominantly reside on the dog neck, we will have to ensure that the battery, wiring, and all components with electricity will not malfunction at any point and injure the dogs or their owners. Once we can ensure that device has a certain level of durability, we do not see the collar raising health and safety issues. Although the potential of our hardware malfunctions while on the dog is a concern, we do not foresee this becoming a huge hurdle for us to overcome since most of our hardware will be low voltage. To further ensure the safety of the dogs, we will be making the enclosure for our circuitry and wiring waterproof and dustproof to the best of our abilities.

Furthermore, our prototypes will include a collar as well as the enclosure for the hardware. The collar will be consistent all throughout at testing at this given moment. However, there should be concerns about the fitting of the actual collars and how the weight of the enclosure might shift on the dog. Once the final product is released, we will keep the weight distribution in mind as well as attempt to make the enclosure suitable for most third-party collars that are not provided by us to ensure the device fits the dog.

Looking deeper into the potential health and safety constraints, we were concerned with the potential of the LEDs to potentially scare or blind the dog while in use. As such, we discussed solutions to this potential problem. We determined that enclosure will be placed either the side or below the dog's neck around where a normal collar would be placed. We decided not to remove the LEDs altogether since this would take away one of the core feature we wanted on the device. The LEDs will serve as a visual indicator for both the pet owners and their children for certain activities. Therefore, we agreed to test variable brightness settings on the LEDs that would be noticeable enough for humans but not distracting or harmful for the dogs.

In addition to the potential of the harmful brightness to the dogs, we also have to take into consideration that the speaker and microphone could potentially distract and damage dogs' ears. Our main point of concern was the potential of feedback that would ring and cause the dog any discomfort. As such, we will take this into consideration while testing our product to ensure that there will be no such noise in the final product more. Furthermore, the volume of our included speaker could be set so that it only produces a capped decibel rating that is unharmful the dog's ears. Though this is a not a major concern for us, it is definitely something to take into consideration while prototyping and testing our device.

5.0.7 Manufacturability Constraints

This section will discuss the manufacturability constraint of our dog collar. Our device's manufacturability will determine how well-suited it is for release as a final product for the general public. The cost and supply chain will be taken into account while determining how well our product can be mass-produced. Furthermore, this constraint will also allow us to explore the longevity of our device to see if it could fit into the market we are entering. Job potential and room for expansion will also be explored since these are critical factors in manufacturing a product. Researching the device's manufacturability will determine if it has a realistic space in an ever-changing market.

As the team was discussing manufacturability constraints, we acknowledged our shortcomings as a team of four and the limitations that comes with it. Since the team is significantly smaller in comparison to that of a larger company, the product will have to go through a seemingly rushed research and development life cycle. If this product is successfully assembled with a team of four, then scaling up the size of the team and reducing other constraint factors will greatly decrease the manufacturability constraint. Additionally, the device we are producing for this project will be at higher cost than if it was mass produced. This is due to the fact that we are ordering parts in a smaller quantity that increases the price of our final product to about 140 dollars per unit of production. However, the price of our device will be noticeably decreased if it were to be mass produced since then a higher order of the components can be placed for a cheaper price.

In addition to the lower cost of production for mass production, the features we plan on including also must be taken into consideration. A significant cost of our product comes from our initial lists of features we would like on our product. With these features, various sensors, modules, and accompanying hardware is needed for the device. If these features are successful and does not exist in competing products, then this drives up the value of our product in comparison. Since we could potentially offer more in our product for a bit higher price, this would drive other competitors to include the feature they are missing in their product into their own. Depending on how the competition approach improving their product, this would naturally lead to a higher price tag on their new version of the dog collar. While offering more features that would lead to a higher price, we also would be encouraging competition in the market as improvements are made and released. In the long-term, this would lead to a more diverse market for consumers. In the case we find the cost of our product too expensive for what we have to offer, we could potentially cut non-essential cost in our device. For example, the rough cost of our 140 collar includes the cost of an included, vinyl collar at 20 dollars. When necessary, we can lower the price of our product by accessing what is superfluous to the finished product.

Furthermore, we will also access how long it takes for a team of four given a three to four months to assemble this product. Time will be a large factor in the manufacturability of this collar. However, if a team of four can turn around a working product in a few months, then mass production site should be able to be more efficient in releasing a large number of collars. Outside of the immediate team, ordering parts from various manufacturers should also be taken into consideration. Since the turnaround time of a finished product is limited by the supply of parts, shipping time and available quantity must be monitored while mass producing this product.

Due to the nature of this device, long-term use and upgrades for this price is extremely extended. The device is built to be durable as is with its in-house build and potential water and dust proofing. As a standalone device, the final collar should work with minimal malfunctions for a reliable amount of time before the hardware fails. Furthermore, the collar as of right now is limited by the available hardware and technology being used today. As time goes on, hardware and software will improve either incrementally or dramatically enough to warrant releasing the next iteration of this device for newer features or better reliability. Therefore, a long-term need for engineers will be necessary to keep this up to date with the improvements being made hardware and software. In turn, this will allow the collar designed today to be more sustainable later on when there are leaps and bounds in technological advancements.

To make sure that a business model for this collar could last long-term, a determination for warranty must be made. Currently, we believe the collar with all its necessary hardware components could last anywhere from 7-10 years (this time span will change as test continues and develop). With this ideal number in mind, a potential for a 2-year warranty being included with this product will help how long a company will last. Should the product fail, a judgment could be made by the team to determine whether the device was overused or defective. Should the device be found defective, then a repair could be made, or a replacement could be sent back to the consumer.

5.1 Related Design Standards

Related design standards give us a scheme of an already existing object or technology that has met the expectations of clients. *ISO* and *ANSI* address more of the technical standards on an international level and within the United States. These standards exist for one reason and that is to provide solutions to emerging problems and questions of products capabilities. Standards are also created in different level of solutions on a basis of specific product needs and application. In return, allowing researchers and developers to find the optimal solution with their development needs. Applying these standards on our research and development gives the consumer, supplier, or client the confidence that the products integrated follow all protocols set by the issuing governing bodies. Within our development we will closely monitor these applicable standards on parts acquisition and product testing ensuring safety and robustness of our final product.

5.1.1 Weather

Varying weather condition is a prominent standard usually discussed within product developments. These conditions allow operability of the device within allowed parameters specified by our team of engineers. As per our development procedure, water protection is the top priority in design of the enclosures. IP65 enclosure is the ideal bound of the development of BarkaRoo collar making sure splashes of water will prevent damage within the electrical unit[4.1.2]. Humidity is a main concern as well in connection with water damage. High humidity can cause water condensation that eventually leads to corrosion if not correctly dried. Low humidity on the other hand is susceptible to electrostatic discharge(ESD) that can result in frying of devices components.

IP Rating	Protection	Description	Test Method
IP65 Enclosures	Able to protect against water jets	Water projected by a nozzle (6.3 mm) against enclosure from any direction shall have no harmful effects.	Test duration: at least 15 minutes
			Water volume: 12.5 litres per minute
			Pressure: 30 kPa at distance of 3 m
IP66 Enclosures	Able to protect against powerful water jets	Water projected in powerful jets (12.5 mm nozzle) against the enclosure from any direction shall have no harmful effects.	Test duration: at least 3 minutes
			Water volume: 100 litres per minute
			Pressure: 100 kPa at distance of 3 m
IP67 Enclosures	Able to protect against Immersion up to 1 m	Ingress of water in harmful quantity shall not be possible when the enclosure is immersed in water under defined conditions of pressure and time (up to 1 m of submersion).	Test duration: 30 minutes
			Immersion at depth of at least 1 m measured at bottom of device, and at least 15 cm measured at top of device

Figure 5.1.1 Budind.com 'Mysteries of IP65, IP66, IP67 Rated Enclosures'

Since electronic devices uses energy, it must then generate heat. With that in mind, enclosures built for BarkaRoo collar must allow for active or passive cooling system. Exposure to heat from outside source is inevitable as we like to take our pets usually on a walk or just to spend time outside the normal house environment. Since batteries are sensitive to extreme heat and extreme cold temperatures insulation is beneficial in diminishing the degrading effects of temperature to batteries. Transitioning from extreme cold to extreme heat environment poses a great introduction for condensation to happen within the enclosure once again threatening a possible electrical damage within the system.

5.1.2 Wi-FI Technology

IEEE(Institute of Electrical and Electronics Engineers) set the standards for Wi-Fi technology. As such, IEEE 802.11 is the current standard in Wi-Fi. However, IEEE 802.11 also contains letter after 802.11 base upon the speed and frequency in which this wireless technology operates on. Standards that are produce by the IEEE specifies the ways of implementing wireless local area network computer network and communication in the 2.4GHz, 3.6Ghz, 5Ghz and other frequency bands that will operate in the future. Currently, Wi-Fi alliance took the steps to make Wi-Fi standards names easier to identify and understand(Wi-Fi 4, Wi-Fi 5, Wi-Fi 6, etc.).[1]

5.1.3 Bluetooth Technology

IEEE 802.15.1 is the governing body who manages such Bluetooth standard but recently no longer actually maintains that standard. Rather, “Bluetooth Special Interest Group oversees development of the specification, manages the qualification program and protects the trademarks.” -Wikipedia.com The oversight of Bluetooth technologies existence allows for private companies to research and develop ways to improve Bluetooth communications.

5.1.4 LoRaWAN

The emerging technology of Low Power, Wide Area networking protocol is designed for IoT technology. Recently, Semtech acquired and patented such wireless communication technique forming LoRa Alliance ensuring interoperability of every single LoRaWAN products and technologies. LoRaWan specification networking protocol is design to wirelessly connect battery operated ‘things’ to the internet in regional, national or global networks.(LoRa Alliance)

5.1.5 Microcontroller Data Communications

Microcontroller developments most important evaluation is which appropriate hardware interface will complement the overall system design. Regarding this hardware interface, three common communication protocols exist and is widely used today, namely, UART(Universal Asynchronouse Reception and Transmission), SPI(Serial Peripheral Interface), and I2C(Inter-Integrated-Circuit). Below is the overview of ESP32 hardware interface.

A Universal Asynchronous Receiver/Transmitter (UART) is a hardware feature that handles communication (i.e., timing requirements and data framing) using widely-adopted asynchronous serial communication interfaces, such as RS232, RS422, RS485. A UART provides a widely adopted and cheap method to realize full-duplex or half-duplex data exchange among different devices. The ESP32 chip has three UART controllers (UART0, UART1, and UART2) that feature an identical set of registers for ease of programming and flexibility. Each UART controller is independently configurable with parameters such as baud rate, data bit length, bit ordering, number of stop bits, parity bit etc. All the controllers are compatible with UART-enabled devices from various manufacturers and can also support Infrared Data Association protocols (IrDA). (Espressif)

I2C is a serial, synchronous, half-duplex communication protocol that allows co-existence of multiple masters and slaves on the same bus. The I2C bus consists of two lines: serial data line (SDA) and serial clock (SCL). Both lines require pull-up resistors. With such advantages as simplicity and low manufacturing cost, I2C is mostly used for communication of low-speed peripheral devices over short distances (within one foot). ESP32 has two I2C controllers (also referred to as ports) which are responsible for handling communications on the I2C bus. Each I2C controller can operate as master or slave. As an example, one controller can act as a master and the other as a slave at the same time. (Espressif)

The ESP32 has four SPI peripheral devices, called SPI0, SPI1, HSPI and VSPI. SPI0 is entirely dedicated to the flash cache the ESP32 uses to map the SPI flash device it is connected to into memory. SPI1 is connected to the same hardware lines as SPI0 and is used to write to the flash chip. HSPI and VSPI are free to use. SPI1, HSPI and VSPI all have three chip select lines, allowing them to drive up to three SPI devices each as a master. (Espressif)

5.1.6 GPS(Global Positioning System)

The USSF is responsible for the design, development, procurement, operation, sustainment, and modernization of the system.(GPS.gov) GPS receiver manufacturers design GPS receivers to use spectrum beyond the GPS-allocated band. In some cases, GPS receivers are designed to use up to 400 MHz of spectrum in either direction of the L1 frequency of 1575.42 MHz, because mobile satellite services in those regions are broadcasting from space to ground, and at power levels commensurate with mobile satellite services.[185] As regulated under the FCC's Part 15 rules, GPS receivers are not warranted protection from signals outside GPS-allocated spectrum.[177] This is why GPS operates next to the Mobile Satellite Service band, and also why the Mobile Satellite Service band operates next to GPS. The symbiotic relationship of spectrum allocation ensures that users of both bands are able to operate cooperatively and freely.(Wikipedia)

5.1.7 Temperature Sensor

Negative Temperature Coefficient(NTC) thermistors suitable for temperature sensing between -55°C and 200°C. NTC standard manufacturing details usually includes:

- Small form factor
- Faster response
- Shock resistance
- Vibration resistance
- Cost effective
- High sensitivity
- ±1°C accuracy between 0°C to 100°C
- ±5°C accuracy on average(over complete range of the NTC)

[1] eepower.com — NTC thermosistor

6.0 Firmware Programming

In this section, we will be discussing the firmware research for our device. This section contains the application and potential tools to be used while developing our firmware. The firmware is the communication link between our hardware and our software to ensure a cohesive final device is produced. This section includes our in-depth analysis on what works best for us.

