Mitigating Autistic Meltdowns

Joshua Kao, Jason Kroslowitz, Noah Lockhart, Andres Perez

Concept of Operations

REVISION – 2

29 April 2022

Concept of Operations

for

Mitigating Autism Meltdowns

Team 12

Approved by:

__Team 12____________________2/08/22__

Project Leader Date

____

Prof. Kalafatis Date

____

T/A Date

Change Record

Rev.	Date	Originator	Approvals	Description
-1	[2/08/2022]	[Team 12]	[Team 12]	Draft Release
-2	[4/29/2022]	[Team 12]	[Team 12]	Final Release

Table of Contents

List of Figures

1. Executive Summary

An autistic meltdown is an involuntary coping mechanism to excessive stimuli, and can be associated with intense and explosive behaviors. These events pose a threat to the safety of these individuals and in some cases others around them. Rather than having a caretaker supervise them 24/7, we propose a device that mitigates meltdowns for these individuals. The 12th Man is a hip wearable device that mitigates meltdowns for individuals with Autism. By monitoring the user's biometrics, the 12th Man uses artificial intelligence to process these signals and predict if a meltdown is imminent. If a meltdown is imminent, an alert is sent to the subject's parent(s) or guardian through an app with the location of the individual and the probability that a meltdown is going to happen. On the device itself, a recording from a loved one is played with grounding exercises to calm down the individual. We are hoping that through this device, the lifestyles of autistic people can be changed.

2. Introduction

The following is an introduction to The 12th Man, a device capable of monitoring, mitigating, and supporting individuals with autism no matter where they are. The 12th Man will provide a sense of comfort for these individuals and their loved ones through its prediction, mitigations, and location tracking of meltdowns.

2.1. Background

Autism spectrum disorder (or ASD for short) is a disorder that affects social, emotional, and communication skills. Although the cause is still unknown, it is believed to develop from a combination of genetic, nongenetic, and environmental influences. Recently, ASD has been more heavily researched, and as a result, more conditions have been connected to the disorder. Although symptoms differ per individual, some common symptoms include anxiety, ADHD, epilepsy, schizophrenia, and others.

It is also common for individuals with ASD to be easily agitated. Due to this, meltdowns are an ailment that affect many autistic people. A meltdown is a result of stressors that cause an overload for the individual. During a meltdown, “involuntary physical and emotional reaction to a situation” can occur. While meltdowns can happen with any individual, people with ASD are more prone to them due to the inability to process the overload of stimulants from the surrounding environment. Some examples of these extraneous stressors could be any variety of loud noises, large crowds, and sometimes just everyday life. Meltdowns can result in devastating effects such as wandering, self harm, harm to others, and in the worst case scenarios death.

Typically when addressing meltdowns, it is best to try to prevent them from even happening. Family and friends are the best suited for this task since these individuals understand when there could be potential stressors to the person expecting a meltdown. They are also the most equipped to develop grounding exercises which can help calm the person down. Grounding exercises are a series of activities, patterns, or words that help calm an autistic person down. The tell tale signs are different for each person who experiences a meltdown, so it is important to tailor an individual grounding plan for each person. Currently the method of preventing a meltdown is having the parent/guardian around to assist with the grounding exercises. The other current option is to leave the individual unsupervised.

While having family and friend supervision is ideal, we all know that it is not feasible 24/7. This is where the 12th Man comes to action to predict, prevent, and calm-down meltdowns through artificial intelligence and biometrics.

2.2. Overview

The 12th Man device will be a wearable device on the hip that will collect biometric and environmental data such as position, body temperature, heart rate, acceleration, and environmental audio. The devices inside the wearable include a gps module, sensors, and a microphone to collect data. The microcontroller will be supplied power by a rechargeable battery. Data received from the sensors will be filtered and sent to a cloud to be stored and classified. It will then be stored in a database, and will be readily available for the mobile application to fetch and display. If the artificial intelligence infers a meltdown is likely, the mobile application will receive a notification of a possible meltdown. Furthermore, an audio recording of a loved one will be outputted through a speaker on the wearable to aid in the mitigation process. Grounding exercise instructions will be included in the audio to calm the user. Finally, the location of the individual will be tracked in the case of the individual wandering which could lead to dangerous outcomes.

2.3. Referenced Documents and Standards

- G., T., Brandon W. | 8 years old, Warren W. | 17 years old, & Andrew S. | 17 years old. (n.d.). *Autism speaks*. Autism Speaks. Retrieved February 8, 2022, from https://www.autismspeaks.org/
- *How many people are diagnosed with autism in the U.S.* Therapeutic Pathways. (2021, February 27). Retrieved February 8, 2022, from https://www.tpathways.org/faqs/how-many-people-have-autism/#:~:text=How%20many%20people%20have%20autism%3F&text=According%20to%20the%20Centers%20for,spectrum%20disorder%20%E2%80%93%20over%2075%2C000%2C000%20people.

3. Operating Concept

3.1. Scope

The 12th Man mitigation system for autistic meltdowns is a wearable device composed of a microcontroller and sensors. The system will monitor the biometrics of the wearer and collect data from the individual's environment. Using artificial intelligence, we will identify the precursors of a meltdown in individuals with autism, and make a prediction about whether a meltdown is likely to occur. If a meltdown is likely to occur, a recording from a loved one will be played through speakers on the wearable. The wearable will also store the data it collects in a remote database. This database will be accessible by a mobile application, which will provide parents/guardians with information regarding the location and meltdown status of their dependent.

3.2. Operational Description and Constraints

The 12th Man is a noninvasive, wearable device that operates without any work from the individual. It is intended to be used by the ASD community and their loved ones. While wearing the device, biometric and environmental data will be collected continuously. The wearer will only interact with The 12th Man if an event occurs. Speakers will begin to play audio of their loved ones walking them through a grounding exercise. While using the app, parents/guardians will be able to receive continuous biometric data, location data, and notifications of an imminent meltdown. In order for The 12th Man to perform properly, all the following criteria must be met.

1. The user is wearing the charged 12th Man device.
2. The user is wearing the heart rate monitor around their chest.
3. Parents/Guardians will have to have The 12th Man app installed on their phone and linked to the device.
4. The device wearer must have cellular or internet connection.
5. Past 5 hours the device will lose power causing the user to be unmonitored.
6. The user must have the speaker module in their pocket.

Additionally The 12th Man is not waterproof or perfect. There is a risk of a damaged sensor sending the wrong biometric data to the app which could potentially cause a fake alert.

3.3. System Description

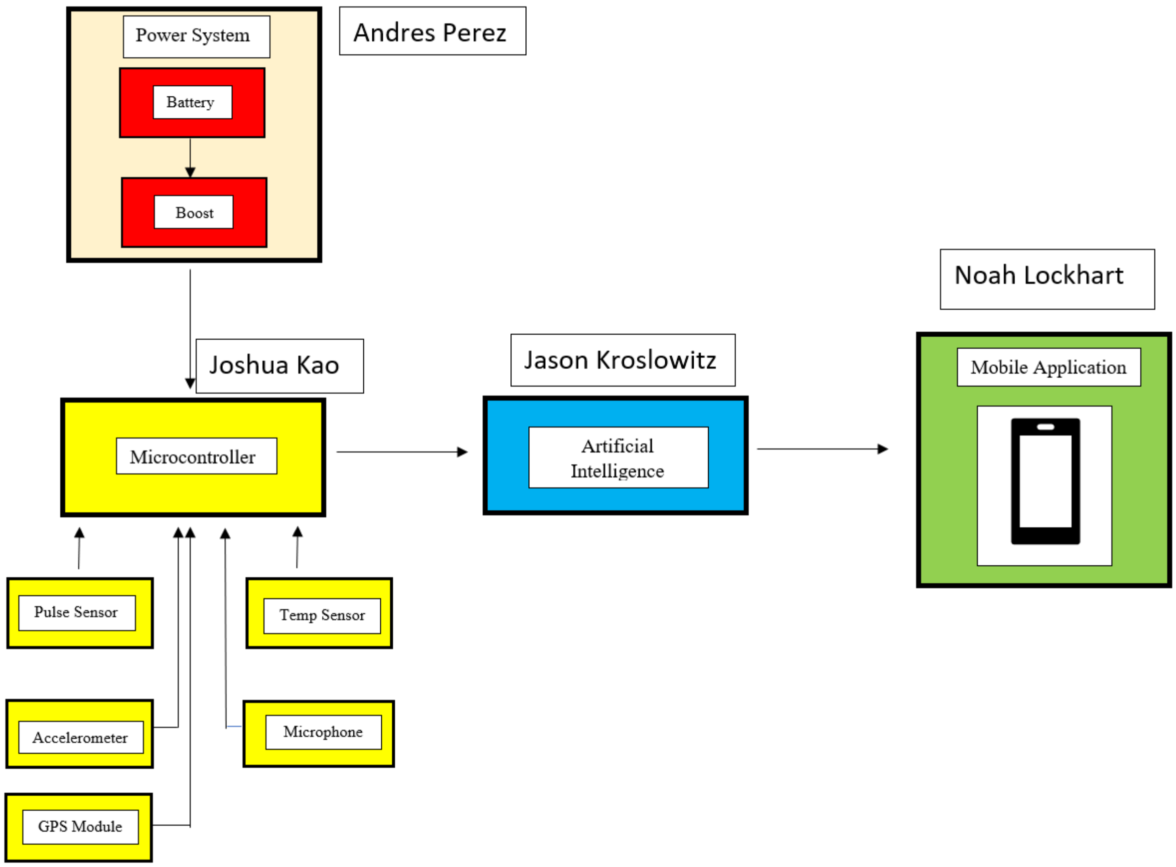


Figure 1: The 12th Man Block Diagram

The 12th Man will be composed of four subsystems. These include artificial intelligence, power supply, mobile application, and sensors/microcontroller featuring a robust enclosure.

Sensors/Microcontroller/Robust Enclosure: The 12th Man device includes 5 different sensors: GPS, temperature, accelerometer, pulse, and audio. Each sensor will be wired and interfaced to a Raspberry Pi 4b microcontroller. The sensors will collect and packet the biometric data automatically and send it to the cloud for inference and display. The goal of the robust enclosure is to hold all wearable components including the PCB, battery sensors, and Raspberry Pi, ensure component security, prevent user electrocution, and device protection.

Power Supply: The wearable device will have a rechargeable battery in place that will be composed of a 3.7V 6000mAh single cell battery. The battery will be connected to a battery charger integrated circuit. The battery will be lithium ion so it will be able to be recharged. The battery will be recharged by an AC/DC adapter to a USB-C input. The rechargeable battery will be connected as an input to a boost converter in order to step up the voltage to meet the required specifications to power the microcontroller as the output of the boost converter will be connected to a USB-C output. This amp-hour specification of the battery will help make battery life last. The battery will also be connected to a fuel gauge integrated circuit. The integrated circuits and USB-C input and output will be designed on a printed circuit board.

Artificial Intelligence: The AI subsystem has several parts. A supervised learning algorithm has been developed by our sponsor to do classification on audio samples. Data preprocessing will entail converting audio samples to mel spectrograms. A CNN will then classify the likelihood of the audio sample originating from one of eight types of environments (small conversation, stimuli, large conversation, etc). Another classification model will be constructed, taking as inputs some/all of the biometrics and the output of the acoustic classifier. This model is the final step in the pipeline to predict a meltdown. Should the environment be deemed stressful or over-stimulative, and the vitals indicative of physiological stress, the model will predict a meltdown is likely to occur. Lastly, we may consider building an additional artificial intelligence based on a generative adversarial network with the hopes of generating convincing “fake” data for biometrics. This will be done should we have issues collecting enough biometric data to train the meltdown prediction model.

App and Database: The database will receive and store all the information the hardware sends it. This will include the GPS location and other readings from the sensors. The app will then present all of this information to the guardian. The guardian will be able to see the location of the individual wearing the device as well as all other sensor readings. If a meltdown is predicted to occur, the app will change and an email will be sent, informing the parent/guardian of the situation.

3.4. Modes of Operations

There is only one mode of operation for the system, which is meltdown mitigation. We are intending to provide end-to-end support for individuals with autism as well as their caregivers. The device will operate the same way at all times. An alert will be generated on the mobile application if the device infers that the user is likely to have a meltdown, and an audio recording will be played on the device. While being worn, the device will remain on at all times. The device features hardware that automatically powers the device down when it is charging. The device, specifically the AI, is intended to be robust enough to generalize to individuals with ASD without calibration.

3.5. Users

The 12th Man is marketed to the guardians/care-givers of individuals who are diagnosed with Autism Spectrum Disorder. The 12th Man will be used by the person with Autism with the intent of improving quality of life. The device will be worn on the waist along with a bluetooth chest band to sense the heartbeat. The app will be easy to use with all the essential information displayed on the home page. The guardian/caregiver and the individual diagnosed with ASD will benefit from the 12th Man since the guardian/caregiver will be more informed of their dependent's status and the user will have more security. User installation will not be complex as instructions will guide the user and guardian/caregiver to utilizing the 12th Man.

3.6. Support

The users will receive a manual with instructions on how to equip the 12th Man wearable on the individual. Provided the sensors are able to collect biometric and environmental data properly, we may include alternative options of wearing the device. The user manual will also include battery life information, cleaning instructions, and warnings. Warnings will be provided to maintain the life span of the 12th Man such as mentioning that the device will not be water-proof and would not be ideal in a water-based environment. Furthermore, we will provide instructions on downloading the mobile application and linking it to the 12th Man device for guardian/caregiver use. The user manual will also have instructions on how to interact with the mobile application to view alerts and the corresponding data retrieved from the device. In the future, it may prove beneficial to have a live call center that a person can receive customer support from.

4. Scenario(s)

4.1. People with Autism

The main usage of the 12th Man is to be worn by people with autism to help them mitigate a meltdown before it occurs. When people with autism are out in public, they can get into situations that can be stressful and overwhelming. When this happens, a meltdown can occur. By monitoring the environment and the biometrics of the wearer, The 12th Man will be able to detect when they are about to have a meltdown. If a meltdown is detected, the device will guide the individual with grounding exercises to calm them down while sending a meltdown alert to the guardian through the mobile application. The guardian will also receive the individual's location for the case of wandering.

4.2. People with Panic Attacks

Another use of the 12th Man could be to detect panic attacks. Since meltdowns can exhibit similar physiological responses as a panic attack does, the device may be used interchangeably. Panic attacks can be caused by an overwhelming amount of stimuli, like crowded areas or loud noises. In addition, panic attacks lead to an increase in heart rate and a shortness of breath. The 12th man can detect when these things are happening and can notify the user of an impending panic attack. The device can also contact a loved one through the app via an alert to let them know that the user is about to or having a panic attack.

4.3. Exercise

The 12th man can also be used by people who are exercising or doing physical activities. By tracking the user's heartbeat and their location, a program can be added to the app to document and show their workout. Additionally, if someone is climbing a mountain, an application can be added to make sure the climber is alright. By tracking the user's heartbeat, the 12th Man can tell the wearer if their heart rate is slowing due to a lack of oxygen.

5. Analysis

5.1. Summary of Proposed Improvements

- The device will have a heartbeat sensor to detect when the individual's heartbeat is speeding up or slowing down. Either scenario could lead to a meltdown.
- The audio inputs, temperature gauge, and accelerometer will give the system more data about the environment the individual is in. For example, determining if the individual is inside or outside, if they are moving, or determining how many people are near them.
- A more advanced GPS in the device will be able to track the user's location more precisely.
- The app will be able to collect all the input data and present it to the guardian of the wearer. The app will show the wearers location and whenever they have a meltdown or if they are about to have a meltdown.
- The AI system in the device will be able to take all the environmental inputs and accurately identify the environment the wearer is in.

5.2. Disadvantages and Limitations

- Sensor data could be unreliable due to bad readings and noise on the signals.
- The device will only be able to last as long as the battery lasts.
- The AI may not be able to detect the environment accurately all the time.
- Input lag from the device to the database to the app and back again may make it difficult to have real time readings.
- Some people may not want to wear the device.
- The wearer could forget to charge the device and the power dies.
- Users may have a hard time understanding how the device works.
- Grounding exercises may not be enough to stop the meltdown.
- Not water resistant

5.3. Alternatives

- An alternate solution could be having someone look after the individual with autism everyday. A trade-off would be having to pay this person every day to watch the individual.
- Launching a large informational campaign with the intent to inform the public about ASD and the challenges it poses. This would rely on public cooperation and would require a great deal of funding.
- The person with autism could also never leave their home and avoid situations that could cause a meltdown entirely. However, this would mean the individual would not live an ideal lifestyle.
- Allowing the individual to have space in a safe environment so that the meltdown can run its course. This is typically how parents currently deal with meltdowns.

5.4. Impact

- This device will help those with autism by preventing and mitigating meltdowns. When the wearer is about to have a meltdown, the device will lead them into grounding exercises to calm the individual.
- Quality of life for people with autism will improve greatly. Now they can go out into public with more security. They know that if they have a meltdown they won’t be alone, that the 12th Man will help them. Also they will feel safe knowing that their guardian will be notified and ready to help if it is needed.

- The guardians of the autistic individual would worry less because they will be alerted when the individual has a meltdown and that a mitigation process has been implemented. The guardian will receive the individual’s location.
- People around the wearer of the device will also learn what situations can cause a meltdown. When someone sees the device calming someone down, then that person will become more aware of what environments can cause a meltdown.
- A negative impact could include stigmatizing the individual wearing the device.
- Another negative impact could include waste disposal of the batteries used to power the devices once they are depleted of their ability to be recharged.

Mitigating Autistic Meltdowns

Joshua Kao, Jason Kroslowitz, Noah Lockhart, Andres Perez

FUNCTIONAL SYSTEM REQUIREMENTS

REVISION - 2
29 April 2022

Functional System Requirements
for

Mitigating Autistic Meltdowns

Prepared by:

Team 12	Date

Approved by:

Josh Kao	Date
Prof. S. Kalafatis	Date
Eric Robles	Date

Change Record

Rev.	Date	Originator	Approvals	Description
-1	[2/23/2022]	[Team 12]	[Team 12]	Draft Release
-2	[4/29/2022]	[Team 12]	[Team 12]	Final Release

Table of Contents

List of Figures

1. Introduction

1.1. Purpose and Scope

Meltdowns in the Autistic community have been a debilitating issue affecting the quality of life for individuals with Autism and their loved ones. The intention of the 12th Man is to improve quality of life for individuals with Autism through a safe, discreet, and user-friendly device that autonomously predicts and mitigates meltdowns. The 12th Man is intended for those who have ASD but it can also be used by individuals who suffer from panic attacks. Figure 1 shows a proposed block diagram of the project from CONOPS.

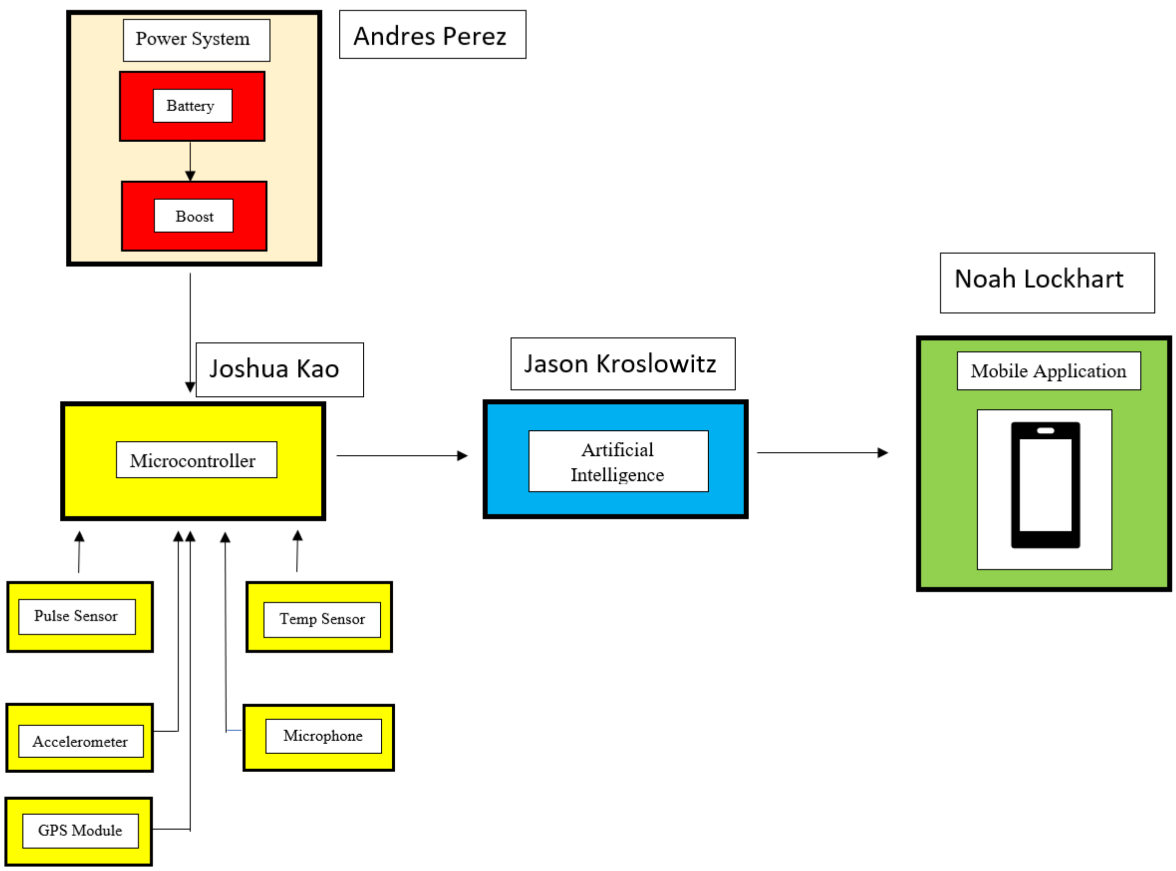


Figure 1: The 12th Man block diagram

The 12th Man is a three stage system composed of a wearable device, artificial intelligence, and mobile application/database. The wearable device shall be composed of a microcontroller, sensors, and a rechargeable battery. Sensors shall monitor the heartbeat, movement, location, and surrounding noise of the wearer. A Raspberry Pi 4 will then send the data to the artificial intelligence, which shall make a prediction about whether a meltdown is likely to occur. If a meltdown is likely to occur, a recording from a loved one will be played through speakers on the wearable. The wearable will also store the data it collects in a remote database. This database will be accessible by a mobile application, which shall provide parents/guardians with information regarding the location and meltdown status of their dependent.

