DP-3
Gait Belt

February 24th, 2020

Team 13

Kaitlyn Clancy - clanck2
Evelyn Cudmore - cudmoree
Faisal Khaleel - khaleef
Chelsea Maramot - maramotc
Meira Morphet - morphem

Academic Integrity Statement:

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Submitted by: Kaitlyn Clancy, 400252165
Signature: Kaitlyn Clancy

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Submitted by: Evelyn Cudmore, 400233138
Signature: Evelyn C

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Submitted by: Faisal Khaleel, 400251143
Signature: Faisal

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Submitted by: Chelsea Maramot, 400246795
Signature: Chelsea Maramot

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Submitted by: Meira Morphet, 400239795
Signature: [illegible]

Executive Summary

Walking patterns reflect the cognitive function of the brain such that gait analysis can be used as an early diagnostic tool for cognitive diseases. Research has shown an inter-related connection between gait and cognition in the adult population. Specifically, neurological diseases such as Alzheimer’s and Lewy Body Dementia can be characterized by changes in gait pattern such as a decline in gait velocity or an asymmetric walk at its early stages [1]. Thus, suggesting that gait evaluation over a course of time can be used as an early prediction for cognitive decline. Early diagnosis of diseases, such as Alzheimer’s disease, can save 7 to 7.9 trillion dollars in health and long-term costs in the United States [2]. Therefore, gait assessment can serve as a diagnostic tool for those with a predisposition for cognitive diseases.

To aid in early diagnosis, an analytic gait belt that operates with an orientation sensor was designed. This belt is worn by the patient at doctors’ offices during their annual checkup. There, the doctor will ask the patient to wear the belt on their hips and demonstrate their gait. Through this motion, the belt measures linear acceleration, which is converted into velocity using a simple kinematics equation. In addition, the belt detects hip angles to check for an asymmetrical walk such as limping. All data collected in the first checkup is stored and used as a baseline in which future gait data will be compared. Once enough data has accumulated, it can be analyzed and used as an aid in early diagnosis. If the data depicts a constant change in walking speed, the patient might be in the early stage of Alzheimer’s disease. On the other hand, frequent changes in gait can be a sign of Lewy Body Dementia [1]. If any of these gait patterns are detected, the doctor will be notified to analyze the patient’s condition.

Currently, there have been other wearable devices in the market that serve as an early diagnostic tool for neurological diseases. However, the observed characteristic is not of gait pattern but speech patterns or eye movement. In comparison to these devices, measuring the gait pattern is convenient due to the portability of the gait belt. In addition, tracking changes over time is an effective way to determine variable patterns. However, due to its longitudinal nature, the patient must wear the belt numerous times over several years to detect subtle changes in gait, thus requiring commitment and risking attrition. For now, the gait belt is limited to the physician's office. Overall, the gait belt provides an alternative for other diagnostic tools.

In summary, the gait belt gathers information longitudinally and is highly beneficial for patients with a predisposition for cognitive diseases. It is programmed to adapt to the user’s unique gait pattern and sense a developing abnormality in walking velocity and symmetry of the hips. With the aid of the gait belt, patients can have access to treatment and a variety of clinical trials to manage their disease at its early stage. Thus, the gait belt has tremendous potential to improve diagnostic and therapeutic approaches with just a small step.

Table of Contents

Summary of Design Objectives

Need Statement: The population with a predisposition (ie. genetic, environmental, lifestyle, etc.) for Alzheimer's Disease or Lewy Body Dementia needs a wearable device which will monitor gait pattern variations common in the early stages of either condition. This would act to increase the likelihood of early diagnosis, therefore allowing the development of an enhanced and effective treatment plan.

List of Criteria:

- Light-weight
- Wearable
- Durable
- Ergonomic
- Easy-to-clean
- Intuitive
- Cost-effective
- Accurate & Reliable
- Provides relevant readings
- Alerts errors / malfunctions
- Biocompatible
- Mobile
- Appropriate size for function
- Powered

Background and Research Summary

The main input device within the physical computing prototype is a 9 degrees of freedom (9-DOF) Adafruit BNO055 absolute orientation sensor which collects data using an integrated accelerometer, gyroscope, and magnetometer [3]. An accelerometer functions by calculating the effect that a force has on the distance between two capacitors. This change in capacitance translates to a change in voltage, which is proportional to acceleration [4]. Gyroscopes respond to changes in angular rotation. They contain small vibrating mechanisms which produce forces perpendicular to the vibration and proportional to the angular velocity. These forces alter the distance between the plates of its capacitors, providing us with the calculated hip tilt angle readings [5]. Lastly, a magnetometer, made of semiconducting layers and thin metallic surfaces, produces a voltage proportional to an intersecting magnetic field, providing direction readings [6]. Overall, the sensor is light, small, and cased with plastic. These physical characteristics allowed the creation of a two-part housing for this sensor which allows motion imitating a patient's gait pattern. Orientation sensors are widely found nowadays, especially within smart devices [4], with applications such as GPS accuracy, compassing, fall detection, and screen orientation.

Output devices within the physical computing prototype consisted of 3 light-emitting diode (LED) lights, designed to respond to any significant change within the data being collected. These diodes emit light by combining a negatively charged N-type semiconductor with a positively charged P-type semiconductor, and electrodes on each end in a circuit [7]. Applying a voltage causes electrons of the N-type semiconductor and holes of the P-type semiconductor to interact, generating energy in the form of light [7]. Different materials used in the diodes allow for different wavelengths of light to be produced, forming different colours. The LEDs used in the prototype are surrounded by coloured epoxy resin [8], such that each is a different colour not due to the source of light, but the colour which the light passes through

Another output device as part of the prototype is an organic light-emitting diode (OLED) monitor. This continuously displays the calculated baseline walking speed, the current walking speed, and the current hip angle tilt. OLEDs do not require backlighting or filters, unlike LCD which do. The general structure of an OLED is a cathode and an anode surrounding the emissive layer, but other layers are added in different devices to meet the needs of the device, such as touch sensors or durability increasing layers. OLEDs are very sensitive to oxygen, so they must be properly encapsulated in order to function [9]. However, what makes OLEDs unique is their carbon-based emitter, which requires less electricity to emit light [10]. Furthermore, their ability to be manufactured at a very thin and miniature size allowed the design of a fitting showcase. The efficiency of these devices has grown them in popularity, now finding common applications ranging from modern cameras to larger-screen televisions.

On the contrary, the theoretical final prototype would only include a single output device. This would be a Bluetooth data logger, which functions using an implemented class 2 Bluetooth chip which relays the logged orientation data over 10-30 meters [11] via radio waves [12] to a pre-established connection with the doctor's or physician's database. This output device would simply be cased with plastic, though it would be small and lightweight enough that it could become embedded within the belt which makes up the final prototype.

Market Analysis

Product 1: Winterlight [13]

This is a recently developed platform which specifically uses speech patterns to identify signs of brain disorders. It is designed to detect hidden clues in the way a patient speaks, highlighting health problems before an official diagnosis is made.

Similarities:

- This device works to monitor early-stage Alzheimer's Disease and dementia symptoms
- This device can be used to help diagnose early-stage Alzheimer's Disease or forms of dementia

Differences:

- This device is not wearable
- This device is not mobile
- This device monitors speech pattern instead of gait pattern

Product 2: Eye-Tracking Technology (under development at Loughborough University) [14]

The idea behind this software is that eventually, eye-tracking approaches will become a leading method of early diagnosis and intervention of Alzheimer's Disease. As described by the sources provided, eye movement impairment within a patient acts as a biomarker for the disease, suggesting said patient is at high risk for developing the subsequent cognitive problems.

Similarities:

- This technology monitors Alzheimer's Disease symptoms that appear before any cognitive issues related to the disease develop
- This software identifies patients at a high risk of developing the disease, though it is unclear as of now whether this technology can be used for diagnosis

Differences:

- This technology is not wearable
- This technology is not mobile
- This software will be based on a previously established eye-tracking model, modified to track eye-movement instead of gait pattern
- This technology will essentially be identifying patients with mild cognitive impairment (MCI) in the aim of also identifying patients with a risk factor for Alzheimer's Disease, thus making it an indirect method of prediction

Description of Design & Design Specifications

Purpose of Design: Our design will be used to help predict early-stage Alzheimer's Disease and Lewy Body Dementia for those that have a predisposition to these cognitive impairment diseases. With early diagnosis, physicians may have the ability to reduce symptoms or prevent them from worsening through a well-managed and effective treatment plan. Ultimately, allowing patients to experience a higher quality of life for a longer period of time.

What Device Does: The technology used in our design combined with the placement of the sensor allows one to monitor the patient's gait pattern. Different patterns are characterized by differences in limb-movement patterns. The pattern that our belt measures specifically is walking velocity and Euler angle of the hips. When a person walks, their hips move with specific angles, and they move at specific accelerations. Our device allows an individual to set their own baseline, through computational calculations, for subsequent readings to be compared against. This baseline suggests their "normal" walking pattern. Research has shown that patients with Alzheimer's Disease and Lewy Body Dementia have varying gait patterns that follow unique paths of development in each case. The changes in measured walking patterns can have a very slow inclination and may take years to diagnose. Patients who are affected by these forms of cognitive impairment are hard to recognize early on. Thus, our device will be used as a within-office tool to aid in early diagnosis.

Device User Intentions: Our device is intended to be used by doctors or physicians for testing patients with a risk factor for Alzheimer's Disease or Lewy Body Dementia. Consequently making the end-user two-fold, doctors/physicians and patients with a risk factor for Alzheimer's Disease or Lewy Body Dementia.

Device Interaction: This device is designed as a belt to be worn around the patient's waist. It will be easily detachable and multi-user friendly. A belt ensures that the user places it around their hips, where random motion is minimized, so the sensor can accurately measure the overall gait pattern. It is important to verify that the sensor is in the proper position (at the hip). For comparison, using limbs will give inaccurate speeds as they are able to move at separate speeds from the body as a whole, and will not account for proper gait patterns. Due to the nature of the device and the data being recorded, this device must be used multiple times over uniquely tailored intervals of when a patient visits their physician for a check-up. Therefore, this device must be used for however long it is deemed necessary by the physician.

Materials: Our final product will be a belt. This will generally be made of traditional leather for comfort, durability, and hypoallergenic traits. However, it is important to note that for some sections, it will be hollow for embedded wiring and device integration. The belt buckle would be made of a zinc-copper alloy that is hypoallergenic and best suited for individuals with sensitive skin [15]. The wiring itself will be within a waterproof cable gland made of nylon fabric, such as Silnylon [16], maintaining its ability to freely move within but preventing any potential danger from the outside. Included inside is an enhanced absolute orientation sensor, embedded within the leather itself. These are typically cased within some variation of plastic, allowing it to be inert with the other present materials. Finally, the device will have an attached battery case on the outside of the leather, which accepts button cell batteries. This casing will be designed to be openable for battery replacement. As well, this casing will be made of a hypoallergenic variation of high-density polyethylene (HDPE) for stability, corrosion-resistance, and chemical inertness [17].

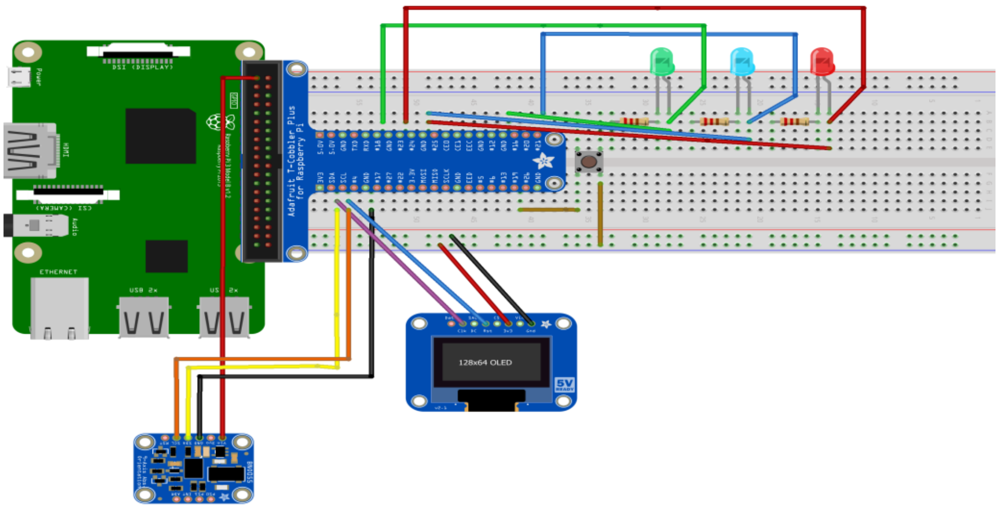
Device Function: To begin, our device is battery-powered and these batteries would be replaceable rather than rechargeable. The data measured is essentially orientation data, analyzed to provide critical gait pattern readings. One of two main readings would be the wearer's walking speed, which is calculated as they walk in a straight line utilizing kinematics. If the patient's walking speed data constantly goes up or down in subsequent visits, showing a significant change from their initial walking speed data (i.e 50% of a difference), the doctor or physician is notified. In addition, the wearer's hip tilt angle is calculated over ten steps using Euler angles. By setting one side of the tilt as a positive one and the other as a negative one, the sum after each trial should be zero, or symmetric. If this is notably asymmetric, the doctor or physician is notified as well. In essence, this system works as a whole without notifying the wearer of any data changes to their gait, but only notifying their doctor or physician, which eliminates practice effects. The data is all sent to the file of the patient within their healthcare provider's database via an embedded Bluetooth chip. Furthermore, this device would only be turned on during the doctor check-ups where it is intended to be used, as described below.

Operating Environment & Length of Operations: The main operating environment would be doctors' or physicians' offices within general hospitals or healthcare clinics as an early diagnostics tool for patients, especially those with a risk factor for Alzheimer's Disease or Lewy Body Dementia. The device is intended to be used at doctor check-ups. The first visit with our device will calculate a baseline specific to the patient, given that there are no evident injuries present (ie. broken foot, sprained ankle etc) that could alter the results. Subsequent checkups, which will differ depending on the patient (ie. annually, semi-annually, or bi-monthly), will continue to monitor the patient's patterns and store the data to their personal file. The belt accommodates numerous users, as long as it is reset before a new patient uses it, and new data taken during subsequent visits are placed in the correct patient's data file for it to be properly analyzed for any major variations. After the data is analyzed by the computational component, any large discrepancies between the new data and the baseline will be notified to the doctor for further analysis. Tests can then be run to aid in full diagnosis.