As the hardware and software team finalize the compilation of components for prototyping, we decided to solely focus on researching and learning the steps for firmware programming on **ESP32 WROVER**. Firmware development for our microcontroller unit is also one of the most sought topics we as a team revisited few times during the research period for microcontrollers that best fit our needs. Luckily for us we are able to narrow down choices based on development environments, plugin features, documentations and open community support.

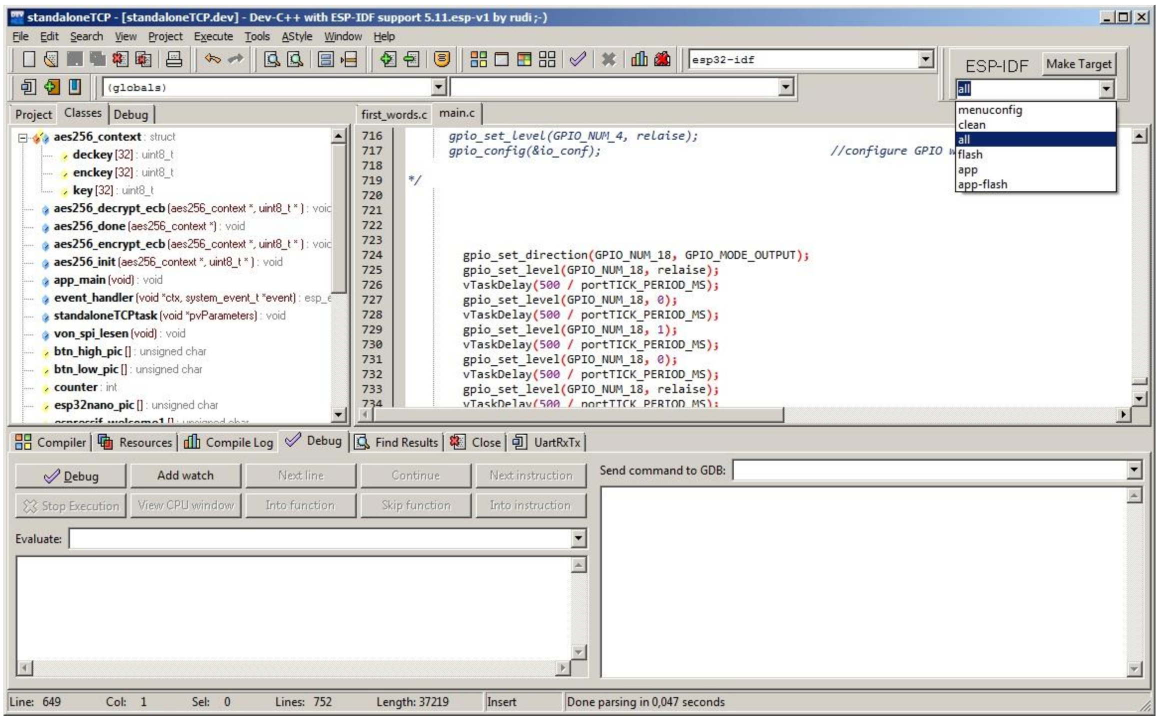
6.1 ESP-IDF

ESP-IDF is Espressif's official IoT Development Framework for the ESP32, ESP32-S and ESP32-C series of SoCs. It provides a self-sufficient SDK for any generic application development on these platforms, using programming languages such as C and C++. ESP-IDF currently powers millions of devices in the field and enables building a variety of network-connected products, ranging from simple light bulbs and toys to big appliances and industrial devices. (Espressif) This production-ready IoT development framework is open-source and freely available on GitHub with extensive documentations for its hardware peripherals and features. Software component examples gives developers an insight of what ESP-IDF has to offer for specific application programming. The GitHub branch of ESP-IDF provides ease of access to jumpstart our team project. Eclipse and VSCode official support gives us a peace of mind that we'll be able to work on an environment that we have some familiarity with. ESP-IDF will provide tools and utilities required for developer and production use cases, like build, flash, debug and measure.

Below are some of the development frameworks our hardware and software team are looking to work with:

- **ESP-IoT-Solutions:** This Internet of Things framework contains libraries of peripheral drivers and usable code frameworks an edge to fire up our rapid development and testing. Support for third-party devices containing drivers for various sensors, screens, audio devices, input devices, actuators, etc.

- **ESP-Jumpstart:** Another GitHub branch provided by Espressif containing an already fully functional build called “Smart Power Outlet”. Convenient firmware development providing workflow from ESP-IDF toolchain and code for the project, building the application, generate executable firmware image and firmware execution on the development board.



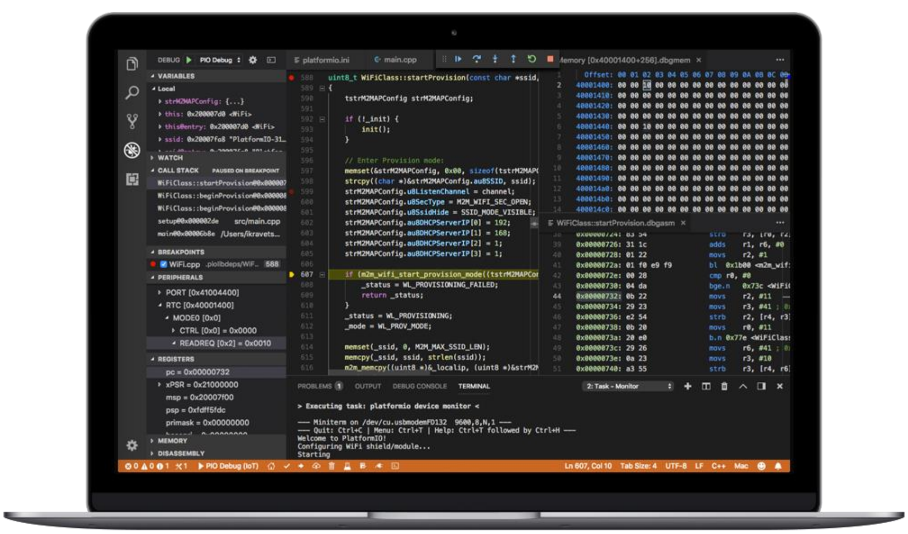
Courtesy of https://github.com/ESP32DE/devC_esp-idf

6.2 PlatformIO IDE

PlatformIO integrated development environment for IoT becomes an runner up building the firmware for the ESP32. Has capability to be fully integrated to Visual Studio Code that allows for full firmware development on ESP32-DevKits. It has its own award-winning build system known as PlatformIO Core (CLI) that ease the development process on software integration, packaging and library dependencies.

Some of its feature highlights:

- **Integrated Debugger:** This allows for quick and extensive debugging techniques for buggy and non-optimized code. Watch Windows allows for viewing of symbols, registers and expressions while code is executed.
- **Unit Testing:** This will allow us to create environments where firmware may be exposed to probability of failures. These failed tests allow our team to quickly identify and target specific errors.
- **Static Code Analysis:** Code analysis might be the most important case during a software development cycle as it decreases the amount of error the code is bound to. This code analysis built within PlatformIO ensures robustness and high industry standards on deployable code.
- **Remote Development:** Allows for remote engagement on development in various environment when device is connected to a remote machine. Allows for test run without extra software, SSH, VPN or opening network ports.(Platformio.org)

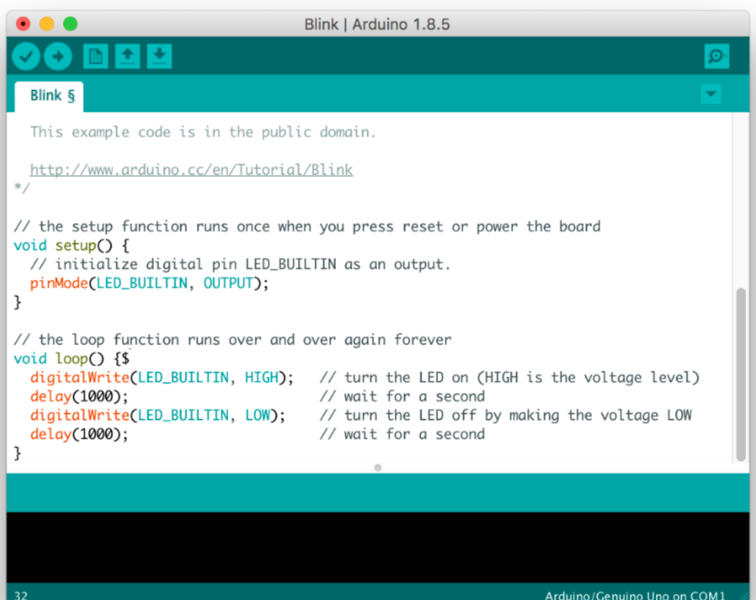


Courtesy of https://platformio.org/platformio-ide

6.3 Arduino IDE

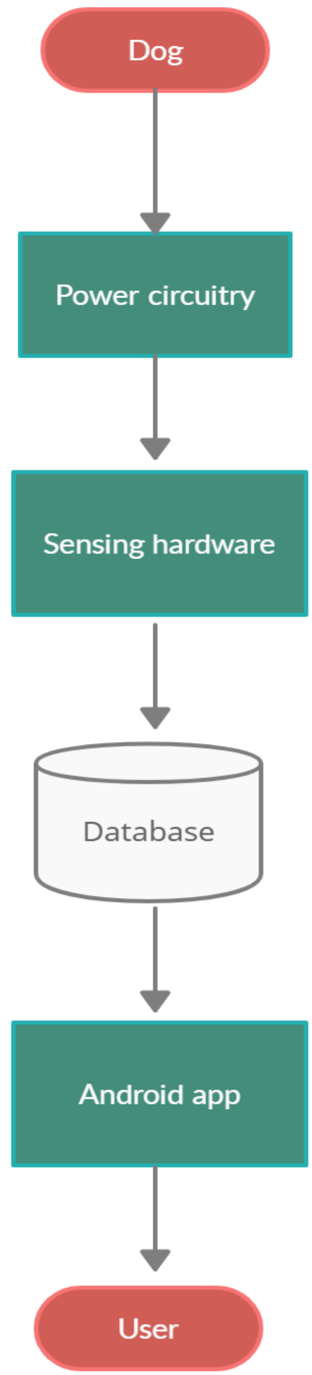
Arguably the most popular DIY and IoT applications Arduino IDE has become a global leader for simple and advance development platform for all ages. The hardware and software concluded that familiarity of Arduino IDE and it's line up of microcontrollers will give our team a great advantage for quickly testing our hardware components. Two of our team members currently have a development kit in hand that is ready for testing as our hardware becomes available at hand. Arduino One specifically that is within our disposal is the most document board from the Arduino family. Arduino IDE support for third-party hardware allows for easy firmware programming on its platform. A serial monitor available within the IDE allows for complete serial communication monitoring and command execution such as AT commands. Some of its highlighted features are:

- Board module options
- Board schematic sketching
- Extensive Documentation
- Integrated libraries(without third party installation)
- External hardware support

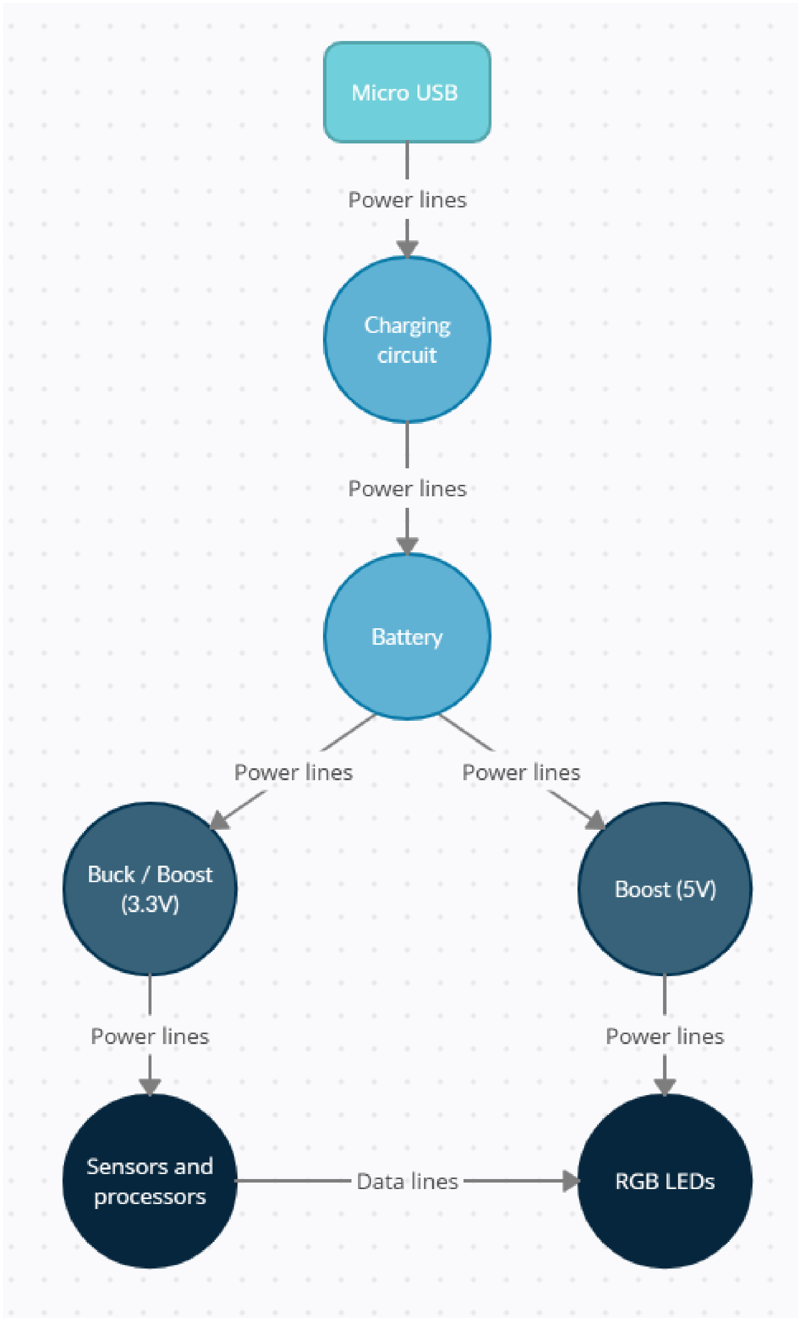


Courtesy of https://en.wikipedia.org/wiki/Arduino_IDE

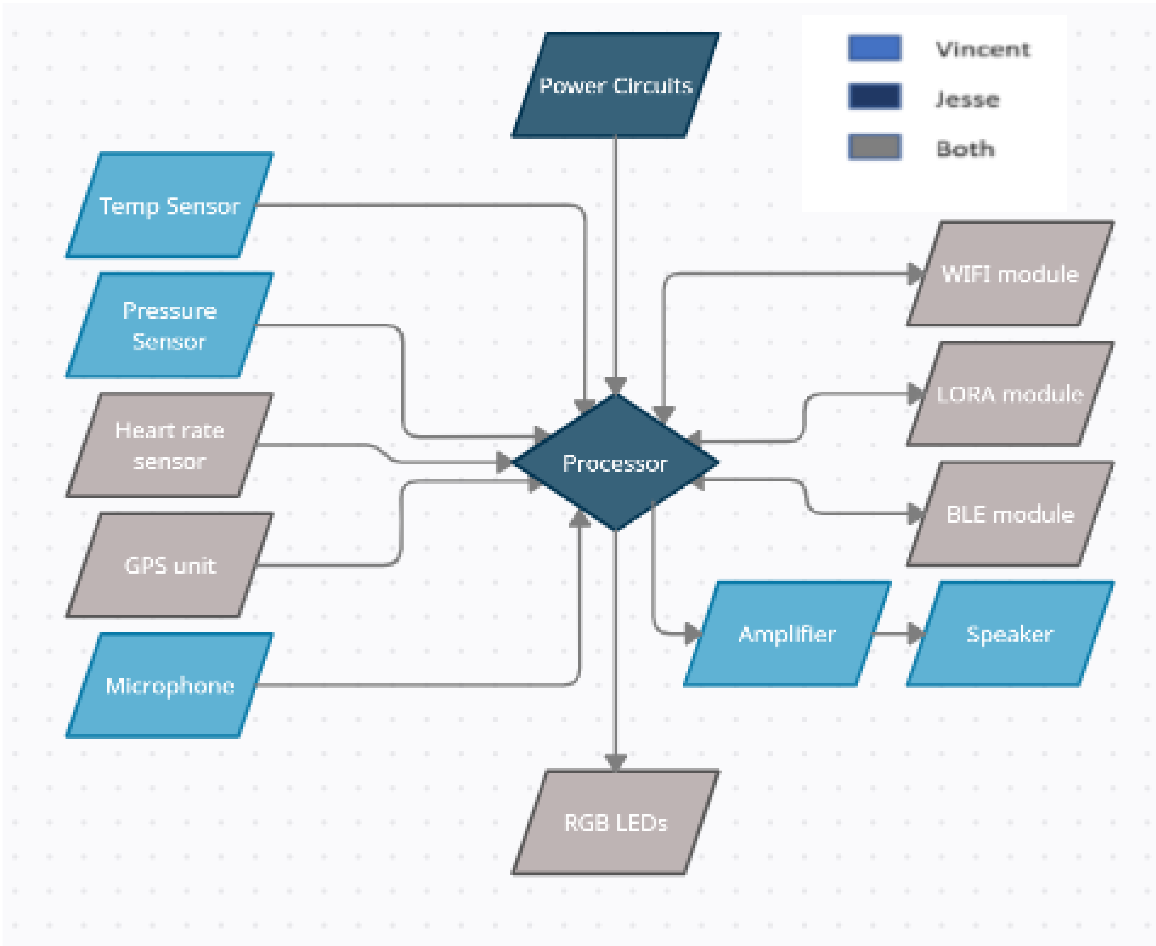
Operational Diagram



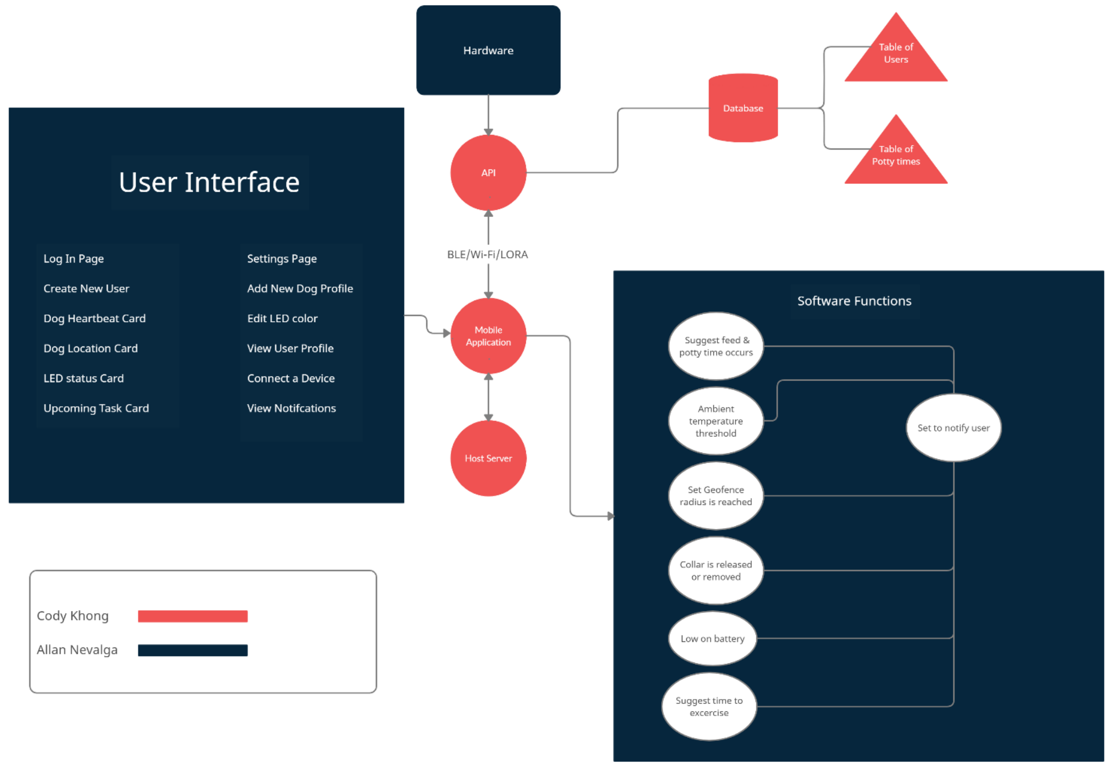
Power Block Diagram



Sensors and Processor Block Diagram



Software Block Diagram



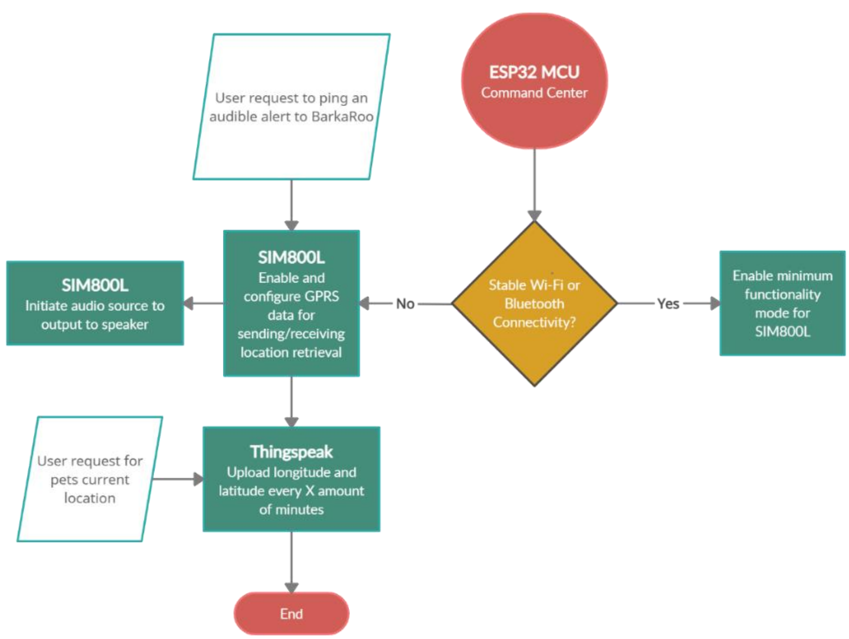
7.0 Software & Firmware Component Design

In this section we'll look over BarkaRoo's main sensory components and wireless communication. This will give a preview of the subsystems functionality and available features that will integrate to our ESP32 Microcontroller. Specifically, this section highlights each components utilization for BarkaRoo's most prominent needs to better serve and care for our dogs and help provide effortless tools for end users needs.