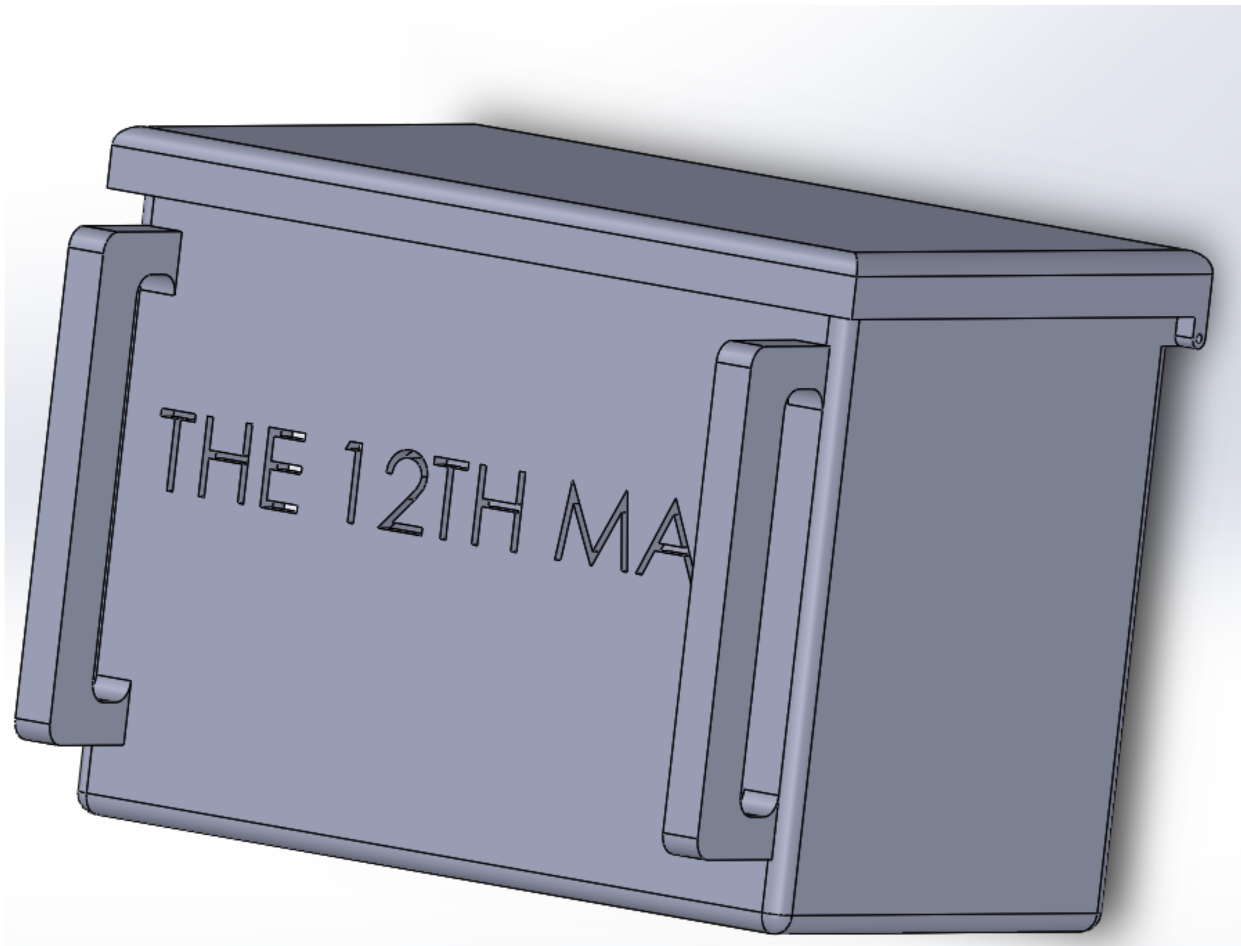


Figure 2: The 12th Man Wearable Device

1.2. Responsibility and Change of Authority

The team leader, Joshua Kao, will have the responsibility of ensuring that all the requirements are met. Any proposed changes to the requirements shall be vetted by each team member, and passed on to our sponsor, MarkusAI, for verification and approval. Team members will each have the responsibility of ensuring that their individual subsystems function as outlined in this document. Subsystem responsibility is as follows:

Joshua Kao	Dataset Collection, Microcontroller/Sensors,

	Robust Enclosure
Jason Kroslowitz	Artificial Intelligence
Andres Perez	Power System
Noah Lockhart	Mobile App and Database

2. Applicable and Reference Documents

2.1. Applicable Documents

Document Name	Revision/Release Date	Publisher
Raspberry Pi Model 4 Datasheet	Release 1 - June 2019	Raspberry Pi Ltd.
TensorFlow Documentation	1.10	Google Brain Team
‘Meltdowns’, surveillance and managing emotions; going out with children with autism	Release 1 - September 2010	US National Library of Medicine and National Institute of Health

2.2. Reference Documents

Document Name	Revision/Release Date	Publisher
50 of most important Raspberry Pi Sensors and Components	July 20, 2020	Raspberry Pi Tutorials
Introduction to Raspberry Pi 4: Tackling the Basic Electronic Kits With the Raspberry Pi 4	May 26, 2020	Device Plus Editorial Team
Meltdown/Tantrum Detection	December 11, 2020	Applied Artificial Intelligence

System for Individuals with Autism Spectrum Disorder		Journal

2.3. Order of Precedence

In the event of a conflict between the text of this specification and an applicable document cited herein, the text of this specification takes precedence without any exceptions.

All specifications, standards, exhibits, drawings or other documents that are invoked as “applicable” in this specification are incorporated as cited. All documents that are referred to within an applicable report are considered to be for guidance and information only, except ICDs that have their relevant documents considered to be incorporated as cited.

3. Requirements

The following section will provide definitions of all three stages for the proof of concept we are developing. “The 12th Man” refers to the system in its entirety, including the wearable device where sensor data is collected and packaged, the mobile app and server where data is sent; stored; and viewed, and the artificial intelligence which performs inference on the data collected. The term “The 12th Man Wearable” refers to the physical device including the sensors, rechargeable battery, microcontroller, and enclosure. “The mobile application and database” refers to the subsystem where collected data is stored and made available to the user. The term “Artificial Intelligence” refers to the inference performed onboard the microcontroller to predict if a meltdown is imminent or not based on sensor readings.

3.1. System Definition

The 12th Man is a wearable device that shall monitor an individual with Autism to predict the occurrence of meltdowns, and mitigate meltdowns should they occur. The project is composed of four subsystems, which include power systems, microcontroller/sensors, artificial intelligence, and mobile application and database.

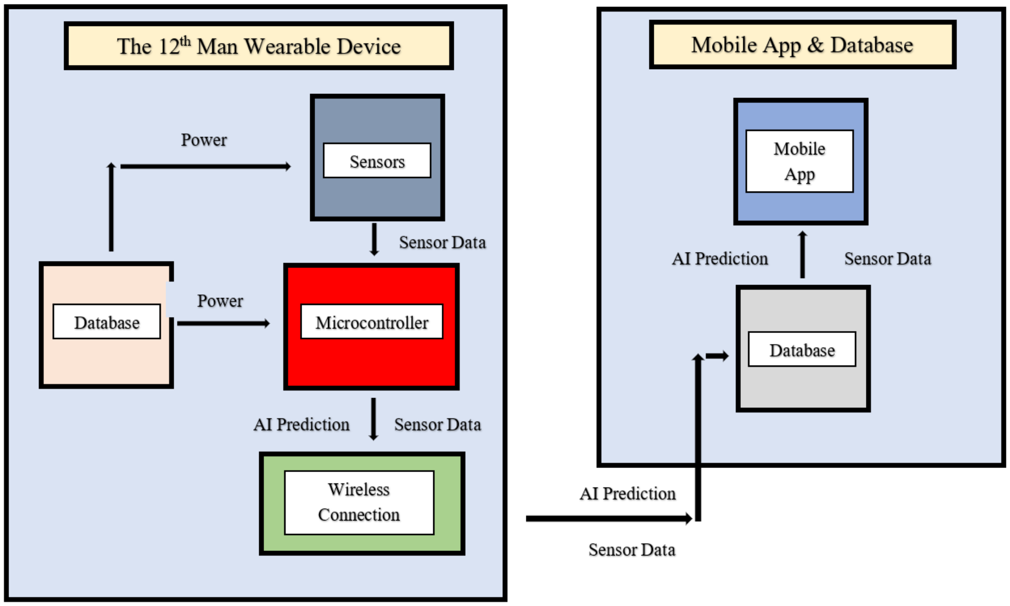


FIgure 3: Functional System Diagram

The power system subsystem shall consist of supplying power to the sensors and microcontroller. The power supply will consist of a lithium-ion battery that will supply power to the GPS module, temperature sensor, and accelerometer. A boost converter shall boost the voltage coming from the battery to supply power to the microcontroller. The system shall have a battery charger and power-path management integrated circuit to support power management to the battery.

The microcontroller, sensor, and robust enclosure subsystem shall gather biometric data and package it into usable data simultaneously and autonomously. The biometric data gathered by the sensor subsystem will include the subject's location, speed, heart rate, temperature, and surrounding audio data. It should also collect gyroscopic data if needed. Following the collection of data, the microcontroller will organize the data into usable .txt packages. The data will then be sent to the server along with the Artificial Intelligence's prediction. The robust enclosure will be a 3D printed box to be worn around the hip of the user. This box will enclose the PCB, battery, all sensors, and the microcontroller by standoff mountings. The robust enclosure will also be finger safe, durable, and wearable.

The artificial intelligence subsystem will be composed of two parts. The environment acoustic classifier shall receive audio data from the microphone sensor, and shall attempt to classify the environment. A separate artificial intelligence will take as input two features from the heartbeat sensor, in addition to the output of the acoustic classifier. This model shall make a prediction on the likelihood of a meltdown occurring based on the similarity of the features it receives to the "trigger" dataset on which it is trained. We hope to be able to generalize to all users in some capacity, while the acoustic CNN can be fine-tuned to provide more specialized audio classification.

The database and application subsystem contains two parts as well. The first part is the database. The database will store all the input signals that are being read to the 12th Man. The most important set of data the database will hold are the GPS coordinates and the AI meltdown prediction. The app will pull the GPS location and the meltdown prediction and display them. The user can also use the app to check the connectivity of the input functions to the database.

3.2. Characteristics

3.2.1. Functional/Performance Requirements

3.2.1.1 Frequency of Measurements

Each sensor in The 12th Man device shall take a measurement every 2 minutes. For the microphone sensor, the data collected will be time-series data, i.e. a 3 second audio recording.

Rationale: Meltdowns that are able to be mitigated will not happen instantly, but rather over an extended period of time, during which warning signs specific to the individual are exhibited. The AI will attempt to learn what the “rumble stage” looks like in an individual and help stop it from progressing.

3.2.1.2 Accuracy of Meltdown Predictions

The 12th Man custom artificial intelligence shall meet an accuracy metric of close to 90% when testing against extrema. There is no fine line between a non-meltdown and a meltdown, thus, we are using are best judgment to identify clear non-meltdown cases, and clear meltdown cases. Testing against these extreme cases should yield an accuracy of 90%.

Rationale: If an individual is going to have a meltdown, it is of the utmost importance that it is predicted with high accuracy so that mitigation can begin immediately. Failing to correctly identify, mitigate, and provide alerts for a meltdown to parents/guardians could be devastating to the individual’s health and safety.

3.2.1.3 System Operation Time Constraint

The 12th Man will have a battery life of five hours. After this amount of time, the device will need to be recharged.

Rationale: The device will be able to function for five hours allowing the guardian of the individual to monitor one in the event that the individual with autism is to be unsupervised.

3.2.1.4 Wireless Communication

The 12th Man and the app must have a strong wireless hotspot connection at the minimum, or be connected to wifi at any given time.

Rationale: The device will send the data to the Dynamo database through the wifi connection. The database will use this data in order to predict if a meltdown will happen. If a meltdown is about to occur then the app will be notified, through the internet connection.

3.2.1.5 System Latency
The data collected by the sensors on the microcontroller and the prediction made by the AI on this data must be made efficiently and transmitted to the database and the mobile application with minimal latency. (20 seconds at the most).

Rationale: When someone is at the beginning stages of a meltdown, then the device must be able to predict this quickly and begin grounding exercises right away (a few seconds could be the difference between the meltdown being mitigated and the meltdown escalating beyond the point of control). Similarly, the parent/guardian will need to be notified as soon as possible if a meltdown is likely to occur.

3.2.2. Physical Characteristics

3.2.2.1 Mass
The mass of The 12th Man wearable shall be less than or equal to 1 kilogram.

Rationale: The device shall not be cumbersome to the individual. Rather, we are attempting to provide a minimal experience that is discreet.

3.2.2.2 Volume Envelope
The volume envelope of The 12 Man wearable shall be less than or equal to 102 mm in length, 99 mm in width, and 89 mm in height. The wearable device will consist of a 3D printed enclosure housing the microcontroller, power system, and sensors.

Rationale: The system must be able to hold all components while still being discreet for the user.

3.2.2.3 Heartbeat Sensor and Device Placement
The heartbeat sensor is unique in that it is not physically connected to the wearable as the other sensors are, but rather, shall be worn around the chest using a strap. The wearable itself shall be placed on the hip and shall feature a strap with a buckle.

Rationale: Heartbeat measurements are most accurate using a sensor that is placed directly across the chest, such as the one that we will be using. Additionally, we determined the most discreet location for the wearable to be the hip.

3.2.3. Electrical Characteristics

3.2.3.1 Inputs
The inputs that The 12th Wearable is able to handle and gather is the power supplied by the components as well as digital data collected from sensors. The sensors will communicate with the microcontroller to receive data .

Rationale: By design, the wearable device is able to be properly powered and readily available to gather data.

3.2.3.2 Power Consumption

The maximum peak power of the system shall not exceed 15W watts. The components will not need to consume a large amount of power to function which shall allow the battery to maintain sustainable battery life. The microcontroller will be the notable component that would draw the most current thus have the highest power consumption.The battery shall then be recharged when there is insufficient energy for the device to consume.

Rationale: The demanding power consumption will not exceed what is required for each component to function properly to ensure the battery shall be sufficient to power the device after a full recharge.

3.2.3.3 Input Voltage Level

The input voltage level of each component shall be in between 3.3V-5V.

Rationale: The battery will be a 3.7V Lithium Ion battery and shall supply power to the microcontroller with 5V after stepping up voltage coming from the battery with a boost converter.

3.2.3.4 System Data Output

Data collected by the wearable device by the GPS module, temperature sensor, heart beat sensor, accelerometer, and microphone. Data will be stored in a database for the mobile application to fetch. The database will hold GPS coordinates, and the AI predictions of a meltdown.

Rationale: The 12th Man shall include a mobile application for users to view alerts and location of the individual wearing The 12th Man.

3.2.3.5 System Audio Output

The 12th Man wearable shall play an audio recording from a loved one when a meltdown is predicted.

Rationale: This would help mitigate a meltdown however the speaker may draw more current.

3.2.3.6 Wiring

The power supply components such as the battery charger integrated circuit, fuel gauge integrated circuit, and boost converter integrated circuit shall be connected on a printed circuit board.

Rationale: A printed circuit board will help make the wearable device as compact as possible.

3.2.4. Environmental Requirements

3.2.4.1 Wind

The CNN for acoustic classification of the environment in which the individual is located is highly susceptible to wind. Should the individual be in a windy environment, the acoustic classifier will classify the environment as such, regardless of if there is a large conversation also occurring (i.e. failing to classify a large outdoor gathering, which poses danger of overstimulation, due to large amount of wind noise).

3.2.4.2 Rain

The 12th Man is not waterproof. The user must be cautious in wet environments.

Rationale: Exposure to water may cause short circuiting in the wearable device resulting in damaged components.

3.2.5. Failure Propagation

3.2.5.1 Diagnostics

The 12th Man will show the biometrics collected by the wearable on the mobile app. If biometrics are not visible the user can identify if any sensors are no longer receiving data or if there is a connectivity issue. Error handling for failed sensors will also be deployed. For example for failure with the pulse monitor the last data collection sample will be used for inference.

3.2.5.2 Database Connection

The user will be able to see if the app is not connected to the database properly.

Rationale: Allows users to see if the app is able to pull data.

3.2.5.3 GPS Connection

The user will be notified if the GPS coordinates are not being sent or received correctly.

Rationale: The location of the wearer must be presented in the app at all times.

3.2.5.4 Input Sensor Signals

A test will be added to make sure that each device that is taking an input, the accelerometer, microphone, GPS, heart rate sensor, is sending data to the database.

Rationale: If any of the input signals are no longer sending data, then the device will not be able to detect meltdowns.

4. Support Requirements

4.1.1 User Interface
Users will need to have an Android smartphone to obtain and run the app. The user should also be able to afford and access a data plan or wifi.

4.2.1 Usage of Wearable Device
The user must be wearing the heart rate sensor on their chest for the system to work reliably.

4.3.1 Maintenance
The user should check reading daily to see if the device is working properly before going into an unsupervised area.

Appendix A

AI	Artificial Intelligence
ASD	Autism Spectrum Disorder
CONOPS	Concept of Operation
GPS	Global Positioning System
I/O	Input/Output
mm	Millimeters
PCB	Printed Circuit Board
SQL	Structured Query Language
V	Volts
W	Watts
4G	Mobile Communication Standard

Mitigating Autistic Meltdowns

Joshua Kao, Jason Kroslowitz, Noah Lockhart, Andres Perez

INTERFACE CONTROL DOCUMENT

REVISION - 2
29 April 2022

Interface Control Document
for

Mitigating Autistic Meltdowns

Prepared by:

Team 12 Date

Approved by:

Josh Kao Date

Prof. S. Kalafatis Date

Eric Robles Date

Change Record

Rev.	Date	Originator	Approvals	Description
-1	[2/23/2022]	[Team 12]	[Team 12]	Draft Release
-2	[4/29/2022]	[Team 12]	[Team 12]	Final Release

Table of Contents

List of Figures

1. Overview

The Interface Control Document or ICD for The 12th Man will go into detail regarding how the subsystems in the Concept of Operations and requests from the Functional System Requirements will be fulfilled. The ICD will provide characteristics of the various stages of our project and how they will interact between each other including communications, component location, and electrical characteristics.

2. References and Definitions

2.1. References

MIL-STD-810F
Environmental Engineering Considerations and Laboratories Tests
1 Jan 2000
Change Notice 2
30 Aug 2002

American National Standard for VME64 (ANSI/VITA 1-1994 (R2002))
4 Apr 1995

American National Standard for VME64 Extensions (ANSI/VITA 1.1-1997)
7 Oct 1998

2.2. Definitions

AI	Artificial Intelligence
ASD	Autism Spectrum Disorder
CONOPS	Concept of Operation
GPS	Global Positioning System
IC	Integrated circuit
I/O	Input/Output
mA	Milliampere
mAH	Microamp hours
mm	Millimeters
mW	Milliwatt
Li	Lithium
PCB	Printed Circuit Board
SQL	Structured Query Language
TBD	To Be Determined
V	Volts
W	Watts
4G	Mobile Communication Standard

3. Physical Interface

3.1 Weight

3.1.1 Weight of the Sensors and microcontroller

The sensors and microcontroller have a combined weight of less than half a kilogram This allows for the wearable device to sit comfortably on the user's waist.