Figure 1: Picture of Assembled Tangible Prototype

Figure 2: Picture of Assembled Physical Computing Prototype

Figure 3: Raspberry Pi Schematic Diagram
Note: Red LED is connected to pin 23 due to the absence of pin 7 in the T-Cobbler image



**Please refer to the appendices for engineering drawings, bills of materials, and parts lists for both the tangible prototype and the physical computing prototype.*

Design Verification

Table 1: Test Plan

Test	Desired Result	Actual Result	Pass/Fail
Velocity >> Baseline Baseline: 4.57 m/s Velocity: 10.8 m/s	GREEN LED ON BLUE LED ON (Out of baseline range)	GREEN LED ON BLUE LED ON	Pass
Velocity << Baseline Baseline: 4.57 m/s Velocity: 0.68 m/s	GREEN LED ON BLUE LED ON (out of baseline range)	GREEN LED ON RED LED ON	Pass
Velocity > Baseline or Velocity < Baseline Baseline: 4.57 m/s Velocity: 4 or 5 m/s	GREEN LED ON BLUE LED OFF (within normal range of baseline)	GREEN LED ON BLUE LED OFF	Pass
Zero acceleration Or Acceleration < $0.3m/s^2$	OLED displays Velocity = 0	OLED displays Velocity = 0	Pass
OsError Sensor malfunction (due to loose wires)	Exception Raised (loop stops and displays “Sensor Malfunction”	Exception Raised (loop stops and displays “Sensor Malfunction”)	Pass
TypeError Euler_angles == None lin_acceleration == None	Value not appended into lists (euler_y or data)	Value not appended into lists (euler_y or data)	Pass
Empty lists: Euler_y = [] Counter = []	GREEN LED OFF (no data received by sensor)	GREEN LED OFF (no data received by sensor)	Pass
Normal hip sway: Maximum left and right shifts for one second each	GREEN LED ON RED LED OFF	GREEN LED ON RED LED OFF	Pass
Abnormal hip sway: Three seconds tilted to the left and one second tilted to the right	GREEN LED ON RED LED ON	GREEN LED ON RED LED ON	Pass
Escalation: Velocity out of baseline range and asymmetrical walk	RED LED BLINKING	RED LED BLINKING (To adjust sensitivity of program, both conditions must be achieved at least three times for the light to start blinking)	Pass

Output Display Summary

Green LED: Indicates status. When it is on, the sensor is receiving and calculating velocity
Blue LED: Turns on when an abnormal velocity is detected
Red LED: A continuous light indicates an asymmetric hip tilt, while a blinking red light indicates escalation. The button must be pressed to restart the program during escalation.
Sensor Malfunction: Often caused by loose wires. As a solution, restart the program.

CAD Model

The sensor housing effectively demonstrates the functions intended by the orientation sensor for our device by imitating a person's gait and mimicking lateral swaying. The CAD model that we built is comprised of the regular breadboard and raspberry pi housing components. We customized a track specific to our device to imitate a person's gait by allowing the sensor to slide and mimic the forward and backward motion of walking. The customized housing specific to the sensor also allows lateral swaying by way of a curved surface rotating within a base that fits the curved shape to resemble hip movements. The lateral shift of the sensor housing is limited to a range of -15 degrees to 15 degrees to encompass a controlled range of hip movement.

Computing Program

The sensor has been programmed to calculate and output appropriate gait values using linear acceleration and Euler angles. To adapt to each person's unique gait pattern, a baseline value is calculated based on the average of the first 10 velocities specific to that person. The sensor measures linear acceleration, but through kinematics, acceleration is converted into velocity. This calculated velocity will then be used as an initial velocity for the next calculation. Likewise, normal hip movement is assumed to be symmetrical and shifts laterally at equal intervals. For instance, the sensor tilts to the left for one second and then shifts to the right for another second. The program was coded to account for slight variabilities in hip sways. Specifically, if the person leans to the left for two seconds and then leans to the right for one second, the data is still within the normal range.

Verification Process

In order to achieve desired results, hold the sensor housing firmly right at its centre with the thumb and index finger. Due to friction and housing material, it can be challenging to slide the sensor smoothly on the track. For best result, push down on the sensor housing as it moves front and backwards to provide a smooth glide across the track. This downward force is exerted by the thumb and index finger as the demonstrator holds and slides the sensor housing. As for the angle measurements, hold the side handles on the sensor housing and tilt it up and down until its maximum angle has been reached.

Calculating Baseline: For demonstration purposes, move the sensor back and forth throughout the track as fast as possible to establish an adequate baseline. This will make it easier for the demonstrator to achieve specific walking conditions. Once a baseline is set, a constant baseline value should be displayed on the OLED.
Output: OLED DISPLAY, GREEN LED ON

Lateral Hip Movements: In this demonstration, a normal hip movement is categorized by symmetric walking. To achieve this condition, move the sensor left and right at equal intervals.

For instance, one second spent on the right side and one second spent on the left side. Repeat this motion until the output is displayed.
Output: GREEN LED ON, RED LED OFF

To achieve an abnormal hip sway, adjust the interval spent on each lateral side such that it depicts an uneven walk. For instance, shift the sensor to the left for three seconds and then to the right for one second. Any unequal and asymmetric time spent on either side of the hips provides a suitable condition to produce an abnormal walk. Repeat this side-to-side motion, until the desired output is displayed.
Output: GREEN LED ON, RED LED ON

Forward and Backward Motion: To exhibit a normal walking velocity, slide the sensor forwards and backwards across the track. The velocity at which this is done depends on the baseline. In general, when the velocity is within the baseline range, it is considered a normal walking velocity.
Output: GREEN LED ON, BLUE LED OFF

To achieve an abnormal walking velocity, slide the sensor forwards and backwards as fast as possible such that the velocity is much greater than the baseline. Likewise, this condition can also be achieved if the velocity is much smaller than the baseline value.
Output: GREEN LED ON, BLUE LED ON

To determine what velocity is within or out of baseline range, a series of calculations is shown below:

Velocity1 = 0.68 m/s (example of an abnormal velocity)
Velocity2 = 0.57 m/s (example of a normal velocity)
Baseline = 4.57 m/s (average of the first 10 velocities)
Baseline Range = 0.5 * 4.57 m/s
= 2.23 m/s

Difference = Baseline - Velocity	Difference = Baseline - Velocity
= 4.57 m/s - 0.68 m/s	= 4.57 m/s - 4.0 m/s
= 3.89 m/s	= 0.57m/s
Difference > Baseline Range	Difference < Baseline Range
3.89m/s > 2.23m/s	0.57 m/s < 2.23m/s
Abnormal velocity	*Normal Velocity*
BLUE LED ON	*BLUE LED OFF*

In general, the sensor is programmed to check if the velocity is within 50% of the baseline. If the velocity is outside that range, it will trigger the blue LED to turn on. Thus, indicating an abnormal velocity.

Escalation: Escalation is achieved when both velocity and hip angles exhibit abnormal patterns. To produce this condition, try to achieve an abnormal walking velocity through a rapid back and forth motion. Then stop and shift the sensor left or right. In general, take turns in achieving both irregular conditions. At some point, both red and blue LEDs will activate to indicate irregularities,

but escalation will not be triggered. In order to call escalation, both abnormal conditions have to be achieved at least three times. The output is displayed and a button should be pressed to restart the program.
Output: RED LED BLINKING, PRESS BUTTON

Design Critique, Discussion, and Recommendations

The gait belt design is effective in providing useful data. By placing the sensor at the hip level, due to the belt-form of the device, accurate readings of the hip angles can be read. The belt is convenient due to its simplicity and wearability. The tool requires little user input and is intuitive, providing all readings by simply wearing it. The tool has no interactions with the skin or orifices of the body as it is worn over the clothing, providing a very low risk for any irritation or adverse reactions with the body. Overall it is a simple yet effective wearable device with the capacity to be an extremely useful gait analysis tool.

The gait belt serves a clear purpose in its initial intentions, providing data to a patient that will aid in the process of early diagnosis of Alzheimer's disease and other forms of dementia. Though it is well suited for this purpose, the gait belt is not limited to only these applications, and instead can be extended into a range of uses.

The concept of the belt is built on research that backs up the relationship between gait changes and cognitive changes. The device could be used to further cement these studies as a convenient and wearable way of amassing this type of data, further strengthening previous research and providing a new method of data collection inside and outside of the office.

Furthermore, this tool is not limited to dementia. Due to its general exploration of the physical gait pattern, this device could realistically be applied to many gait studies. There is a range of diseases and conditions which produce abnormalities in gait, and early detection of these abnormalities by the use of sensitive technology may allow for the detection of these known gait issues before they are entirely apparent to the human eye. In hemiplegia the patient experiences extension of one leg, which could be detected by the rocking pattern aspect of the belt, as the patient would take much longer steps on one side than the other due to circumduction of the leg when walking [18]. Acute cerebellar Ataxia causes stumbling in the direction of the affected cerebellar hemisphere [19], which could potentially be detected by the rocking pattern aspect as well, as the patient would stumble heavily on one side to a degree that could be detectable by the device. Cerebral palsy causes walking an abductor spasm, causing the patient's legs to be pulled close together, causing a stark forward stumbling [18]. This pattern could be read through the speed monitoring system in the gait belt, to due quick starting and stopping, which could be recorded and tracked for an increase in severity. Myopathy, which can be caused by muscular dystrophy as well as other conditions, causes the weakening of muscles [20]. This can produce a leaning or waddling gait pattern as the body tilts to help lift the legs due to weakness in pelvic stabilization. The result of this gait would be deepening hip angles being read, and over time the recording of hip angles by the gait belt could indicate the increase of severity of myopathy.

The gait belt device has the capacity to expand the way we currently look at gait patterns and their relationship to life-altering conditions. The success of this device could allow for great improvements in gait research and gait-related diagnosis.

References

[1] M. Montero-Odasso, J. Verghese, O. Beauchet, and J. M. Hausdorff, “Gait and Cognition: A Complementary Approach to Understanding Brain Function and the Risk of Falling,” *Journal of the American Geriatrics Society,* vol. 60, no. 11, pp. 2127–2136, Oct. 2012.

[2]“Why Get Checked?,” *Alzheimer's Disease and Dementia*. [Online]. Available: https://www.alz.org/alzheimers-dementia/diagnosis/why-get-checked. [Accessed: 23-Feb-2020].

[3]K. Townsend, “Adafruit BNO055 Absolute Orientation Sensor,” *Adafruit Learning System*. [Online]. Available: https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor/overview. [Accessed: 20-Feb-2020].

[4] R. Goodrich, “Accelerometers: What They Are & How They Work,” *LiveScience*, 01-Oct-2013. [Online]. Available: https://www.livescience.com/40102-accelerometers.html. [Accessed: 20-Feb-2020].

[5]J. Blankenship, “Sensing Orientation,” *Servo Magazine,* Jul-2016. [Online]. Available: https://www.servomagazine.com/magazine/article/July2016_Sensing-Orientation-for-Balance. [Accessed: 23-Feb-2020].

[6] V. Mabe, “How Does a Magnetometer Work?,” *Sciencing*, 24-Apr-2017. [Online]. Available: https://sciencing.com/a-magnetometer-work-4913575.html. [Accessed: 20-Feb-2020].

[7]T. Harris, C. Collette, and W. Fenlon, “How Light Emitting Diodes (LEDs) Work,” *HowStuffWorks,* 31-Jan-2002. [Online]. Available: https://electronics.howstuffworks.com/led.htm. [Accessed: 23-Feb-2020].

[8]W. O. Abdullah, “Basic led,” *BikesRepublic*, 02-Jun-2020. [Online]. Available: https://www.bikesrepublic.com/featured/halogen-versus-led-should-you-upgrade-to-led-lighting/attachment/basic-led/. [Accessed: 23-Feb-2020].

[9]“Home,” *OLED info*, 23-Dec-2018. [Online]. Available: https://www.oled-info.com/oled-technology. [Accessed: 20-Feb-2020].

[10]“OLED Basics,” *OFFICE of ENERGY EFFICIENCY & RENEWABLE ENERGY*. [Online]. Available: https://www.energy.gov/eere/ssl/oled-basics. [Accessed: 20-Feb-2020].

[11] bluAir, “Bluetooth Range,” *BluAir*. [Online]. Available: http://www.bluair.pl/bluetooth-range. [Accessed: 23-Feb-2020].

[12] C. Franklin and C. Pollette, “How Bluetooth Works,” *HowStuffWorks*, 11-Nov-2019. [Online]. Available: https://electronics.howstuffworks.com/bluetooth2.htm. [Accessed: 23-Feb-2020].

[13]“This is the Place to Predict Alzheimer's,” *University of Toronto News*. [Online]. Available: https://www.utoronto.ca/entrepreneurs/winterlight. [Accessed: 23-Feb-2020].

[14]Ndivya, “Eye-tracking technology offers hope for predicting Alzheimer's,” *Verdict Medical Devices*, 23-Aug-2019. [Online]. Available: https://www.medicaldevice-network.com/news/eye-tracking-technology-alzheimers/. [Accessed: 23-Feb-2020].

[15] “Is Zinc Alloy Hypoallergenic?(Quick Pro Answer),” *A Fashion Blog*, 30-Dec-2019. [Online]. Available: https://www.afashionblog.com/is-zinc-alloy-hypoallergenic/. [Accessed: 23-Feb-2020].

[16]Ripstop by the Roll, “Silnylon,” *Ripstop by the Roll*. [Online]. Available: https://ripstopbytheroll.com/collections/silnylon. [Accessed: 23-Feb-2020].

[17]“High-Density Polyethylene (HDPE) Labware: Thermo Fisher Scientific - US,” *ThermoFisher Scientific*. [Online]. Available: https://www.thermofisher.com/ca/en/home/life-science/lab-plasticware-supplies/plastic-material-selection/high-density-polyethylene-hdpe-labware.html. [Accessed: 23-Feb-2020].

[18]Stanford Medicine, “Gait Abnormalities,” *Stanford Medicine 25*. [Online]. Available: https://stanfordmedicine25.stanford.edu/the25/gait.html. [Accessed: 23-Feb-2020].

[19] A. Delgado, E. Boskey, and M. Beaugureau, “Acute Cerebellar Ataxia (ACA),” Healthline. [Online]. Available: https://www.healthline.com/health/acute-cerebellar-ataxia [Accessed: 20-Feb-2020].

[20] “Myopathy Information Page,” *National Institute of Neurological Disorders and Stroke*. [Online]. Available: https://www.ninds.nih.gov/disorders/all-disorders/myopathy-information-page. [Accessed: 23-Feb-2020].