7.1 GSM/GPRS System

The heart of our wireless communication lies on the power of SIM800L, a chip from SimCom capable of connecting to GSM networks supporting the communication basics of a cellular device. Our team of engineers decided to work with SIM800L because of its ability to send and receive GPRS data with quad-band support that works around the globe. Because of BarkaRoo's application we will utilize it's ability to obtain geo-coordinate locations that can output the approximate latitude and longitude of the object. Since AT commands are fairly simple to use and most devices are well documented with these commands, SIM800L allows for using these AT commands to collect location data, send/receive messages or calls, and has external support to create audible alerts.

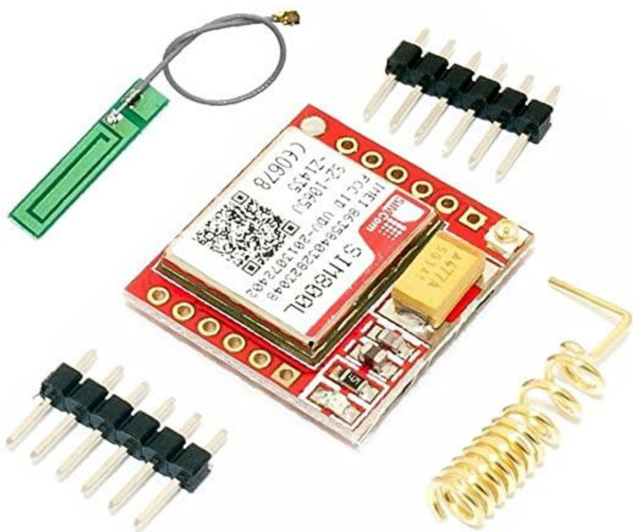
A tentative view of BarkaRoo's GSM/GPRS functional flow:



Research and data indicate that GSM usually dominates energy-consumption in mobile phones and with this information it is safe to say that this will apply to BarkaRoo's power management. SIM800L allows for 'minimum functionality mode' using an AT command disabling RF and SIM card functions. This is huge since if our device is well within stable Wi-Fi or Bluetooth connection, we can disable SIM800L. If on the other hand GSM/GPRS data is needed, SIM800L 'sleep mode' function can be enabled using an AT command disabling serial port and will trigger a wake up routine when external interrupt or pull down on DTR(Data Terminal Ready) pin.

Since location tracking is very prevalent to BarkaRoo's feature, we generated an idea of implementing an audible ping when requested by the user for tracking purposes. Dense environment can pose challenges on specific locations of the device especially in a constant motion and this audible ping can be the solution we are looking for. SIM800L available speaker output pin connection allow for this feature expansion be readily implemented.

Geofence application is also one of the interests our software development team to implement alongside with our GSM/GPRS module. Customizable per client geofencing needs allow for automatic alert tools to locate and respond when a dog reaching beyond the specified imaginary geofences.



Courtesy of https://www.amazon.com/HiLetgo-Smallest-Breakout-Quad-band-3-7-4-2V/dp/B01DLIJM2E

Changed over to the AI Thinker A9G because the SIM 800L protocol for location acquisition is no longer working in the USA.

7.2 Ambient Temperature Sensor

The temperature sensor NXRT15XV103FA1B040 have an operating temperature of -40°C to 125°C. It is a Negative Temperature Coefficient thermistors that uses the Steinhart-Hart equation to measure temperature reading.

```
1/T = 1/To + 1/B * ln(Rt/Ro), so
T = 1 / (1/To + 1/B * ln(Rt/Ro))
```

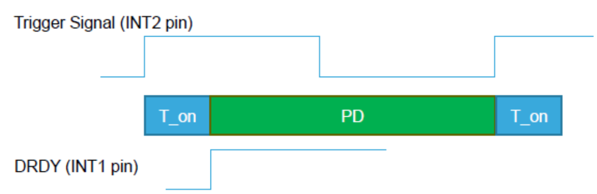
Where:
T is the temperature to be measured in Kelvin;
T_o is the reference temperature in Kelvin for 25 degree Celsius;
R_o is the thermistor resistance at To;
B is Beta or B parameter, provided by manufacturer in their specification.

Implementation of the thermistor won't be particularly difficult as ESP32 GPIO pins can be configured as ADC (Analog to Digital Converter). Luckily for us we know that ESP32-IDF has built in functions for ADC applications. Our team has also noted the non-linearity of ESP32 ADC where there are instances that an inaccurate temperature reading above certain temperature will occur. ESP32 high CPU clock and small factor is also receptive to noise as we collect data from the thermistor for analysis and this can usually be smoothen out by filtering algorithm or adding a small capacitor. (e-tinkers.com)

One of the safety features BarkaRoo intends to showcase is to alert users of low or high quick changes of temperature in a given period of time we specify as possibly endangering to our furry friends. Paw damage from exposure to incredibly high surface temperature gave us an idea to implement an alert to the owner of possibly injuring their dogs. This will be achieved by determining the current location of the tracker and collecting data from a trusted weather reporting services online following a predetermined calculations of normal pavement or asphalt surface temperature.

7.3 Accelerometer Sensor

Our team proceeded to integrate LIS2DW12 accelerometer because of its availability and also widely available supporting documents to fire up implementation with the ESP32. This accelerometer is a 3-Axis accelerometer able to sense tilt angle and finding how fast and direction the device is moving. This sensor from STMicroelectronics uses I^2C or SPI, I^2C requiring a two-wire interface before accessing the sensors while one SPI interface however is needed to properly access the sensors. The module containing also single data conversion on demand meaning this will give us huge advantage in processing data because MCU ultimately get to decide when the sensor should measure it. The following data can be measured and output by the sensor unit:



- Motion and acceleration detection
    - Free-fall
    - Wakeup
    - Configurable single/double tap recognition
    - Activity/inactivity
    - Stationary/motion detection
    - Portrait/landscape detection

Digital output interface provides high speed transmission on demand and can enter low power mode with given the right parameters. LISDW12 high configurability allows to meet the demands for BarkaRoo's needs. This device has also an embedded temperature sensor that is stored as two's complement data and left justified.

As part of the intended software application for our accelerometer, configuration of the X and Y axis will serve to produce a direction for when a longitude and latitude data is generated from the GPRS module. It's intended to alleviate the need for guessing the direction our dog is moving. Since LIS2DW12 can be configured for either stationary or motion detection, this useful measurement of activity will be implemented to enable power saving features of our device.

8.0 Software Research

This section discusses the software research for our dog collar. In this section, various software decisions will be discussed to explain the reasoning behind our choices. Multiple applications, programs, and platforms will be researched and compared to put together the best possible workflow for our software team. Aspects such as mobile application, cloud computing, and database platforms are explored to give a better understanding of the topics. Various solutions will be compared, and an analysis will be given for each choice. With many tools at our disposal, proper research is required to ensure that the proper tools are chosen to integrate into a complex project.

8.1 Cloud Computing

Cloud computing technology will be an important factor with embedded and software design of the project. Utilization of this computing technology will allow information to be exchanged with the microchip and the internet. Speaking of which, Internet of Things(IoT) platform enables a suite of features for client interactive experience, data analytics and on the go access of information of BarkaRoo. This open-source platform also allows an unimaginable amount of support for various sensors and in our case the ambient temperature sensor, GPS tracking sensors, Geofence technology and accelerometer sensors. It is a powerful tool that allow localization for data analysis and over the network. In return, BarkaRoo can be diagnosed, altered or receive updates in real-time without physically having the device.

BarkaRoo will take advantage of everything Internet of Things has to offer to ease the process of software development and end-user interfaces. Many cloud platform services today are interoperable with any Arduino, Raspberry Pi, ARM, x86, etc. microcontroller units. Not only interoperability is at its best with these platforms, it also excels in cross software platform applications (i.e. Android, iOS, Web). In making these interactions with embedded systems, developer, and end user possible two web services are going to be used. First, PubNub.com is the solution for gathering data interpretations and pushing notifications. Interpreted data can then be modified to meet certain specifications such as ambient temperature reaching dangerous numbers, collar's location is out of boundary of the specified geofencing and battery status just to name a few. And the other will be Grafana, which will serve as a real-time data visualization produced by the microcontroller components. The overall objective for this cloud platform service is to provide insights of functioning components, send alerts and give users visual indicators.

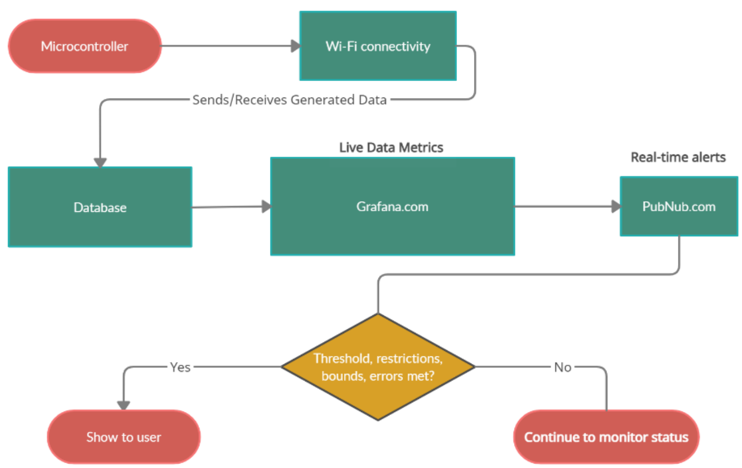


Figure 1.0 Cloud Platform

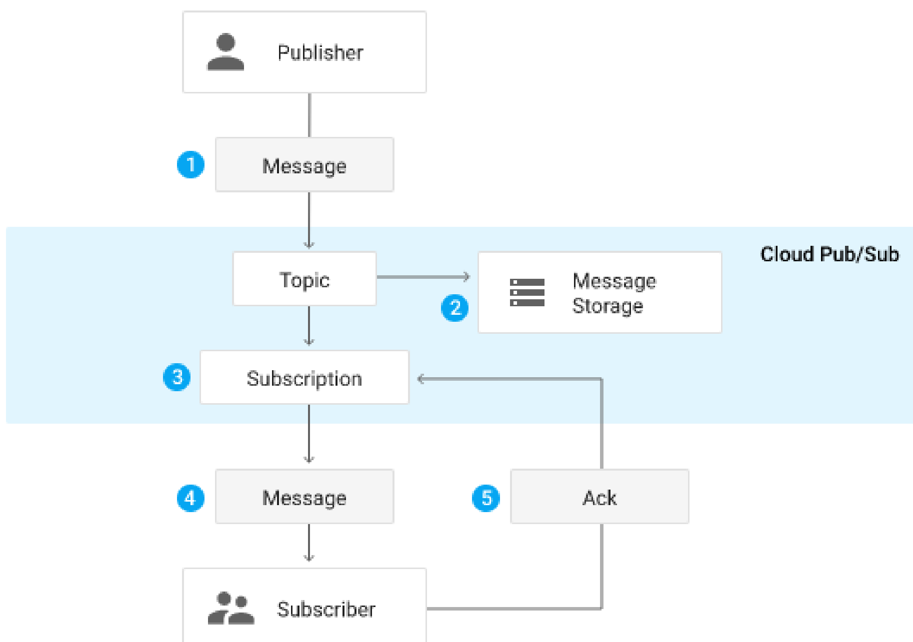
8.1.1 Grafana

Grafana.com is a cloud platform service built to support Internet of Things technologies. The sensors incorporated in BarkaRoo like ambient temperature sensors, accelerometer, GPS tracking, battery life, etc. that generates data will be displayed on a dashboard for analysis in Grafana.com. These customizable dashboards allow for real-time data engagement given that the BarkaRoo collar is within reliable internet connection. Specifically, a GPS map can be created through Grafana to display real-time position of the collar. Ambient temperature will be displayed with an option of Fahrenheit or Celsius also in real-time including a temperature threshold indicating a possible dangerous temperature where the collar is located. Accelerometer data can be plotted to determine its speed and position. This interactive visualization of data allows developers to rapidly test and verify sensors capabilities as well as accuracy.