Component	**Weight(grams)**
Raspberry Pi 4 (microcontroller)	46 grams
MPU6050(accelerometer)	6.208546
GPS Module GPS GT-U7	5.668
Polar H9 Heartrate Monitor	100
USB 2.0 Mini Microphone	22.9631
DS18B20 Temperature Sensor	22.96

3.1.2 Weight of Power System Components

Component	**Weight(grams)**
Li-ion Battery	158.757
Battery Charger IC	0.05
Boost Converter IC	0.02
Fuel Gauge IC	0.03
MOSFET	0.03
Printed Circuit Board	21

3.2 Dimensions

3.2.1 Dimensions of Sensor, Microcontroller, and Robust Enclosure

The biometric sensors will fit on a PCB board inside a 3D printed enclosure. The enclosure will be less than or equal to 170 mm in length, 112 mm in width, and 50 mm in height.

Device	Length (mm)	Width (mm)	Height (mm)
Accelerometer	20.066	16.51	9.906
Temperature	60.96	15.24	2.54
Heart-Rate	34.036	65.034	9.906
Microphone	22.29	18.45	7.12
GPS	27.686	1.01	22.606
Prototype A	111	85	45
Prototype B	85.598	56.388	39
Wearable	100	99	89

3.2.2 Weight of Power System Components

Component	**Length(mm)**	**Width(mm)**	**Height(mm)**
Li-ion Battery	96.5	66.5	8.0
Battery Charger IC	2.8	2.8	1
Boost Converter IC	2	2	2
Fuel Gauge IC	2.5	4	2
MOSFET	2.49	2	3
Printed Circuit Board	83.82	88.9	1

3.3. Mounting Locations

3.3.1. Mounting of Sensors

All sensors will be placed on the PCB board such that there is no interference with their data collection. The microphone and temperature sensors will be partially external to allow for accurate noise and temperature collection without interference. The other sensors are not location dependent.

3.3.2. Mounting of Microcontroller

The microcontroller will be inside a 3D printed enclosure below the PCB board. The wiring from the microcontroller to the sensors will also be tucked inside said enclosure and spaced apart to avoid any potential for short circuits.

3.3.3. Mounting of Heartbeat Sensor

The heartbeat sensor should be around the user's chest to ensure the best heartbeat possible.

3.3.4. Mounting of the device

The 12th man device should be attached to the user's pants or belt around their waist without anything covering the microphone. This to ensure the best audio quality of the surrounding noise is collected.

4. Electrical Interface

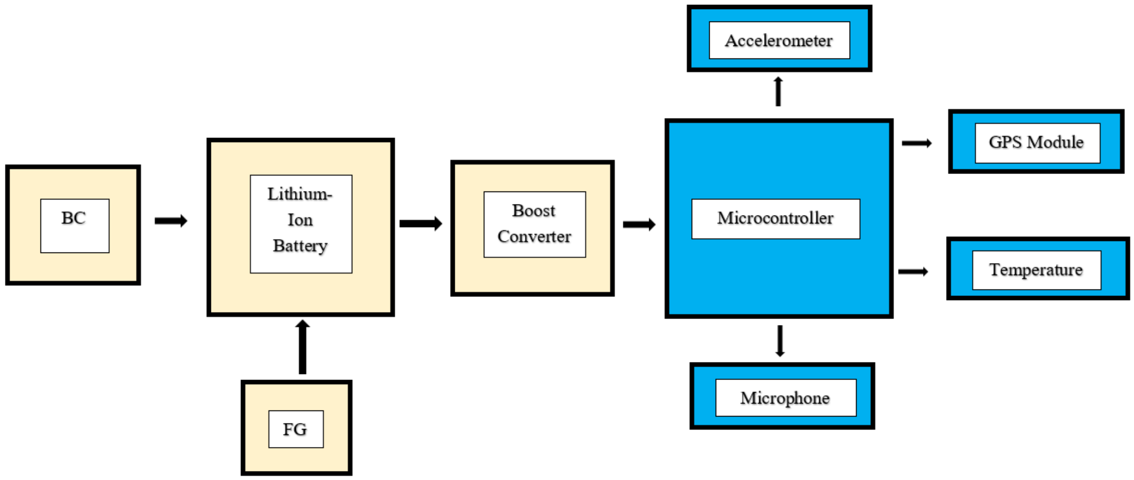


Figure 1. Electrical block diagram

4.1. Primary Input Power

The Lithium Ion battery shall be charged and recharged by an AC/DC adapter in which a battery charger integrated circuit shall be connected to the 3.7V 6000 mAh battery. The battery shall provide input voltage to a boost converter to boost the voltage output to 5V and supply power to the Raspberry Pi. The microphone shall be powered by the microcontroller via USB. The heart rate sensor is an internally powered sensor.

4.2. Voltage and Current Levels

4.2.1. Voltage, Current, and Power Consumption

Component	Voltage(V)	Current(mA)	Power(mW)
Raspberry Pi 4b	5	3000	15000
GPS Module	5	45	225mW
Temperature Sensor	3.3	1	3.3mW
Accelerometer	3.3	3.9	12.87mW

Microphone	5	1	5mW

The current drawn by the GPS module, temperature sensor, accelerometer is assumed to not draw significant current. The Raspberry Pi 4b will draw a significant amount of current. The battery shall have a capacity of 6000mAh to ensure sustainable battery life for the device.

4.3. Signal Interfaces

4.3.1. Input Sensors

The GPS module, ambient temperature sensor, heart rate sensor, accelerometer, and microphone shall communicate with the Raspberry Pi so that data can be collected.

4.3.2. Output Signals

For the purpose of mitigation, The 12th Man device should have an output speaker that can be used to play audio recordings. Said audio recordings are grounding exercises recorded by loved ones. The Raspberry Pi should be able to communicate with the speaker so that it can be utilized as an output device.

4.4. User Control Interface

4.4.1. Database Signals

The mobile application will provide the user with a control interface. This interface will provide the user the ability to tether their mobile application to the 12th Man device their dependent will be wearing. The parent/guardian shall then be able to view the data collected by the device and an indication of the meltdown status of their dependent. The main information displayed will be the user location and status. The location of the device will be displayed on the front page on a google map. The AI predicted status will also be displayed, saying whether the wearer is either stable or about to have a meltdown. If the AI does predict that a meltdown may occur, the app will send a notification to the guardian.

The user will be able to check the status of each input signal. The user will see if all the database is receiving data from the GPS, accelerometer, temperature sensor, heart rate sensor, and AI predictions.

4.4.2. Physical Device

The individual with Autism who is wearing the device is not provided a control interface whatsoever, and is simply responsible for wearing the device.

5. Communications / Device Interface Protocol

5.1. Wireless Communications (WiFi)

Wifi connection will be achieved using the Raspberry Pi 4 microcontroller. The Raspberry Pi 4 microcontroller utilizes a Wifi package that allows for wireless connectivity. The wifi connection will allow the device to take inputs from the GPS, accelerometer, heart rate sensor, temperature gauge, and microphone and send that information to the database. The database will use this data to identify if a meltdown will occur. If a meltdown is about to happen, a signal will be sent to the device to start the grounding exercises. The database will also send information from the database to the app using a HTTPS connection, while the app is connected to the internet.

5.2. Microcontroller Input and Output

The raspberry pi includes an assortment of digital and analog inputs. Analog Input pins take in voltage at 5v and also take in signals from an accelerometer and GPS. The digital input pins receive microphone data via bluetooth and USB ports. All the following pins meet the requirements for the sensors used above.

5.3. Sensor Communication

The sensors will primarily communicate through pins with the Raspberry Pi using an I2C connection. The heartbeat sensor will use the bluetooth module attached to the Raspberry Pi and the microphone will use the USB module included with the Raspberry Pi.

5.4. Device Peripheral Interface

The database will be connected to the mobile app through a HTTPS connection. This is a reliable way to send data from a Dynamo database to an application.

Mitigating Autistic Meltdowns

Joshua Kao, Jason Kroslowitz, Noah Lockhart, Andres Perez

Execution and Validation Plan

REVISION – Draft 2

29 April 2022

1. Execution Plan

2. Validation Plan

Task	Status	Responsibility
Read Accurate data from sensors and microcontroller, Heart Rate 60-110 bpm, microphone max sensitivity, accelerometer movement on all 3 axis, temperature below 110 fahrenheit, gps accurate location	Completed	Joshua
Validate Polar H-9 against Apple watch to show the same trends with less than 1 standard deviation	Completed	Joshua

Sensors Collect data automatically every 30 seconds	Completed	Joshua
Validate Microphone against Mac Microphone using spectrogram in MATLAB	Completed	Joshua
Validate GPS location using Google API with less than 10 meters deviation	Completed	Joshua
Validate Accelerometer by moving on all axes graphing the trends to show significant movement on certain axes	Completed	Joshua
Validate temperature sensor by using the weather app for outdoors and the thermostat for indoors with less than 3 degrees of deviation	Completed	Joshua
Compile a synthetic meltdown dataset containing samples that represent meltdown and non-meltdown scenarios.	Completed - Continually augmenting dataset for better classification.	Joshua & Jason
Interview and research data from Special Education Specialists	Completed	Joshua & Jason
Application is hosted and can be accessed by multiple devices	Completed	Noah
Verify sensor sizes are smaller than 3.75 x 6 x 1.13 inches	Completed	Joshua & Andres
Create Robust and secure board for sensors through terminal blocks, soldering, and mounting	Completed	Joshua
Create and print robust enclosure no larger than 110mm x 99mm x 100mm that allows sensors to be mounted	Completed	Joshua
Evaluate acoustic CNN performance on correctly classifying environments	Completed - Confusion Matrix	Jason
Verify that correct data has been packaged into .txt file	Completed	Joshua
Preprocess and clean ML dataset.	Completed	Jason

Min-Max Scaling & 1-Hot Encoding		
Send Latitude and Longitude to database from API and see the same data in the database	Completed	Noah
Receive Latitude and Longitude from API and see the same data in the API variables	Completed	Noah
Train a sequential NN to recognize meltdown triggers (combination of biometric inputs and acoustic CNN output).	Completed - Can always improve accuracy by augmenting the training set.	Jason
Access application on a mobile device	Completed	Noah
Press notification button on app and get a notification on mobile	In Progress	Noah
Send GPS coordinates from API to app to display coordinates on google maps	Completed	Noah
Provide Proper Input Voltage and Current Levels(Sensors)	Completed	Andres and Joshua
Provide Proper Input Voltage and Current Level(Microcontroller)	Completed	Andres & Joshua
Fine tune prediction models	In Progress - Classification accuracy of the acoustic CNN could be improved	Jason
Verify PCB is within 3.3x3.5 inches	Completed	Andres
Solder all components to the PCB	Completed	Andres
Verify Proper Output Voltage and Current from Boost Converter	Not Completed/Design issues when testing on PCB	Andres

Mitigating Autistic Meltdowns

Joshua Kao, Jason Kroslowitz, Noah Lockhart, Andres Perez

Subsystem Reports

REVISION – Original

4 April 2022

Subsystem Report

for

Mitigating Autism Meltdowns

Team 12

Approved by:

__Team 12____________________4/28/22__

Project Leader Date

____

Prof. Kalafatis Date

____

T/A Date

Table of Contents

List of Tables

List of Figures

1. Introduction

The 12th Man is a wearable device capable of monitoring and supporting individuals with autism no matter where they are. The 12th Man will provide a sense of comfort for these individuals and their loved ones through meltdown prediction and mitigation. The 12th Man gathers data through the sensor and microcontroller subsystem which collects biometric data simultaneously every 30 seconds. This data is then packaged and sent to the Machine Learning subsystem where prediction occurs. The data is also sent to the App and Database subsystem where the information is displayed to the user's guardian and saved. The wearable device is powered by the power subsystem through a USB-C charger and lithium ion battery. Each subsystem was validated through careful and rigorous testing. The full path of integration for these subsystems are specified in the Conops, FSR, and ICD.

2. Microcontroller, Sensors, and Robust Enclosure

2.1. Subsystem Introduction

The purpose of the sensor subsystem is to collect and aggregate biometric and environmental data at the same time on a scheduled basis. This data is then sent to the Machine Learning subsystem for inference and sent to the App/Database for permanent storage. Data is used by artificial intelligence to predict the likelihood of a meltdown occurring and display the status to the parent/guardian through the app. The data collected from this subsystem includes heart-rate, temperature, audio, acceleration, and GPS location. The goal of the robust enclosure is to create a stable, non-conductive, and wearable housing for all of the components including the PCB, battery, sensors, and microcontroller.

2.2. Subsystem Details

Data collection is fully automated via a cron-job and collected in 30 second samples and then instantly packaged into a .txt file. It is critical to verify that each sensor is operating properly. After interfacing with the microcontroller, each sensor was tested to validate that it collected the proper data and performed according to the manufacturer's specifications. The second part of this subsystem included creating a robust enclosure to securely hold the sensors, making a wearable device, and having all wires securely placed. Below is an image of the wiring diagram for the sensor subsystem, please note that the images are not to scale. The wearable was 3D printed with the ability for all components to be securely enclosed.

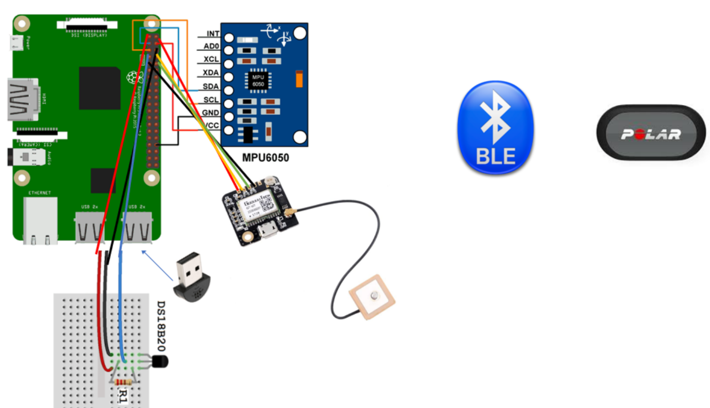


Figure 1. Sensor and Microcontroller Interface

2.3. Sensors

2.3.1. Operation Heart-Rate Sensor

A heart-rate sensor is the most valuable biometric in the prediction of a meltdown. A Polar H9 sensor was used to collect the individual's heart rate . The Polar consists of a pulse meter which is strapped across the chest. The sensor tracks the heart-rate of the individual and relays the information to the microcontroller via a bluetooth low energy connection. Advantages of the Polar H9 compared to other pulse meters is that it is directly on the user's chest, externally powered, and industry proven. This allows for more accurate data along with not using any power from the wearable device itself. The connection between the Raspberry Pi 4 and Polar H9 was achieved through terminal commands as seen below.

```
# Collect data from heartrate monitor
timeout 30s gatttool -t random -b D9:B0:FA:49:40:87 --char-write-req --handle=0x0011 --value=0100 --listen |
stdbuf -o0 cut -d ' ' -f 7 > hr_hex.txt
```

Figure 2. Heart-Rate Command Line Script

The terminal command used to connect via bluetooth energy is called gattool, which allows reading and writing to bluetooth low energy devices. The device's mac address was found using a command called hcitool. To read the data, the handle and register value has to be specified unique to the Polar H9. This data is read from the Polar H9 in hex, and is pipelined into a .txt file. These values are then passed to a hex to decimal converter script to put the heart rate in a more readable form. The final output of the heart-rate data is in BPM. Below is a diagram with an overview of the process; the full script can be found in the Appendix A.

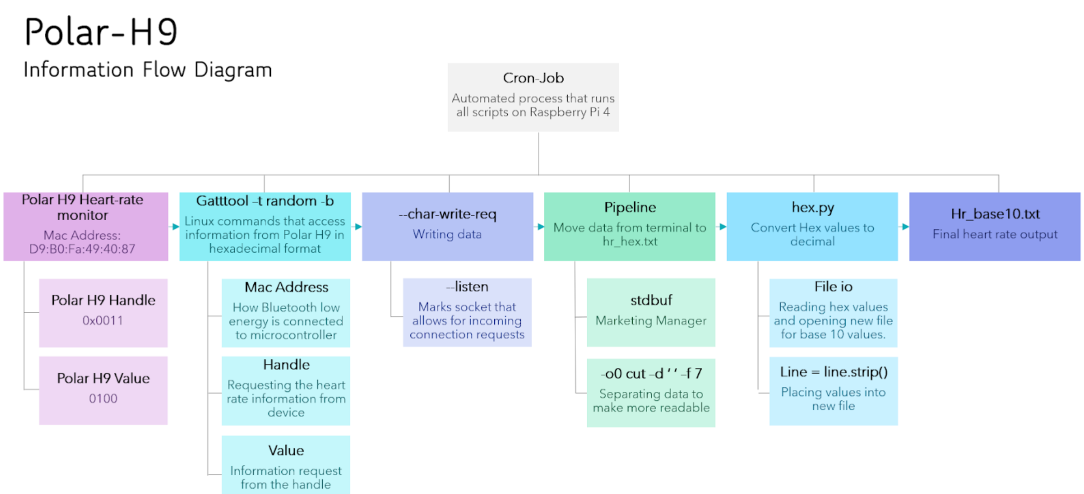


Figure 3. Polar H-9 Code Flow Chart

2.3.1 Validation Heart-Rate Sensor

Although the Polar H9 is an industry grade heart-rate sensor, it still has to be validated. The Polar H9's data was validated by using another industry proven pulse meter, an Apple watch. Both the Apple watch and the Polar H9 were worn at the same time and in the graph below the trend of both sensors was generally the same with a standard deviation of ±2 BPM.

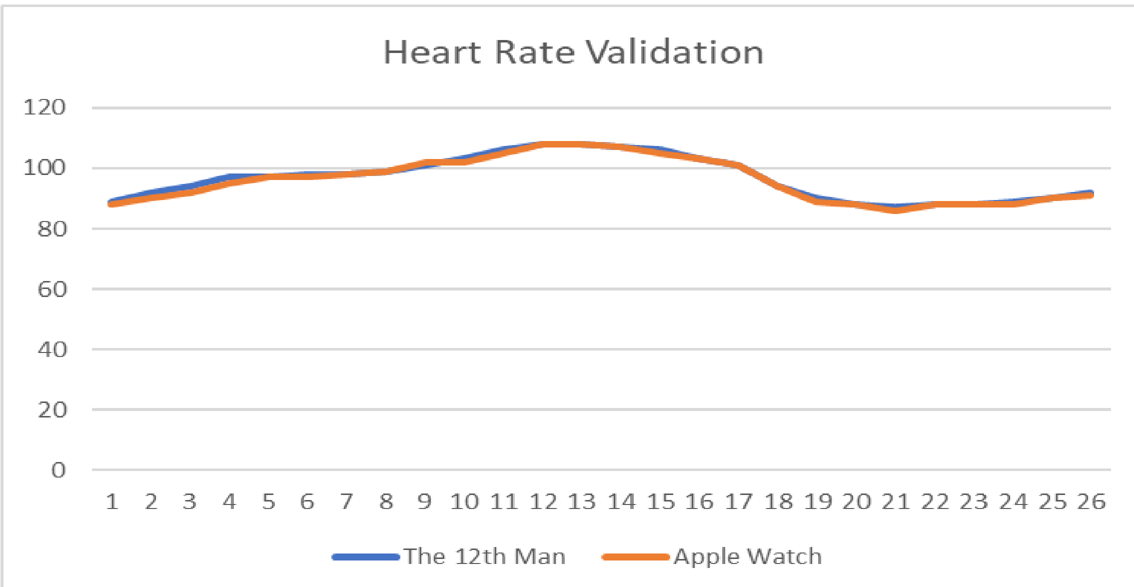
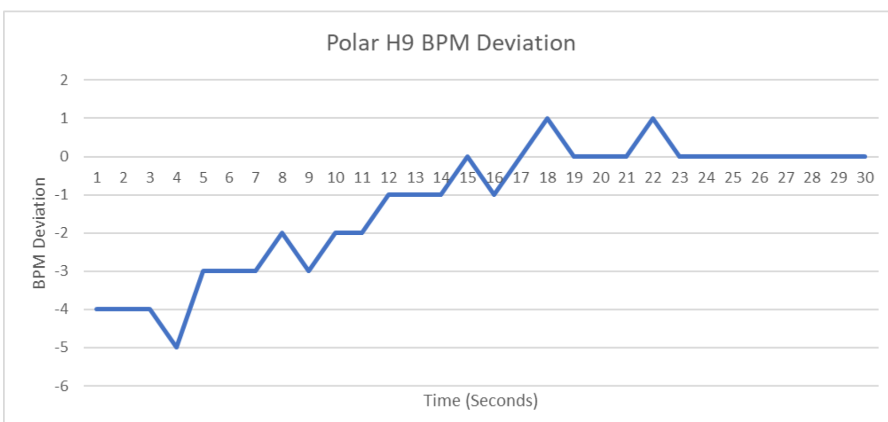


Figure 4. Heart-Rate Validation Graph



Polar H9 Deviation from Apple Watch

While this was just one case, multiple cases were run and resulted in the same trends validating the proper collection of heartbeat.