Appendices and Supporting Documentation

Tangible Prototype Supporting Documentation

Figure 4: Preliminary Drawing

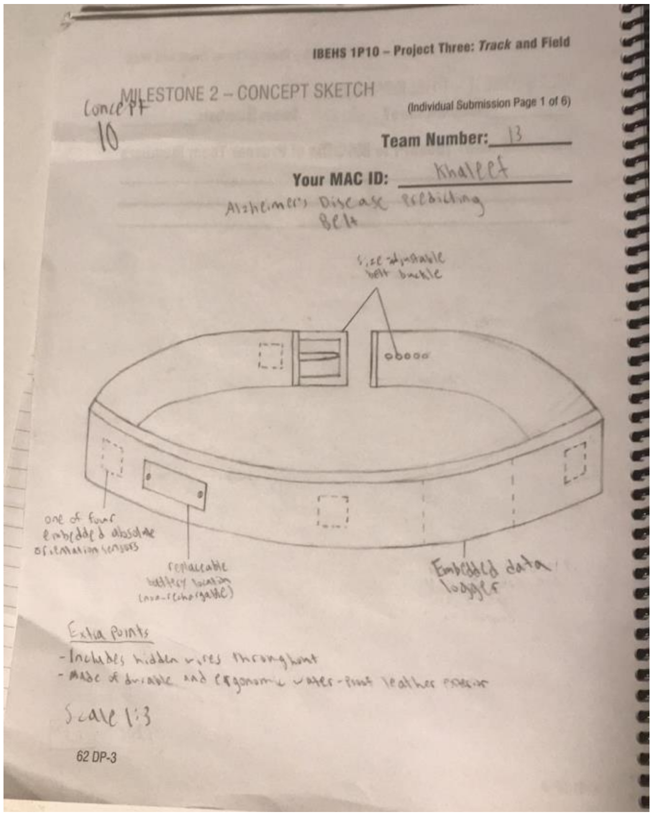


Figure 5: Preliminary Flowchart

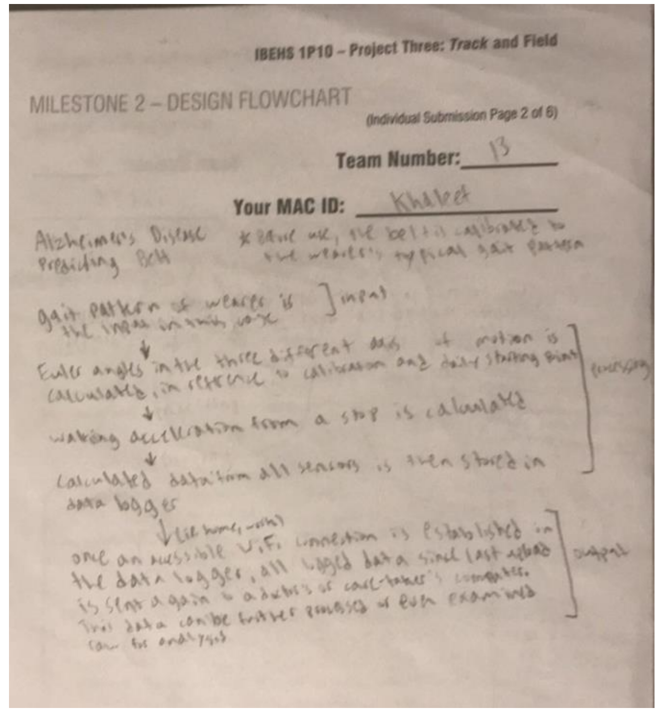


Table 2: Parts List for Tangible Prototype

Part	Quantity	Cost Per Item
Bluetooth Data Logger	1	~$150.00
Absolute Orientation Sensor	1	~$55.00
Replaceable Button Cell Batteries	2	~$2.00
Absolute Orientation Sensor Case	1	Modelled and 3D printed
Replaceable Battery Case	1	Modelled and 3D printed
Push Button	1	$0.30