Courtesy of https://grafana.com/grafana/dashboards

8.1.2 PubNub

In the age of modern technology, we are always plugged in with our smartphone devices. PubNub.com cloud platform will perform real-time notifications through the actual mobile application. Moreover, PubNub can be setup to detect online status of the application in order to create a fallback by pushing a direct SMS notifications or email when this happens. It uses a Publish and Subscribe model in which a publisher(device) sends a message to a particular topic (i.e. dangerous temperature, low battery, unknown location, etc.) located in the cloud(PubNub) finally forwarding this message to the subscribers(users). An acknowledge is sent to the cloud platform in order to verify that a notification or message has been received by the subscriber. The software team particularly love this cloud platform in terms of its library of client SDKs supporting reliable connectivity with various protocols and infrastructure we will throw at it. PubNub excellent documentation with variety of SDKs help ease our software teams mind on how to successfully make it applicable to BarkaRoo's fully functioning components.

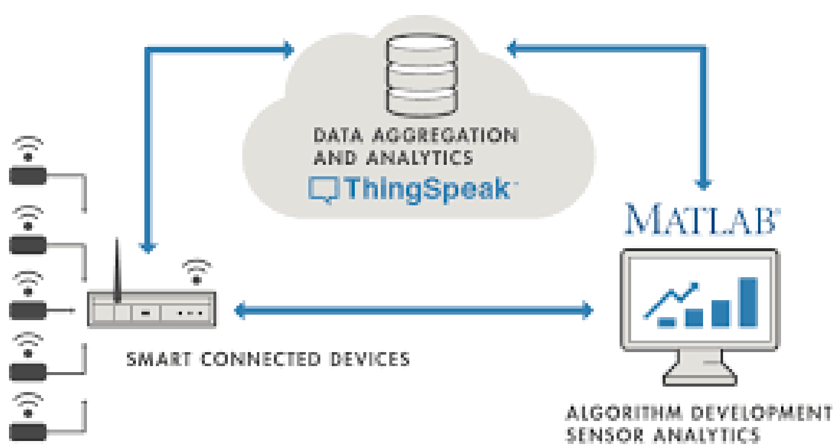


8.1.3 ThingSpeak

ThingSpeak IoT platform is going to be our suitable choice for the BarkaRoo purpose and implementations. Having a free limited option to sending no more than 3 million messages each year earning this platform the highest rank in that category. Same as PubNub.com, ThingSpeak uses channels to send and retrieve data from the microcontroller unit. It has a social media plugins compatible with Twitter for automatic “tweet” notifications of triggered events as well as Twilio APIs and support most if not all mobile and web applications. Some of its features includes:

- Live data collection and analytics
- MATLAB integration for IoT data computations and visualization
- Automated response for data analyzation
- Open-source API
- Geolocation tracking
- Public channel available for rapid device prototyping and testing

ThingSpeak ability to transform our collected data from ESP32 to a visualize data representation on a cloud platform solidifies our components working status. This visualize data also give our team feedbacks on outputs that could be inaccurate, wrong or simply disabled components for debugging purposes. This rapid prototype testing also gives a quick assurance that our sensors will have no issues in cooperation of our software application.



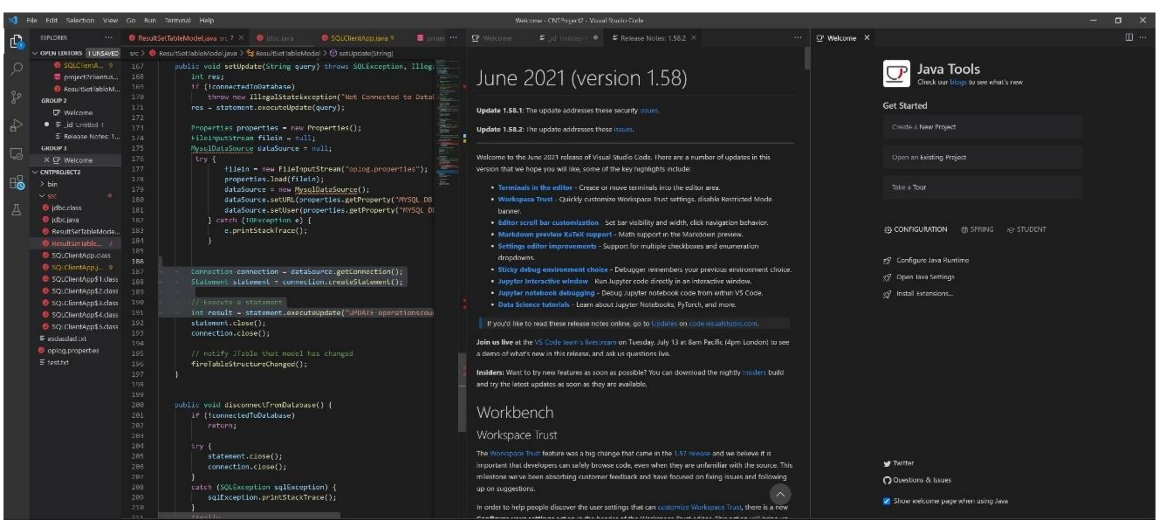
Courtesy of https://thingspeak.com/login?skipSSOCheck=true

In the end, ThingSpeak ended up being our IoT, Cloud Computing platform that was used throughout our project. Their visual layouts of the data being sent made it simple for us to track and test our data points for our sensor readings.

8.2 Mobile Application Development

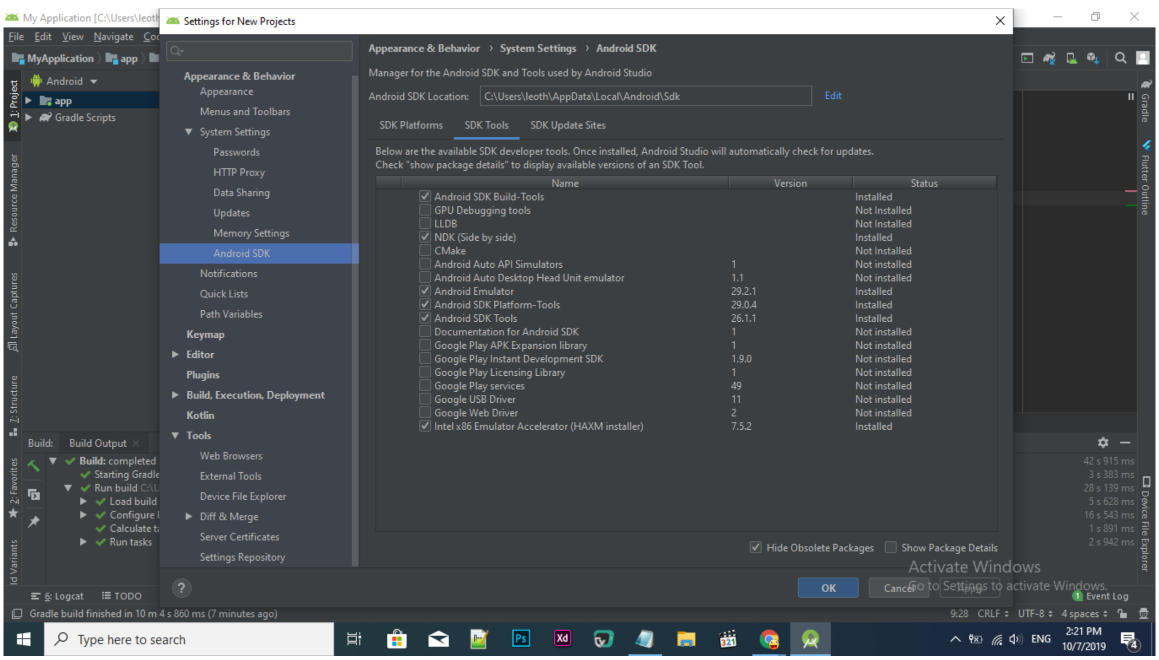
The mobile application development aspect of this project will be a key component in addition to the hardware design. This application will be the main mode of interaction users will have to get the hardware we design to be usable for the features we plan on including. We chose to make a mobile application for this project since the collar we are designing would be best suited for interaction through the phone and on-the-go. As such, a stationary web application that requires users to sit at a personal computer was ruled out. For the platform, the team decided the development on the Android platform would be most beneficial for us throughout the project. In large part, this is due the open-sourced nature of Android. Android allows development, integration, testing, and deployment to be more manageable with the ample documentation online that will help us with any issues that may arise during the process.

After narrowing down our options to an Android, mobile application, we looked at tools we believe would best suit this project. In order to develop an Android application, we decided to use a mix of both Android Studio and Visual Studio Code as our IDE for this project. Android Studio will allow us to see a glimpse of our final application as we code with its built-in Android Emulator. Based on IntelliJ, Android Studio offers a platform that allows us to code with included templates, integrated version control tools using GitHub for team development, Java/C/C++ support, and much more included. Since the Android Studio is supported by Google and has been extensively used over the years for other application development, this established platform allows us to fully utilize the ample documentation and online resources while developing our mobile application. In tandem with Android Studio, we will also be incorporating Visual Studio Code into our workflow since most of our team members have experience with this IDE. With Visual Studio Code, we will be able to keep our options open and ease into the Android Studio environment.



Screenshot from Team Member, Cody Khong

Due to the open-source nature of Android Studio and the well-established support backing this platform, we were easily able to find supporting tools and software development kits (SDK) to boost our app development. The most prominent SDK we were able to find was Flutter. We wanted our mobile application to look well-designed, visually simplistic, and well-equipped to handle the features of our hardware. The SDK we found best-suited for fulfilling our criteria was Flutter. Flutter is an open-source SDK specifically tailored for designing the user interface. Written in C/C++/Dart, this SDK was created by Google and has complete integration with the Android Studio IDE. This integration makes Flutter a near seamless integration into development as we dive deeper into prototyping our mobile application. In order to do so, we need to install both Android Studio (version 3.0 or later) and Flutter. After setting up both to properly work on the computer, Android Studio will have a setting preference for plugin. In this setting, select Flutter and coding can start from there. Like Android Studio, Flutter was chosen due to the massive support online with plenty of documentation from Google and additional resources from other developers using this SDK. The massive support online for both Android Studio and Flutter help make our choices for the tools we are going to use for this project straightforward. With any new topic, there will always be a learning curve. With the tools we have chosen to work, this learning curve will not be as steep and allows us to be more efficient with our time.



Courtesy of https://stackoverflow.com/questions/58269495/new-to-programming-flutter-sdk-is-not-found-in-the-specified-location

The tools we have at our disposal, at this point in time, seems best suited for how we want to proceed with our development process. At this moment (6/24/2021), the mobile application development aspect of our project is constrained mostly by the lack of hardware to work on. As such, we are starting mobile application development with a barebone prototype. This prototype will be with Android Studio with a superficial, UI design using Flutter that will act as a placeholder for the features we plan on including once the hardware is completed. This barebone app will give us a bit more perspective on how we want our finished product to operate with the functionalities we plan on having with the placeholders within in the app. Additionally, we will be able to simultaneously work on the application and the hardware while adding and removing aspects of our project we find achievable or otherwise. From our bare prototype with dummy buttons and tabs, we will integrate working features as they are done to test on the Android application (blinking an LED, receiving a Bluetooth transmission, etc.) one-by-one to test functionality. After ensuring the feature we are testing works in a controlled, isolated environment, we will bring the application with the feature we are testing to a “live” environment to simulate what the end user’s experience will be like without the developers interfering with the code and hardware. From this process, we will see how well this testing methodology work and repeat the process throughout development with each feature filling in the placeholder on the protype until we have a stable, feature-filled, mobile application. In the case we find this testing method inefficient or unreliable, the process will be revisited and revised to better suit the constraints we have and/or will encounter.