2.3.2 Operation Temperature Sensor

To measure the environmental temperature, a DS18B20 temperature sensor was used. This sensor utilized one-wire communication and “parasite power mode”, meaning all the power used is through the serial connection. The sensor is accurate from a data range of 55-125 degrees celsius with a 64 bit unique address. The advantage of this sensor is that it is waterproof and located externally on the wearable to more accurately collect the data of the surrounding environment rather than the device itself. Data for this sensor is collected in 10 values over the time frame of 30 seconds via the python script called temp.py located in Appendix A. This script accesses the gpio pin module and activates the temperature sensor module, reading the data line by line and saving it to file. This data is then converted to Fahrenheit and saved into a final .txt file. Below is a diagram overviewing the temp.py script.

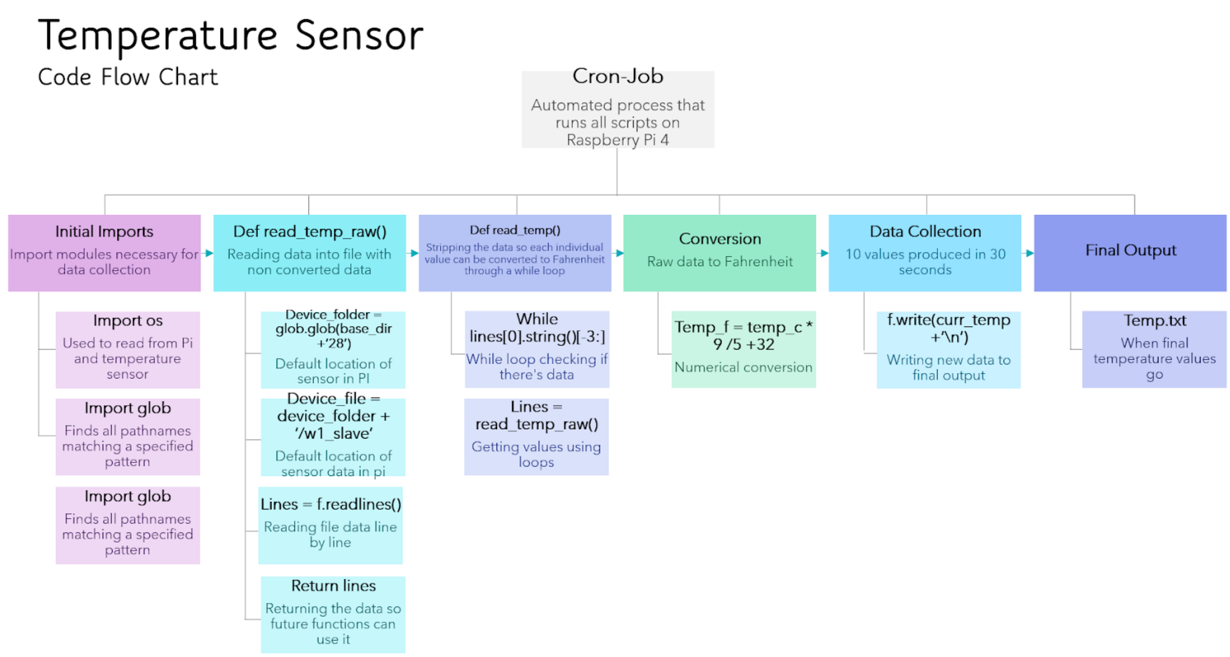


Figure 5. Temperature.py Code Flow Chart

2.3.2 Validation Temperature Sensor

The DS18B20 sensor was validated by comparing its measured temperature with the daily weather report for outdoor conditions and indoor thermostat for indoor conditions. The sensor collects 10 values in 30 seconds and the trends for the temperature sensor are shown below.

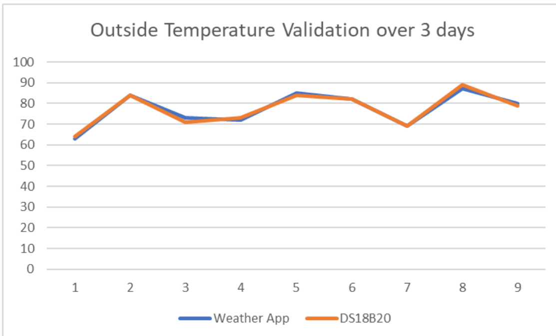


Figure 6. Outdoor Temperature Validation

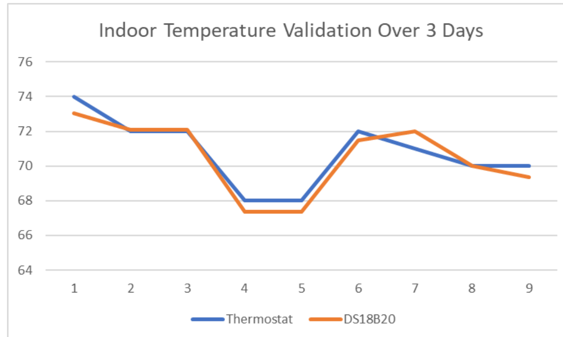
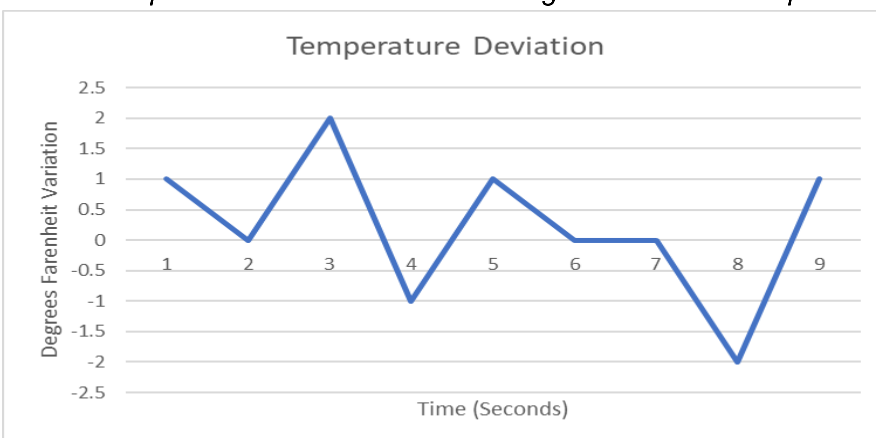


Figure 7. Indoor Temperature Validation



Degrees of Variation from Industry Tested Sensor

The following graphs map the data between the DS18B20 and other temperature sensors in multiple conditions. Data was collected over a period of three days at different times of the day and then compared. As you can see the data maps together with a standard deviation of about ±1 degree Fahrenheit per point recorded. With the high degree of corroboration between the two values, the sensor was validated.

2.3.3 Operation Microphone

Microphone data is another critical piece of information for meltdown prediction. Audio can be used to ascertain environmental stimulants and overall stress levels, and artificial intelligence can use this input to make a prediction. The microphone used to collect environmental data is a SunFounder USB 2.0 Mini Microphone. This microphone was chosen due to its small size. Using a python script called recorder.py, the microphone collects environmental audio in 3 second intervals at 44.1 kHz with a sampling rate of 4096. The python script utilizes a python module named pyaudio. This allows for the audio to be read into a stream that is then packeted into a .txt file. Below is a diagram overviewing the recorder.py script, the full script can be found in Appendix A.

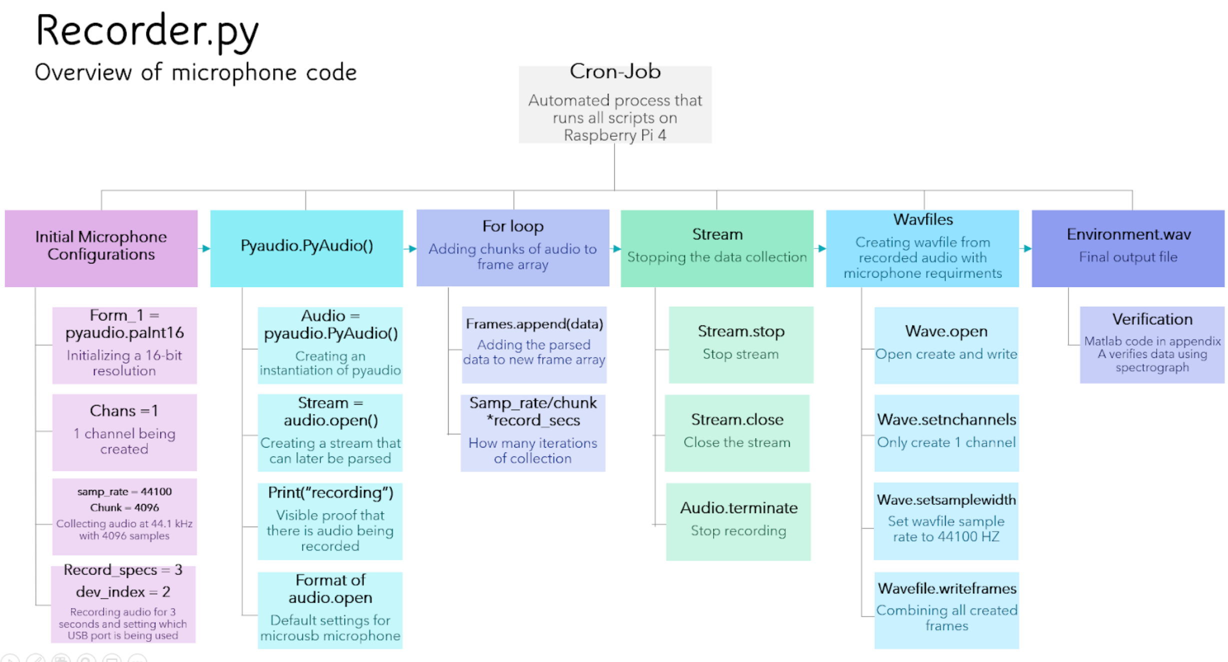


Figure 8. Recorder.py Code Flow Chart

2.3.3 Validation Microphone

The microphone's validation was quite unusual compared to the other sensors. While the .wav file is created and can be listed to, validation does not solely rely on being able to hear the audio. When validating the microphone, a mac's microphone was used. While most expensive microphones collect audio at 48.0 kHz, ours collects at 44.1 kHz. For this reason validation had to be used with a spectrogram as seen below. Using MATLAB (code can be found in Appendix A) the spectrograms trends were plotted in terms of magnitude to verify that they trended the same.

Despite the difference in the sampling rates between the microphone sensor and the Macbook Pro microphone, the spectrograms qualitatively validate that our sensor is performing as expected. Frequency domain analysis reveals the strength of the signal at various frequencies. While the two microphones may be collecting audio data at different magnitudes/sensitivities, the similar distribution of the frequency components conveys that the sensor microphone is “hearing” the same thing as the Macbook Pro microphone, albeit at different volumes. Thus, the microphone is validated. Additionally the artificial intelligence acoustic classifier was used to validate output data by recording the same audio on two different microphones and then checking to see if the classification was the same.

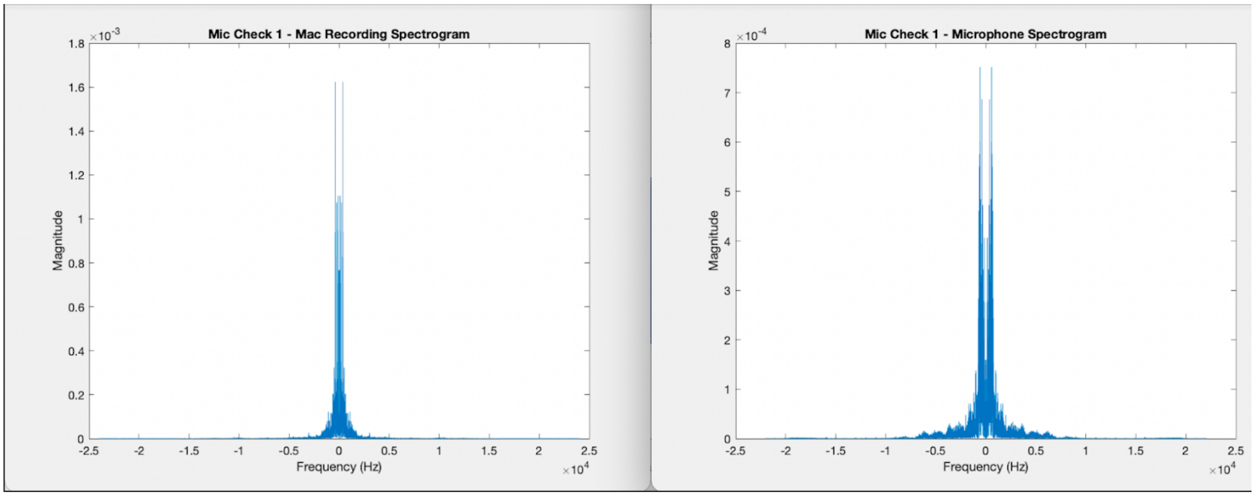


Figure 9. MATLAB Audio Validation Spectrographs

2.3.4 Operation GPS

Knowing the location of the user is very important for the caretakers of individuals with autism. The GPS receiver used is a Gouuu-Tech GT-U7 with a NEO-6M gps module. This module was selected due its affordability, performance, and generally low usage of power. Through a python script called Location.py, which uses the I2C interface and the serial0 port, the GPS module collects longitude and latitude to continuously collect the position of the user. This sensor performs most accurately when the user is outside or in a smaller building. Below is a diagram outlining the location.py script, the full script can be found in Appendix A.

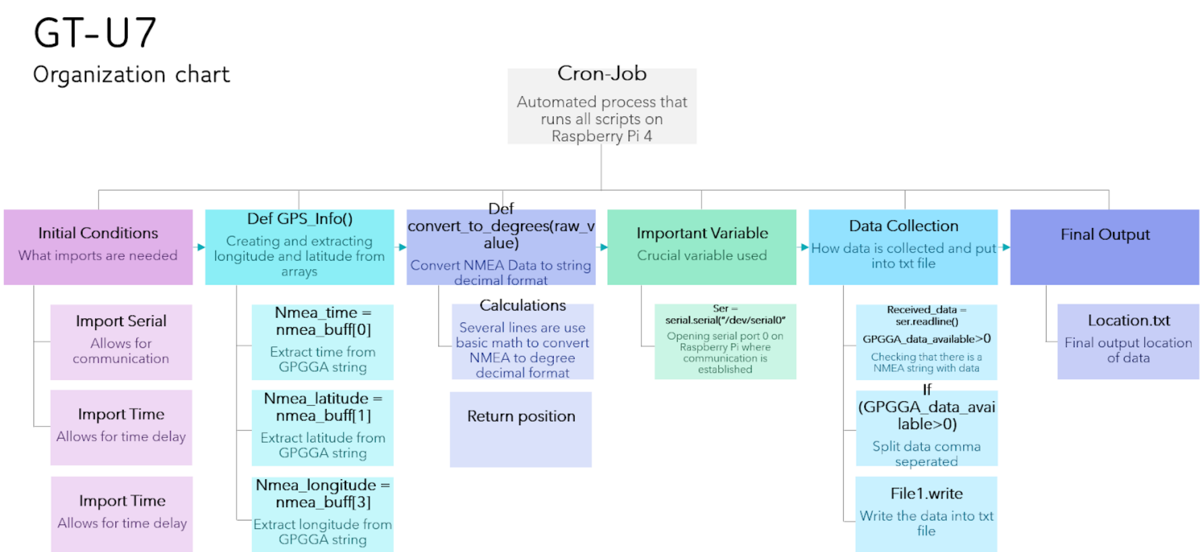


Figure 10. Location.py Code Flow Chart

2.3.4 Validation GPS

Validation of the gps is quite important, as it is important to know that you are in the right place at all times. The GT-U7 was validated by recording the location of the sensor in multiple locations. These coordinates were then put into google maps API to verify that the location presented was within 70 feet of accuracy. Below you can see the data validation of when testing was done outside of Zachry Engineering building.

```
Latitude  :  Longitude  :  Time
30.6208      -96.3393      0.43449368500000674
```

Figure 11. GT-U7 Coordinates

Figure 12. GT-U7 Validation

Below is a graph matching the trends between the GT-U7 and Google maps. In the beginning, there was a larger difference of about 50 meters, but over the span of 10 minutes the satellites adjusted and corrected the difference to about 1 to 5 meters. Only two lines can be seen on the graph because the trend was so similar that the GT-U7 lines were covered by the Google API's.

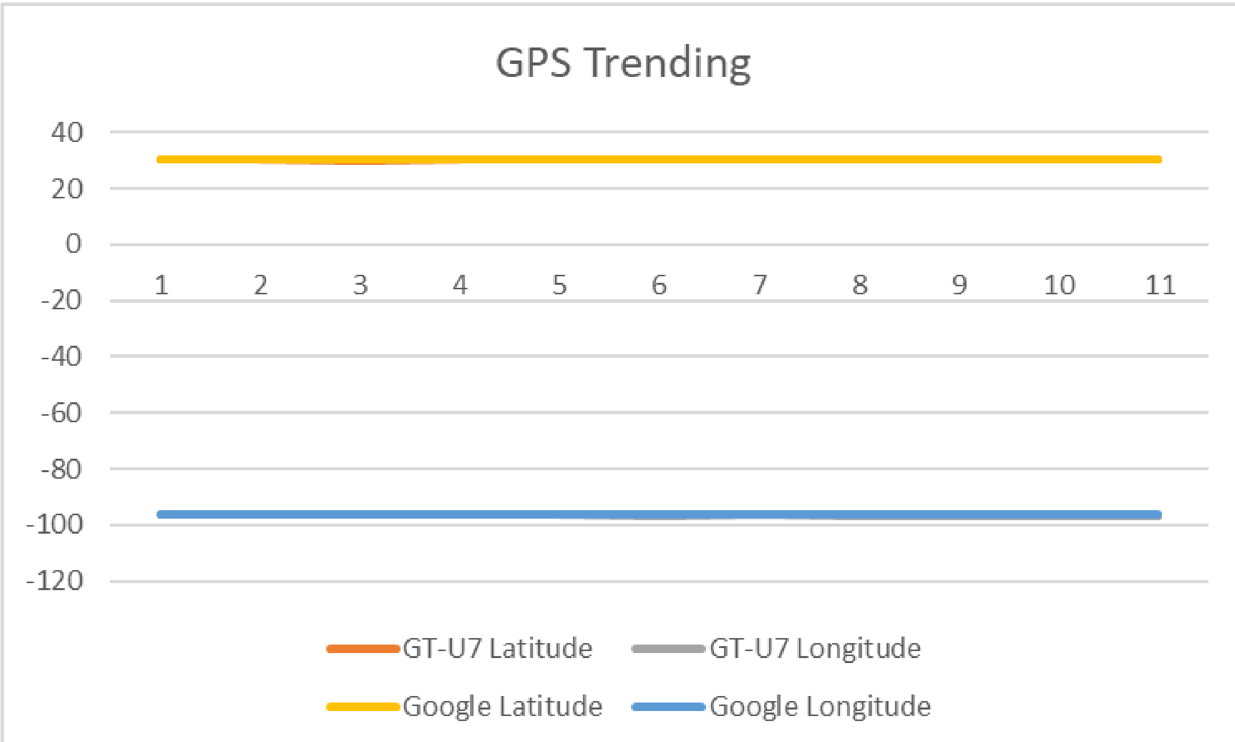


Figure 13. GPS vs Google API Validation

2.3.5 Operation Accelerometer

The MPU6050 is an accelerometer module that consists of a three-axis accelerometer and a three axis gyroscope. This sensor was selected due to it having additional sensors that can be used if needed. It allows for the measurement of the velocity, orientation, acceleration, and displacement of the user. The accelerometer is connected to the Raspberry Pi using an I2C communication method and using code called accel.py found in the Appendix A and through the smbus module the data is able to be extracted. Standard accelerometer measurement units for the MPU6050 is 2g, giving it a sensitivity of 16,384. 16,384 is the value that all movement on every axis is scaled by. Below is an overview of the accel.py script.