Figure 6: Bluetooth Data Logger Model

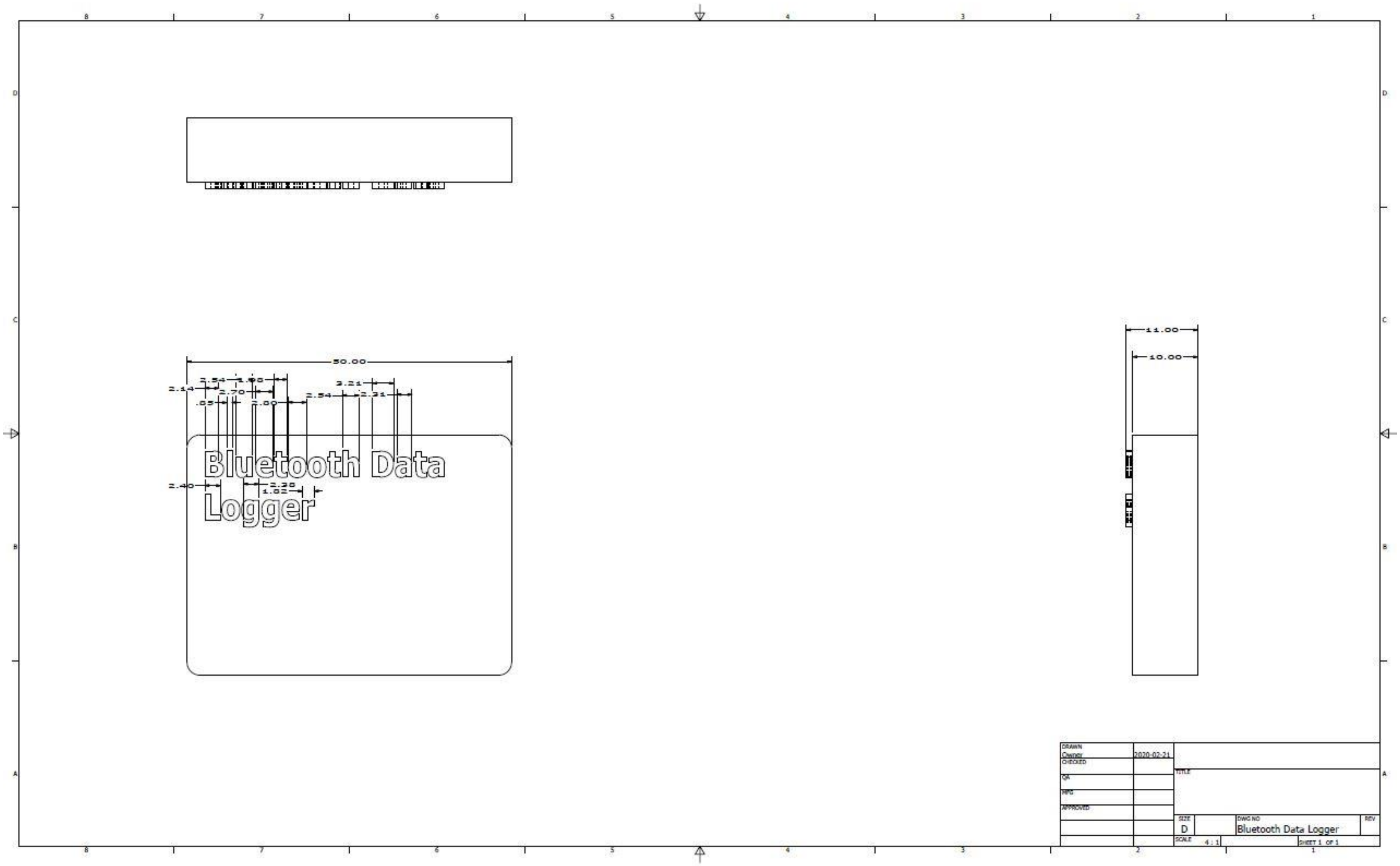


Figure 7: Orientation Sensor Case Model 1

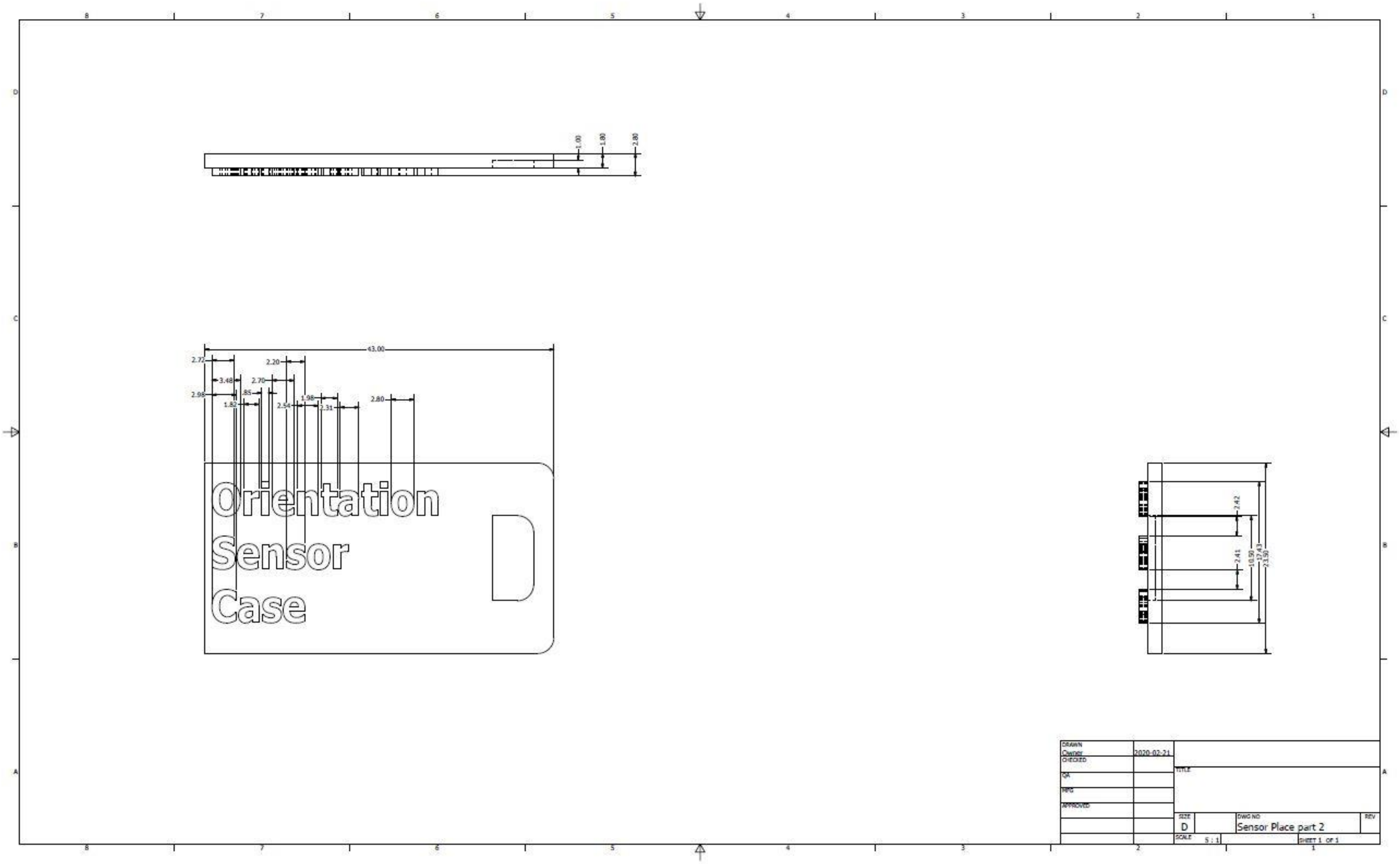


Figure 8: Orientation Sensor Case Model 2

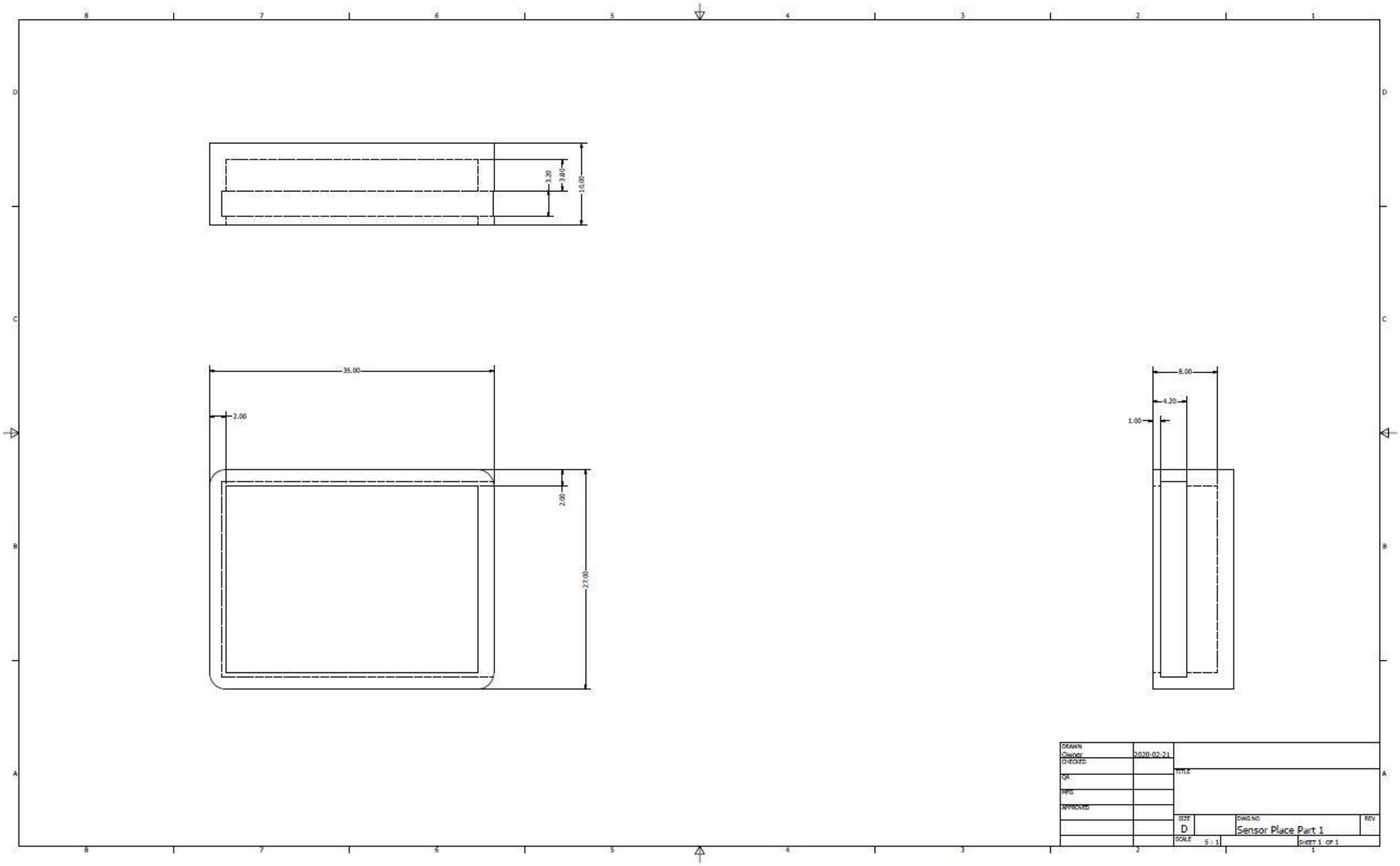


Figure 9: Replaceable Battery Case Model 1

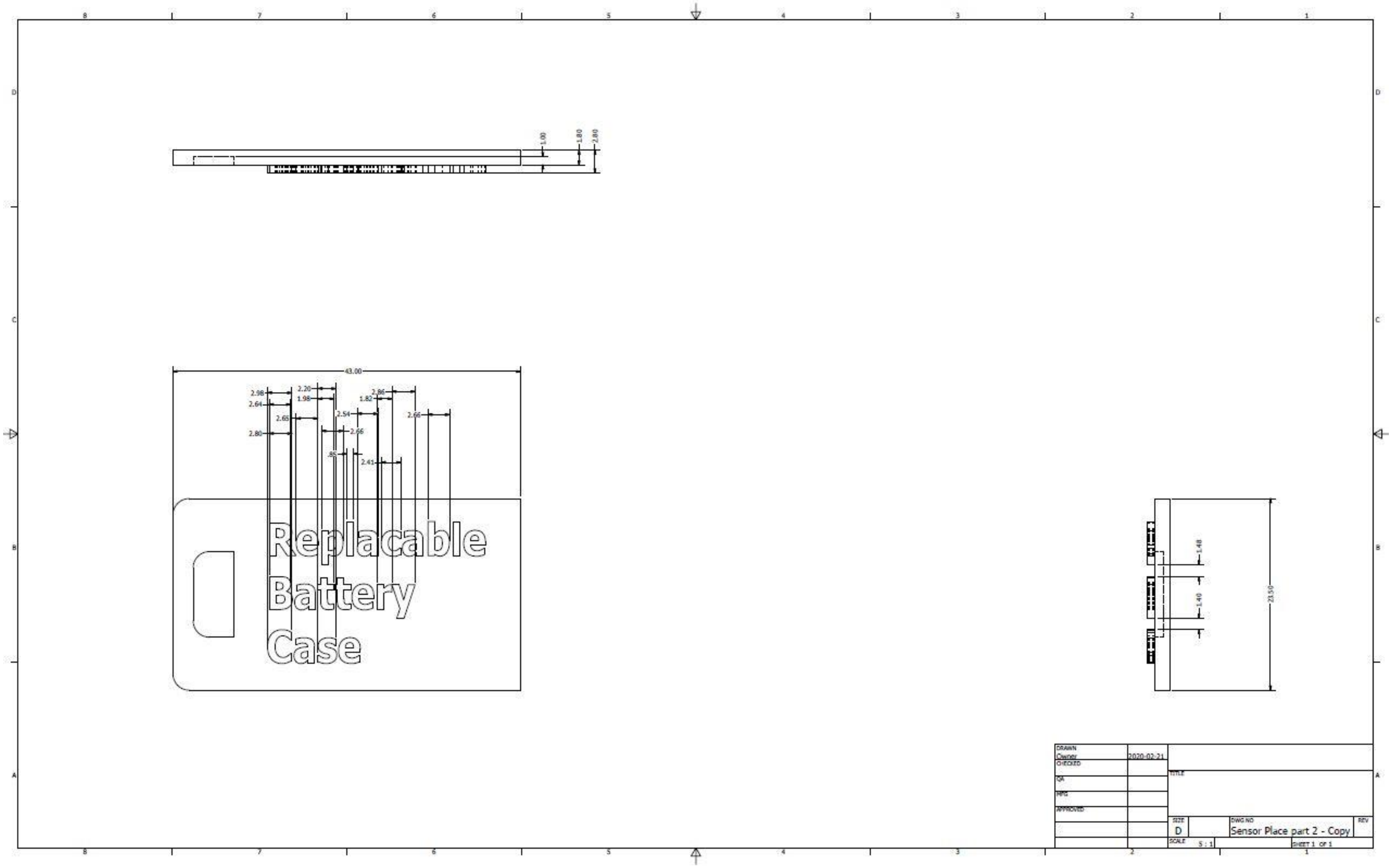
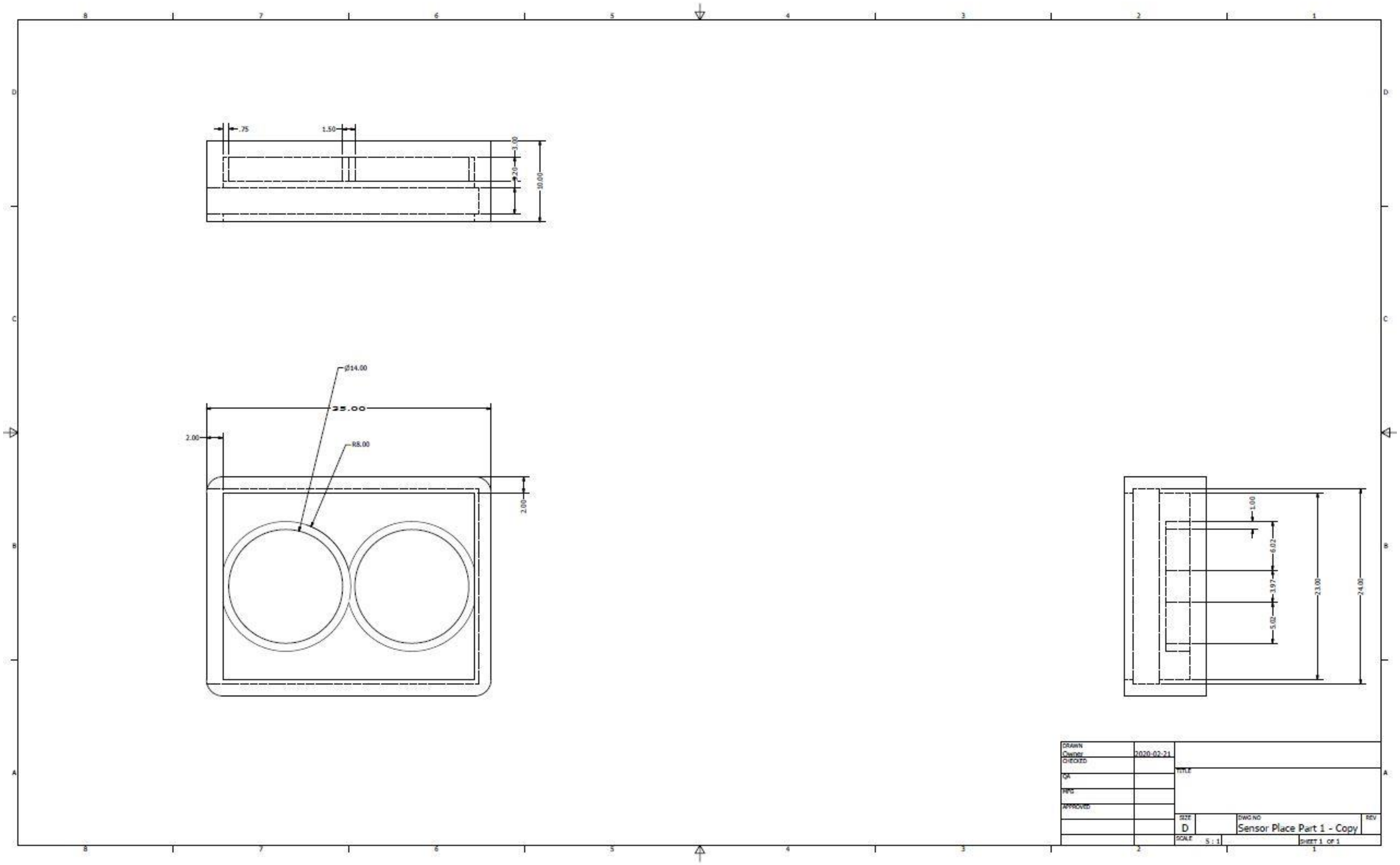


Figure 10: Replaceable Battery Case Model 2



Physical Computing Prototype Supporting Documentation

Table 3: Parts List of Electronics for Physical Computing Prototype

Parts	Quantity	Cost per item
Breadboard	1	$9.00
Raspberry Pi Board	1	$70.00
Absolute Orientation Sensor	1	$10.00
T-Cobbler	1	$11.00
USB Wire	1	$10.00
Computer Mouse	1	$13.00
Breadboard Jumper Wires	8	~$5.00
Computer Monitor	1	$150.00
Keyboard	1	$25.00
Resistor	3	~$1.00
Push Button	1	$0.30
LED (Red, Green, Blue)	3	~$1.00
OLED Monitor	1	$7.00
Mini HDMI to HDMI Adapter	1	$11.00