8.3 Database

	Pros	Cons	Candidate
MySQL	• Reliable • Vast online community • High data handing • High traffic handling	• Not the fastest • Some performance issue • Might not play with cloud servers	1
SQLite	• Fast • Lightweight • Easy to integrate • Great for testing	• Severely limited multiuser support • Limited table size • Will need replacement for future release	3
MongoDB	• High scalability • Easy setup • Flexible database • Fast	• High memory usage • Limited data size	2
PostgreSQL	• High security • Great error handling	• High learning curve • Slow data transmission	X
RealmDB	• Lightweight	• Limited documentation • Occupies small market space	X

For our device, we had to take in the consideration of having our own database to store user and other supplemental information for the features for our finished product. We plan on having our mobile application being able store and create users so that it can have a personalized profile for each dog and their owner instead of a general “one size fit all” page. In addition, the tentative plan is to have the application be able to notify users when it is time to for different activities for the dogs. For example, we wish to have both our application and collar notify/indicate when it is time to take the dog out for the bathroom. This function will be as personalized as possible using a table in the database that stores the American Kennel Club’s information for each dog’s weight, breed, or size. With this information for the American Kennel Club, we can have the user input their dog’s information to predict the appropriate time for bathroom breaks as well other activities and send a notification and/or indicator. With our current plan for the list of features and functionalities we are planning to have on this device, we concluded that a database was needed for this project.

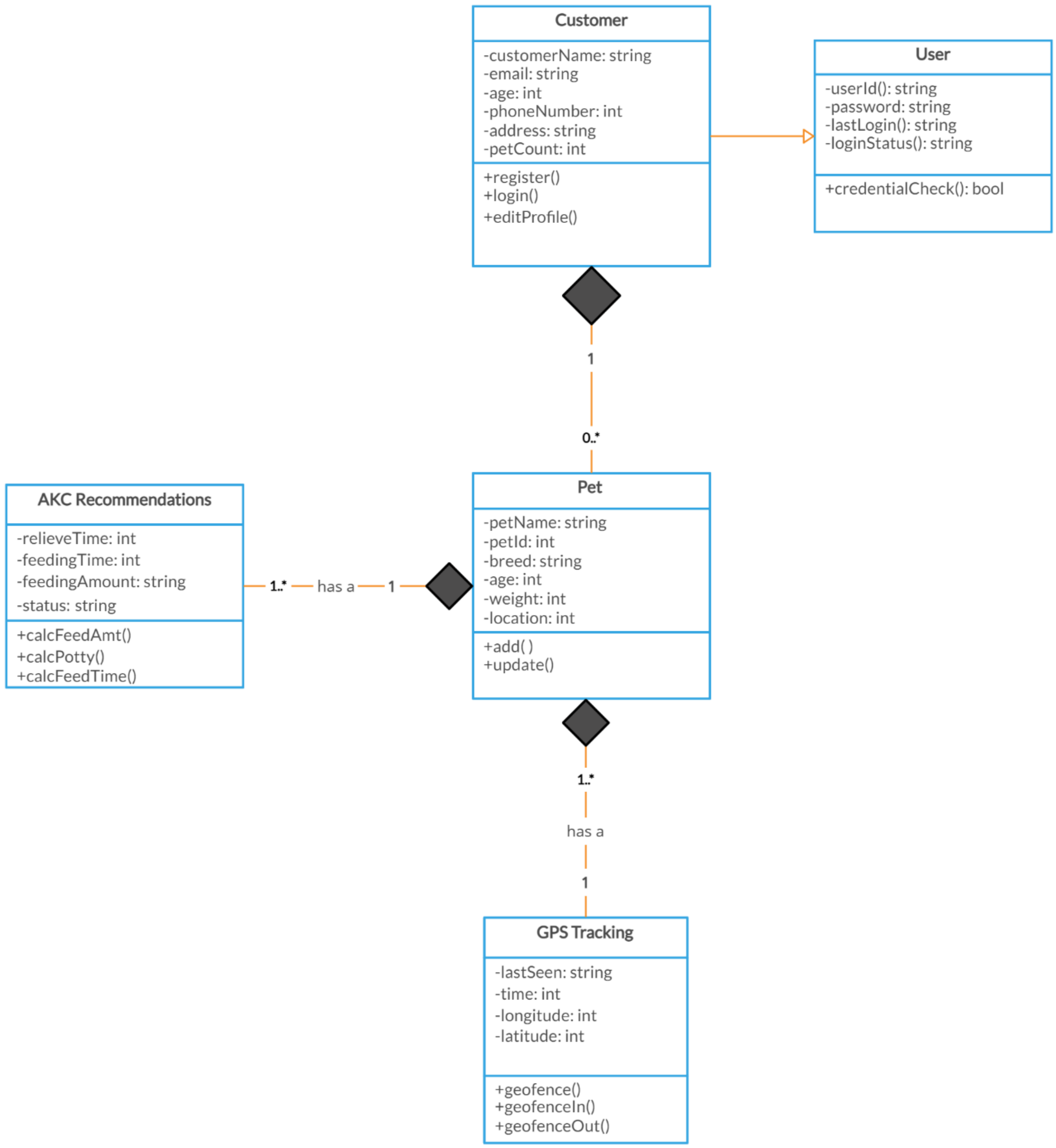
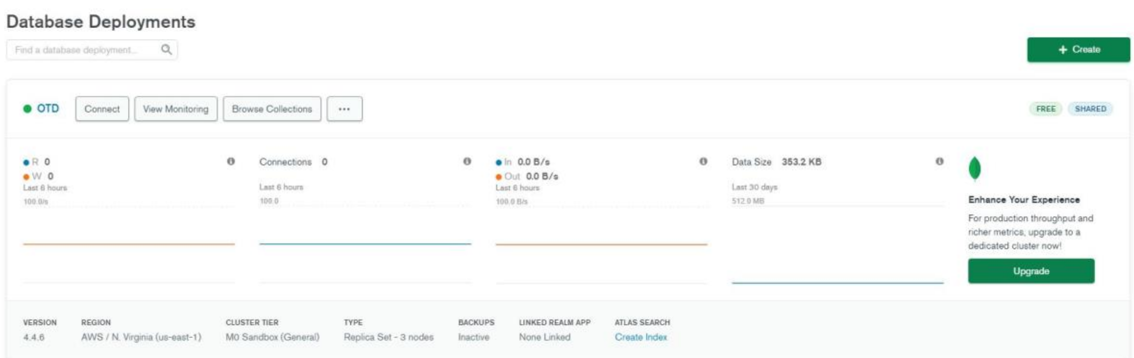


Figure 83 Database UML Class

Once it was decided that a database is necessary for our mobile application, we needed to find and choose the proper database for our project. During a preliminary search for a database to choose, we were able to find a list of databases that could potentially be integrated into our application. The options found were placed into a table to see the benefits and drawbacks of each. The mobile application that will be used in tandem with the hardware collar will be Android-based. Therefore, the database chosen needs to be the fit for an Android application. Another criterion for the proper database we were going to choose is the ease of use and the documentation available. This is especially important due to the fact that time is limited. Narrowing the lists down with our requirements, the best options for this project, for the time being, are MongoDB, MySQL, and SQLite.

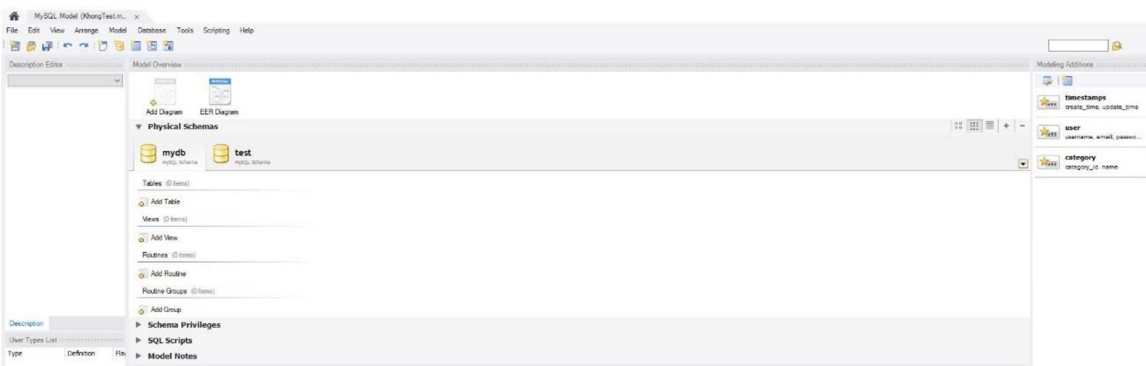
MongoDB is a popular database for the development Android applications due to large market space occupation and ease of use. MongoDB allows applications to have a flexible database for more complex relationships. Additionally, the popularity of this database gives MongoDB ample documentation in the case any issues were to arise. The learning curve is also small with the documentation and the easy environment setup. Another benefit of MongoDB would be the speed at which this database operates. This allows updates and alterations to be fast and responsive. However, these benefits do not come without drawbacks. MongoDB has limitations on both memory usage and data size. Since our project will ideally have numerous users with tailored and large quantities of data, this could possibly lead to a hinderance of to our application's user experience if any bottlenecks were to be exhibited. Nonetheless, the features of this database still makes it highly placed on our list of options.



Screenshot from Team Member, Cody Khong

SQLite is lightweight database that offers a great platform for prototyping. The database is based on the SQL database engine in a smaller, faster engine. SQLite allows us to have a database that is easy to setup and apply for a mobile application. Moreover, since this SQLite is based on SQL, a fully flushed database could be made from another SQL database like MySQL. Since testing this mobile application will be primarily for prototypes, a faster database for a smaller test group could potentially be the most efficient for tests. While SQLite would be potential candidate for this project, there are limitations that should be considered if it were to be integrated. One of drawbacks to using SQLite would be its limitation for the size of data available for transfer and alterations. SQLite is extremely limited by its multi-user capabilities. This could potentially be a problem when the application is fully finished, but a different database could be used when the application is to be published for the public and not just for testing. Furthermore, the data tables on SQLite is vastly limited. Since the application will need a table on the database with a high amount of information from the American Kennel Club, a large table with room for additions will be needed. This could potentially hinder progress if SQLite cannot handle this table. With the high potential for ease of integration and lightweight nature of SQLite, this database would be the ideal candidate for testing, but different database will almost be a must for final release of the mobile, Android application.

From the list of best contenders for the database that will be used for this mobile application, MySQL appears to be the most likely candidate. MySQL is an extremely popular and powerful database that has a vast community and ample documentation. MySQL has a high market space occupation that provides reliability and sustainability. An immediate draw to this database would be its data handling capabilities. With MySQL, larger tables can be made and stored with very few issues that would be best suit our Android application. On top of the data handling, MySQL also handles high traffic optimally. This would allow us to have a final application that allows multiple, concurrent users online at once. In turn, any alterations to the table at any moment would be done so in an efficient manner that does affect the end users' experience on the application. Despite the numerous benefits of MySQL, this database also has its downsides. Since this database can handle a higher amount of data, the speed of this data transmission is a bit slower in comparison to some of the competitors. When its vast support and stability, MySQL is the leading contender for integration into our application despite not being the absolute fastest. This tradeoff is well worth the downsides if this application is to have long-term sustainability.