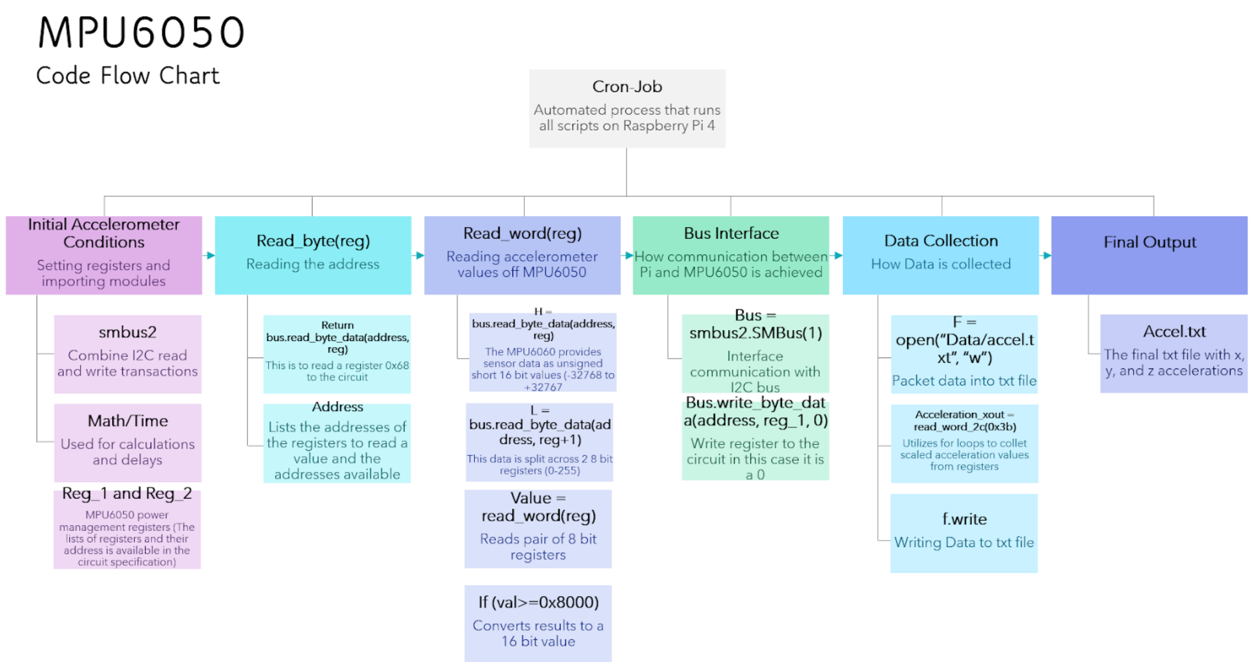


Figure 14. Accel.py Code Flow Chart

2.3.5 Validation Accelerometer

The MPU6050 measures with a 2g scale range along 3 axis's X, Y, and Z. Values recorded by the accelerometer can be multiplied by 9.8 to give the result in meters per second squared. The validation was completed by moving the accelerometer along each of these axes and verifying that there was a change in values. For example, when the accelerometer is stationary there should be a value of 1 on the Z axis. This means that in the Z direction, the accelerometer is measuring 9.8 meters per second squared of acceleration, which can be attributed to gravity. Below are charts following the movement of the accelerometer on all axes.

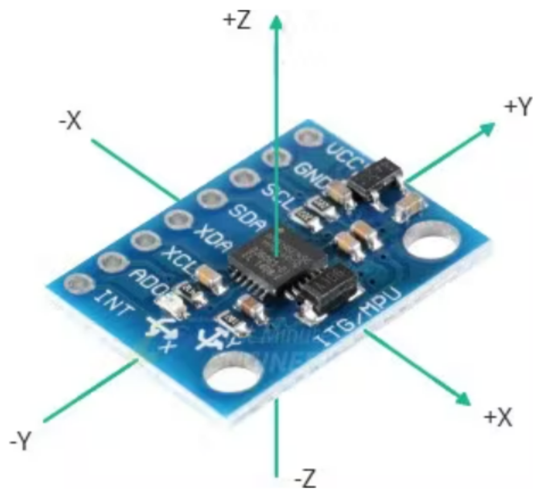


Figure 15. MPU6050 Axes of Measurement

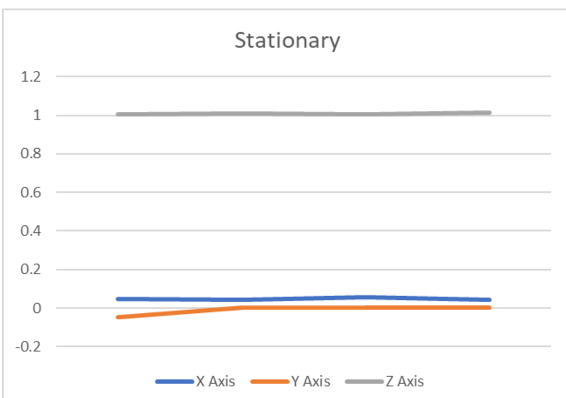


Figure 16. Stationary Accelerometer Validation

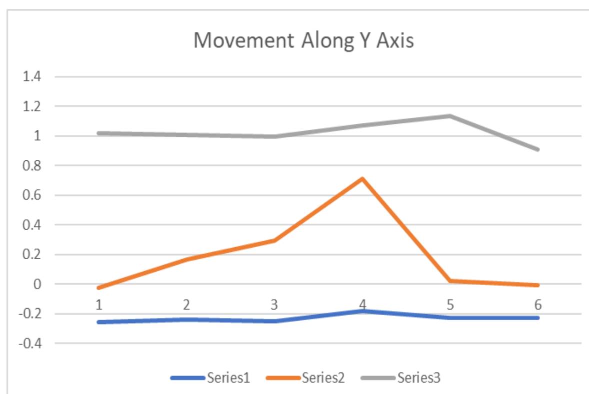


Figure 17. Y Axis Accelerometer Validation

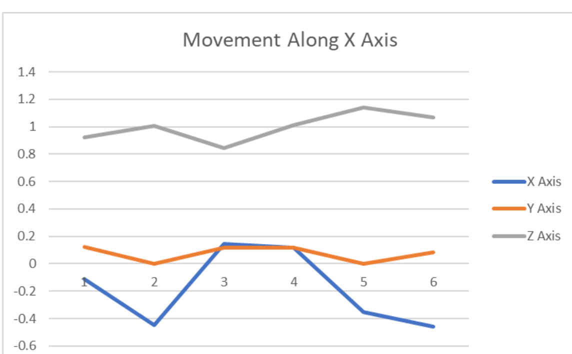


Figure 18. X Axis Accelerometer Validation

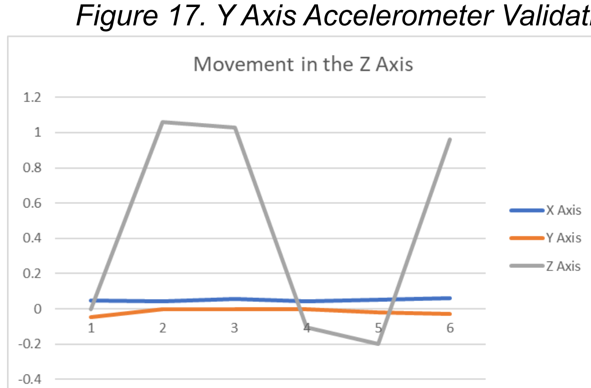


Figure 19. Z Axis Accelerometer Validation

2.3.6 Operation Robust Enclosure

The second part of the subsystem was designing a robust enclosure. Below are two different prototypes. Prototype A was the original design, which is easy to debug and add more sensors too. It was created by mounting 2 prototype boards to the Raspberry Pi. The terminal blocks were then soldered into the proto board, connecting to the IC pins and the prototype board on the left with sensors mounted onto them with standoffs. The connection was then completed by using snappable dupont wire which when fixed in place would not move and allow the circuit to be non conductive.

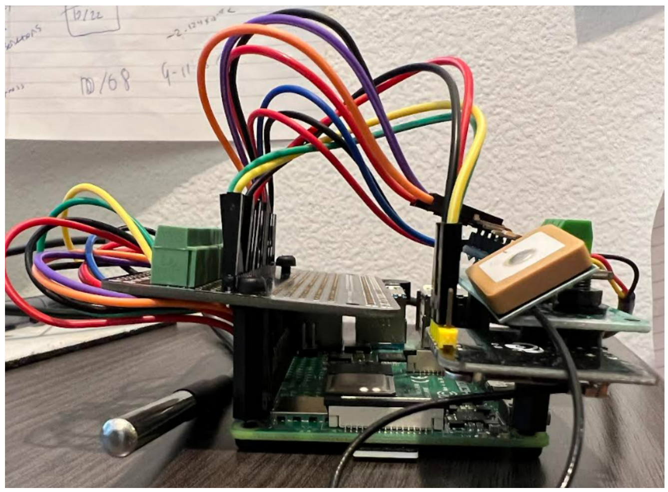
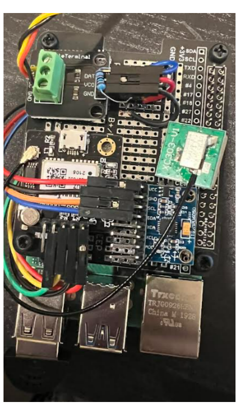


Figure 20. Prototype A Image

The second prototype was completed using only one protoboard. The advantages of this model was that it is much smaller and able to be compacted into a smaller space. This board was created by mounting the one board on top of the Raspberry Pi with the exact dimensions. The sensors are also mounted on the board with an insulator to prevent any crossover of the wires. All connection points between the sensors and wires were then soldered into place using blue wire. This model is also safe to touch without the risk of electric shock. This is the model that was used for our final design due to its compactness.



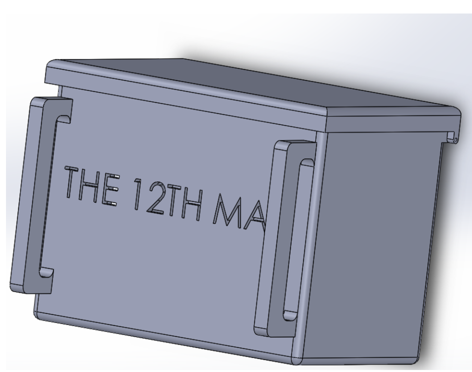


Figure 21. Prototype B Image Figure 22. Electrical Box Diagram Figure 23. 3D Print Electrical Box

Finally a 3D auto cad model was created on solidworks and printed. The prototype has a removable snap-on lid with several cutouts for the microphone, temperature probe, and speakers to perform to their optimal standards.. On the back of the device there are two loops which allows for the wearable to be mounted through the users belt loops. Once all components are placed inside the box, a snap-on lip is placed at the top so that the user can see into the device while being unable to touch the components.

2.3.6 Validation Robust Enclosure

The goal of the robust enclosure is to ensure component security, preventing user electrocution, and device protection. Through the use of dupont wires, the circuit itself is safe to touch with no open wires and the box itself also does not allow the user to access it. Below is a chart with all the specifications of our components and the box. Based on these sizes all parts fit into said enclosure.

Device	Length (mm)	Width (mm)	Height (mm)
Accelerometer	20.066	16.51	9.906
Temperature	60.96	15.24	2.54
Heart-Rate	34.036	65.034	9.906
Microphone	22.29	18.45	7.12
GPS	27.686	1.01	22.606
PCB	88.9	56.388	.5
Battery	99	67	8.1
Prototype A	111	85	45
Prototype B	85.598	56.388	39
Wearable	101.6	76.2	63.5

Table 1. Chart of Component Sizes

2.4. Subsystem Conclusion

After validation, each one of the sensors operated and performed according to the manufacturers specifications. Despite small differences between the other industry proven sensors used to validate our system, the differences were negligible. Each sensor worked generally as specified and displayed the correct trends. The ability of each sensor to interface with the microcontroller allowed the proper data files to be created and updated autonomously.

The robust enclosure part of the subsystem proved to be durable, wearable, finger safe, and capable of holding all current components necessary for our design.

3. Machine Learning

3.1. Subsystem Introduction

The machine learning subsystem will perform inference on the data collected by the Sensor/Enclosure subsystem to determine the likelihood of a meltdown. The objective of the ML subsystem is to make highly accurate predictions so that mitigation techniques can be initiated when necessary, and so parents/guardians can always be aware when their dependent is having an episode. In order to accomplish this complex task, extensive interviews were conducted with specialists from A&M's department of psychology and special education.

3.2. Subsystem Details

A high level overview of the ML subsystem, which features two neural networks working in tandem, is loosely depicted below. Both of these networks were created using the Tensorflow Keras library.

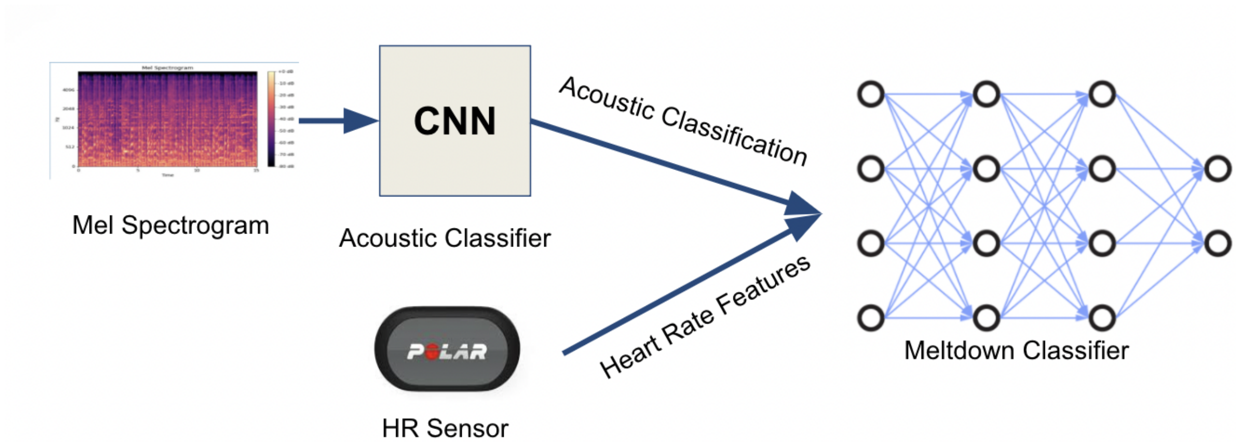


Figure 24. ML Subsystem - High Level Overview

The first network is an acoustic classifier model based on a convolutional neural network (CNN). The premise of this network is to “listen” to the environment of the individual wearing the device, and to make a prediction about what type of environment they are currently in. The eight categories are as follows: crowd conversations, large conversations, small conversations, quiet environments, stimulative environments, noisy environments, windy environments, and unknown environments. By inferring the acoustic environment class, we can get an idea of how stressful/stimulative the world surrounding the wearer is at any given moment.

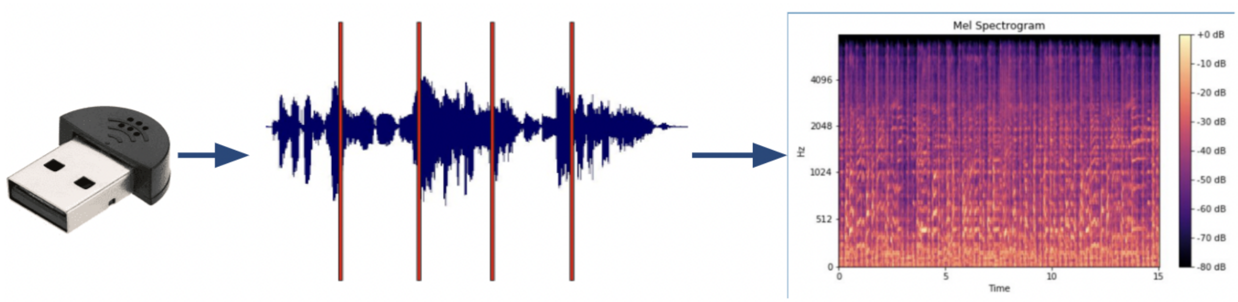


Figure 25. Audio Data Preprocessing

Prior to making any predictions, audio recordings from the sensor subsystem need to be preprocessed. Each .wav file is cut into one and a half second segments before being converted into the frequency domain via the short time Fourier transform. The spectrograms undergo one final preprocessing step, in which they are converted into Mel Spectrograms. This step helps the CNN interpret sounds as a human would, associating pitch values based on the frequency components of the signal. All of the digital signal processing was accomplished using the Python Librosa library.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
rescaling (Rescaling)        (None, 314, 235, 3)       0
_________________________________________________________________
conv2d (Conv2D)              (None, 314, 235, 64)      1792
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 104, 78, 64)       0
_________________________________________________________________
batch_normalization (BatchNo (None, 104, 78, 64)       256
_________________________________________________________________
dense (Dense)                (None, 104, 78, 256)      16640
_________________________________________________________________
dropout (Dropout)            (None, 104, 78, 256)      0
_________________________________________________________________
dense_1 (Dense)              (None, 104, 78, 128)      32896
_________________________________________________________________
flatten (Flatten)            (None, 1038336)           0
_________________________________________________________________
dense_2 (Dense)              (None, 8)                 8306696
=================================================================
Total params: 8,358,280
```

Figure 26. Output of model.summary() call

The model architecture of the CNN is illustrated in the model summary output above, and is depicted loosely in the CNN diagram below. Inputs to the network are Mel Spectrograms, which can be thought of as 2D images. Rescaling is applied at the input to ensure that each input is of size (314, 235). Next, the inputs are passed through a Convolutional 2D layer and a subsequent MaxPooling layer before entering a Batch Normalization layer. These three layers work to extract features from the input in the form of feature maps. Once these features have been extracted, several fully connected layers are used to make a classification. Within the fully connected layers exist a dropout layer and a flattening layer. A sample output is also shown below.

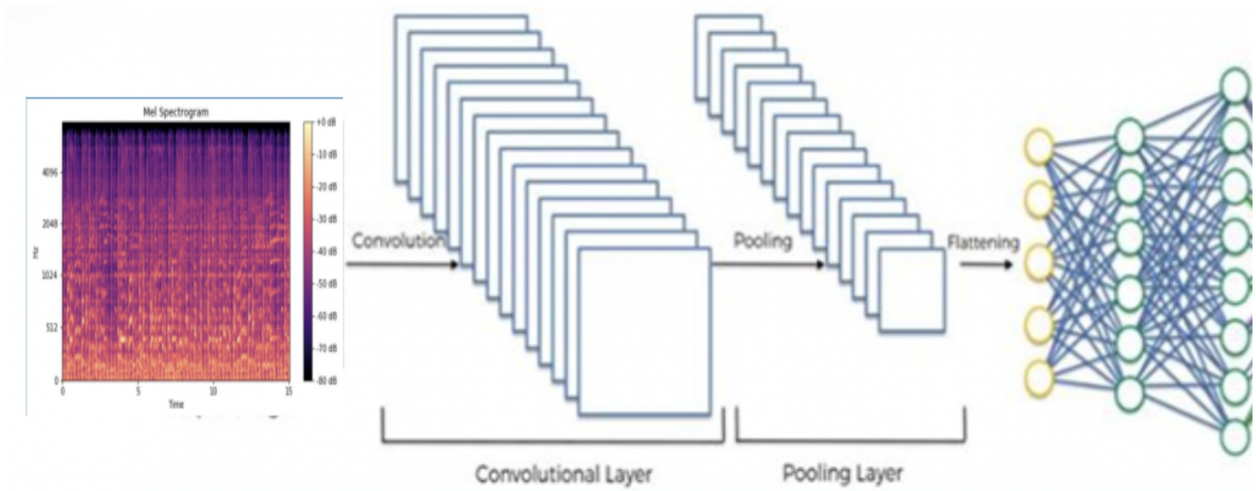


Figure 27. CNN - High Level Overview

```
Crowd_Conversation:tf.Tensor(0.010878092, shape=(), dtype=float32)
Large_Conversation:tf.Tensor(0.988926, shape=(), dtype=float32)
Noise:tf.Tensor(1.6076714e-10, shape=(), dtype=float32)
Quiet:tf.Tensor(3.0713311e-16, shape=(), dtype=float32)
Small_Conversation:tf.Tensor(0.000120142584, shape=(), dtype=float32)
Stimuli:tf.Tensor(7.57578e-05, shape=(), dtype=float32)
Unknown:tf.Tensor(0.0, shape=(), dtype=float32)
Wind:tf.Tensor(1.4008635e-15, shape=(), dtype=float32)
```

Figure 28. Sample Output of CNN

While the audio data can be used to ascertain the external stressors/stimuli present, the Sensor subsystem also collects biometrics which can be used to determine the physical stress the individual undergoes. The meltdown predictions are done by feeding several biometric features and the acoustic class outputted by the acoustic classifier model into another neural network. An output from the network is a probability bounded between [0,1] that represents the likelihood of a meltdown. The goal of the subsystem is to consistently make predictions based on the data collected by the Sensor subsystem so that the meltdown status individual can be continuously monitored without a human present.