Figure 11: Average Flowchart

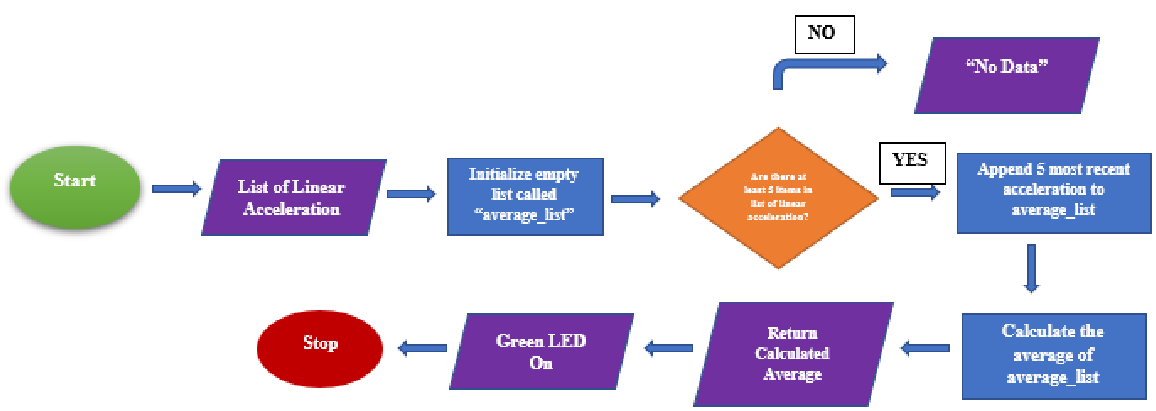


Figure 12:Velocity Flowchart

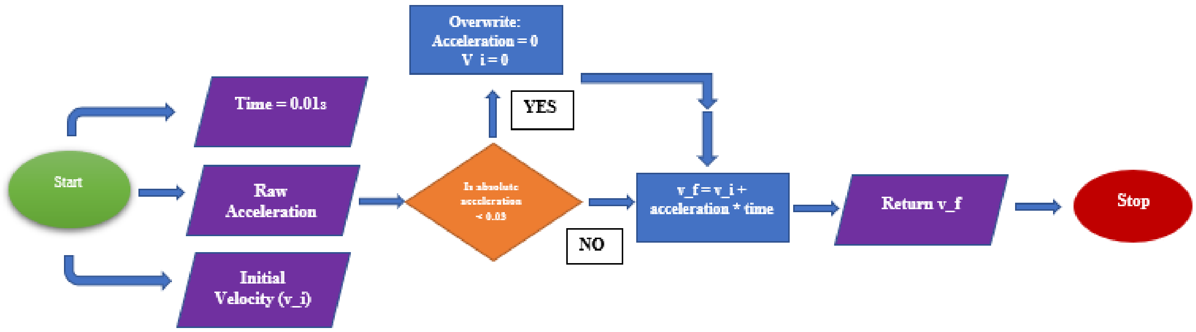


Figure 13: Speed Notification Flowchart

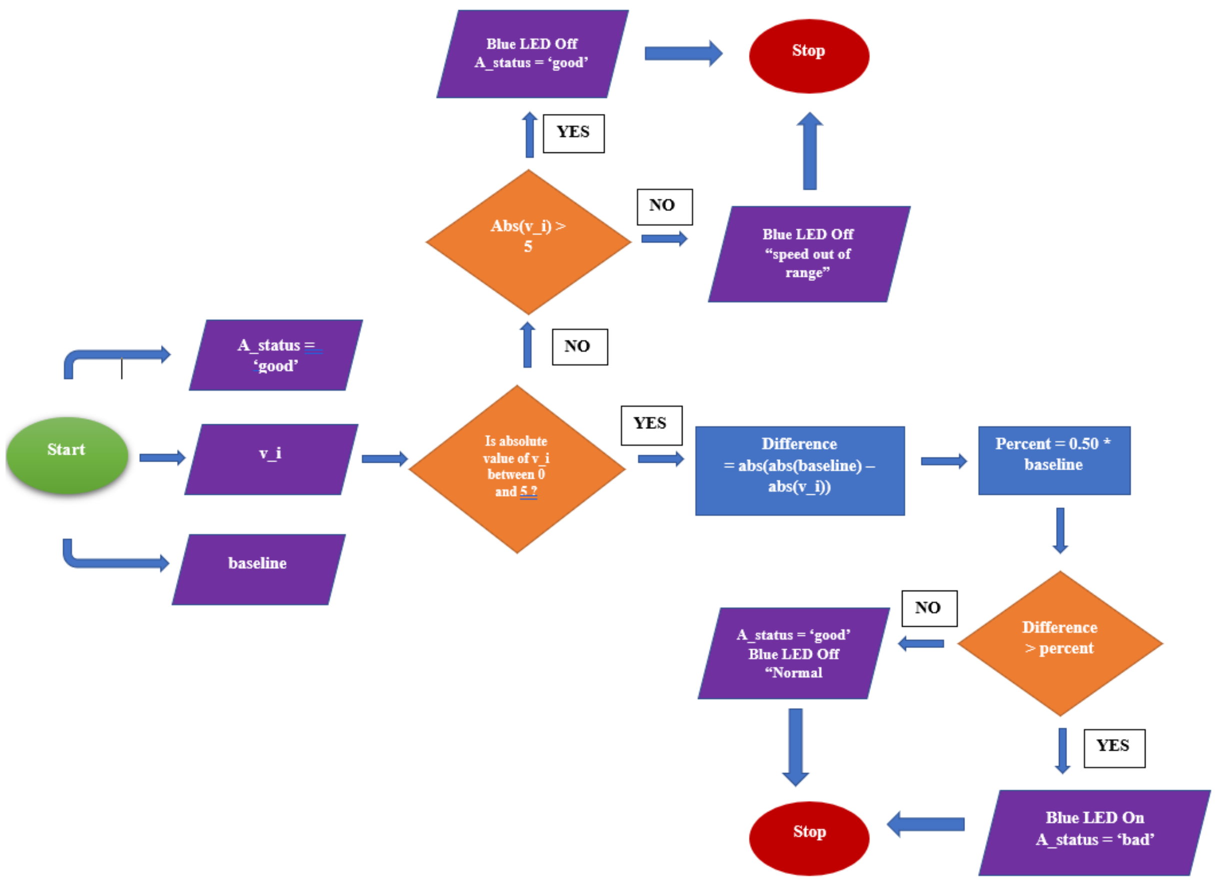


Figure 14: Angle Notification Flowchart

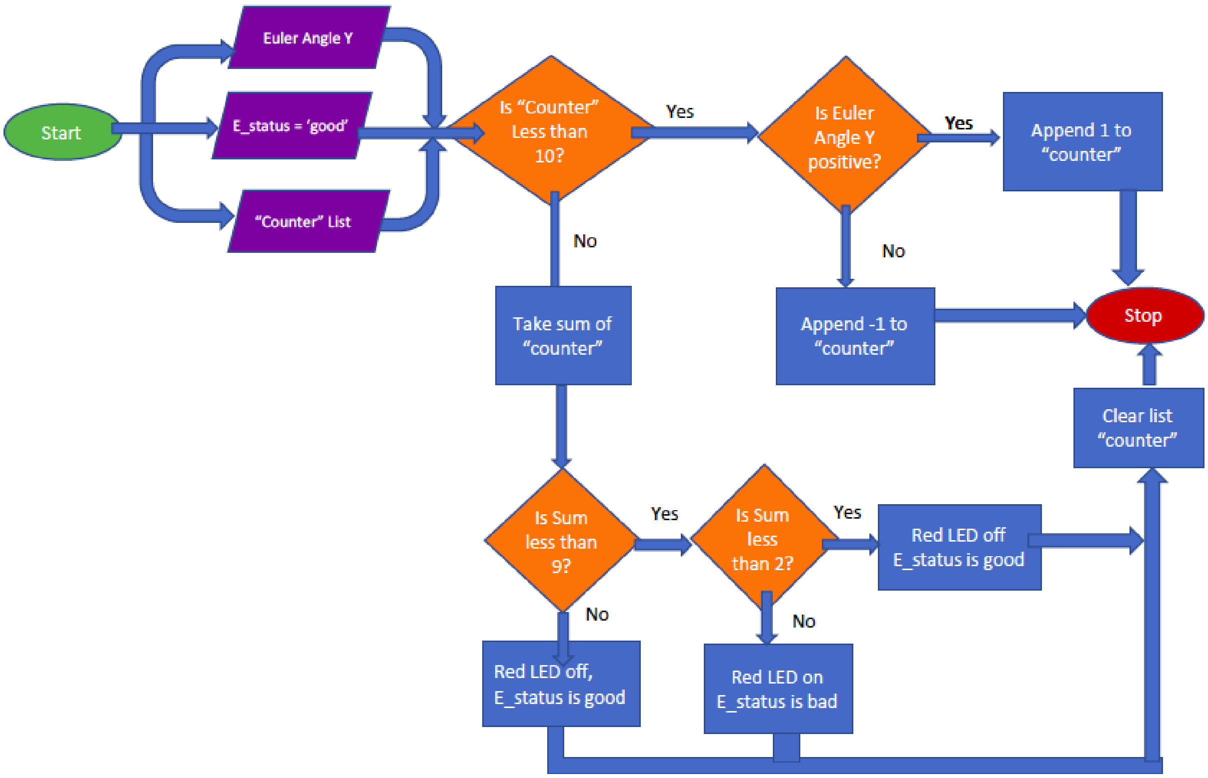


Figure 15: Escalation Flowchart

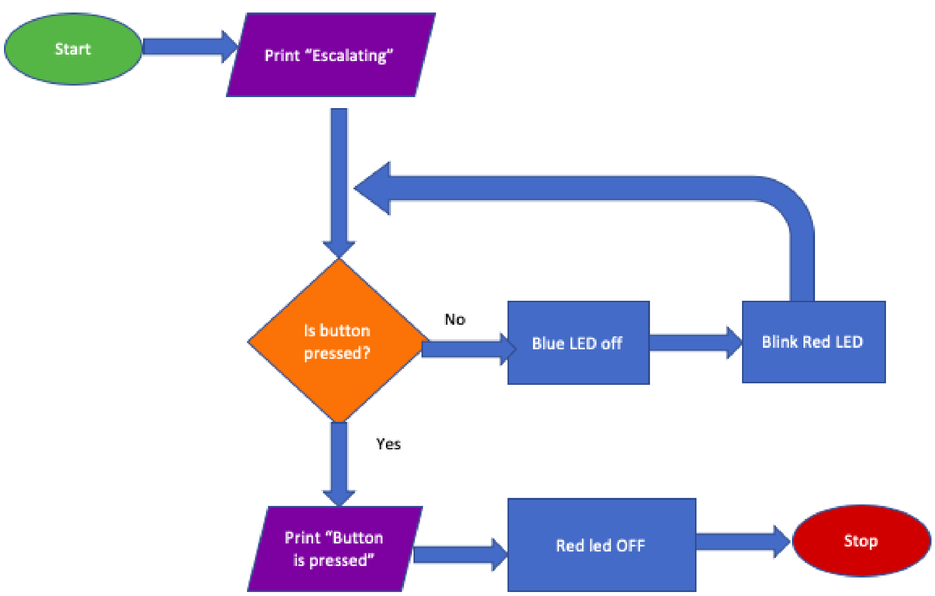
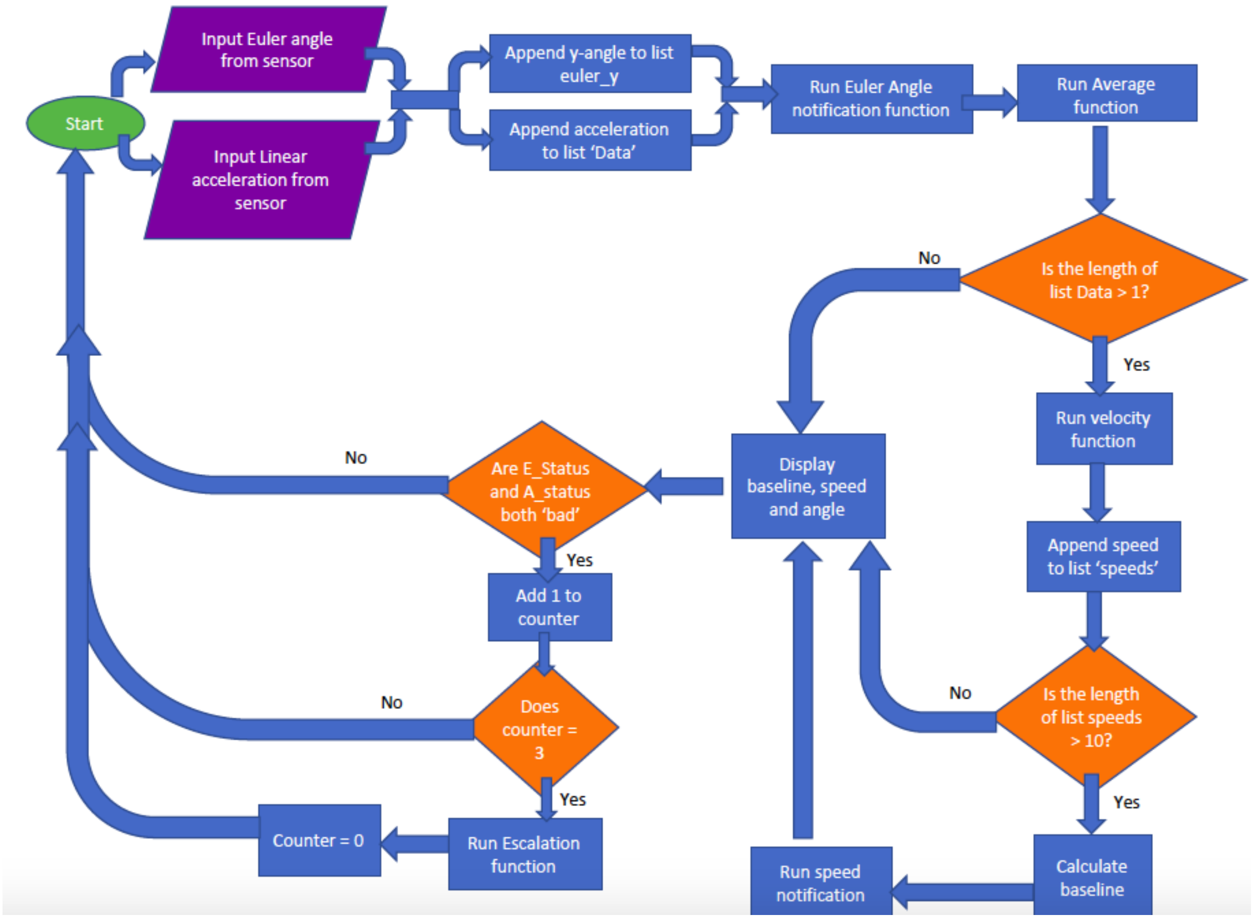


Figure 16: Main Function Flowchart



Python Code Printout

```
import time
import busio
import board
import smbus
import Adafruit_GPIO.SPI as SPI
import Adafruit_SSD1306

from PIL import Image
from PIL import ImageDraw
from PIL import ImageFont
from math import pow
from sensor_library import Orientation_Sensor
from gpiozero import LED
from gpiozero import Button

## We can now read values from the sensor by calling the appropriate
method:
## Calling each method only returns a single value. To retrieve data
continuosly, call the method inside an infinite loop. #while True
sensor =  Orientation_Sensor()
sensor.euler_angles()
sensor.lin_acceleration()

#Install Library files by typing "sudo pip3 install adafruit-
circuitpython-bno055" at the Terminal.
class Orientation_Sensor(object):
    def euler_angles(self): # 3-axis orientation in angular degrees
        return self.bno055.euler
    def lin_acceleration(self): # 3-axis vector of linear acceleration
in m/s^2 (acceleration - gravity)
        return self.bno055.linear_acceleration



#Raspberry pi (LED):
green_led = LED(18)
red_led = LED(7)
blue_led = LED(21)

#Raspberry pi (Button):
user_button = Button(26)

##OLED
# Raspberry Pi pin configuration:
RST = 24
# Note the following are only used with SPI:
DC = 23
SPI_PORT = 0
SPI_DEVICE = 0

# 128x32 display with hardware I2C:
disp = Adafruit_SSD1306.SSD1306_128_32(rst=RST)

# Initialize library.
```

```
disp.begin()

# Create blank image for drawing.
# Make sure to create image with mode '1' for 1-bit color.
width = disp.width
height = disp.height

image = Image.new('1', (width, height))

# Get drawing object to draw on image.
draw = ImageDraw.Draw(image)

# First define some constants to allow easy resizing of shapes.
padding = 2
shape_width = 20
top = padding
bottom = height-padding

# Move left to right keeping track of the current x position for
drawing shapes.
x = padding

# Load default font.
font = ImageFont.load_default()
###########################

# Function displays baseline, speed, and angle values determined in
each loop on the OLED
def display(baseline, v_i, euler_y):
    draw.rectangle((0,top+10,width,height), outline=0, fill=0)
    draw.text((x, top),    'Baseline',  font=font, fill=255)
    draw.text((x, top+10),str(abs(baseline)), font=font, fill=255)
    draw.text((x+80, top),    'Speed',  font=font, fill=255)
    draw.text((x+80, top+10),str(abs(round((v_i),2))), font=font,
fill=255)
    draw.rectangle((0,20,width,height), outline=0, fill=0)
    draw.text((x, top+20),    'angle Value',  font=font, fill=900)
    draw.text((x+80, top+20),str(euler_y[-1]), font=font, fill=255)
    disp.image(image)
    disp.display()



## Task 1: Status
#Finding the average of the n most recent items in the list
#It appends the n most recent into average_list, n being 5 in this
case
#If the number of items in the list equals 5, the average will be
calculated
#This is done by taking the sum of the list values, then dividing it
by the length, which is fixed at 5
#If the average is coming in a green light will turn on
def average(data): # data is the list where we put all of our sensor
values
    average_list = []
    avg_sensor_value = 0
```

```
    if len(data)>=5:
        for i in range(-5,0): ## The 5 most recent items.
            average_list.append(data[i])

        if len(average_list) == 5:
            avg_sensor_value = sum(average_list)/len(average_list)
            avg_sensor_value =round(avg_sensor_value,2)
            green_led.on()


    else:
        print("No data")


    return avg_sensor_value

## Task 2: Speed Notification

#assuming initial velocity is zero
#Use kinematic formula Vf = Vi + at
#A new final velocity value will be produced and overwrite the old v_i
variable
##This value is called v_f and returned as such, but reassigned to v_i
when the actual function is called
##v_i is used to process the need for a speed notification in the next
function
#The function then loops
#When acceleration is a near-zero value, we assume that we have
completely stopped (v_i = 0)
def velocity(v_i, acceleration):
    time= 0.01

    if abs(acceleration) < 0.3:
        acceleration = 0
        v_i = 0

    v_f = v_i + acceleration * time

    return v_f

##Inputs baseline from the main function and the speed calcualted in
the velocity function
#Determines what is 30% of the baseline annd assigns that to "percent"
#Calculates the difference between baseline and secondary
#Normal: green LED is on and blue LED is off
#Abnormal: if the difference is greater than percent, blue LED is on
def speed_notification(baseline, v_i):
    A_status= 'good'
    if abs(v_i) <= 5 and abs(v_i) > 0: ##avg is value from
average(data)
        difference = abs(abs(baseline) - abs(v_i))
        percent = 0.50* abs(baseline)

        if difference > percent:
            blue_led.on()
            A_status = "bad"
```

```
        else:
          print("Normal results")
          A_status = 'good'
          blue_led.off()
    elif abs(v_i)> 5:
        blue_led.off()
        print('Speed out of range')
    else:
        print ("you're not moving")
        blue_led.off()

        A_status='good'


    return A_status



#Task 4: additional notification of the euler angle value
##Detects an abnormal hip angle while walking
##As hips rock, axis on y-plane of body will slant from one side to
the other, prducing negative or positive angles
# If a negative angle is read, a value -1 is appended into the list
counter
## similarily, if a positive angle is read, a value +1 will be
appended into the list counter
##The list sum is taken once the counter has 10 values
#Normal walk: Equal amount time spent on each hip side. Therefore sum
of counter is zero (Red LED off)
#Abnormal walk: Unequal amount of time spent on each hip. Therefore,
sum accumulates as negative or positive (Red LED on)
##For the sake of data speed input and potential errors in readings,
theres an allowance of an absolute value of the sum of 2 before the
sensor is set off
## A variable E_status is defined and given a string of 'good' or
'bad', related to normal or abnormal respectively. This is returned
for the escalation portion later

def euler_angle_notification(counter, euler_y):
    E_status='good'
    if len(counter) < 10:
        if euler_y[-1] < 0:
            counter.append(-1)
        elif euler_y[-1] > 0:
            counter.append(1)

    else:
        angle_difference = sum(counter)
        if abs(angle_difference) > 2 and abs(angle_difference) <= 9:
            red_led.on()
            list.clear(counter)
            E_status = 'bad'


        elif abs(angle_difference) > 9:
            E_status = "good"
```

```
            red_led.off()
            list.clear(counter)
        else:
            red_led.off()
            list.clear(counter)
            E_status = 'good'


    return E_status

##Task 3 Escalation
#Requires user input to acknowledge abnormal speed and angle detected
##being called only when both A_status and E_status are assigned 'bad'
#If both speed and angle are bad (RED & BLUE light ON)
##The escalation portion causes the program to stop signal to the user
that theres an issue with the gait pattern
#A button needs to be pressed which is indicated by flashing red light
#When pressed red LED turns off
def escalation():
    print ("Escalating!")
    while user_button.is_pressed == True:  ## while the button is NOT
PRESSED
        blue_led.off()
        red_led.on()
        time.sleep(0.5)
        red_led.off()
        time.sleep(0.5)

    print("Button is pressed")  ##button becomes False -> pressed
    red_led.off()

def main():
    try:
        ##These blank/initial values allow for these prexisting value
sto change over time
        data = []
        euler_y = [0]
        counter = []
        counter_2 = 0
        speeds = [0]
        A_status = 'good'
        E_status= 'good'
        v_i = 0
        baseline = 0
        acceleration = 0

        ##while true loop allows for function to run indefinitely as
sensor value is looped and redefined with every loop, allowing for new
calculations
        while True:

            #Assigns the the euler angle from the sensor into a
variable called "euler_angles"
            euler_angles = sensor.euler_angles()


            #If the y-euler angle is not "None", it is added into a
```

```
list called "euler_y"
            if euler_angles[1] != None:
                euler_y.append(int(euler_angles[1]))


                #If the last value euler_y is not none
                #Calls the returned value from Euler notification
function and overwirtes the old E_status value
                if euler_y[-1] != None:
                    E_status = euler_angle_notification(counter,
euler_y)

            #Assigns linear acceleration from the sensor to
lin_acceleration
            #Adds values that are not none into a list called data
            lin_acceleration = sensor.lin_acceleration()

            if lin_acceleration[2] != None:
                data.append(lin_acceleration[2])

                acceleration = average(data)

            #Calculates the velocity from the raw acceleration values
from the sensor
                if len(data)>1:
                    v_i = velocity (v_i, lin_acceleration[2])
                    speeds.append(round(v_i,2))

            #Calculates baseline with the first 10 items on the list
"speeds"
            #Overwrites A_status with returned value from
speed_nofication
            if len(speeds)>10:
                baseline = sum(speeds[0:10])/10
                baseline = round(baseline,2)

                A_status = speed_notification(baseline, v_i)

            #Activates OLED and displays baseline,angle, and speed
            if euler_y[-1] != None:
                display(baseline, v_i, euler_y)

            #Calls escalation when E_staus and A_status are both
"bad".
            if E_status == 'bad' and A_status == 'bad':
                counter_2 +=1
                if counter_2 == 3:
                    escalation()
                    counter_2=0

            print('\n',"Acceleration", acceleration, '\t',
'speed',v_i, '\t', "Velocity Status", A_status)
            print ('\n','angle', euler_angles[1], '\t', "Euler
Status:", E_status)
            print('\n', "counter",counter)

            time.sleep(0.01)    #allows for a pause in the program
```

```
before loopingthis allows for calculation of speed in the velocity
function using the same value for change in time
    except OSError:
        print("Sensor Malfunction, please restart the program")
    #Stops the code when there is OS Error

main()
```