Screenshot from Team Member, Cody Khong

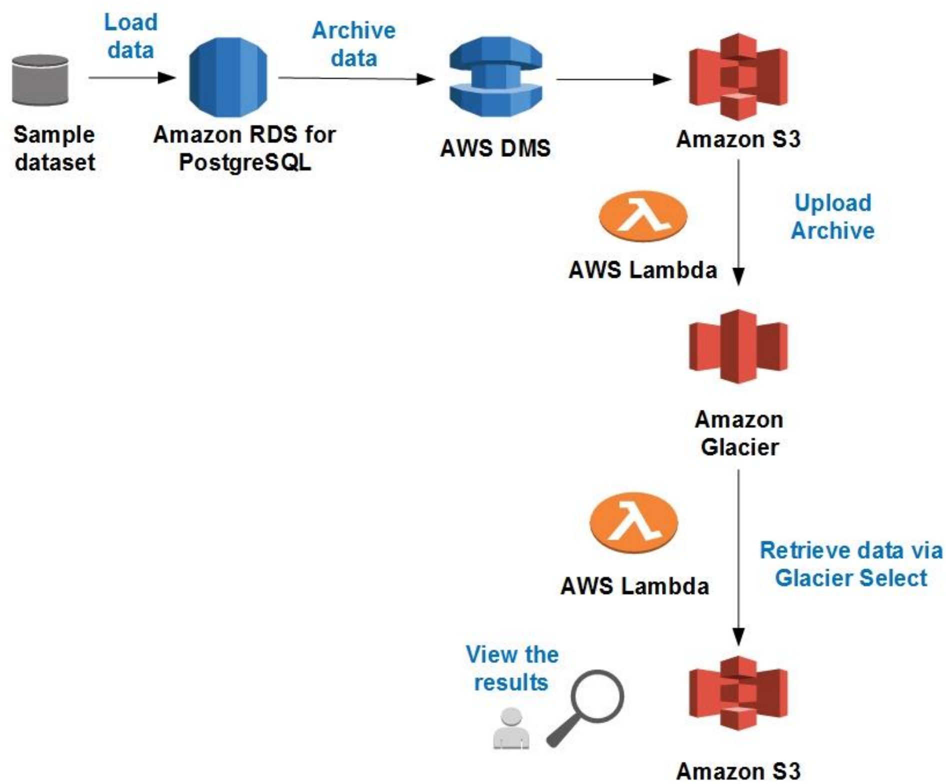
Despite having previous knowledge and making a preemptive choice on database platforms, this project ended up utilizing a platform that was not even previously considered. We decided to use Firebase as our choice for storing information since it came seemingly integrated into our workflow. Firebase is developed by Google like Android Studio. When the time came to integrate a database to store user and pet information, Firebase was the most seamless to use and provided all the necessary tools for our development and data tracking.

8.4 Server

	Pros	Cons
Amazon Glacier	• Free • Reliable	• Dependent on third party • Compatibility issues
Raspberry PI Server	• Easy maintenance • More control • More certainty	• Cost • Potentially time consuming

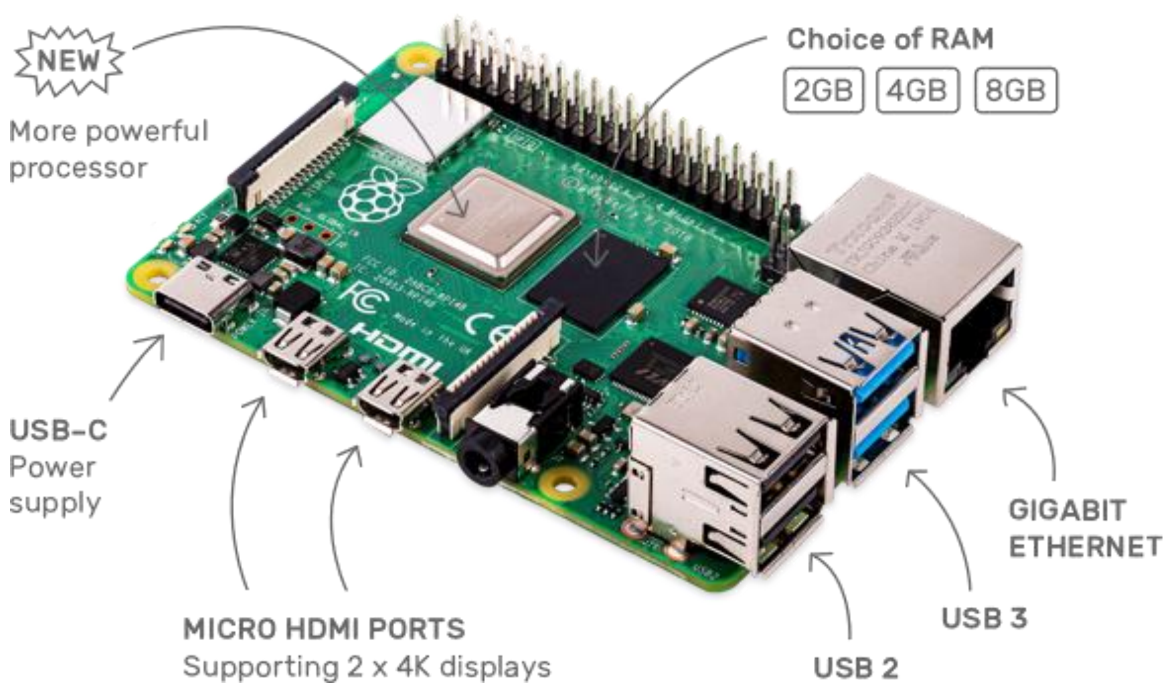
For Android, mobile application, our project will need a server to host when it is online. The collard will not only communicate the pet owner's android device, but the users will also be online and have data transfers when the product is in a real-world setting. Since the development and testing of the application will be at a smaller scale then a final, mass-produced =release of this product, two smaller scale solutions for the server were taken into consideration to see where we would want to store this information. The first option was to have a cloud server setup from Amazon to just test and see how the hardware interacts with our software when in a controlled, online environment. The second option would be to build our own locally stored server using a Raspberry PI.

In terms of cost, the free, Amazon cloud server would seem like the best option to avoid any cost obstacles. This solution would be stable enough that there would be no very little server-side errors. However, we also had to take into consideration how our software would run in a cloud server environment. For example, MySQL would have compatibility issues running on the cloud server if this should be the route we decided to take. Furthermore, running on the cloud server would mean there could be potential issues of being hindered on the server side if Amazon were to perform maintenance or experience a period of being offline. Despite being free, these drawbacks must be taken into consideration while choosing where we are going to host our project. On the other hand, this is still a real option for our project as of right now, so it will not be ruled out.



Courtesy of https://aws.amazon.com/blogs/database/archiving-data-from-relational-databases-to-amazon-glacier-via-aws-dms/

Our second option seems to be potentially the best option for us since we get more control with building our server with the Raspberry PI. With a Raspberry PI server, server-side tuning and monitoring becomes much easier and reliable. Having full access to the server hardware allows us to better isolate the root cause of potential issues that may arise while we test both of our hardware and software in an online environment. This also eliminates any reliance on a third-party server client like the Amazon cloud server. The team members will also be able to control when the server is online or offline. A Raspberry PI server will provide the overall project a layer of reliability and consistency that will overall benefit the project when time is a major factor on getting everything done. However, one of the barriers in this becoming our final option would be the cost. When the option to lower the price of the overall project is available, this makes the decision a bit more difficult when compared to an already reliable cloud server service. In addition, an on-site server could potentially be an issue should there be a need to for any server maintenance. With both the Amazon, cloud server and the Raspberry PI server being strong contenders with their own benefits and drawbacks, we will keep both options open for the potential of integrating both into our project.



Courtesy of https://www.raspberrypi.org/products/raspberry-pi-4-model-b/

For our project, a server was not needed to complete a working, prototype product. However, our previous selection is still be considered to further expand the future of this product. Since this project will eventually be bought online, a server will be needed in the future to support multiple users.

9.0 Initial Project Milestones

This section contains the projected milestones for this device. The following graphs are the projected timeframe it takes to complete our design document and final product assembly. These timeframes are subject to change.

9.0.1 Design Document Project Milestones

9.0.2 Final Product Project Milestones

9.1 Estimated Budget and Financing

This section is our financing and budgeting for our dog collar. The following tables will provide insight to our projected spending in researching and developing our device. Additionally, a table will be provided to see the breakdown of the cost per unit of our devices for release. The information found in this section helps with budgeting the overall cost for the project between four team members. The information found in the cost per device will help with determining project margins for the final product if it ever gets mass-produced.

Table 9.1.2: Development Cost

Item	Quantity	Price	Availability
Microcontroller	1	~$10.00 per unit	TBD
Custom PCB/Layout	5	~$8.00 per unit	readily
Multipurpose LED	1**	~$0.80 per unit	readily
Speakers	1**	~$2.43 per unit	readily
Vinyl Collar	1	~$12.45 per unit	readily
GPS module	1	~$39.50 per unit	TBD
Heartbeat Sensor	1	TBD	TBD
Pressure Sensor	1**	~$6.15 per unit	readily
Temp Sensor	1**	~$1.35 per unit	readily
BLE module	1	TBD*	readily
Wi-Fi module	1	TBD*	readily
Lithium Batteries	2	~$9.99 per unit	readily
Dev Boards	9	~$10.00 per unit	readily
Server	1	~$100.00	TBD
Shipping Cost	10	~$5 per unit	-
Microphone	1	~$10.00 per unit	
Misc. (Wires, Connectors, etc.)	-	~$20.00	-
Misc. Components (Resistor, Caps, etc)	_	~$20.00	-
Total	-	**$442.68**	-

**Quantity needed is subject to change

*Awaiting part approval before estimating price. (Some microcontrollers have bundled Wi-Fi and BLE modules.)

Table 9.1.3: Cost per Production Unit

Item	Quantity	Price	Availability
Microcontroller	1	~$10.00 per unit	TBD
Custom PCB/Layout	1	~$1.60 per unit	readily
Multipurpose LED	1**	~$0.80 per unit	readily
Speakers	1**	~$2.43 per unit	readily
Vinyl Collar	1	~$12.45 per unit	readily
GPS module	1	~$39.50 per unit	TBD
Heartbeat Sensor	1	TBD	TBD
Pressure Sensor	1**	~$6.15 per unit	readily
Temp Sensor	1**	~$1.35 per unit	readily
BLE module	1	TBD*	readily
Wi-Fi module	1	TBD*	readily
Lithium Batteries	2	~$9.99 per unit	readily
Dev Boards	1	~$1.00 per unit	readily
Microphone	1	~$10.00 per unit	
Misc. (Wires, Connectors, etc.)	-	~$20.00	-
Misc. Components (Resistor, Caps, etc)	–	~$20.00	-
Total	-	**$135.28**	-

**Quantity needed is subject to change

*Awaiting part approval before estimating price. (Some microcontrollers have bundled Wi-Fi and BLE modules.)

10.0 Testing

This section will be discussing the tentative testing plan and procedures in place for this dog collar. These plans and procedures are subject to change as various time, money, and other constraints are taken into account and encountered. As of this moment (7/22/2021), the plan is to produce and iterate hardware prototypes first to ensure there is a physical product to be tested. From there, the software team will be working with the hardware prototype to add working software features to the product. The completed hardware and software device will be then tested with various, experimental group to ensure that device is working as advertised and account for any unexpected errors. The timing for each iteration for the hardware and software team will be planned so that a newer, better iteration can be release at the end of the testing cycle until there a final version the team is content with releasing.

10.1 Prototyping

For this section, prototyping will be discussed to get a better understanding about how testing will be implemented. The hardware team will be tasked with completing an initial, completed collar. The collar will be given to the team to implement and test the software functionalities that are planned to be included. From there, the completed collar will be tested as if it was completed product by various test groups. While the software team is adding the functionalities to hardware prototype, the hardware team will be working on the next iteration of the collar for better hardware improvements and awaiting the software team's feedback for any improvements that could be made before testing the initial collar. Although subject to change, the tentative plan for prototyping our device for testing is as follows:

- Hardware Team creates working prototype (1)
- Software Team implements functionalities with the prototype from Hardware Team (1)
- Software Team relays feedback for any potential improvements from implementing features back to Hardware Team (1)
- Hardware Team begins next iteration of the dog collar (2)
- Software Team finishes functionality implantation on initial dog collar prototype
- Testing begins on completed prototype (1)
- Feedback from testing groups given to Full Team for next iteration (2)
- Hardware Team and Software Team takes feedback from Testing Groups to improve prototype (2)
- Repeat until satisfactory

10.1.1 Hardware Testing

This section discusses the plan, as of right now, to build a working hardware prototype. Overall, the plan is to make sure that PCB design will work with all accompanying hardware. The goal for the hardware team will be to get all the sensors, microcontroller, and other parts to be put together and be bought online to be ready to be programmed. During this hardware testing, the completed prototype's reliability and stability will be tested to ensure that device does not malfunction and potentially cause harm to users or wearers. The hardware will be stressed test with different environments and varying electricity throughput. Once the hardware can be definitively proven, to the best of our knowledge, to be not harmful, the hardware team will pass that prototype on to the software team to add the features and functionalities we want to be included on the final device.

From releasing the initial prototype, the hardware team can begin putting together the next iteration of the dog collar prototype and await feedback from the software team and the testing group. The feedback given from the software team and testing group can be used to improve the way the hardware is put together. These changes can be implemented all throughout the testing period. The next iteration will then be released to the software team again for feature implementation. This process can be repeated until resource run out or there will be too minor improvements to make. The following is the tentative plan for hardware testing:

- Put together hardware
- Stress test hardware
- Pass hardware to software team
- Wait for feedback from software team and testing groups
- Begin next iteration of device
- Repeat until satisfactory

10.1.2 Software Testing

This section discusses the tentative plan to implement the software features on the dog collar. The general goal for this plan is to get the functionality we plan on including to fully work on the hardware prototypes being developed. Unlike hardware testing, the software testing could potentially be “one and done” by the end of the first prototype’s life cycle. Software testing could also potentially be iterative process where the software team will focus primarily on getting the barebones function to work with the software to be release. Keeping this in mind, a majority of the software team’s time would be spent on optimization. However, this optimization can be separated from the hardware team’s iterative process. The only caveat would be that the software team will need a fully functional prototype with all the necessary core components to optimize the software. Once this is assured, the software team can work with one or any iteration of the prototype to optimize and test the features.

The optimization of the software will include the responsiveness of the environment, speed at which data can be received or transmitted, and user friendliness for the end users. While coding the software features to be used on the device, the software team will attempt to create a system where the code is modular for every iteration the dog collar. The code will be accommodating for any extra hardware that could potentially be added and/or removed during the testing period. After the core software features are running, improvements to the code can be made separately from the hardware team by the software team. This optimization period will be a period of “polishing” for the software where we get the basic, running software to be better suited for wider use by the general public. Outside of speed, the user experience can be tailored during this time by designing a user-friendly application. While the initial software development could potentially be time-consuming, the optimization and iterative prototype of the software can take place unobtrusively in tandem with the hardware development as long the software team has possession of a fully working prototype to test. The tentative plan for the software testing is as follows:

- Develop initial software system to use with hardware prototype
- Provide feedback to hardware team
- Release fully working, basic prototype for testing
- Retrieve prototype from testing
- Optimize software
- Test optimized software with hardware team’s newest iteration
- Release fully working, improved prototype for testing
- Take feedback, improve software, release for testing, and repeat until satisfactory

10.2 (Not) Animal Testing

This section discusses our preliminary tests before releasing the device to be test by humans. As discussed in the ethical constraints (5.5), live animal testing is completely off the table for this project per the United States government and animal rights regulatory bodies. Each completed iteration of the dog collar will be stress tested to ensure that no harm will come to the user or wearer. However, the testing period holds an amount of uncertainty that leaves room for error. There could potential for the hardware to malfunction and burn or injure the dog or the dog owner while testing. Additionally, the LEDs' brightness level would potentially need adjustment and could blind or stress out the dog during testing. With a myriad of potential negative outcomes, it is best to not test unperfected hardware on live animals.

Despite being unable to test on a living dog, the prototypes of the dog collar still need to be tested but in a more controlled, unharmful manner. As such, a synthetic model of a dog will be made to test the dog collar and all its functionalities. A model can be made from scratch using a Styrofoam cutout of a dog. Another method could potentially be buying a stuff animal the size of a dog and testing out our collar on it. The major benefit of this overall device would be that it is aimed to just be worn by the dog to help adjust the owner's behavior. This means that if the collar can be placed around an object, then the device can still be tested with little hinderance. This testing period will be primarily focused on the fitment of the collar on the dog substitute and if the hardware and software implementations work as planned. There are many substitutes for an actual dog when it comes time test the completed device. Not being able to conduct animal testing will prove to not be a huge concern overall.

10.3 Human Testing

This section discusses the tentative plan for releasing our device for testing outside of the team members. By releasing the device outside of our group, the testing groups will be able to provide useful feedback on errors that might have potentially glossed over during development. The testing for each testing groups will be standardized to see if any deltas occur. Since the device has a wide target audience to be used with, it would be ideal if there were a variety of users with different backgrounds and needs that could test our dog collar.

To ensure the safety of our users, the team will thoroughly test the completed device before releasing the dog collar to the testing groups. Although the device can be tested with a degree of certainty that it will not injure or harm the users, the potential for malfunctions is still present. As such, the various testing groups will be informed of the potential danger in using the product in its preliminary stages. The feedback from each testing group will be taken into consideration while testing the next iteration of the completed device.

This testing period with humans will the area with the most valuable information for improving our final device. The entire purpose of our device is to adjust the behavior of dog owners to improve the quality of life for dogs with this owner on them. Since this dog collar is aimed at adjusting human behavior, the goal of testing with various groups of humans with different background is to see the overall effectiveness of our device. While we aim to please a variety of users with different backgrounds and needs, we can compensate and adjust our device to suit the needs of any minority groups that may arises.

10.3.1 Family Household Testing

As of right now (7/22/2021), the team is still determining the testing group types we are trying to establish. Initially, the idea was to release the product to different household with different family size and types to test the collar. However, this could potentially be superfluous to both the household and our experiment since it was decided that live animal testing will not occur while testing. On the other hand, this might also hinder our ability to test the effectiveness of our features and functionalities on different age groups. Since the main purpose of this dog collar is to make better pet owner habits and improve the quality of life for the dog, there is a potential here of being unable to collect valuable data on the effectiveness of our development. If we were to go down this route of testing using different household types as our testing groups, adjustments will have to be made for the tests conducted. The groups for this testing group would be as follows:

- Small house area
- Large house area
- Big family (size 5+ members)
- Small family (size less than 5 members)
- Family with kids ages 4-9
- Family with kids ages 10-15
- Family with kids ages 16+
- Combinations of the above criteria

10.3.2 Various Background Testing

Should there be any issues with arranging testing with the use of families and their households, there will be another testing group arrangement that could be made. The proposed testing groups would comprise of people with various background knowledge in engineering, varying ages, different petkeeping background, and other criteria that could be added later as we approach a more completed device. By utilizing this type of testing groups, the value of live animal and family household testing is reduced significantly.

These testing groups with a variety background will provide an even more diverse look into the different environments are devices could potentially be in. The test subjects could be potentially picked from our peers, coworkers, and family members. The only area of information collection that would be highly impacted would be the absence of varying younger age groups testing our product. However, this lack of information in this one area can be supplemented by the different older age groups and living conditions provided by the proposed testing groups.

With these testing groups comprised of peers, coworkers, and family members, the team's initial testing plans can still be enacted but with more ease. The risk of destruction of our property is significantly reduced with the removal of younger age groups in our testing trials. The plan for testing our dog collar will remain the same. Each group will be given the lists of potential risk of testing the device in its uncompleted stage. Additionally, test groups will also be asked to test out different features and functionalities and provide feedback for the tests. In order to increase our insight to the benefits and drawbacks of our dog collar, the feedback from each testing groups will also have an area where we can be notified if there were any unpredicted outcomes from the tests outside of the feature and functionalities tested. If there were any unexpected outcomes, then we can these into account to better improve the hardware and ensure there is less unexpected outcomes. The testing groups for the testing trials will be as follows:

- Vastly knowledgeable in technology
- Mildly knowledgeable in technology
- Less knowledgeable in technology
- Lives in dorm
- Lives in apartment
- Lives in house
- Novice pet owner
- Experienced pet owner
- Combination of the above criteria

11.0 Citations and References

- **Past Senior Design Paper**

https://www.ece.ucf.edu/seniordesign/fa2015sp2016/g33/Senior%20Design%202%20Final%20Paper.pdf

- **Bluetooth Low Energy**

https://help.locusmap.eu/topic/is-gps-over-bluetooth-low-energy-le-bt-4-0-supported

- **Similar Products**

https://www.amazon.com/LINK-AKC-Smart-Dog-Collar/dp/B01M5JZ1VO/ref=as_li_ss_tl?ie=UTF8&linkCode=ll1&tag=caninejournal-20&linkId=6945ab19ddcca7c46924e117ddb93d50&language=en_US

https://www.amazon.com/PetPace-Smart-Collar-Large/dp/B01N5YFHBW//ref=as_li_ss_tl?ie=UTF8&linkCode=ll1&tag=caninejournal-20&linkId=610fec344692de7b7091ccf2c35480af

https://www.whistle.com/

https://technobark.com/best-dog-fitness-trackers-this-year/

- **Hardware References**

https://www.mouser.com/ProductDetail/Espressif-Systems/ESP32-C3FN4?qs=iLbezkQI%252BsjnhgSmJXMnMQ%3D%3D

https://www.digikey.com/en/products/detail/espressif-systems/ESP32-S2/11613132

https://www.digikey.com/en/products/detail/espressif-systems/ESP32-S2-WROOM-I/11613141

https://www.digikey.com/en/products/detail/espressif-systems/ESP32-S2-WROVER-I/11613134

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6316215/

https://learn.adafruit.com/whistle-dog-activity-monitor-teardown

https://www.sfcircuits.com/pcb-school/pcb-design-software-comparison-guide

- **Software References**

https://flutter.dev/docs/get-started/editor?tab=vscode

https://blog.panoply.io/mongodb-and-mysql

https://www.mongodb.com/compare/mongodb-mysql#:~:text=Both%20databases%20support%20a%20rich,syntax%20for%20common%20database%20operations.

https://www.freecodecamp.org/news/what-is-flutter-and-why-you-should-learn-it-in-2020/

https://www.tekrevol.com/blogs/database-for-app-development/

https://www.quora.com/Is-it-necessary-to-know-databases-in-order-to-make-an-Android-application-If-yes-why

https://data-flair.training/blogs/advantages-of-mongodb/

https://www.quora.com/Is-MongoDB-good-for-Android-development

https://www.trustradius.com/products/sqlite/reviews?qs=pros-and-cons

https://www.smartfile.com/blog/the-pros-and-cons-of-mysql/

https://www.keycdn.com/support/sqlite-vs-mysql

https://www.appcentricsolutions.com/mongodb-vs-mysql/

https://www.techgalery.com/2019/11/mysql-database-pros-and-cons-why-mysql.html

https://www.aalpha.net/blog/pros-and-cons-of-using-postgresql-for-application-development/

https://stackshare.io/realm

- **Discussion References**

“The Evolution of WiFi Standards: a Look at 802.11a/b/g/n/Ac.” *Actiontec.com*, 19 May 2021, www.actiontec.com/wifihelp/evolution-wi-fi-standards-look-802-11abgnac/.

“What Is LoRaWAN® Specification.” *LoRa Alliance*®, 6 May 2021, lora-alliance.org/about-lorawan/.

“UART ESP32.” *ESP*, Espressif, docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/uart.html.

“I2C Driver ESP32.” *ESP*, Espressif, docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/i2c.html.

“SPI Master Driver ESP32.” *ESP*, Espressif, docs.espressif.com/projects/esp-idf/en/release-v3.0/api-reference/peripherals/spi_master.html#overview.

“Is PLA Recyclable? – The Truth.” *All3DP*, 3 Apr. 2020, all3dp.com/2/is-pla-recyclable/.

“Global Positioning System.” *Wikipedia*, Wikimedia Foundation, 2 July 2021, en.wikipedia.org/wiki/Global_Positioning_System#Regulatory_spectrum_issues_concerning_GPS_receivers.

“ESP-IDF.” *IoT Development Framework I Espressif Systems*, www.espressif.com/en/products/sdks/esp-idf.

“PlatformIO.” *PlatformIO.org*, https://platformio.org/.

Cheung, Henry, et al. “Using a Thermistor with Arduino and Unexpected ESP32 ADC Non-Linearity.” *ETinkers*, 29 Apr. 2020, www.e-tinkers.com/2019/10/using-a-thermistor-with-arduino-and-unexpected-esp32-adc-non-linearity/.