```
print(df.head()) # Visualize dataset

   Label  Avg. HR  Final - Initial HR  Acoustic Class
0      0     74.1                   4               3
1      0     82.6                   1               6
2      0     83.1                   2               1
3      1     90.7                  15               1
4      1    110.4                   6               0
```

Figure 29. Dataset Visualization using Pandas

Due to certain ethical and health standards, biometric data was not able to be collected from actual human subjects. Additionally, there is virtually zero data on austistic meltdowns available. Thus, a synthetic dataset, which was modeled off of panic attacks and the information gathered from interviewing specialists, was created in order to train the meltdown classifier. A label of 0 indicates a non-meltdown scenario, while a label of 1 indicates a meltdown. The biometric features used for training the network are the average heart rate measured, and the difference between the first heart rate value measured and the last heart rate value measured over one data collection cycle. One data collection cycle is of duration 30 seconds and occurs every minute to continuously monitor the individual.

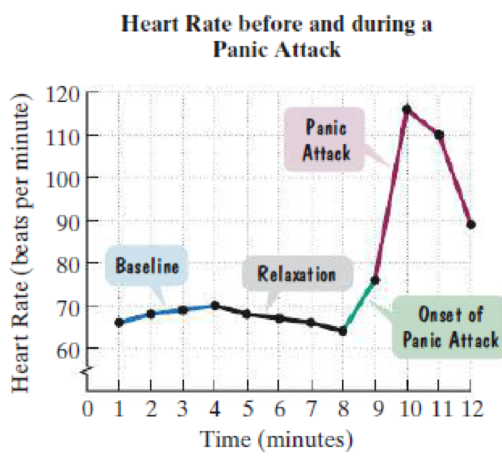


Figure 30. Heart Rate Trend during Panic Attacks

Average heart rate was chosen as a feature because an anomalously elevated average heart rate can indicate a stressful state. Individuals with ASD tend to have slightly higher average heart rates, thus this was accounted for when creating the dataset so that it could generalize to many individuals with ASD. However, averages have a tendency to be deceptive, therefore, it was decided that another feature was necessary. By computing the difference between the heart rate measured at the beginning and the end of one data collection cycle, we could ascertain whether the heart rate was trending upwards, indicating a stressful state. While someone might exhibit a relatively normal average heart rate (due to the deceptive nature of averaging), their heart rate trending upwards at an anomalous speed is an indication that they are in a state of distress. The inspiration for this came from studying heart rate trends during the course of a panic attack, which can be seen in the figure above. In addition, the acoustic class of the environment outputted by the acoustic classifier CNN model was utilized as a feature.

Once again, it was necessary to aggregate and preprocess that data before feeding it into the neural network. While our subsystems are yet to be interfaced, the time series heart rate data from the sensor subsystem will be aggregated via averages and final minus initial computations. To help the network train more efficiently, all continuous numerical data was normalized using Min-Max scaling technique. The highest average heart rate included in the dataset was 200 BPM, while the highest change in heart rate over a 30 second interval was 28 BPM. These values are extremum observed in panic attack scenarios. Furthermore, the acoustic classes (crowd conversation, noisy environment, etc) are categorical data that can be represented by a number 0-7. In order to avoid biasing the network, all acoustic classes were converted from base 10 values (0-7) to base 2 one hot encoded 8-bit strings. Preprocessing thus ensured that all inputs to the network were between [0, 1]. All of the data wrangling and preprocessing was accomplished using the Pandas library.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Apply for Avg. HR and (HR_F - HR_I)

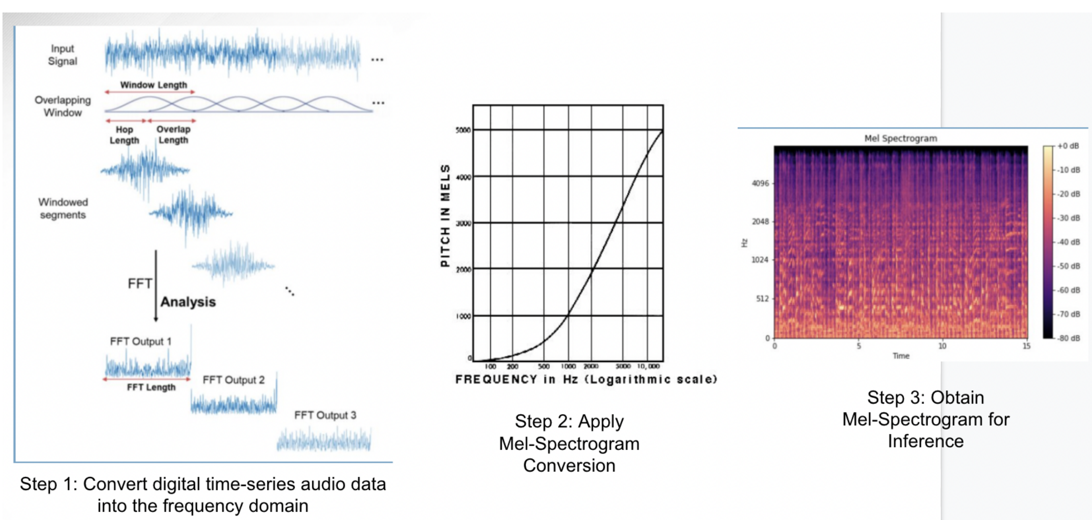


Figure 31. Meltdown Data Preprocessing

Finally, the meltdown classifier neural network is composed of two fully connected hidden layers and an output layer featuring a sigmoid activation function. This sigmoid activation function was used to scale the singular output of the network to a value between [0,1], representing the probability of a meltdown occurring based on the data collected by the sensor subsystem. All of the code used for data preprocessing and neural network training and validation can be found in Appendix A.

3.3. Validation

The meltdown classifier was trained and validated using a train-test split of the synthetic dataset, in which 80% of the data was used to train on, while the other 20% was used to validate. The supervised learning process utilized binary cross-entropy due to the binary nature of the labels (meltdown vs. non-meltdown), the Adam optimizer with a learning rate of .001, and a batch size of 1. Experimentation with the number of hidden layers, number of neurons per layer, and the number of epochs was conducted in order to create the best model possible. Tensorboard, a utility provided by Tensorflow, was used to visualize the progression of the training/validation accuracy/loss over the course of the epochs. The number of epochs (12) was chosen such that the network achieved the best performance possible without overfitting to the data.

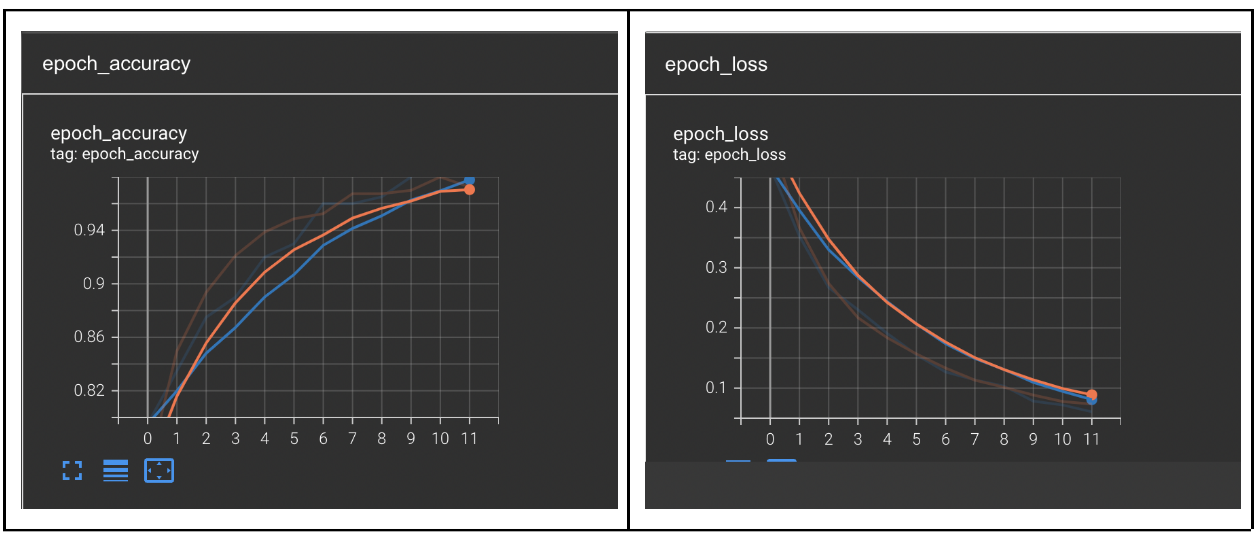


Figure 32. Meltdown Classifier Tensorboard Visualization (Note: Orange=Train, Blue=Test)

The final accuracy of the meltdown classifier network on the validation dataset plateaued at around 97%. Furthermore, for the lab demonstration, 50+ test cases were created in order to further validate the performance of the network. In each of the test cases, the network correctly identified the meltdown and non-meltdown scenarios, outputting probabilities that align with what I personally expected to see given the inputs.

For the acoustic classifier CNN, sparse categorical cross-entropy was used as the loss function in conjunction with the Adam optimizer with a learning rate of .0001 and a batch size of 64. The acoustic classifier CNN model was validated on audio files that were provided by the sponsor, MarkusAI. These audio recordings were unlabeled, and thus the categories to which they were assigned was subject to my discretion. A Confusion Matrix was created in order to demonstrate the accuracy of predictions made by the CNN network. This figure, which is shown below, depicts the percentage of correct classifications, as well as the percentage of misclassifications for each category. Furthermore, it provides insight into specifically what categories are predicted when the model makes an incorrect classification.

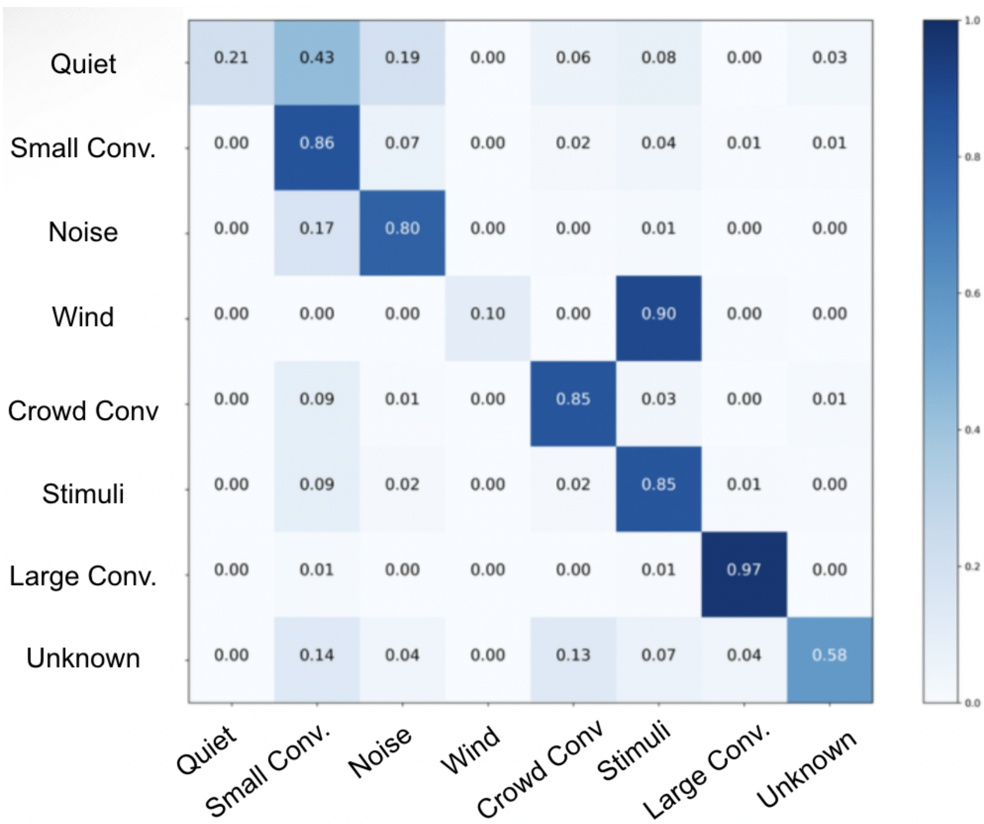


Figure 33. CNN Confusion Matrix

As is evident from the figure, there is still some work to be done for the acoustic classifier. Currently, the issues with classification and the potential repercussions associated with misclassification can be summarized as follows:

I. Quiet environments are often classified as small conversations. This may be due to the fact that both environments are characterized by low volume. While misclassification is happening, it is not worrisome, as both of these environments are low stress, and should thus not greatly influence meltdown prediction.

II. The network is highly susceptible to wind. Misclassification as a stimuli may be due to the fact that the majority of audio recordings of wind are reminiscent of the loud windy sound that can be heard while driving in a car with only one the back windows rolled down. While this undoubtedly should be classified as wind, it is possible that the network is classifying it as a stimulus event because the sound is extremely loud and unpleasant, and has the potential to set off an individual with ASD. Thus, while misclassification is occurring, it could potentially work in our favor.
III. Unknown environments are misclassified into a plethora of different categories. This can be associated with the fact that the labeling process is very subjective. Overall, this should not be that big of an issue, as the neural network usually always outputs one of the seven other classes when making a prediction.

3.4. Subsystem Conclusion

After validation, the subsystem is functioning properly and is meeting expectations. The meltdown classifier proved to perform extremely well. Meltdown scenarios were classified with extremely high accuracy, and predictions exhibited high levels of confidence in classification. Furthermore, the acoustic classifier did a satisfactory job of classifying the environments, making incorrect predictions occasionally. When interfaced with the other subsystems, it will enable the system to accurately predict meltdown scenarios so that parents/guardians can be alerted and meltdown mitigation procedures can be initiated for the individual with ASD.

4. Power System

4.1. Subsystem Introduction

The power system is designed in order to supply power to the Raspberry Pi 4b in which the sensors will be powered by the Raspberry Pi 4b. The power system consists of a printed circuit board that contains a battery charger integrated circuit, boost converter integrated circuit, and a P-Channel MOSFET. The printed circuit board is powered using a lithium ion battery as well as a USB-C input to charge the battery. The printed circuit board shall go inside the casing along with the battery and microcontroller when interfacing with the other subsystems.

4.2. Subsystem Details

Below is a block diagram of the printed circuit board.

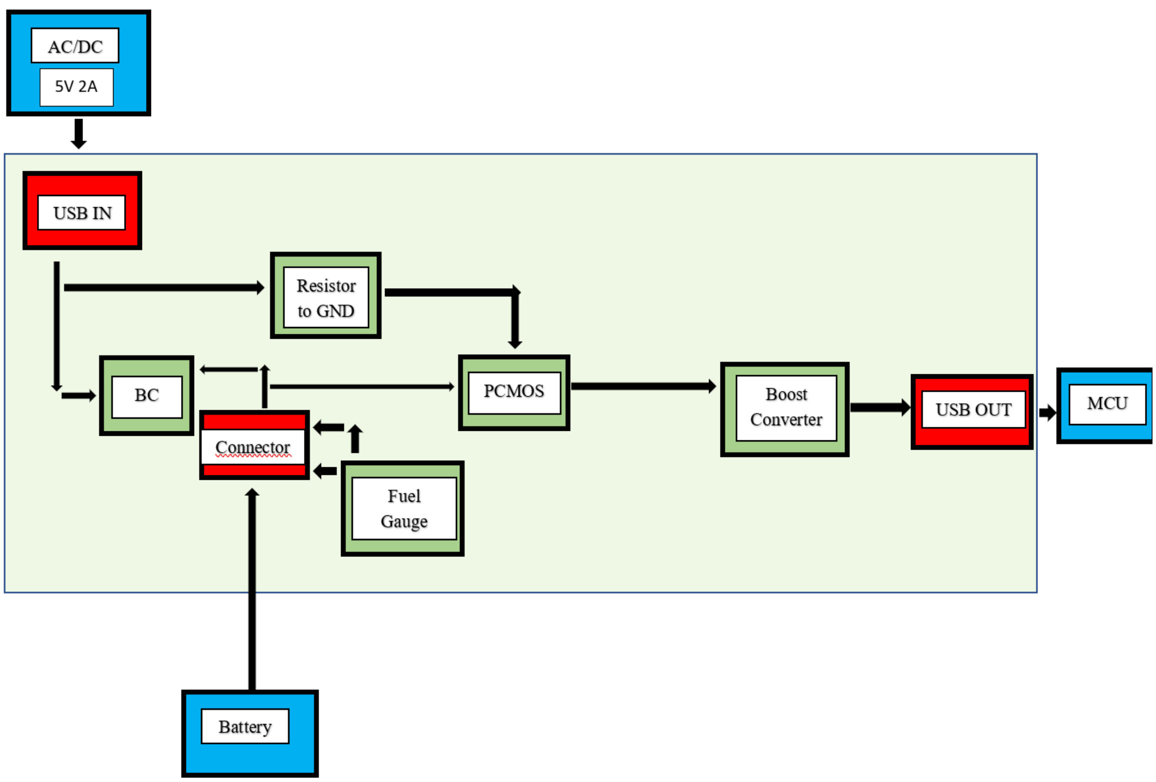


Figure 34. Printed Circuit Board Overview

The power system is designed for a 3.7V 6000mAh to supply power to the board in which the voltage shall be stepped up to 5V using a boost converter integrated circuit and thus supply power to the microcontroller with a USB-C connector. The battery charger shall be recharged at