Figure 17: Breadboard Housing Engineering Drawing

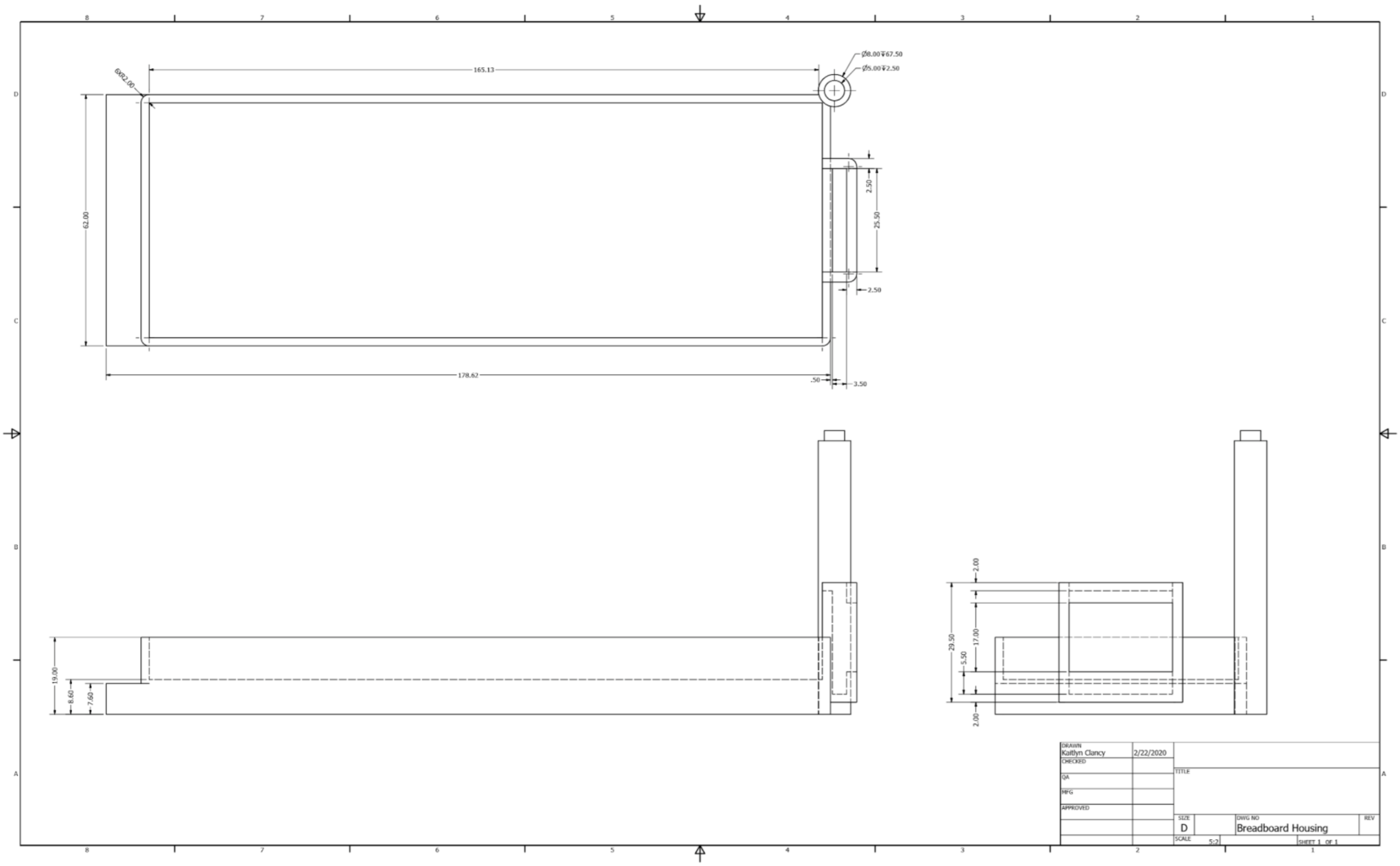


Figure 18: Raspberry Pi Housing Engineering Drawing

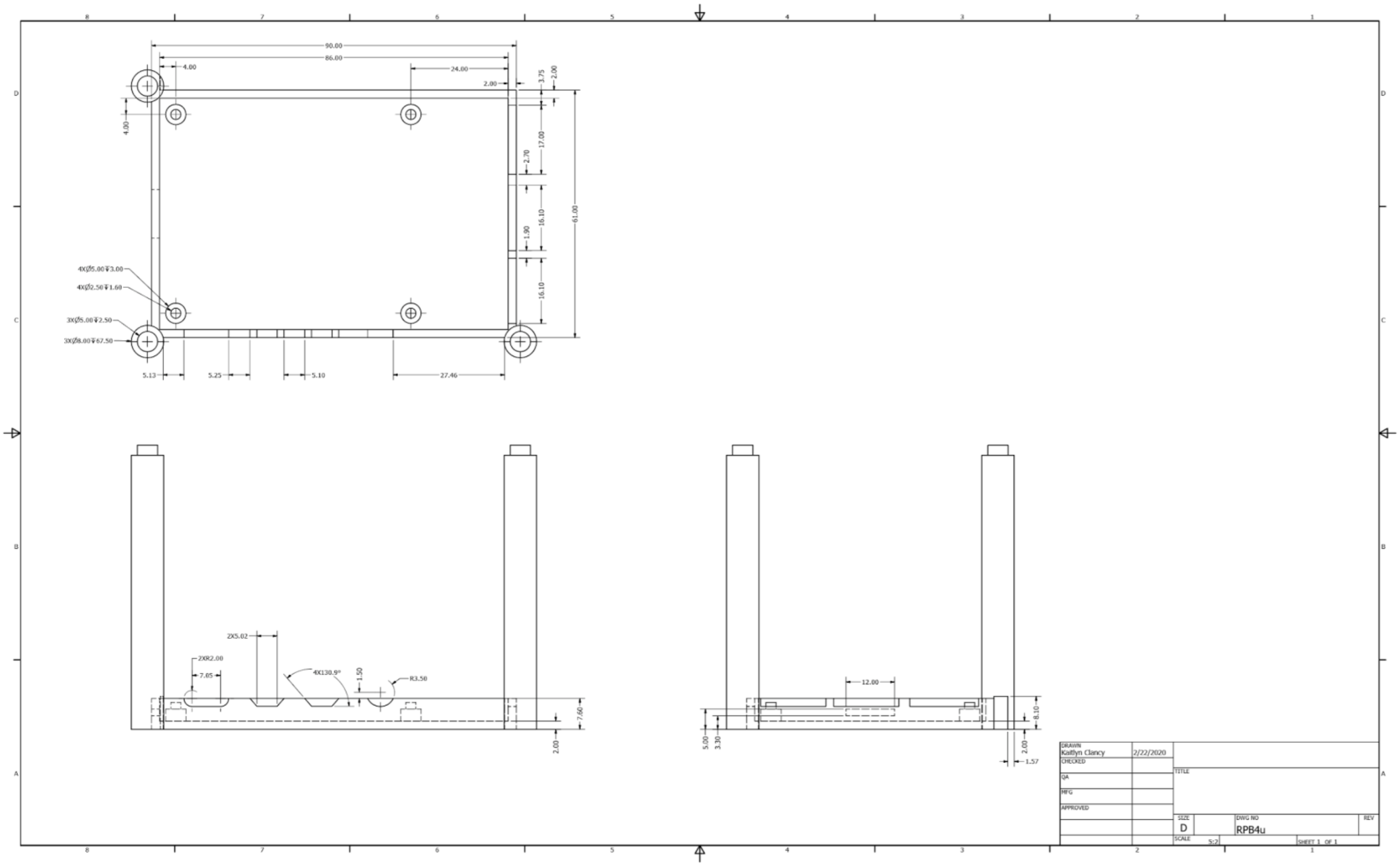


Figure 19: Sensor Housing 1 Engineering Drawing

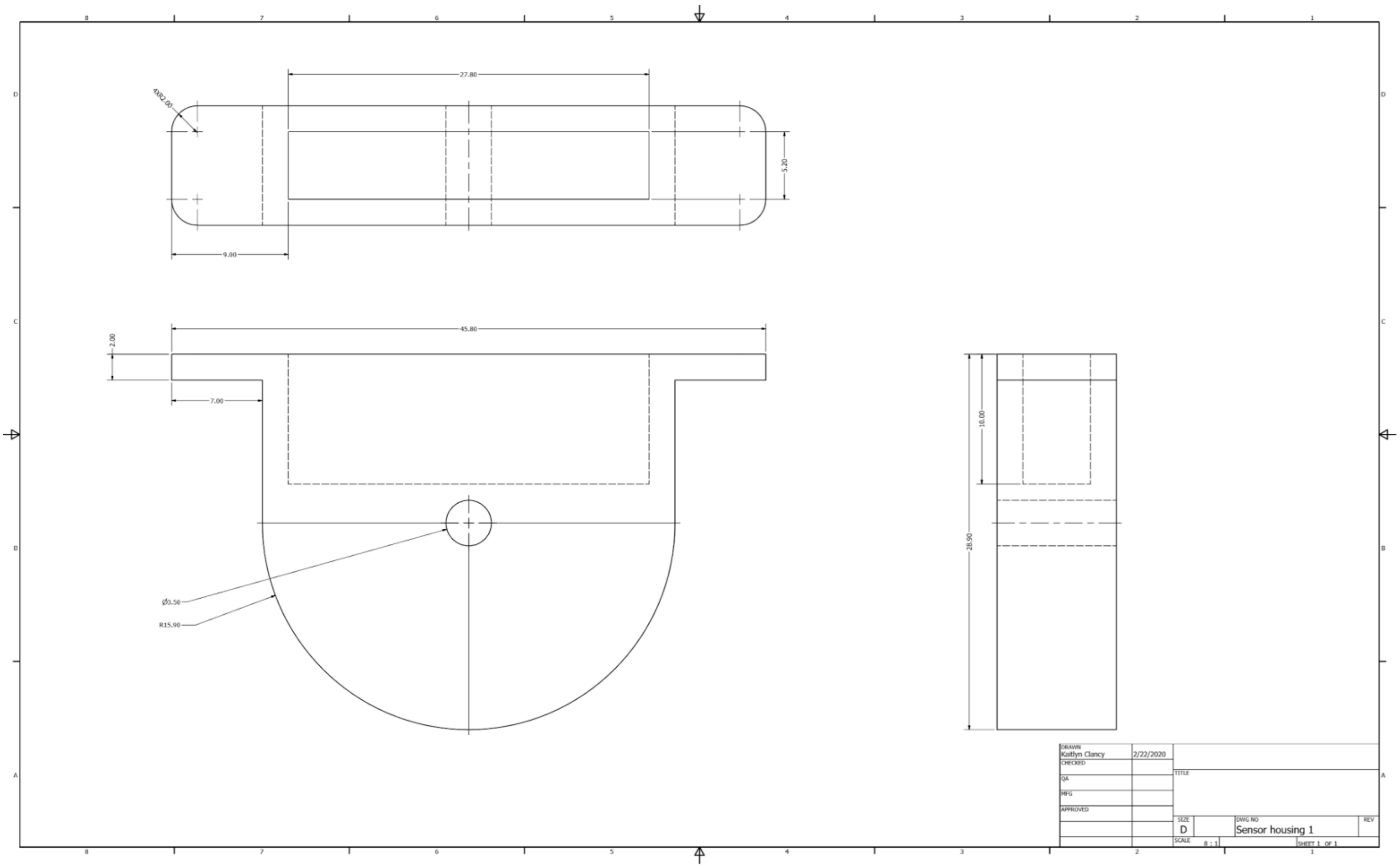


Figure 20: Sensor Housing 2 Engineering Drawing

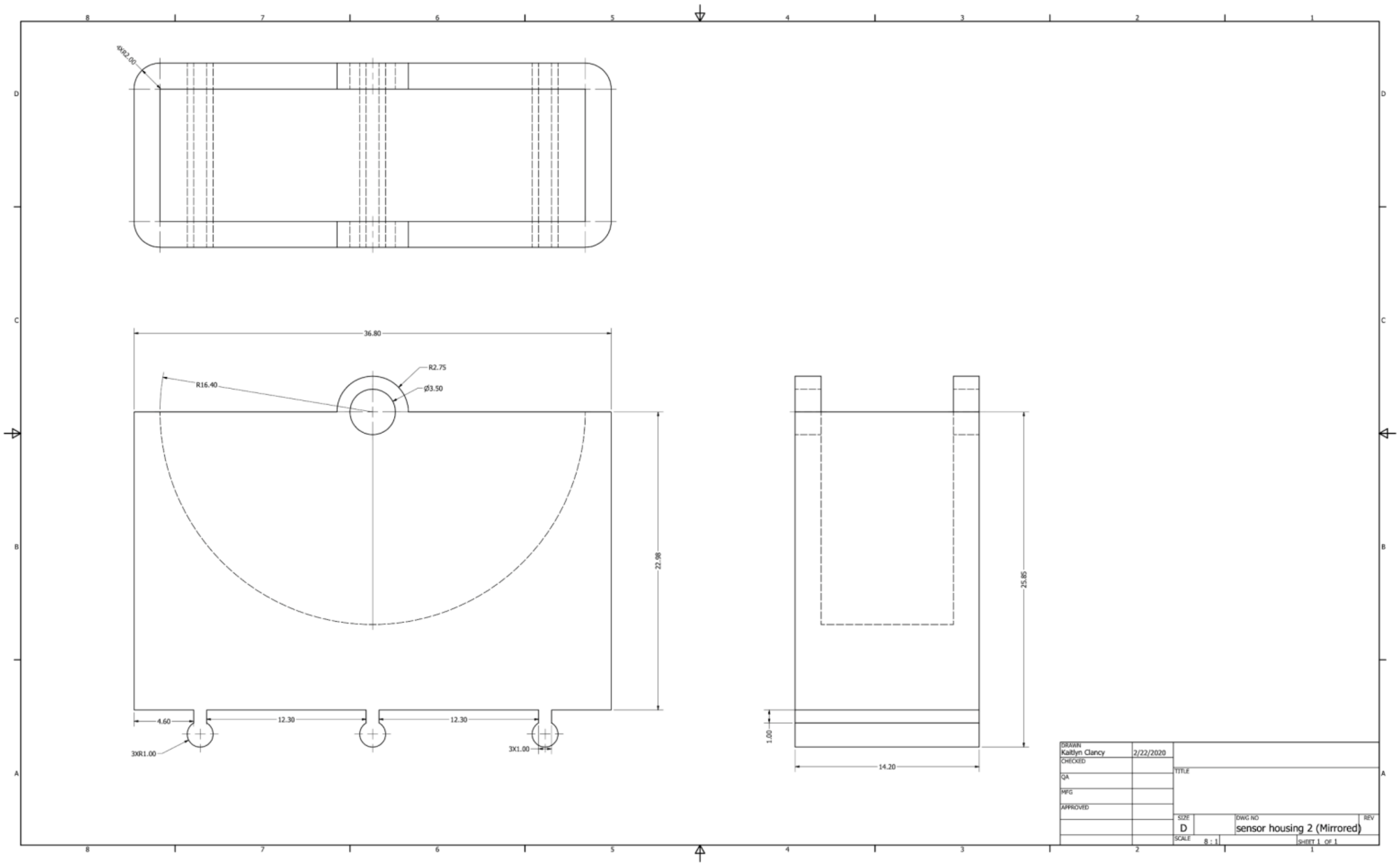


Figure 21: Track Engineering Drawing

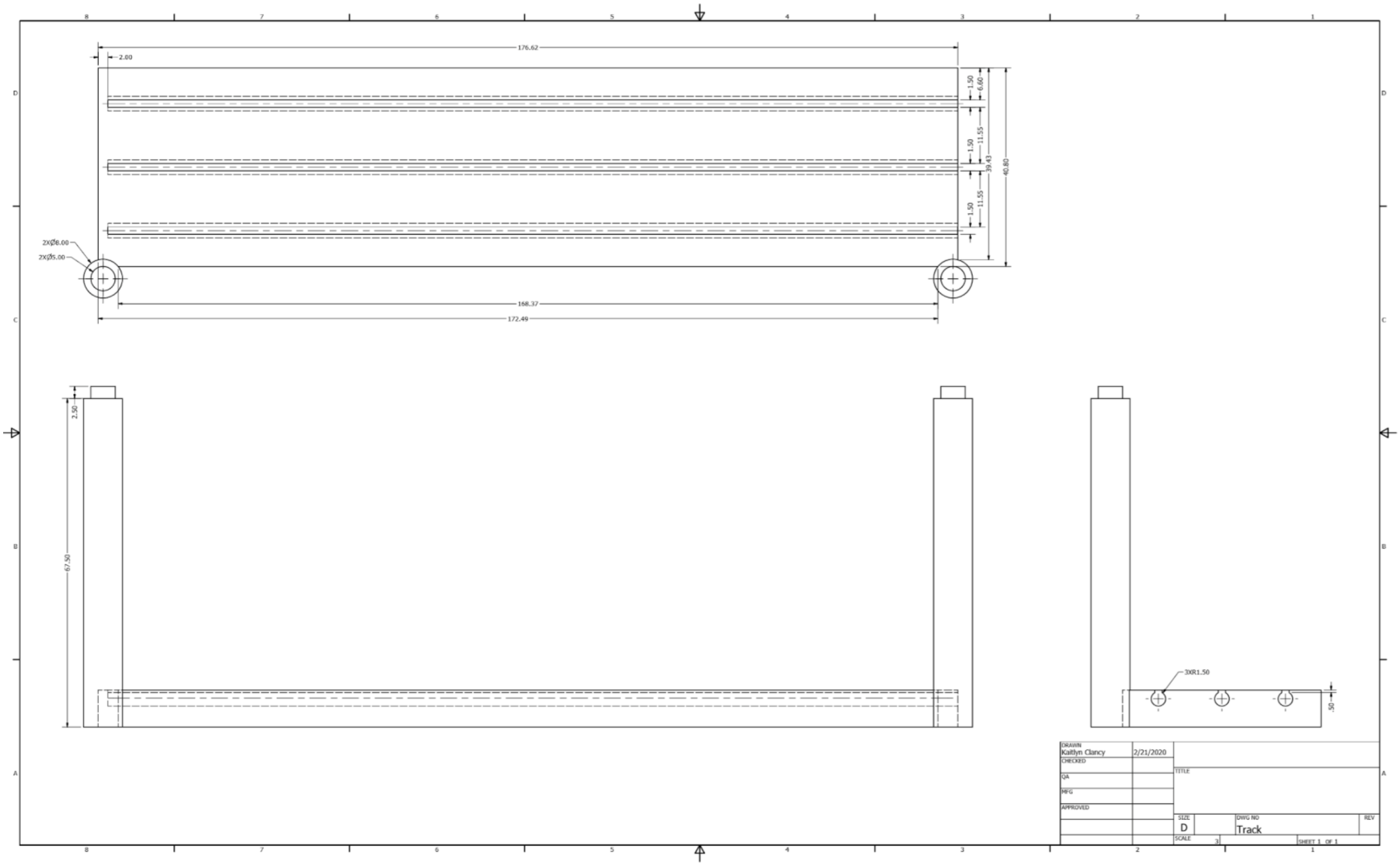


Figure 22: Peg Piece Engineering Drawing

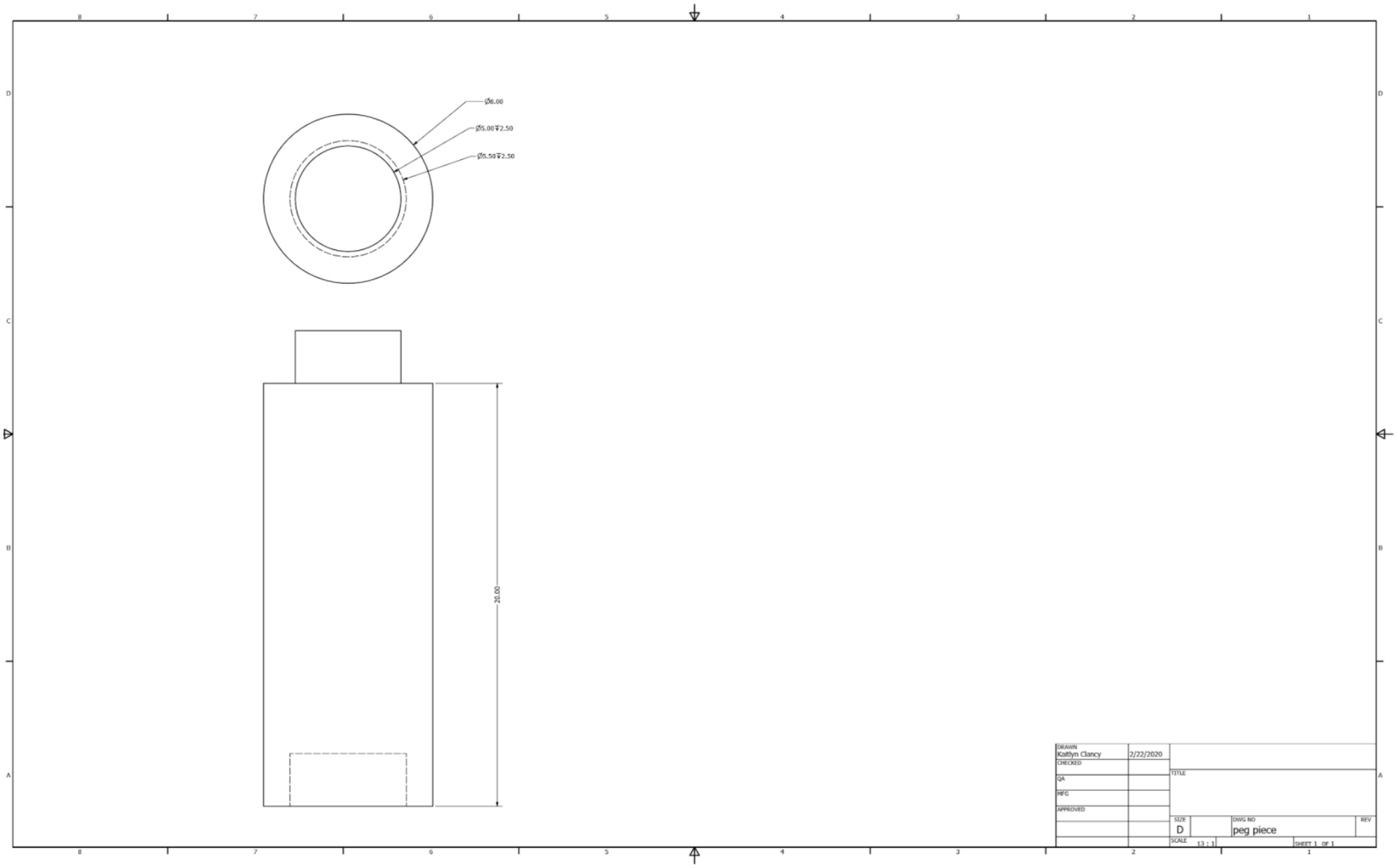
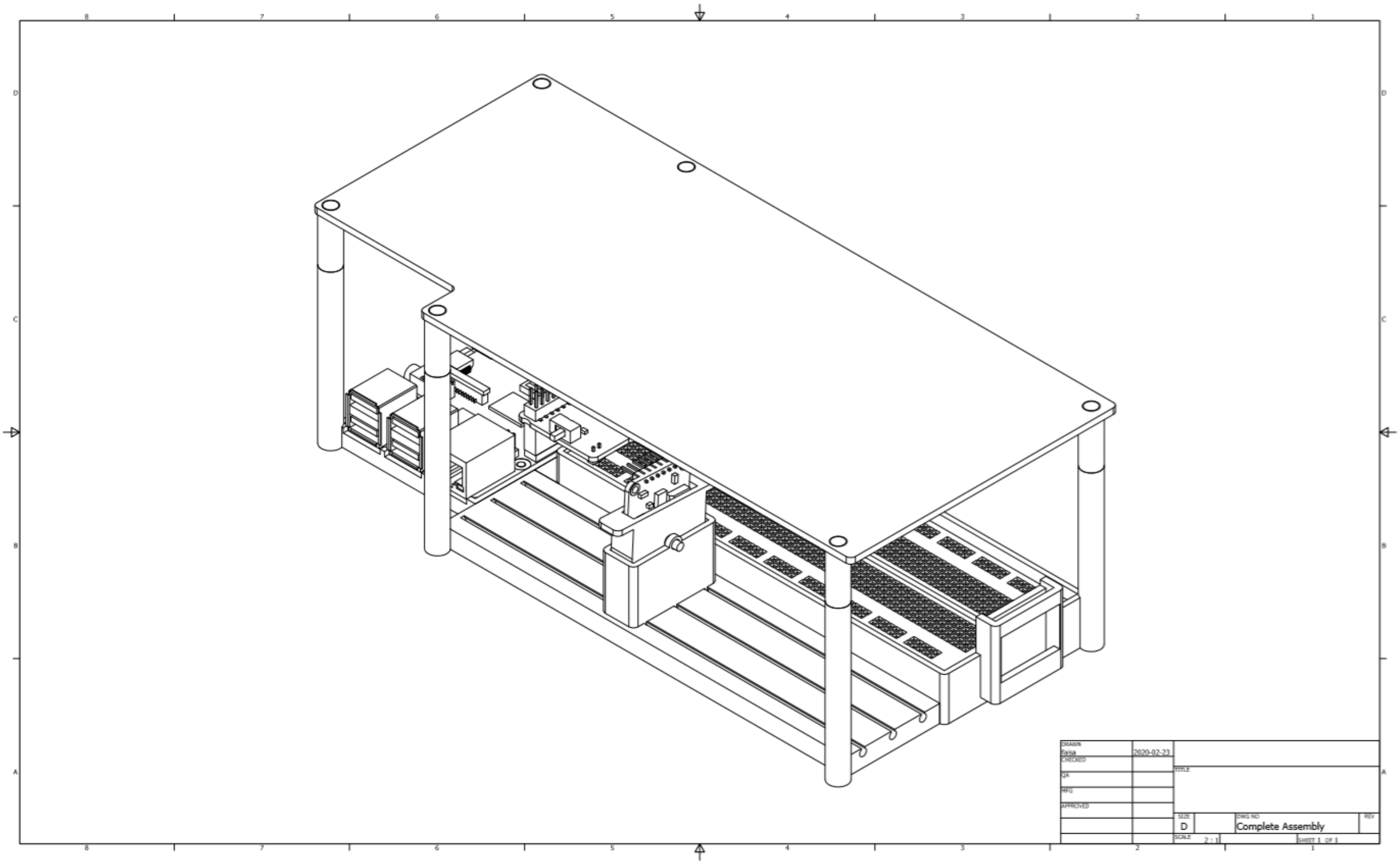


Figure 23: Physical Computing Prototype Assembly Drawing



Additional Documentation

Table 4a: List of Modelling Sub-Team Contributions

Kaitlyn Clancy	Brainstormed CAD model housings Actively participated in retrieving any additional materials needed Modelled components of CAD housing Conducted measurements on online modelled pieces as well as 3D-printed pieces and other tangible components (ie raspberry pi and breadboard) Manipulated dimensions to account for 3D print errors Created final tangible CAD assembled model Helped and assisted in the creation of Engineer Drawings Co-wrote Design Description and Design Specifications Contributed to editing and formatting documentation Created and Designed Preliminary and Final Gantt Charts
Faisal Khaleel	Developed original wearable device idea using preliminary sketch and preliminary flowchart. Brainstormed and modelled CAD housings for the orientation sensor and the breadboard. Modelled physical computing prototype motion design, including a track as well as a 2-piece sensor housing system. Modelled and printed 3D parts for the tangible prototype. Assembled tangible prototype and helped to create a video.Wrote a background and research summary. Integrated the background and research summary into the written report. Wrote the market analysis section within the written report and finished the design specifications section. Organized the appendices and overall formatting.