4.2V and 2A using a battery charger integrated circuit and a USB-C connector for an input of voltage and current from a wall adapter. The MOSFET with a pull-down resistor is used as a load switch application. When voltage is applied to the gate of the MOSFET, voltage coming from the USB-C input, the MOSFET will not allow conductivity and thus the load will not draw current and the device will be in an off-state. In contrast, when there is zero volts at the gate of the MOSFET, the MOSFET will allow conductivity and the load will draw current and the device will be in an on-state. This load switch application of the MOSFET is applied in order to turn off the device when the battery is recharging so the battery can recharge efficiently without the load drawing current. The user would not need to use the device while the device is recharging.

4.3. Battery Charger

4.3.1. Design

Below is the schematic of the battery charger integrated circuit and the USB-C Input.

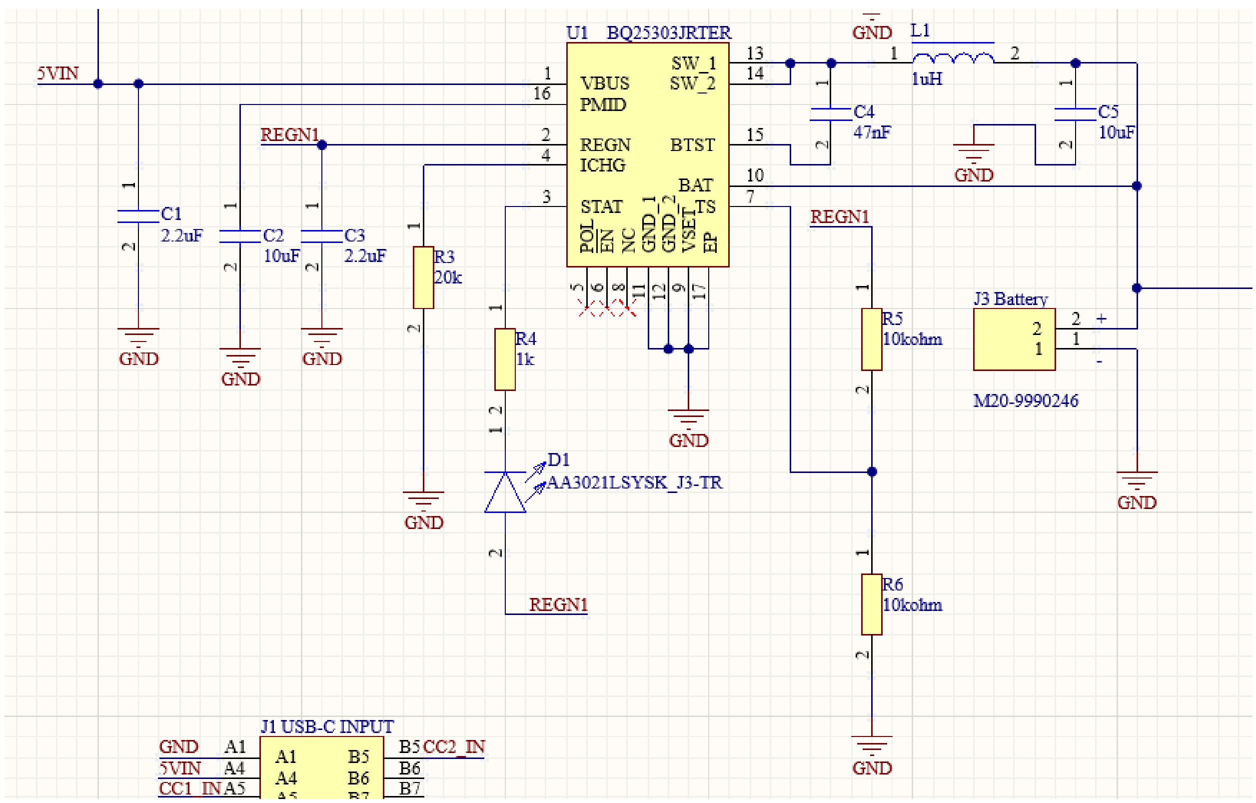


Figure 35.Battery Charger Schematic BQ25303JRTER

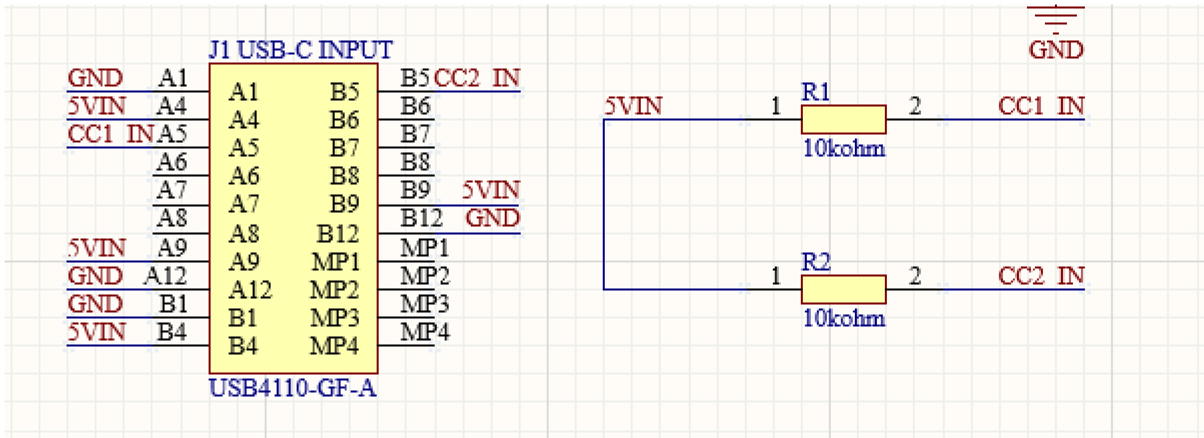


Figure 36. USB-C Input Schematic

The lithium-ion battery is connected to the battery charger using a header and pins that are represented as J3 on the schematic. The positive end of the battery connects to the BAT pin of the battery charger circuit and the negative terminal connects to ground. Utilizing a USB-C connector, CC1 and CC2 were configured to withstand an input voltage of 5V and 2A by choosing the resistor values to be 10k ohms for the upstream facing port as this usb-c would sink power from the wall outlet for the battery charger to operate. The Vin pins of the USB-C are connected to the Vbus pin of the battery charger integrated circuit. The input current and charging current were programmed using the equation Ichg(A) = Kchg(AΩ)/RICHG(Ω). Knowing that Kchg is typically 40,000(AΩ) from the characteristic table and 2A is the ideal input current, a resistor of 20kΩ was chosen to placed coming out of the ICHG pin of the integrated circuit. An inductor value of 1uH was chosen to be placed coming out of the switching node pins due to the Vbus max having a voltage less than 6.2V. The inductor saturation current should be higher than the ICHG current plus half of the ripple current so the ripple current was calculated using the known input voltage of 5V and the duty cycle in which the duty cycle is calculated by dividing the battery voltage over the input voltage. The ripple current was calculated to be 0.56A so the saturation current should be greater than 2.28 amps. The capacitor RMS input and output current ICIN and ICOUT was also calculated to be 1.35A and 0.1624A to size the input and output capacitors properly. An effective capacitance of 14.4uF was chosen to be at the input and an effective capacitance of 10uF was chosen to be at the output as well as a bootstrap capacitor of 0.047uF from the BTST to the SW pins. To set the charging voltage to 4.2V, I shorted the Vset pin to ground as the battery is recommended to be charged at 4.2V from the battery's datasheet. A status indicator was implemented to indicate the battery charging. To implement this, a diode was placed out of the STAT pin as well as a current limiting resistor of 1kΩ. The current limiting resistor was calculated by sizing the forward current properly for the diode to operate. The anode and cathode placement of the diode is important as the cathode would need to face the STAT pin. The diode would have a voltage drop of 2V and the current limiting resistor would have a voltage drop of 3V as the bus would have a voltage of 5V coming from the

FET driver supply output of the REGN pin.. The forward current that is aimed to achieve is 3mA for the diode to operate so utilizing 1kΩ current limiting resistor would enable a current of 3mA to flow after calculating with ohm's law. The thermal pad is utilized to provide an electrical ground connection. The POL and EN pins are floating to enable the integrated circuit to operate when the Vbus has voltage. I programmed the TS pin to have a voltage divider with two 10kΩ resistors. Below is the PCB layout design of the battery charger integrated circuit and the USB-C input.

Below is the PCB Layout of the battery charger integrated circuit and the USB-C Input.

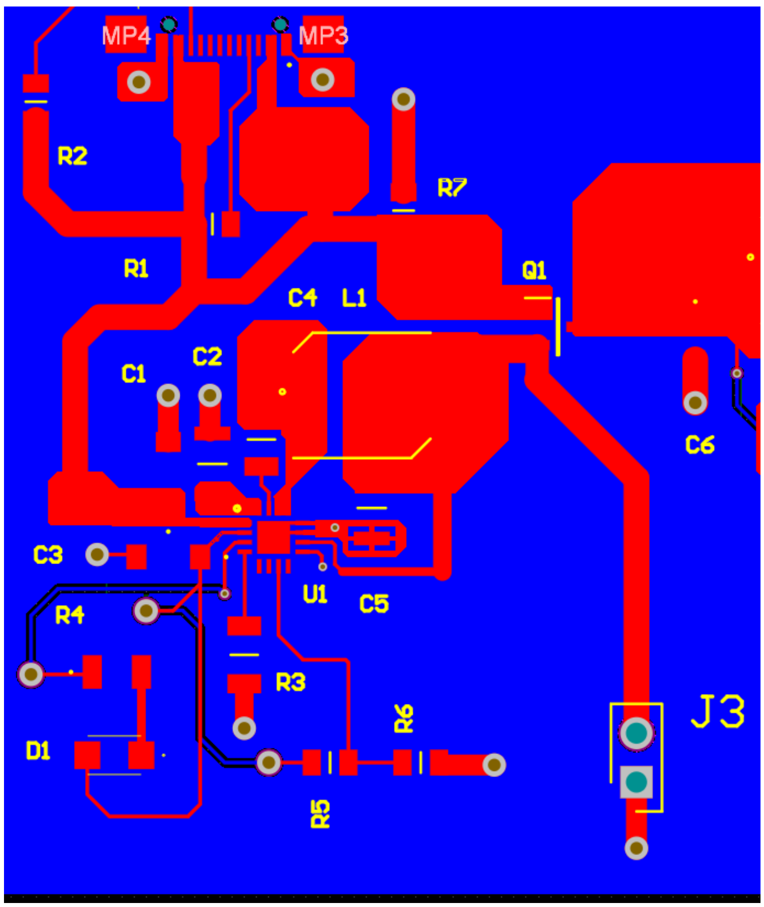


Figure 37. PCB Layout for Battery Charger and USB-C Input

The traces of the USB-C input to the battery charger and the battery charger to the battery were properly sized to handle up to 3A. All power pins were sized to handle 3A and analog pins were sized much smaller to be effective as well as to conserve space in the layout design. Input and output capacitors were placed as close as possible to the battery charger integrated circuit for effectiveness. Polygon pours were utilized as much as possible with the idea of having no ninety degree corners to prevent failure. This board has two main layers as the first layer is dedicated for mostly power and analog lines. The second layer is dedicated to a ground polygon pour.

4.4. Boost Converter

4.4.1. Design

Below is the schematic of the boost converter and USB-C output.

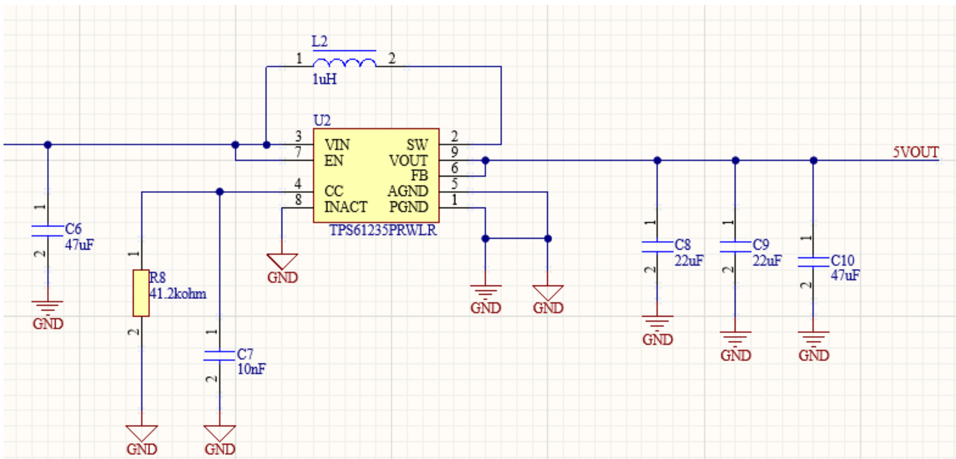


Figure 38. Boost Converter Schematic TPS61235PRWLR

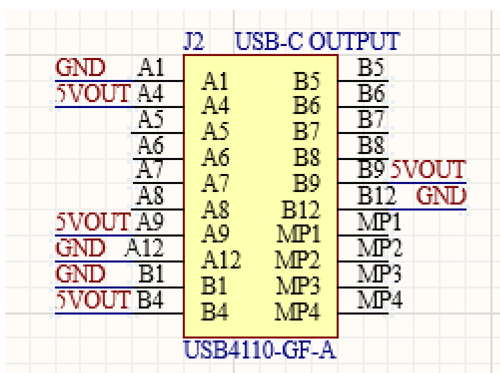


Figure 39. USB-C Output Schematic

The input of the boost converter comes from the positive terminal of the lithium-ion battery. The boost converter integrated circuit is programmed to have an output voltage 5.1V with the feedback pin, FB, connected to the output voltage pin, VOUT.. The effective input capacitance is chosen to be 47uF. The effective output capacitance is chosen to be 97uF. The enable pin is tied to the input pin of VIN so that the boost converter is operating when there is an input voltage detected. An inductor value of 1uH is chosen to be placed from the VIN to the switching node pin, SW. The value of 1uH is chosen to minimize control loop instability. To find the maximum output current capability, the duty cycle was calculated using the worst case input voltage and output voltage. The duty cycle was calculated to be 0.3833. The inductor ripple current was then calculated using the known inductance 1uH, 85% efficiency estimation, and switching frequency value of 1250kHz in which the value of 1.1346A was achieved. The peak current and input average current was calculated to ensure the inductor's saturation and RMS current was sized properly. This was done utilizing the equation below in which a peak current of 5.81A and an input average current of 5.2427A was calculated. The saturation current was at least 20-30% more than the peak current and the RMS current should be higher than the average input current.

$$I_{L\_peak} = I_{IN\_avg} + \frac{\Delta I_L}{2} = \frac{I_{OUT}}{1-D} + \frac{V_{IN} \cdot D}{2 \cdot L \cdot f_{sw}}$$

With the switch valley current limit being 6.5A, the max output current is calculated by using the equation below.

$$I_{OUT\_MAX} = (1-D) \cdot (I_{LIM} + \frac{\Delta I_L}{2})$$

The maximum output current calculated is 4.3587A. The maximum current that is ideal is 3A so a constant current output was programmed to be at 3A utilizing at 41.2kΩ and a 10uF capacitor in parallel coming out of the CC pin. The resistor was calculated using the voltage reference inside of the integrated circuit. The output of the boost converter is engineered to have an output of 5.1V and sized to withstand up to 3A. The analog and power grounds were connected separately but one single connection to ensure the same potential between them. The output of the boost converter is connected to the power pins of the USB-C output connector

Below is the PCB Layout of the boost converter integrated circuit and the USB-C output.

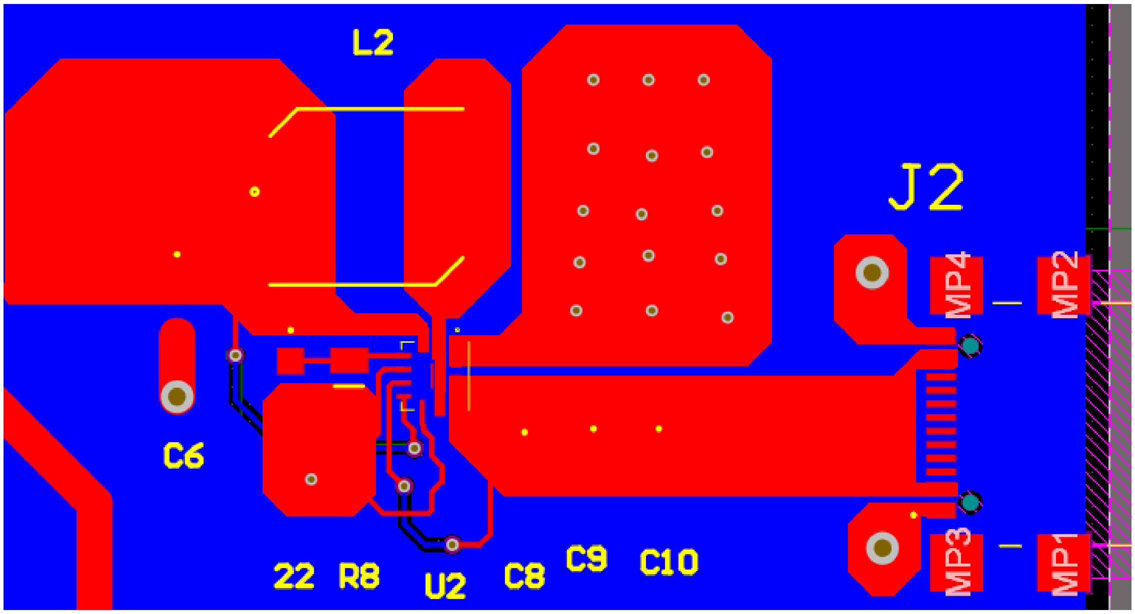


Figure 40. PCB Layout for Boost Converter and USB-C Output

The power traces were sized to handle at least 6A for a worst case scenario. The power and analog grounds were separated and connected with a trace to act as a single point connection. The boost converter is connected to the USB-C using the power pins. A polygon pour was implemented using multiple vias to cover a significant amount of area on the first layer for power ground.

4.6. Mosfet

4.6.1. Design

The mosfet was designed in order to withstand current and perform at the operating voltage. The gate is connected to the input of the USB-C, the source is connected to the battery, and the drain is connected to the input of the boost converter. The mosfet was chosen to have a drain-source breakdown voltage of 30V, continuous drain current of 7.6A, gate-source threshold voltage of 2.5V, drain-source resistance of 29 mOhms, power dissipation of 2.5W, and a gate-source voltage of -20V, +20V. These characteristics were sufficient as the MOSFET is calculated to handle 3.7-3.9V coming from the battery and at most 3A drawn to the output of the boost converter for operation purposes. The mosfet was chosen to be a P-Channel MOSFET with one channel. There is a pull-down resistor at the gate of the MOSFET with a value of 10kΩ for it to be pulled down when there is no power supply coming from the USB-C input so that the MOSFET will allow current to conduct through it. Calculations and simulations were conducted to ensure the MOSFET would function properly with the overall power system design. The MOSFET was simulated to ensure that the idea of 0V applied to the gate vs 5V applied to the gate would make the load draw current and not draw current respectively. Zero volts at the gate represent when the battery is not charging and thus the load will not draw current. Five volts applied at the gate simulates the battery being recharged and thus the load will not draw current. Below are the figures representing when there is 0V applied to the gate and 5V applied to the gate.

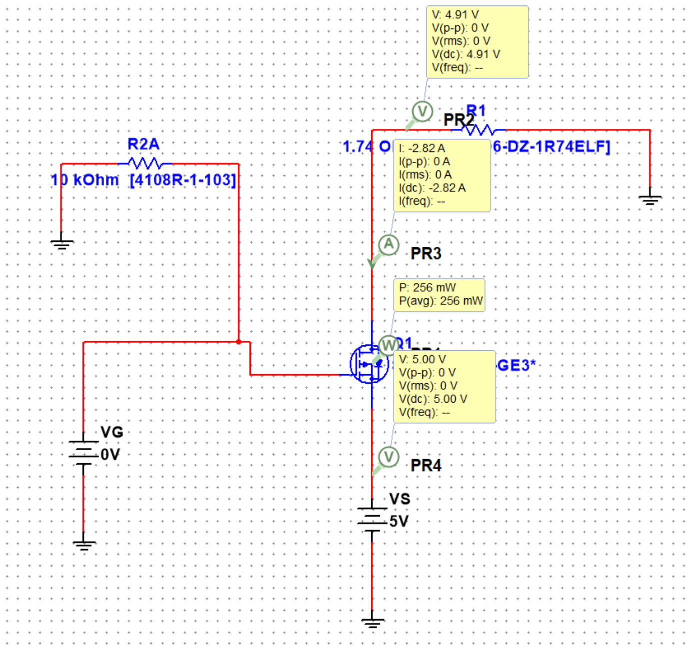