Meira Morphet	Brainstormed CAD model housing Took measurements for multiple components of the housing Proposed idea for track component and sensor housing of CAD model Created Engineering Drawings for housing parts and tangible prototype Assisted with the 3D printing for all of the modelled parts Co-wrote/edited Design Description and Design Specifications Shot and edited the tangible prototype video Wrote Summary of Team Meetings

Table 4b: List of Coding Sub-Team Contributions

Evelyn Cudmore	Brainstormed angle function and speed calculation function Coded Angle, escalation and display functions Co-Coded main function Breadboarded Wrote background research summary Wrote “Design Critique, Discussion, and Recommendations” Worked on report formatting Organized main, escalation, and angle notification flowcharts Tested and troubleshot code Completed all citations Wrote source Materials database
Chelsea Maramot	Wrote Design Verification Finalized and wrote the test plan Designed the schematic diagram of the Raspberry Pi Organized speed notification, acceleration, and average flowcharts Wrote Executive Summary Co-Coded main function Debugged and tested the code Breadboarded Coded task 1 (status) and task 2 (speed notification)

Summary of Team Meetings

Designated Design Studio Time:

Tuesday Jan 6, 2020
- Project roles and chairs/coordinators for the duration of the project discussed
- Preliminary list of ideas touched upon and further meeting date discussed

Tuesday Jan 13, 2020
- Individual ideas discussed and Go No Go Screening chart created to narrow down list of ideas

Tuesday Jan 20, 2020
- Parts list generated
- Test plan created
- Support Frame sketches created

Tuesday Jan 27, 2020
- Met with our two DP advisors to discuss our proposed design and get feedback

Extra Meetings

Friday Jan 10, 2020
- All members present at this meeting to discuss our brainstorming ideas, narrow it down and come up with a need statement as well as an end-user

Friday Jan 24, 2020
- Members of the CAD team met to discuss potential housing options
- Concluded that we wanted to CAD separate housing for the breadboard and raspberry pi and then assemble them together
- Decided to brainstorm the creative component individually and reconvene later to present our ideas and decide which one to pursue

Friday Jan 24, 2020
- Members of the coding team met to discuss aspects of the code
- Agreed to code each of the parts individually and reconvene at a later time to discuss the methods used and decide upon the best option

Monday Jan 27, 2020
- Members of the coding team met online to discuss certain aspects of the code that were confusing or could be executed a number of different ways

Friday Jan 31, 2020
- Members of the CAD team present to discuss the creative aspect that they had brainstormed; ultimately decided to use a glass sheet
- Modelling responsibilities were divided up
    - Kaitlyn: raspberry pi housing, poles to support glass sheet, and gathering glass materials
    - Faisal & Meira: breadboard housing, orientation sensor housing, o-LED housing and CAD-ing the outline of the glass sheet

Saturday Feb 1, 2020
- Meira and Faisal present to begin CAD on the breadboard and sensor housing

Monday Feb 3, 2020

- Kaitlyn in the design studio printing the raspberry pi housing and the sensor piece

Thursday Feb 6, 2020

- Meira and Faisal present to CAD the housing for o-LED

Friday Feb 7, 2020

- Meira and Kaitlyn present to print the breadboard housing and the sensor track with o-LED housing
- Sensor housing was also reprinted with the newly filleted edge model

Friday Feb 7, 2020

- All members of coding team present for online meeting to discuss last minute fixes as well as a time to meet to go over the code one last time

Thursday Feb 13, 2020

- Meira and Kaitlyn present to reprint the sensor housing with the hole (to increase security)
- The “lego pieces” to give the poles an extra few centimeters of height were also printed
- Pieces were shaved down and modified to fit appropriately

Friday Feb 14, 2020

- All members present to discuss presentation details
- Document created with general answers to questions asked during the EXPO
- Coding team described operational conditions to CADing team
- “Mock” expo was rehearsed and dress code discussed

Figure 24: Preliminary Gantt Chart

DP-3 PRELIMINARY GANTT CHART

PROJECT TITLE: TRACK AND FIELD	Design Solutions for Applications in Health, Fitness, and Active Liv	**TEAM:**	Kailtyn Clancy, Evelyn Cudmore, Meira Morphet,
PROJECT MANAGER	Chelsea Marmot		Chelsea Maramot, and Faisal Khaleel

	Milestones	TASK OWNER	START DATE	DUE DATE	WEIGHT	PCT OF TASK COMPLETE
	Milestone 0: Team Development				5%	
	Team Charter	Team	1/7/20	1/7/20		100%
	Milestone 0 Submission	Chelsea	1/7/20			100%
	Milestone 1: Problem Definition				5%	
	Criteria Definition	Team	1/7/20	1/7/20		100%
	Need Statement	Team	1/7/20	1/10/20		0%
	Preliminary List of Ideas	Team	1/7/20	1/10/20		0%
	Milestone 1 Submission	Kaitlyn	1/7/20	1/14/20		0%
	Milestone 2: Conceptual Design				15%	
	Consept Sketch and Deisgn Flowchart	Individual	1/9/20	1/14/20		0%
	Go, No-Go Screening	Team	1/14/20	1/14/20		0%
	Design Matrix	Team	1/14/20	1/14/20		0%
	Concept Sketch and Design Flowchart Submission	Individual	1/14/20	1/14/20		0%
	Go, No-Go Screening and Design Matrix Submission	Kaitlyn	1/14/20	1/14/20		0%
	Milestone 3: Preliminary Design				P/F	
	Parts List	Team	1/21/20	1/21/20		0%
	Test Plan	Computation Team	1/21/20	1/21/20		0%
	Support Frame Sketch	Modeling Team	1/21/20	1/21/20		0%
	Milestone 3 Submission	Kaitlyn	1/21/20	1/21/20		0%
	Milestone 4-A: Detail Design				P/F	
	Low-Fidelity Prototype	Team	1/21/20	1/28/20		0%
	Preliminary Design Review	Individual	1/28/20	1/28/20		0%
	Presentation	Team	01/28/20	01/28/20		0%
	Milestone 4-A Sumission	Kaitlyn	1/28/20	1/28/20		0%
	Milestone 4-B: Detailed Design				5%	
	Computer Program	Individual	1/27/20	1/27/20		0%
	Test Plan	Individual	1/27/20	1/27/20		0%
	Milestone 4-B Submission	Individual	1/27/20	1/27/20		0%
Submissions due on the given dates by administartor	**Final Submission: Design Demonstration and Verification**				60%	
	Computer Program	Computation Team	1/27/20	2/14/20	30%	0%
	Assembly Model	Modelling Team	1/27/20	2/14/20		0%
	3D-Printed Assembly	Modelling Team	1/27/20	2/14/20	30%	0%
	Tangible Prototype	Team	1/27/20	2/24/20	10%	0%
	Written Technical Report	Team	1/27/20	2/24/20	15%	0%
	Final Computing Prototype	Computation Team	1/27/20	2/24/20		0%
	Learning Portfolio and Reflection	Individual	1/27/20	2/28/20	10%	0%
	Self- and Peer-Evaluation	Individual	2/24/20	2/28/20	5%	0%

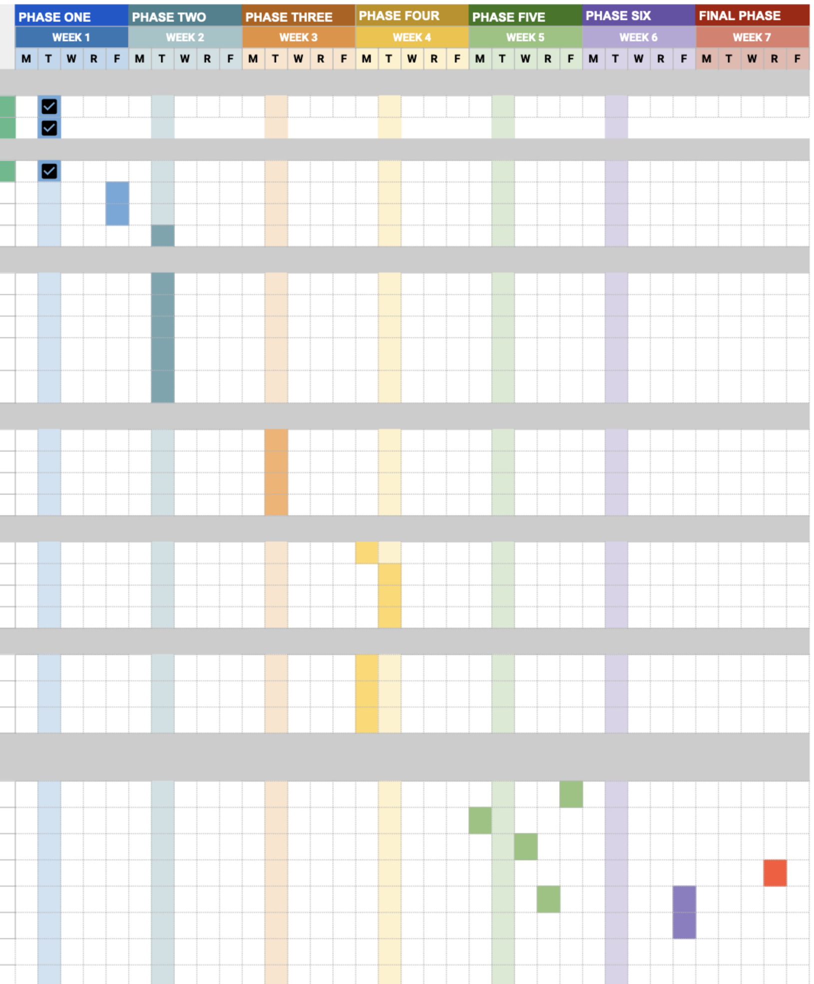
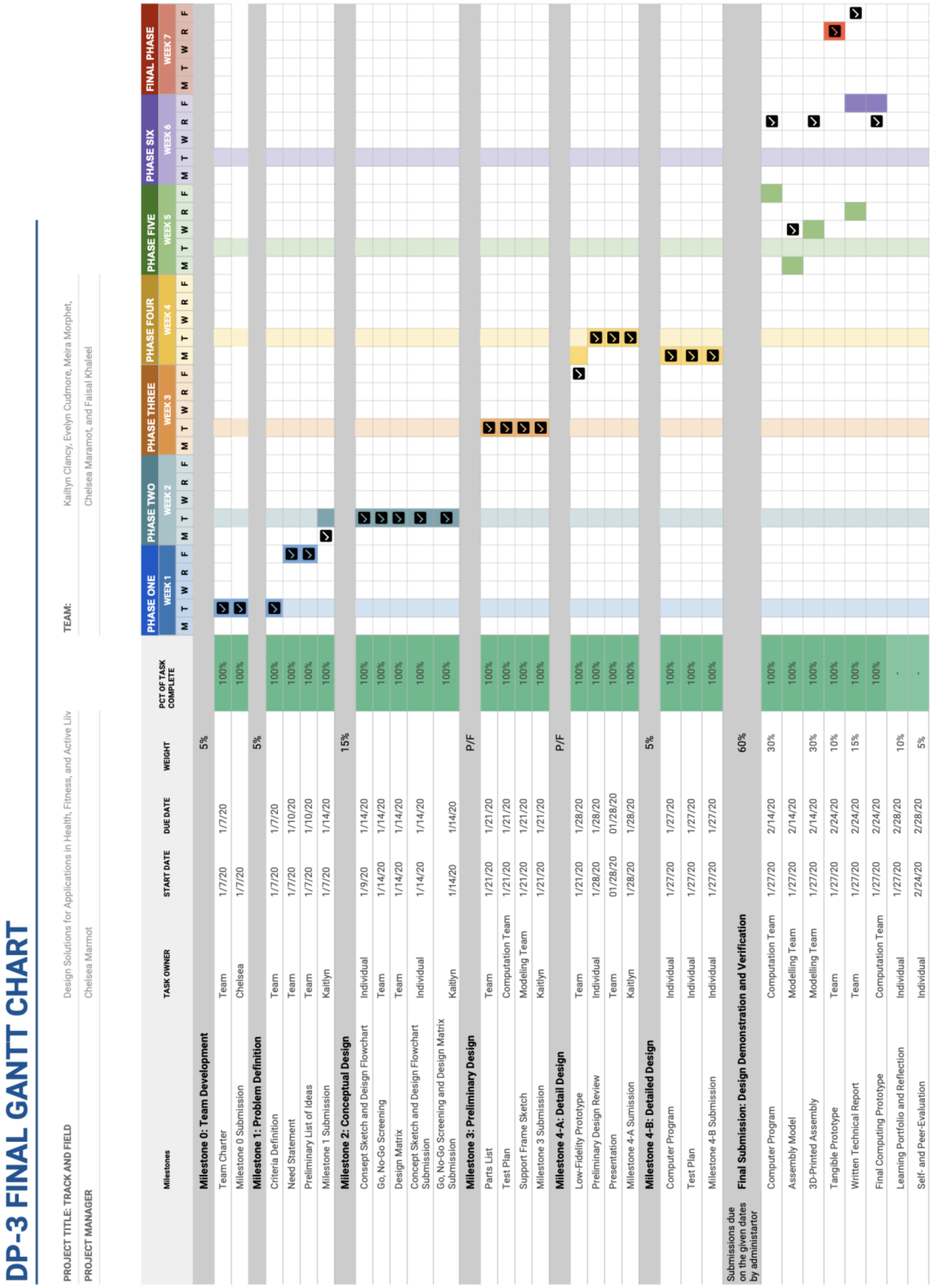


Figure 25: Final Gantt Chart



DP-3 FINAL GANTT CHART

PROJECT TITLE: TRACK AND FIELD — Design Solutions for Applications in Health, Fitness, and Active Liv

PROJECT MANAGER — Chelsea Marmot

TEAM: Kaitlyn Clancy, Evelyn Cudmore, Meira Morphet, Chelsea Maramot, and Faisal Khaleel

PHASE ONE (WEEK 1) · PHASE TWO (WEEK 2) · PHASE THREE (WEEK 3) · PHASE FOUR (WEEK 4) · PHASE FIVE (WEEK 5) · PHASE SIX (WEEK 6) · FINAL PHASE (WEEK 7) — M T W R F

Milestones	TASK OWNER	START DATE	DUE DATE	WEIGHT	PCT OF TASK COMPLETE
Milestone 0: Team Development				5%	
Team Charter	Team	1/7/20	1/7/20		100%
Milestone 0 Submission	Chelsea	1/7/20			100%
Milestone 1: Problem Definition				5%	
Criteria Definition	Team	1/7/20	1/7/20		100%
Need Statement	Team	1/7/20	1/10/20		100%
Preliminary List of Ideas	Team	1/7/20	1/10/20		100%
Milestone 1 Submission	Kaitlyn	1/7/20	1/14/20		100%
Milestone 2: Conceptual Design				15%	
Consept Sketch and Deisgn Flowchart	Individual	1/9/20	1/14/20		100%
Go, No-Go Screening	Team	1/14/20	1/14/20		100%
Design Matrix	Team	1/14/20	1/14/20		100%
Concept Sketch and Design Flowchart Submission	Individual	1/14/20	1/14/20		100%
Go, No-Go Screening and Design Matrix Submission	Kaitlyn	1/14/20	1/14/20		100%
Milestone 3: Preliminary Design				P/F	
Parts List	Team	1/21/20	1/21/20		100%
Test Plan	Computation Team	1/21/20	1/21/20		100%
Support Frame Sketch	Modeling Team	1/21/20	1/21/20		100%
Milestone 3 Submission	Kaitlyn	1/21/20	1/21/20		100%
Milestone 4-A: Detail Design				P/F	
Low-Fidelity Prototype	Team	1/21/20	1/28/20		100%
Preliminary Design Review	Individual	1/28/20	1/28/20		100%
Presentation	Team	01/28/20	01/28/20		100%
Milestone 4-A Sumission	Kaitlyn	1/28/20	1/28/20		100%
Milestone 4-B: Detailed Design				5%	
Computer Program	Individual	1/27/20	1/27/20		100%
Test Plan	Individual	1/27/20	1/27/20		100%
Milestone 4-B Submission	Individual	1/27/20	1/27/20		100%
Submissions due on the given dates by administartor — **Final Submission: Design Demonstration and Verification**				60%	
Computer Program	Computation Team	1/27/20	2/14/20	30%	100%
Assembly Model	Modelling Team	1/27/20	2/14/20		100%
3D-Printed Assembly	Modelling Team	1/27/20	2/14/20	30%	100%
Tangible Prototype	Team	1/27/20	2/24/20	10%	100%
Written Technical Report	Team	1/27/20	2/24/20	15%	100%
Final Computing Prototype	Computation Team	1/27/20	2/24/20		100%
Learning Portfolio and Reflection	Individual	1/27/20	2/28/20	10%	-
Self- and Peer-Evaluation	Individual	2/24/20	2/28/20	5%	-