Figure 45. Zero Gate Voltage MOSFET

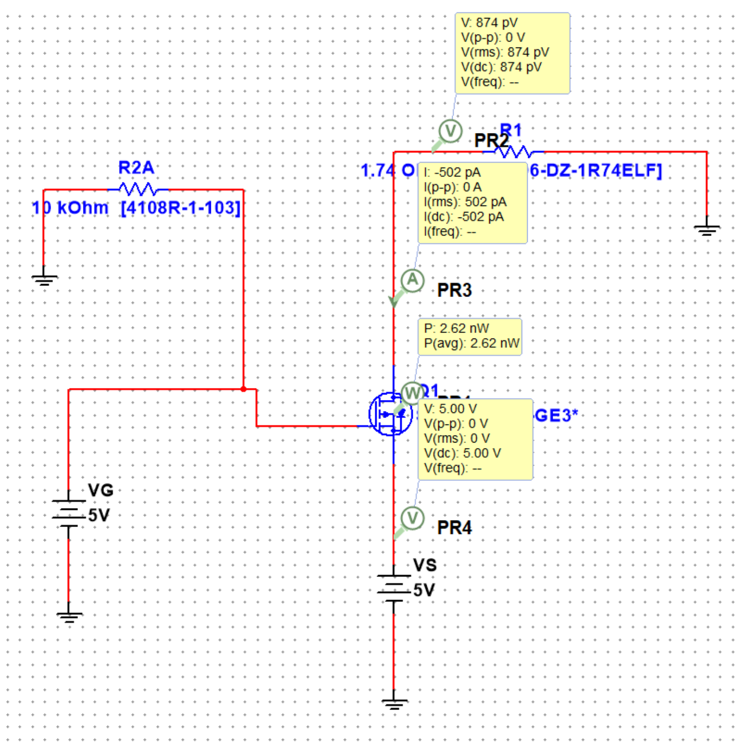


Figure 46. Five Volts Gate MOSFET

The resistance on the drain is to ideally draw 3A when being able to draw current from the source. From Figure 45, it can be seen that when 0V volts is applied to the gate, the load will draw current and there will be power dissipated at the resistor. From Figure 46, it can be seen that when 5V is applied to the gate, the load is not drawing current thus there is zero power being dissipated by the resistor. This shows that the load is ideally going to be turned off when the battery is recharging and the load is going to be powered when the battery is not recharging.

The next step was to calculate the power dissipation of the MOSFET to ensure it does not exceed 2.5W. By using the power dissipation formula P = I^2(R) and replacing I with the current draw of 3A and R with the drain-source resistance, I was able to calculate a power dissipation of 0.261W in which it is lower than 2.5W. Using the equation P=V^2/R, we can expect the voltage drop to be 0.087V at the MOSFET. Note that although voltage decreases coming from the source to the drain, the boost converter is designed to have an input in the range of 2.7V-4.35V.

Below is the PCB Layout of the MOSFET.

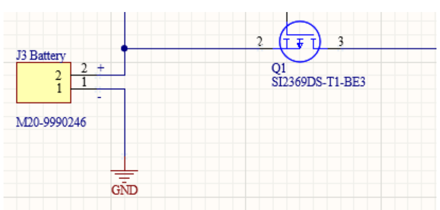


Figure 47. MOSFET Schematic

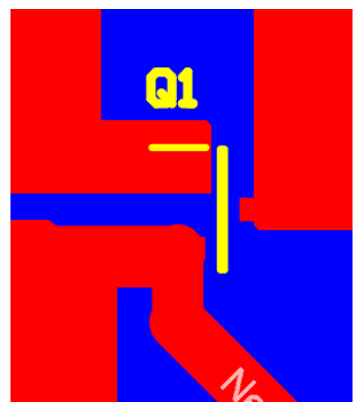


Figure 48. PCB Layout for MOSFET

The MOSFET input and output traces were designed to withstand current and voltage to their respective expected input and output current at the drain, source, and gate of the MOSFET.

4.7. PCB Design//Validation

4.7.1. Design

Below is the schematic capture of the full printed circuit board design.

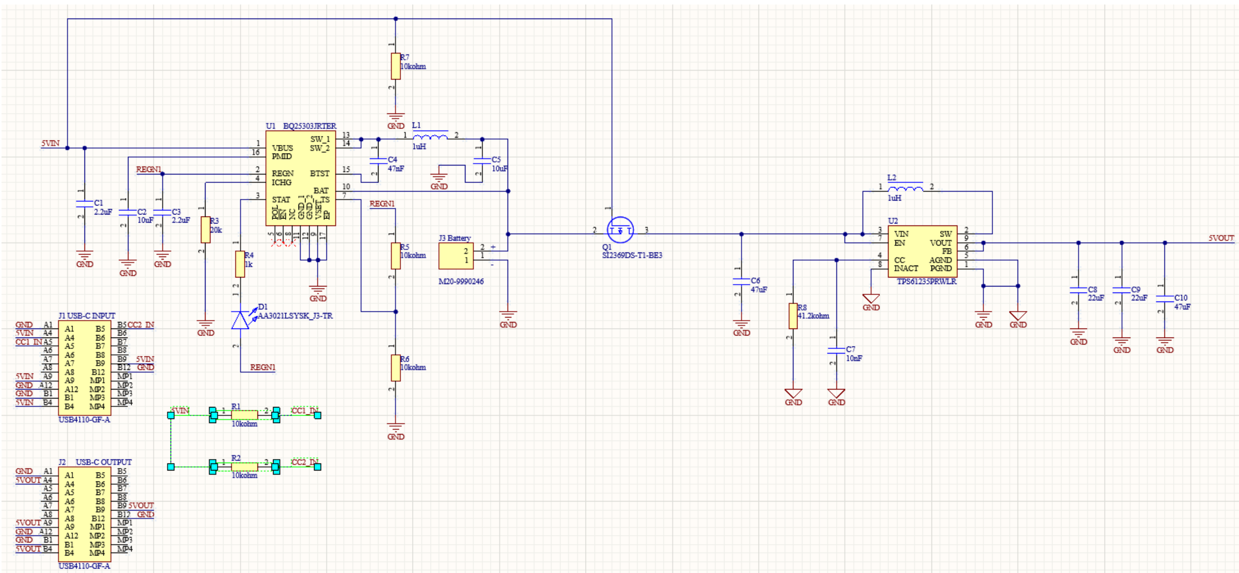


Figure 50. Full Schematic Capture of PCB

Below is the PCB Layout.

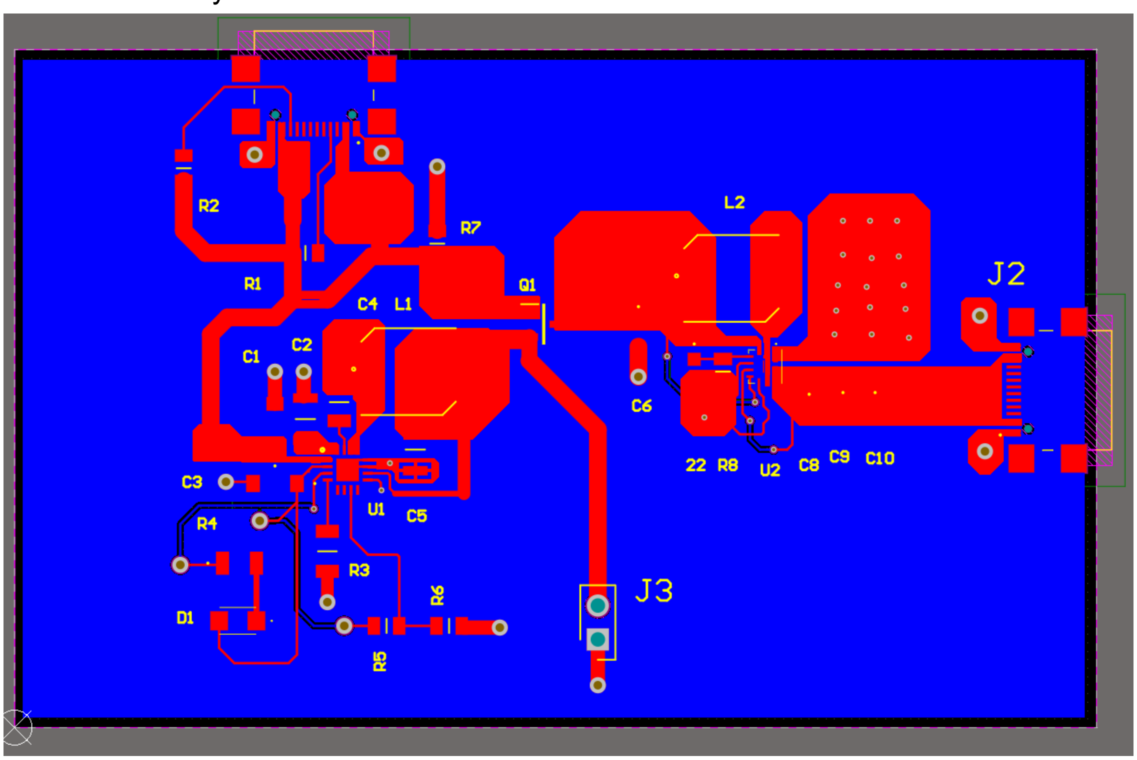


Figure 51. PCB Layout for PCB

Below is the 3D model of the printed circuit board.

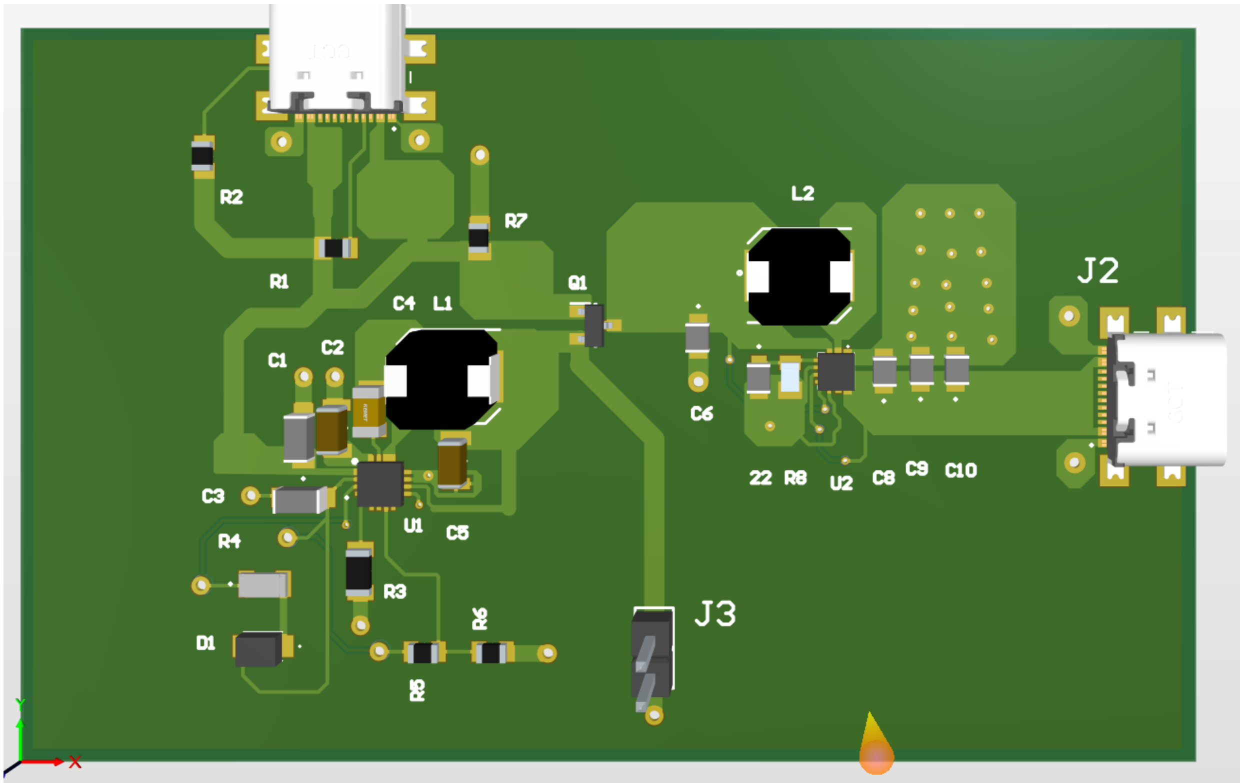


Figure 52. 3D Model of PCB

Below is the PCB on hand after individually soldering all the components.

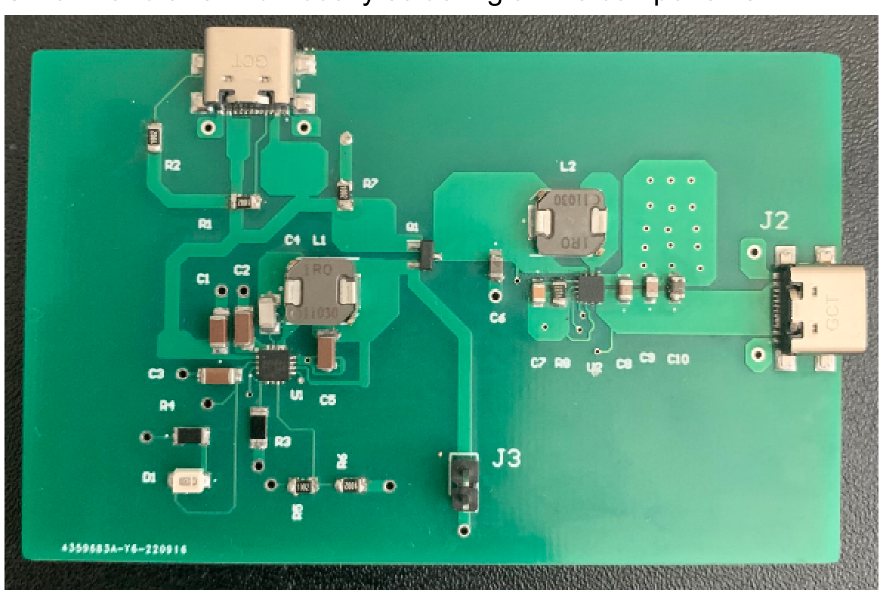


Figure 53. On Hand PCB

4.7.2. Validation

All connections on the board were tested for continuity as well as ensuring the resistors and capacitors were of their true values using a multimeter.

4.7.2.1 Battery Charger Validation

The input voltage and input current coming into the USB-C connector from a 5V/3A adapter on the wall was tested using a digital multimeter tester. This test is ideal to make sure the input voltage and input current are satisfactory for the battery charger to operate. The test method is ideal as the digital multimeter tester has a USB-C adapter and is connected in series to the input from the wall. The battery charging voltage was tested using a voltage meter. The battery charging current was tested using a digital clamp meter.

4.7.2.1 Boost Converter Validation

4.7.2.1 MOSFET Validation

4.8. Subsystem Conclusion

The battery charger, boost converter, and load switch method are all fully functional. The input voltage and current from the USB-C are as expected for the battery charging integrated circuit and the p-channel MOSFET to operate as expected. The input voltage and current from the battery is connected to the boost converter properly to efficiently step up the voltage to 5V and power the load up to 3A at around 5-5.15V. The load switch method works as intended as there is no power being drawn on the drain side of the MOSFET when the battery is recharging. The battery charges at values that are expected. There is sufficient battery life that is more than five hours. This subsystem will enable our device to be portable for the user to have the ability to wear around their waist.

5. Database and Application Subsystem Report

5.1 Subsystem Introduction

The database and application subsystem will store all the data sent by the 12th man device and display the wearer's location and status. The main user of this subsystem will be the parents or guardians of the person with autism. With the app the guardian will be able to have the constant location of the wearer and if they are about to have a meltdown. This subsystem is meant to provide a sense of comfort for the guardians, so they will always know what is going on with their dependent without having to physically be there.

5.2 Subsystem Overview

The subsystem works first when the 12th man device sends its data to the database. The data will be sent and organized into specific tables to be used by the app later. The next part of the system is the Lambda function. The Lambda function, hosted on AWS, will be the connection between the database and the app. It has calls that will pull data from the database whenever the app needs that information. The application is the last part of the system. The application will be run on Android phones. It shows the map and the status of the device wearer. The application will be constantly updated by making calls to the API every thirty seconds, to get accurate data. This overview can be seen in figure 54.

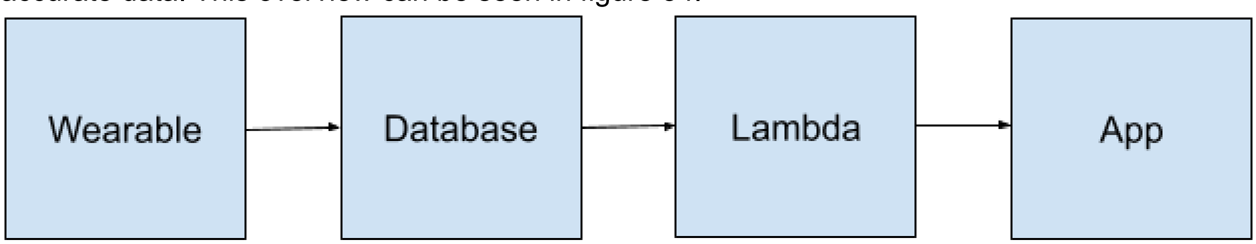


Figure 54. Database and App Layout

5.3 Database

5.3.1 Details

The database to store the devices data was created using Dynamodb. It is currently being hosted on Amazon Web Services (AWS). This allows the database to be accessible at all times, meaning that the application will have access to up to date information. The database has tables for all of the data sent from the device including acceleration, AI prediction, GPS coordinates, heartbeat, and temperature. Each table will hold the necessary data for each category as outlined in table 3. Each column will hold a VARCHAR(25) or a variable character of at most 25 characters long. Using this variable I can hold any type of data sent by the device. The variable type can be changed later in the API or application if needed.

Table Name	**Column 1**	**Column 2**	**Column 3**
Acceleration	X-axis	Y-axis	Z-axis
AI Prediction	Percentage		
GPS Coordinates	Longitude	Latitude	
Heatbest	BPM		
Temperature	Fahrenheit		

Table 3. Database Entities

An API was also created to make calls to the database. The API is coded in javascript and runs using Node.js. It will connect to the database once the application is open using Express, a back-end framework for Node.js. The Express framework is used to connect the application and the database. The API has calls that will pull data from the database and calls to send data to the frontend of the application.

5.3.2 Validation

To validate the database and API I tested sending data between the two systems. I created calls in the API that would add longitude and latitude coordinates into the GPS coordinates. This test worked and I was able to see the database updated with new coordinates. I also tested pulling that data from the database to the API. Once the data was collected from the database it was printed to the console, as seen in figure 55.

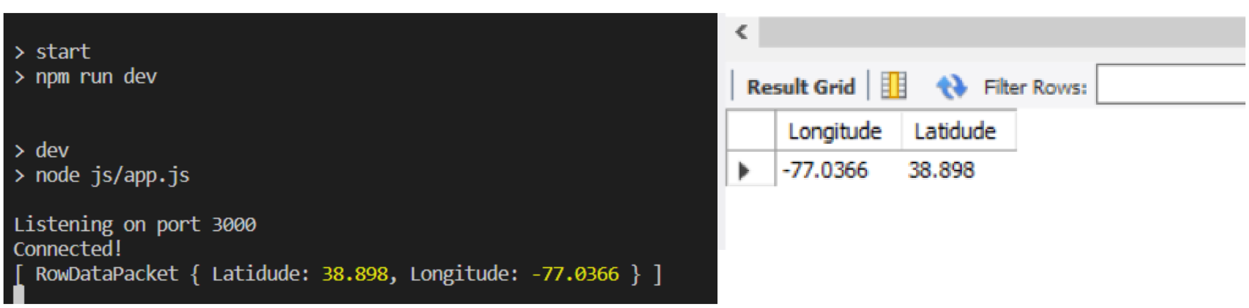


Figure 55. API Call and Database Data

5.4 Application

5.4.1 Details

The app I created using ejs, a JavaScript language that lets the user generate HTML, CSS and js. It is being hosted by Heroku and can be accessed using a link. The website is accessible by both desktop and mobile devices as seen in figure 56. Once the user opens the website on their Android mobile device they can add the app to their home screen and use it like any other app.

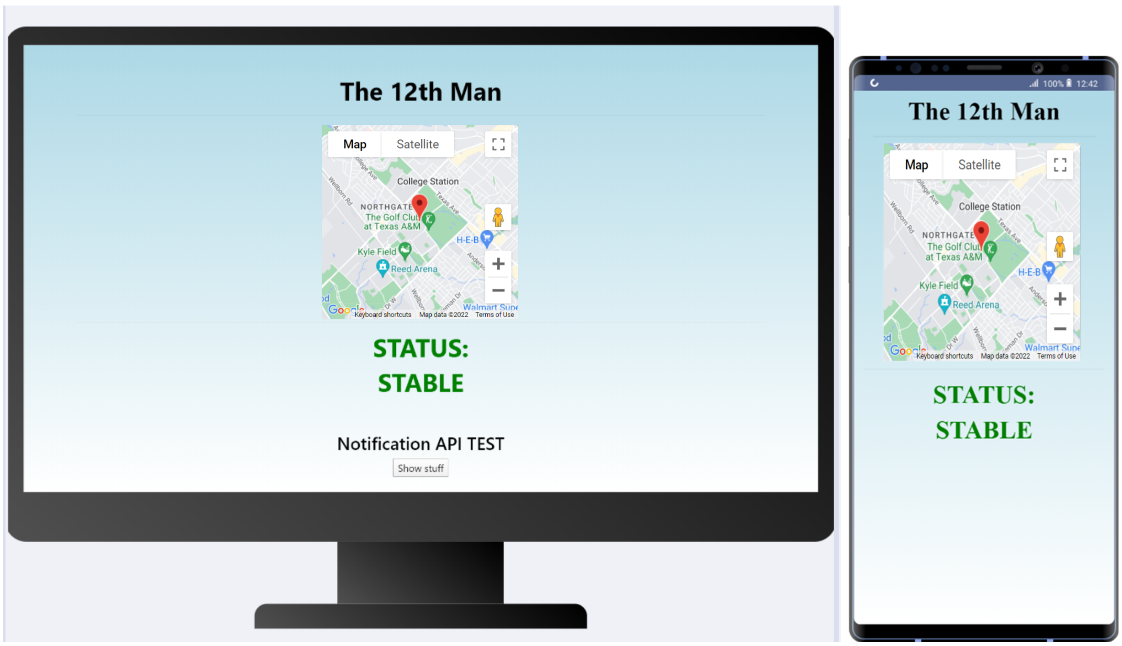


Figure 56. App View on Desktop and Mobile

The app displays the title, a Google map, and the status of the wearer. The map will display the user's location based on the coordinates it takes from the database. Once the coordinates are taken, they are sent to a function that calls the Google Maps API to display the location of the device. The status currently is just a placeholder, but will display the AI predicted probability of a meltdown. A button was added to display a push notification. This feature currently works on desktop devices but not for mobile.

5.4.2 Validation

To validate the application I changed the coordinates in the database to see if it changed in the app. Since the database is being hosted on AWS and the app is hosted on Heroku, both can be accessed at any time. By testing the application in this way it will show that the app can pull the most relevant data. During my validation I changed the coordinates in the database from texas A&M and it was reflected in the app as seen in figure 57.

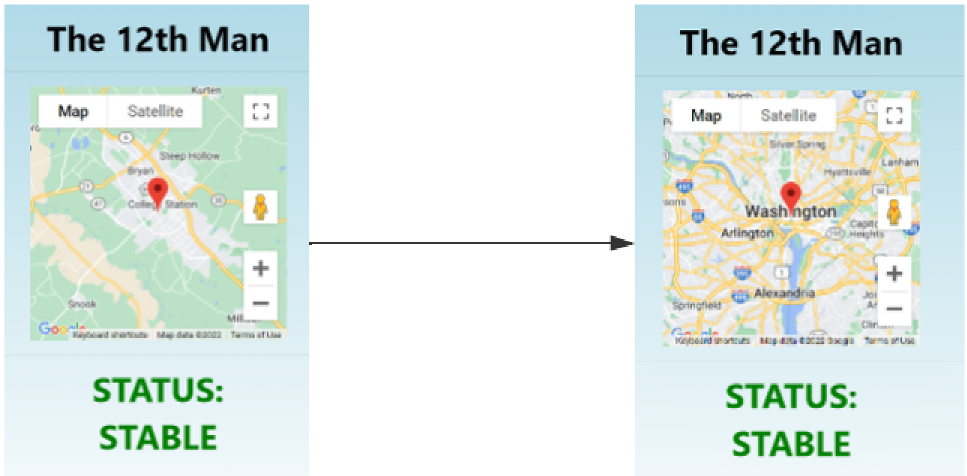


Figure 57. Changing Coordinates on the App

5.5 Subsystem Conclusion

The app and database have both been created and hosted, being able to be accessed at any time. All the parts of my subsystem are connected and can effectively communicate with each other. The application does need some work including adding a login and having access to more data. The database also needs to be changed to store local data from the user and keep it separate from other users.

Appendix A

ML/Sensor Code	https://github.com/krosljas/Capstone
App/DB Code	https://github.com/NoahLock10/the12manApp
AI	Artificial Intelligence
ML	Machine Learning
CNN	Convolutional Neural Network
ASD	Autism Spectrum Disorder
CONOPS	Concept of Operation
GPS	Global Positioning System
IC	Integrated circuit
I/O	Input/Output
mA	Milliampere
mAH	Microamp hours
mm	Millimeters
mW	Milliwatt
Li	Lithium
PCB	Printed Circuit Board
SQL	Structured Query Language
TBD	To Be Determined
V	Volts
W	Watts
4G	Mobile Communication Standard’
AWS	Amazon Web Services
API	Application Programming Interface
ejs	Embedded JavaScript
js	JavaScript
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets