

Remote Penguin Pulse Oximeter

EEE4113F

Group 34



**Prepared by:**

Priya Moodley MDLPRI040

Kashan Pillay PLLKAS005

Liyana Singh SNGLIY001

**Prepared for:**

EEE4113F

Department of Electrical Engineering

University of Cape Town

May 25, 2025

# Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.



May 25, 2025

---

Liyana Singh

Date



May 25, 2025

---

Kashan Pillay

Date



May 25, 2025

---

Priya Moodley

Date

# Contents

|  |           |
|--|-----------|
| <b>List of Figures</b>   | <b>vi</b> |
| <b>1 Introduction</b>  | <b>1</b>  |
| 1.1 GitHub Link . . . . .  | 2         |
| <b>2 Problem Analysis</b>  | <b>3</b>  |
| <b>3 Literature Review</b>   | <b>5</b>  |
| 3.1 Introduction to the Monitoring of Health Status of Birds . . . . . | 5         |
| 3.2 Blood Pressure . . . . .   | 5         |
| 3.2.1 Catheterization . . . . .  | 6         |
| 3.2.2 Oscillometric Monitors . . . . .                                 | 6         |
| 3.2.3 Doppler Ultrasonic Flow Detectors . . . . .                      | 6         |
| 3.3 Heart Rate . . . . .   | 6         |
| 3.3.1 FM Radio Transmitter and Electret Microphone . . . . .           | 7         |
| 3.3.2 Eulerian Video Magnification of IRT (EVM) . . . . .              | 7         |
| 3.3.3 Electrocardiograms (ECG) . . . . .                               | 7         |
| 3.4 Weight . . . . .   | 8         |
| 3.4.1 Tag Identification . . . . .                                     | 8         |
| 3.4.2 Weighbridge . . . . .  | 8         |
| 3.4.3 Body Girth . . . . .   | 8         |
| 3.5 Body Temperature . . . . .   | 9         |
| 3.5.1 Thermocouple . . . . .   | 9         |
| 3.5.2 Infrared Thermography . . . . .                                  | 9         |
| 3.5.3 Passive Integrated Transponder (PIT) Tags . . . . .              | 10        |
| 3.6 Breathing . . . . .  | 10        |
| 3.6.1 Pulse Oximeter . . . . .   | 11        |
| 3.6.2 Magnetic Beak Sensors . . . . .                                  | 11        |
| 3.7 Conclusion . . . . .   | 11        |
| <b>4 Subsystem 1</b>   | <b>13</b> |
| 4.1 Introduction . . . . .   | 13        |
| 4.1.1 Context . . . . .  | 13        |
| 4.1.2 Scope and Limitations . . . . .                                  | 13        |
| 4.2 Requirements Analysis . . . . .                                    | 14        |
| 4.2.1 Requirements . . . . .   | 14        |
| 4.2.2 Specifications . . . . .   | 14        |
| 4.2.3 Testing Procedure . . . . .                                      | 15        |
| 4.2.4 Traceability Analysis . . . . .                                  | 15        |

|          |  |           |
|----------|--|-----------|
| 4.3      | Design . . . . .                                   | 16        |
| 4.3.1    | Light Emission Module . . . . .                    | 16        |
| 4.3.2    | Light Detection and Amplification Module . . . . . | 18        |
| 4.3.3    | Final Design . . . . .                             | 20        |
| 4.3.4    | System Integration and Interfacing . . . . .       | 21        |
| 4.4      | Results . . . . .                                  | 22        |
| 4.4.1    | Tests . . . . .                                    | 22        |
| 4.4.2    | Results Analysis . . . . .                         | 22        |
| 4.5      | Conclusion . . . . .                               | 24        |
| 4.5.1    | Recommendations . . . . .                          | 24        |
| <b>5</b> | <b>Subsystem 2</b>                                 | <b>25</b> |
| 5.1      | Introduction . . . . .                             | 25        |
| 5.1.1    | Context . . . . .                                  | 25        |
| 5.1.2    | Scope and Limitations . . . . .                    | 25        |
| 5.2      | Requirements Analysis . . . . .                    | 26        |
| 5.2.1    | Requirements . . . . .                             | 26        |
| 5.2.2    | Specifications . . . . .                           | 26        |
| 5.2.3    | Testing Procedure . . . . .                        | 27        |
| 5.2.4    | Traceability Analysis . . . . .                    | 27        |
| 5.3      | Design . . . . .                                   | 28        |
| 5.3.1    | Power Module Component Selection . . . . .         | 28        |
| 5.3.2    | Location Module Component Selection . . . . .      | 29        |
| 5.3.3    | Processing Module . . . . .                        | 29        |
| 5.3.4    | Final Design . . . . .                             | 30        |
| 5.3.5    | System Integration and Interfacing . . . . .       | 32        |
| 5.4      | Results . . . . .                                  | 32        |
| 5.4.1    | Tests . . . . .                                    | 32        |
| 5.4.2    | Results Analysis . . . . .                         | 33        |
| 5.5      | Conclusion . . . . .                               | 36        |
| <b>6</b> | <b>Subsystem 3</b>                                 | <b>37</b> |
| 6.1      | Introduction . . . . .                             | 37        |
| 6.1.1    | Context . . . . .                                  | 37        |
| 6.1.2    | Scope and Limitations . . . . .                    | 37        |
| 6.2      | Requirements Analysis . . . . .                    | 38        |
| 6.2.1    | Requirements . . . . .                             | 38        |
| 6.2.2    | Specifications and Testing Procedure . . . . .     | 38        |
| 6.2.3    | Traceability Analysis . . . . .                    | 39        |
| 6.3      | Subsystem Design . . . . .                         | 40        |
| 6.3.1    | Design Decisions . . . . .                         | 40        |
| 6.3.2    | Final Design . . . . .                             | 44        |
| 6.3.3    | Failure Management . . . . .                       | 46        |
| 6.4      | Results . . . . .                                  | 47        |

|   |           |
|---|-----------|
| 6.4.1 Tests . . . . .                           | 47        |
| 6.4.2 Analysis of Testing and Results . . . . . | 48        |
| 6.5 Conclusion and Recommendations . . . . .    | 49        |
| <b>7 Conclusions</b>                            | <b>50</b> |
| 7.1 Recommendations . . . . .                   | 51        |
| <b>Bibliography</b>                             | <b>52</b> |
| <b>A Bill of Materials</b>                      | <b>55</b> |
| <b>B GA Requirements</b>                        | <b>56</b> |
| <b>C Subsystem 3: Appendices</b>                | <b>59</b> |

# List of Figures

|      |   |    |
|------|---|----|
| 2.1  | Context Diagram . . . . .   | 4  |
| 3.1  | An example of the use of an electret mic and fm transmitter in an artificial egg. [1] . . . . .   | 7  |
| 4.1  | Context diagram highlighting the interactions of Subsystem 1 . . . . .                            | 13 |
| 4.2  | Circuit diagram for Module 1 Option 1 . . . . .   | 16 |
| 4.3  | Circuit diagram for Module 1 Option 2 . . . . .   | 16 |
| 4.4  | Circuit diagram for Module 2 Option 1 . . . . .   | 19 |
| 4.5  | Circuit diagram for Module 2 Option 2 . . . . .   | 19 |
| 4.6  | Circuit diagram for Module 2 Option 3 . . . . .   | 19 |
| 4.7  | Final Schematic of Subsystem 1 . . . . .  | 21 |
| 4.8  | Vero-board for Module 1 . . . . .   | 21 |
| 4.9  | Vero-board for Module 2 . . . . .   | 21 |
| 4.10 | Interfacing Diagram of Subsystem 1 . . . . .  | 21 |
| 4.11 | Voltage across light emitter circuit ( $V_1$ ) . . . . .  | 23 |
| 4.12 | Voltage across light detection circuit ( $V_2$ ) . . . . .  | 23 |
| 4.13 | Current through light emitter circuit ( $I_1$ ) . . . . .   | 23 |
| 4.14 | Current through light detection circuit ( $I_2$ ) . . . . .                                       | 23 |
| 4.15 | PWM signals for red and infrared LEDs . . . . .   | 23 |
| 4.16 | Output signal from detection circuit . . . . .  | 24 |
| 5.1  | Context diagram that features the inputs and outputs for various modules of Subsystem 2 . . . . . | 25 |
| 5.2  | Subsystem 2 Interfacing Diagram with other Modules and other Subsystems . . . . .                 | 32 |
| 5.3  | Acceptance Test 01 Setup and Measurement . . . . .  | 34 |
| 5.4  | Measured Voltage (4.9063V) supplied to GPS Module . . . . .                                       | 35 |
| 5.5  | Measured Voltage (5.0522V) supplied to Light Detection and Amplification Module . . . . .         | 35 |
| 5.6  | Measured Voltage (3.2859V) supplied to Light Emission Module . . . . .                            | 35 |
| 5.7  | PWM signals generated by ESP32 . . . . .  | 35 |
| 5.8  | ADC Sampling reconstruction of square wave . . . . .  | 36 |
| 6.1  | A diagram indicating how Subsystem 3 interacts and fits in with the system as a whole. . . . .    | 37 |
| 6.2  | Isometric views of the three considered designs. . . . .  | 40 |
| 6.3  | Considered colour palettes. . . . .   | 44 |
| 6.4  | Orthographic views of the final design. . . . .   | 44 |
| 6.5  | Final System Housing Design. . . . .  | 45 |
| C.1  | The Home Page . . . . .   | 59 |
| C.2  | The Data Page . . . . .   | 59 |
| C.3  | The Tracking Page . . . . .   | 60 |
| C.4  | The cloud back end with random test data in it. . . . .   | 60 |

|   |    |
|---|----|
| C.5 The strap/harness utilised in section 6.4.2 . . . . .   | 60 |
| C.6 The retrieval of data on different networks as in section 6.4.2 . . . . .                       | 61 |
| C.7 The display of all historical data as required in section 6.4.2. . . . .                        | 61 |
| C.8 The accurate tracking of the gps coordinates (Figure C.9) as required in section 6.4.2. . . . . | 62 |
| C.9 The GPS coordinates required as per section 6.4.2. . . . .                                      | 62 |
| C.10 The illuminated warning panel as required in section 6.4.2. . . . .                            | 62 |
| C.11 The minimized window without any distortion as required in section 6.4.2. . . . .              | 63 |
| C.12 The hover feature as required in section 6.4.2. . . . .  | 63 |
| C.13 The hover feature as required in section 6.4.2. . . . .  | 63 |
| C.14 The weight of the system housing as per section 6.4.2. . . . .                                 | 64 |

# Chapter 1

## Introduction

This project was undertaken as part of the EEE4113F Design Class of 2025. African Penguins are critically endangered and are projected to become extinct by 2035 if effective conservation measures are not urgently implemented. In light of this crisis, BirdLife South Africa partnered with the EEE4113F class to communicate their conservation priorities, with the goal of inspiring innovative engineering solutions to support their work.

Our group, Group 34, chose to work with Alistair McInnes from the Seabird Conservation Programme at BirdLife South Africa. He highlighted the need for a system capable of monitoring the body condition and molt stage of African Penguins. Such a system would support a wide range of research and conservation activities. The collected data could inform ecological assessments by evaluating molt condition in response to prey availability and changing oceanographic conditions. Additionally, it would be useful for demographic studies, including survival and population modelling. The data would also help assess the efficacy of management interventions. Furthermore, information on the timing and synchronicity of molting could contribute to understanding breeding cycles and breeding success.

Monitoring African Penguins during molt presents key challenges. Identification is possible through RFID tags or spot patterns visible in later molt stages. Birds cannot be handled or disturbed, limiting device types and requiring non-invasive methods. Environmental factors like tides, swell, and weather affect device placement. Although molting birds move, they typically stay within a local area such as the slipway at Stony Point which is a promising monitoring site.

A literature review, which can be seen in [chapter 3](#), was conducted to explore various methods for assessing avian health metrics, with an emphasis on identifying techniques that are both feasible and minimally invasive. The research highlighted that blood oxygen saturation and pulse rate are critical indicators of a penguin's body condition. A remote pulse oximeter can be employed to measure these parameters and wirelessly transmit the data, minimizing disturbance to the bird during its molt. The device will be worn throughout the 21-day molting period to monitor changes in body condition across different stages of the molt. It will be attached prior to the molt and removed afterward. To ensure minimal impact, the device must be lightweight and unobtrusive.

To summarise, **Problem Statement:** Monitor the health during the molting stages of African penguins to understand how their condition varies throughout the molt and to inform ecological assessments, benefiting researchers, wildlife managers, and conservationists while ensuring that the birds are not disturbed during the molt.

**Solution Statement:** A lightweight remote pulse oximeter will monitor the pulse and blood oxygen saturation of African Penguins to understand how their condition varies throughout the molt to inform ecological assessments, benefiting researchers, wildlife managers, and conservationists while ensuring that the birds are not disturbed during the molt.

## 1.1 GitHub Link

Link to the [GitHub Repository](#)

# Chapter 2

## Problem Analysis

At the Design School (D-School), we learnt techniques for identifying and understanding problem spaces from a human-centered perspective. Rather than focusing on the solution, we focused on developing empathy for Alistair McInnes and understanding his needs. This process helped us frame a meaningful problem statement that aligned with the overall conservation goals. Next, we learned how to create effective prototypes to demonstrate our solution's functionality and received feedback to iterate accordingly. This skill was beneficial for our consultations with Stephen Paine where solutions were evaluated and improved.

Some design considerations were discussed and ultimately a complete design was developed and divided among us.

The lightweight remote pulse oximeter requires an external power source, as it will be attached to the penguin throughout the molting period. The placement of the device was carefully considered, with two options explored: the webbed foot or the wing. While placing the oximeter on the foot would allow light to pass through and be detected on the opposite side, this configuration would interfere with the penguin's ability to walk. Therefore, the wing was chosen as the optimal placement site. Since the wing is too thick for the light to pass entirely through, both the light emitter and detector are placed on the same side of the wing, relying on the reflection of light from the skin.

The micro-controller processes the detected light signals to calculate the penguin's pulse and blood oxygen saturation, transmitting this data wirelessly to a digital user interface for monitoring. During a consultation with Stephen Paine, it was suggested that a location-tracking unit be added to the design. This unit would also interface with the micro-controller, allowing real-time location data to be transmitted wirelessly alongside the physiological measurements.

The system is split into three subsystems. The context diagram in [Figure 2.1](#) indicates what each subsystem contains and how the different subsystems interact.

**Subsystem 1 - Light Emission Module and Light Detection and Amplification Module:** It is responsible for the hardware design of the pulse oximeter. It emits red and infrared light onto the penguin's wing and detects the reflected light as an electrical signal. This signal is amplified and used in Subsystem 2.

**Subsystem 2 - Power Module, Location Module and Processing Module:** This is responsible for providing power for all components, tracking the location of the penguin and using a micro-controller to facilitate signal processing and transmission of signals from Subsystem 1 to be displayed in Subsystem 3.

**Subsystem 3 - System Housing and Front-end GUI Development:** This subsystem involves the design and manufacture of a minimally invasive housing unit that stores the delicate electronic

equipment (Subsystem 1 and Subsystem 2) and ensures that the necessary components of Subsystem 1 have access to the flipper of the penguin for accurate sensor readings. Additionally, an intuitive graphical user interface is developed to display the sensor readings and other information in a simple and informative manner to the user.

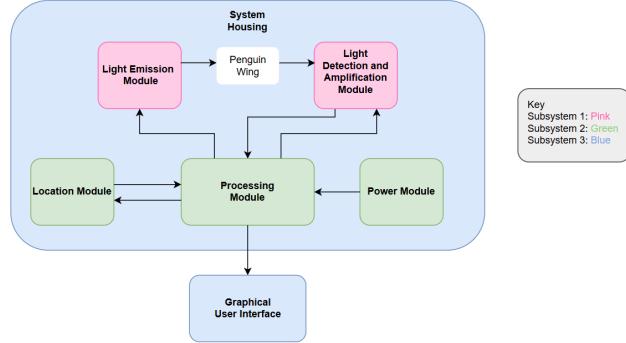


Figure 2.1: Context Diagram

# Chapter 3

## Literature Review

### 3.1 Introduction to the Monitoring of Health Status of Birds

By conserving wildlife, we're ensuring that future generations can enjoy our natural world and the incredible species that live within it.

—NWF

Ottinger et al. state that ‘avian health metrics can serve as reliable indicators of long-term system restoration and success’ [2]. These health metrics are essential for assessing the overall health of bird populations as they not only reflect individual well-being but also offer insights into the broader health of ecosystems [3]. This literature review will explore various methods for assessing key health metrics such as blood pressure, heart rate, weight, body temperature and breathing. It will analyse each method in terms of its accuracy, feasibility, and potential impact on the bird’s natural behaviour.

Given the delicate nature of many bird species, particularly those that are endangered or threatened, it is essential to employ methods that do not interfere with their natural behaviour. Minimally invasive techniques ensure that researchers can gather accurate data while minimising stress and disturbance to the animals. While existing methods for monitoring avian health are valuable, many traditional techniques can be invasive or logistically challenging, especially when monitoring wild populations. These limitations emphasise the need for more accessible and non-disruptive solutions. The data gathered from these health metrics can be applied to a wide range of ecological assessments, demographic studies, evaluations on the efficacy of management interventions and phenological research [3]. By analysing the accuracy, feasibility, and disruption of these techniques, this review aims to provide valuable insights for researchers and conservationists seeking effective ways to monitor bird health in the wild.

### 3.2 Blood Pressure

Arterial blood pressure is the force exerted by flowing blood on vessel walls [4]. According to Schnellbacher et al., ‘Blood pressure assessment is a crucial component of ... general health assessments’ [5]. This section will examine the feasibility and limitations of three commonly used techniques for blood pressure monitoring in birds: catheterization, oscillometric monitoring and Doppler ultrasonic flow detection.

### 3.2.1 Catheterization

Direct blood pressure measurement involves inserting a catheter into an artery and connecting it to a pressure transducer [4]. This method is considered the gold standard because it accurately measures intra-arterial pressure [5]. Despite its accuracy, catheterization is an invasive procedure that requires specific technical skills and equipment. It is generally not feasible for routine or field use as it often requires sedation or anaesthesia.

### 3.2.2 Oscillometric Monitors

Oscillometric blood pressure measurement is a non-invasive technique that uses an inflatable cuff to detect oscillations in arterial pressure, estimating systolic, diastolic, and mean arterial pressures [6]. Studies conducted on animal species have shown inconsistent accuracy with this method. Acierno et al. found that oscillometric devices failed to produce reliable results in Hispaniolan Amazon parrots [7]. Similarly, Zehnder et al. concluded that oscillometric blood pressure measurements were unreliable in Red-tailed hawks, likely due to differences in vascular structure [6]. The method's effectiveness is highly dependent on proper cuff placement and limb size, making it less feasible for small animals.

### 3.2.3 Doppler Ultrasonic Flow Detectors

The Doppler ultrasonic flow detector is another non-invasive method that uses ultrasonic waves to detect blood flow and determine systolic blood pressure. Unlike oscillometric devices, Doppler technology provides real-time auditory feedback, which can help in manual blood pressure estimation.

Compared to oscillometric methods, Doppler measurements are often considered more reliable, especially in small animals. Lichtenberger reported that Doppler ultrasonic detectors can be an ‘easy, inexpensive, and accurate’ [4] alternative to direct measurement. However, this method has limitations: it only provides systolic pressure and requires manual operation, which can introduce user variability. Additionally, proper probe placement is crucial for accurate readings, and external factors such as motion or weak pulse signals may affect results [6].

Blood pressure monitoring in birds presents unique challenges, especially when it is crucial to minimise disturbance. While non-invasive methods like Doppler ultrasonic flow detectors and oscillometric devices may be used, both require handling the animal, which can induce stress. Birds are particularly sensitive to stress, and physical restraint can lead to elevated heart rates and altered blood pressure, which may skew the results [6]. Despite its value as a health metric, blood pressure monitoring is not suitable in situations where minimizing disturbance is a priority due to the need for handling and restraint.

## 3.3 Heart Rate

According to Rzucidlo et al., ‘Simpler indices such as heart rate (HR) and respiration rate (RR) that are tightly correlated with energetic expenditure across taxa are thus used as common indices of animal health’ [8]. This supports the use of heart rate measurements as a measure of the health of wild animals. Measuring the heart rate of penguins, during their molting stages involves a good share of ingenuity and is required to be as minimally invasive as possible. Giese et al. note the importance of

reducing the confounding effects of investigator disturbance [1].

### 3.3.1 FM Radio Transmitter and Electret Microphone

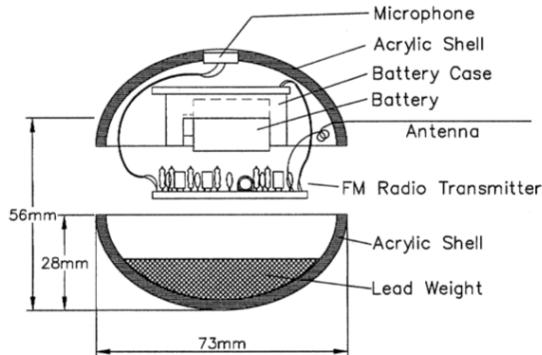


Figure 3.1: An example of the use of an electret mic and fm transmitter in an artificial egg. [1]

A commercially available frequency modulated radio transmitter could be utilised to transmit the sound (heart-beat) to any FM domestic receiver (frequency band of 88-108MHz) [1]. The sensitivity of the microphone would have to be adjusted - using a resistor - to optimise the signal-to-noise ratio and allow the microphone to have a stethoscopic effect. Utilising this method was successful in a series of field tests for Giese et al., ‘Penguin heart rates were able to be recorded as soon as birds settled on the eggs’ and they noted that their results were in line with those of other studies. While successful, they did note that some limitation of this method are the drifting of frequency which results in a limited number of units being deployed and functional as well as reliance on the artificial egg being in contact with the birds’ brood patch and having someone present in close range with a receiver and recorder [1].

### 3.3.2 Eulerian Video Magnification of IRT (EVM)

EVM is, according to Rzucidlo et al., ‘a spatial decomposition applied to videos such that variation in pixel colour at a given location can then be amplified using temporal filtering’ [8]. In order to apply this tool, infrared imaging is required and Rzucidlo et al. utilised a FLIR T540 camera to capture this (as well as a GoPro Hero 4 for RGB video). Rib cage expansion and nostril flaring were key indicators utilised in the EVM process in order to determine the heart rate of the animal.[8]. While Rzcidlo et al. experienced less than a 5% error rate, some limitations of this method are that thick skin or fur severely impacts the effectiveness of the EVM process, and the presence of water on the animal results in the degradation of the quality of the IRT video - limiting this method of heart rate detection to dry land mammals with a minimal coating or fur. The largest drawback according to Rzidlo et al. was animal movement, making rib cage expansion and nostril flaring difficult to detect [8].

### 3.3.3 Electrocardiograms (ECG)

A commercially available pico EMG (electromyogram) - which, when placed close to the thoracic region allows for high quality recording of ECG traces with a fixed sampling rate (2000Hz) [9]. The logger has the capability to transmit signals over WiFi to a receiver station connected to a laptop or

work as an autonomous logger using its internal memory. The data does need to be extrapolated in order to be useful. Bonadonna et al. noted success in their experiment with very accurate and clear results. However they did note some limitations with the use of the EMG logger ranging from battery and memory specifications, to logger programming and the interfacing of the physical logger with a laptop/computer [9].

## 3.4 Weight

According to Viblanc et al., ‘Body mass and body condition are often tightly linked to animal health and fitness in the wild and thus are key measures for ecophysiologists and behavioural ecologists’ [10]. This supports the use of weight as a measure of the health of wild animals. In addition to this, knowing the weight of a penguin, and its identity, allows for extended knowledge on the breeding state of the penguins according to Bird Life SA, ‘If conditions are good these birds can gain over a third of their average body mass in one day, but if conditions are poor, they can lose this weight; if this happens repeatedly, they may abandon their breeding effort’ [11].

### 3.4.1 Tag Identification

In order to accurately track the weight of individual birds, an electronic identification tag can be used and can theoretically last the lifetime of the bird [12] - since it is passive. While this method is invasive, since tags had to be injected in to the bird, Kerry et al. noted that, ‘There was no evidence from recapture of tagged birds that the tags migrated away from the injection site or caused any physical damage’ [12]. Alternatively, the use of RFID (radio frequency identification transponders) - which has been used since 2003 - is available[13]. While the initial chipping of the penguins is rather invasive - ‘...injected in to the bird..’ [12], this method has been tested and in use for a long period of time, indicating its effectiveness.

### 3.4.2 Weighbridge

The appeal of a weighbridge to record the weight of wild animals is due to the lack of handling it requires (minimally invasive), while also recording the specific animal and various other characteristics. The use of load cells (to track the weight), infrared beams (to detect direction) and passive transponders (to identify the animal) has been proven effective by logging over 80 000 crossings and recording weight variations with an accuracy of around 95% while minimising the disturbance of the colony while eliminating the need for manual handling [12]. Alternative weighbridge specifications could be RFID readers and RFID chips, as well as environmentally protected rated load cells or even a solar panel recharging power supply as suggested by Afanasyef [13]. Possible limitations highlighted by Knowles et al. are that multiple trips by untagged animals are indistinguishable, and local movement of animals across the bridge as well as multiple animals on the bridge at once are drawbacks of this method [12].

### 3.4.3 Body Girth

Here an alternative to physical mass measurements is proposed in order to circumvent the common logistical challenges and technological limitations associated with it. Viblanc et al. indicate that body girth is a reliable proxy for mass measurement [10]. What this method involves is the physical

measurement of the abdomen of the bird. While there is a strong correlation between girth and mass and statistical modelling can allow for accurate estimations, this method is proven to be invasive. Further limitations involve inaccuracies involved with manual measurements as well as dirt caught in the fur of the bird and additional girth as a result of a recent feeding [10].

### 3.5 Body Temperature

Temperature is a vital indicator of health in birds, as fluctuations in body temperature often reflect changes in physiological status, stress levels, or illness. As stated by McCafferty et al., ‘body temperature is, therefore, a key parameter in physiological’ research[14]. Accurate temperature monitoring can provide important insights into a bird’s well-being, particularly during demanding periods such as molting or migration. This section will focus on three methods commonly used to assess avian body temperature: thermocouples, infrared thermography, and passive integrated transponder (PIT) tags.

#### 3.5.1 Thermocouple

Thermocouples are widely used to obtain core body temperature readings in birds. It is ‘an inexpensive, fast, and attractive method for body temperature recording’[14]. This method ‘relies on inserting a thin temperature sensitive probe into the rectum via the anus or cloaca,’ [14] providing a direct measurement of internal body temperature.

This method has some limitations, in particular a concern for animal welfare and data reliability. The stress caused by the thermocouple insertion may bias the temperature reading [14], as handling and restraint can artificially raise body temperature. This raises concerns about data reliability and animal welfare. However, Welman et al. ensured the penguins showed ‘no visible signs of distress or discomfort’ while the thermocouple was recording data during their experiment. Additionally, if physical attachment is only viable for inactive, or relatively inactive animals [14], it should only be used on birds during temporary, less active periods such as the molting stage. Therefore, when used appropriately, thermocouples can serve as a valuable tool which ‘provides valuable insights’[15] into the body temperature of birds.

#### 3.5.2 Infrared Thermography

Infrared thermography (IRT) is a non-invasive method for assessing surface temperature in birds and is especially useful in situations where capturing animals is not feasible or necessary. IRT works by detecting infrared radiation, which is naturally emitted from all surfaces with a temperature above absolute zero (0 K) [14]. By capturing this emitted radiation with an infrared camera, researchers can generate thermal images that reveal surface temperature, which may correlate with underlying physiological processes such as stress.

When using infrared thermography (IRT), it is essential to ensure that the correct anatomical region of the bird is captured from a consistent and known distance, that all images are processed uniformly with orthogonal alignment and environmental factors. Gauchet et al. and Tabh et al. have shown that the bill and the medial canthus of the eye provide the strongest correlations with internal body temperature but are not entirely accurate[16] [17]. However, the temperature of the bill and the medial

canthus of the eye indicate stress responses of birds[17]. To maintain accuracy, shooting distances must ideally be under 3 metres, as measured by a laser rangefinder or another suitable device, but this approach is only feasible for birds with a high tolerance for human presence [16]. Additionally, images must be carefully selected to omit any in which the anatomical region of interest was not perpendicular to the lens of the thermographic device, as such angles can introduce measurement errors [17]. A weather station, such as ‘the HOBO H21-USB weather station’[16], is required to record ambient temperature, relative humidity, wind speed, and solar radiation at one-minute intervals.

There are several factors to consider when using infrared cameras as discussed above, but when applied correctly and in suitable conditions, they can yield valuable insights. Infrared cameras are particularly effective for detecting signs of stress in birds[14]. However, when the objective is to accurately assess core body temperature, ‘using combined measurements’ [16] has been shown to be more reliable.

### 3.5.3 Passive Integrated Transponder (PIT) Tags

PIT tags are small, devices commonly used for animal identification and monitoring. These tags consist of an antenna coil, capacitor, and circuit board encased in a durable glass capsule. They can be implanted subcutaneously. PIT tags work by emitting a unique identification code when they come into proximity with a reader [18]. For temperature sensing, certain PIT tags are designed to include temperature-sensitive elements that allow them to detect and record the animal’s body temperature. When the transponder is within the electromagnetic field of a transceiver’s antenna, it transmits data, including temperature readings, which can be analysed to monitor the animal’s health or environmental conditions over time[14].

While PIT tags can be useful for monitoring body temperature, there are several factors to consider when evaluating their suitability for measuring bird body temperature. First, the placement of the PIT tag can affect the accuracy of the readings, as temperature measurements can vary depending on whether the tag is implanted subcutaneously, in muscle tissue, or deeper within the body. Secondly, PIT tags require the animal to be within a relatively short range (approximately 0.3 m) of the receiving unit to transmit data, which may limit their use in free-ranging or migratory species. Furthermore, if more than one ‘transponder is contained within the same electromagnetic field at any given time’[14], the data is arbitrary. Additionally, PIT tags do not provide continuous temperature data like some other methods and require manual retrieval or periodic access to the receiving unit for data collection. Although implantation is minimally invasive and does not typically cause significant stress or discomfort, it is important to assess whether the procedure is appropriate for the species being studied[14].

Therefore, while PIT tags can be useful in specific controlled environments but they are not be the most effective tool for long-term, continuous monitoring of bird body temperature in the wild.

## 3.6 Breathing

Breathing is a key indicator of avian health, as changes in respiratory patterns often signal underlying stress or illness. As noted by Orosz et al., ‘respiratory distress is usually a life-threatening emergency in any species and this is particularly important in avian species because of their unique anatomy and physiology’ [19]. This is especially crucial for birds during their molt phase, when they are

physiologically stressed and fasting. This review explores the use of Pulse Oximeters for measuring oxygen saturation, and Magnetic Beak Sensors for monitoring breathing patterns.

### 3.6.1 Pulse Oximeter

The pulse oximeter was developed, according to Morey, as a less invasive alternative to arterial blood sampling, providing continuous, real-time monitoring of oxygen saturation, though with slightly reduced accuracy compared to traditional invasive methods[20]. The pulse oximeter consists of LEDs placed on one side of a body extremity with a photodetector positioned on the opposite side. The photodetector measures and compares the intensity of red and infrared light, which is then used to determine the absorption by oxyhemoglobin and deoxyhemoglobin molecules [20].

For this method to be effective, the wavelength of the transmitted light must be clearly detected. To minimise motion artefacts, the pulse oximeter should be placed on a body part with minimal movement. Morey selected the ear as the preferred site, observing that individuals tend to move their ears significantly less than their fingers[20]. Similarly, Schmitt et al. reduced artificial signals by ‘designing a black elastic tape to which the probes’ were attached on the birds leg[21], ensuring stable positioning during measurements. Furthermore, when adapting this method for avian patients, it is recommended to use ‘an avian calibration curve’ [21] to accurately interpret the ratio of transmitted red and infrared light.

A pulse oximeter is ‘the only noninvasive method estimating arterial oxygen status, that appears to meet the requirements of avian anesthetic monitoring,’ as confirmed by Shmitt et al.[21]. With the consideration of artificial signals and the use of an avian calibration curve, the pulse oximeter will be suitable for monitoring avian health.

### 3.6.2 Magnetic Beak Sensors

Magnetic beak sensors, based on Hall effect technology, offer a non-invasive method to monitor avian breathing patterns by detecting subtle beak movements[22]. These sensors typically involve placing a small magnet on one part of the beak and a hall sensor on the other. The hall sensor is connected to a logger via a cable. The ‘magnetic field (orthogonal to the Hall plate) is converted into an output voltage’[14] allowing precise measurement of the beak movements associated with breathing.

This device is successful as Wilson et al. noted, ‘individual breaths could be identified in the changing beak angles, which took on a distinct wave form’[22] By analysing these angular patterns, magnetic beak sensors can effectively isolate breathing movements, providing valuable insight into avian respiratory health. However, the ‘magellanic penguin was observed to be stressed by the beak sensors’[22]. After 24 hours, the penguins had removed the device and, subsequently, no data could be retrieved. Therefore, it is fair to conclude that the bird must have a good temperament to successfully record their breeding patterns using this method.

## 3.7 Conclusion

This literature review aimed to explore various methods for assessing avian health metrics, with a focus on evaluating how feasible as well as minimally invasive the techniques are. The review highlighted the

effectiveness and challenges of several methods.

Blood pressure monitoring in birds is essential for assessing cardiovascular health. Non-invasive methods such as Doppler ultrasound are commonly used to measure blood pressure, offering a reliable way to assess cardiovascular function. However, they require direct handling and positioning of the animal and can be influenced by the animal's stress levels or movement.

Heart rate measurement can be done using FM radio transmitters, which transmit heartbeat sounds to receivers. While effective, this method suffers from signal drift and requires close-range monitoring. Electrocardiograms (ECGs) provide precise heart rate data but may require invasive attachments. Eulerian Video Magnification (EVM) is a non-invasive approach that analyses rib cage movement for heart rate data, but it is limited by fur thickness, water exposure, and animal movement.

Weight tracking can be done through electronic identification tags and RFID systems, which allow long-term monitoring with minimal handling, though initial tagging can be invasive. Weighbridges, using load cells and infrared beams, are effective for non-invasively monitoring weight without disturbing animals, although challenges such as distinguishing untagged animals or multiple animals on the bridge can occur. An alternative method, body girth measurement, serves as a proxy for body mass but is invasive and can be influenced by external factors like feeding or dirt.

Body temperature can be measured using thermocouples, providing direct internal readings, but they can cause stress and are best used when animals are inactive. Infrared thermography (IRT) offers a non-invasive way to capture surface temperature, though its accuracy for core body temperature is limited. Passive integrated transponder (PIT) tags also measure temperature, though they require animals to be close to the reader and offer limited continuous data.

For breathing, pulse oximeters are non-invasive tools that measure oxygen saturation and offer continuous tracking, though they are sensitive to motion artifacts and require careful placement. Magnetic beak sensors, which detect subtle beak movements, are effective for monitoring breathing but may cause stress in some species, limiting their use.

Although each method has its own benefits, non-invasive techniques are particularly valuable for reducing stress and disturbance in birds. Understanding these health metrics and using minimally invasive approaches is essential for advancing avian conservation, especially for species that are endangered or at risk.

# Chapter 4

## Subsystem 1

Author: Liyana Singh (SNGLIY001)

### 4.1 Introduction

The solution, as discussed in [chapter 2](#), is to develop a remote pulse oximeter to measure the penguin's pulse and blood oxygen saturation. Subsystem 1 focuses on the hardware design of the pulse oximeter. It involves emitting red ( $\pm 660nm$ ) and infrared light ( $\pm 910nm$ ) onto the penguin's wing and detecting the reflected light as an electrical signal, which is then amplified. The output of this subsystem is an electrical signal that represents how much light was detected. The pulse is determined by tracking the periodic fluctuations in the detected signal over time, while the blood oxygen saturation ( $SpO_2$ ) is calculated using the ratio of the red to infrared light absorption during pulsatile flow.

#### 4.1.1 Context

Subsystem 1 interacts with the Processing Module in Subsystem 2. The Light Emission Module receives power from the Processing Module and controls the LEDs via PWM signals. The Light Detection and Amplification Module also draws power from the Processing Module and transmits an amplified electrical signal back which represents the intensity of the reflected light detected. This signal serves as the input for further processing and analysis in Subsystem 2. Subsystem 1 interacts with the System Housing in Subsystem 3. Subsystem 1's modules are placed inside the System Housing and positioned to interact with the penguin and the other modules. The overall interaction and signal flow of Subsystem 1 is illustrated in [Figure 4.1](#).

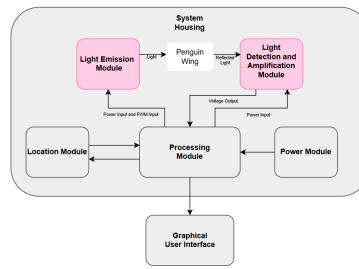


Figure 4.1: Context diagram highlighting the interactions of Subsystem 1

#### 4.1.2 Scope and Limitations

The scope of this subsystem is to design the hardware of the pulse oximeter. This excludes any data processing or analysis. There are also several limitations to consider:

- The design will be implemented on vero-board not on a PCB due to time constraints. The vero-board will be bulky but can be improved in future iterations.
- The final complete and working overall system (Subsystem 1, 2 and 3) is to be demonstrated on 26/05/2025.
- There is a cost restriction of R500.
- The testing will be done on humans not penguins which will cause some inaccuracies.

## 4.2 Requirements Analysis

### 4.2.1 Requirements

The requirements for Subsystem 1 are described in [Table 4.1](#).

Table 4.1: Requirements of Subsystem 1

| Requirement ID | Description  |
|----------------|--|
| R01            | The subsystem must be externally powered.  |
| R02            | The subsystem must not exceed the current limitations.                                     |
| R03            | The subsystem must emit infrared light and red light.                                      |
| R04            | The subsystem must output a voltage signal indicating the intensity of the light detected. |
| R05            | The subsystem output must be readable by the micro-controller's ADC.                       |
| R06            | The subsystem must be an appropriate size for the housing.                                 |
| R07            | The subsystem must adhere to the budget.   |

### 4.2.2 Specifications

The specifications based on the requirements in [Table 4.1](#) for Subsystem 1 are described in [Table 4.2](#).

Table 4.2: Specifications of Subsystem 1

| Specifications ID | Description   |
|-------------------|---|
| S01               | The subsystem will be supplied with 3.3V or 5V from the micro-controller.   |
| S02               | The subsystem will draw a maximum current of 400mA supplied by the micro-controller.  |
| S03               | The subsystem must allow for the Processing Module to control the infrared and red light.   |
| S04               | The subsystem must output a square waveform voltage signal with periodic fluctuations indicating the reflected light intensity of the infrared light and red light. |
| S05               | The output must be between 0V and 3.3V to be read by the micro-controller's ADC.  |
| S06               | The light emitters and detector must be on a vero-board of size 33mm × 26mm × 8mm. The other components of the subsystem must be less than 90mm × 40mm × 10mm.      |
| S07               | It must cost under R500.  |

### 4.2.3 Testing Procedure

A summary of the testing procedures detailed in subsection 4.4.1 is given in Table 4.3.

Table 4.3: Summary of Testing Procedures for Subsystem 1

| Test ID | Description  |
|---------|--|
| AT01    | Measure the voltage over the circuits.   |
| AT02    | Measure the current through the circuits.  |
| AT03    | Simultaneously measure the voltage over the infrared LED and the voltage over the red LED. |
| AT04    | Measure the voltage output signal.   |
| AT05    | Measure the length, width and height of the vero-boards.                                   |
| AT06    | Add the cost of each component.  |

### 4.2.4 Traceability Analysis

Table 4.4 shows the link between the requirements, specifications and testing procedures.

Table 4.4: Requirements Traceability Matrix

| # | Requirements | Specifications | Acceptance Tests |
|---|--------------|----------------|------------------|
| 1 | R01          | S01            | AT01             |
| 2 | R02          | S02            | AT02             |
| 3 | R03          | S03            | AT03             |
| 4 | R04          | S04            | AT04             |
| 5 | R05          | S05            | AT04             |
| 6 | R06          | S06            | AT05             |
| 7 | R07          | S07            | AT06             |

#### Traceability Analysis 1

The subsystem must be externally powered (R01) by the micro-controller with 3.3V OR 5V (S01). This will be tested by measuring the voltage over the circuits (AT01).

#### Traceability Analysis 2

The subsystem must not exceed current limitations (R02) of  $400mA$  which is supplied by the micro-controller (S02). This will be tested by measuring the current through the circuits (AT02).

#### Traceability Analysis 3

The subsystem allow for control of emitting infrared and red light (R03), turning each on and off in succession S03). This will be tested by simultaneously measuring the voltage over the infrared LED and the voltage over the red LED (AT03).

#### Traceability Analysis 4

The subsystem must output a square waveform voltage signal with periodic fluctuations indicating the reflected light intensity (R04) of the infrared light and red light (S04). This will be tested by measuring the voltage output signal (AT04).

### Traceability Analysis 5

The subsystem output must be between 0V and 3.3V (S05) to be read by the micro-controller's ADC (R05). This will be tested by measuring the voltage output signal (AT04).

### Traceability Analysis 6

The light emitters and detector must be on a vero-board of size  $33mm \times 26mm \times 8mm$ . The other components of the subsystem must be less than  $90mm \times 40mm \times 10mm$  (S06) to fit into the housing (R06). This will be tested by measuring the length, width and height of the vero-boards (AT05).

### Traceability Analysis 7

The subsystem must adhere to the R300 (S07) budget (R07). To test this the cost of each component must be added (AT06).

## 4.3 Design

### 4.3.1 Light Emission Module

#### Circuit Design

The LEDs must provide intense light to be able to penetrate the skin which requires high currents. The Light Emission Module has access to the micro-controllers GPIO pins, the 3.3V output pin and 5V output pin. The circuit diagrams for Option 1 and Option 2 can be seen in [Figure 4.2](#) and [Figure 4.3](#), respectively.

Option 1: In this configuration, the PWM input from the GPIO pins are directly connected to the red and infrared LEDs. This setup allows the LEDs to be turned on and off by the micro-controller for processing and analysis.

Option 2: In this configuration, a constant voltage from the 3.3V supply is connected to the red and infrared LEDs, which are placed at the collector of an NPN BJT. The PWM signals from the micro-controller's GPIO pins are connected to the base of the BJT through appropriate resistors. The emitter of the BJT is connected to ground. This setup allows the microcontroller to switch the LEDs on and off using PWM control for processing and analysis.

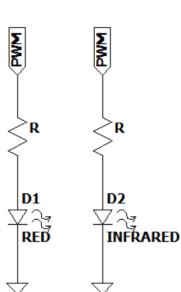


Figure 4.2: Circuit diagram for Module 1  
Option 1

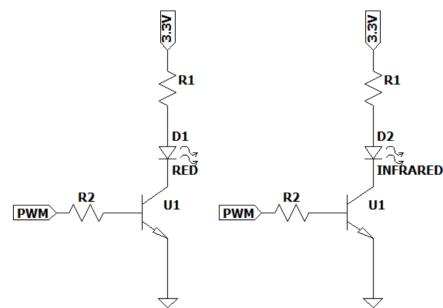


Figure 4.3: Circuit diagram for Module 1  
Option 2

Using a BJT to control an LED instead of connecting a PWM signal directly from the micro-controller's GPIO pin to the LED is important when driving high currents. The GPIO pins do not have sufficient current for powering high-intensity LEDs. A BJT acts as a current switch, allowing the LED to draw higher current from the  $3.3V$  pin, while the GPIO pins only supply a small base current to control the BJT. Therefore, Option 2 was chosen as the final design for the Light Emission Module.

### Component Choice

The red LED, D1, must emit high intensity light at a wavelength of  $\pm 660nm$ . Based on the comparison in [Table 4.5](#), the L-53SURCK is selected. While it has a slightly lower maximum power than the L-483HDT, it emits light much closer to the target wavelength and delivers a higher light intensity at  $660\text{ nm}$  (0.7 compared to 0.4).

Table 4.5: Comparison of Red LEDs

| LED Model | Peak Wavelength (nm) | Max Power (mW) | Max Current (mA) | Intensity at 660 nm | Price (ZAR) |
|-----------|----------------------|----------------|------------------|---------------------|-------------|
| L-483HDT  | 700                  | 105            | 25               | 0.4                 | 2.65        |
| L-53SURCK | 645                  | 75             | 30               | 0.7                 | 2.19        |

The infrared LED, D2, must emit high intensity light at a wavelength of  $\pm 910nm$ . Based on the comparison in [Table 4.9](#), the L-53F3C is selected. The L-71135F7BT operates at a peak wavelength of  $850nm$ , which is below the target, and delivers a relatively low intensity of 0.2 at  $910nm$  while also being more expensive. In contrast, the L-53F3C has a peak wavelength of  $940nm$ , but its emission intensity at  $910nm$  is higher (0.5), and it is significantly more cost-effective. Despite the mismatch in peak wavelength, the higher intensity at  $910nm$  and lower cost make the L-53F3C the more suitable choice.

Table 4.6: Comparison of Infrared LEDs

| LED Model   | Peak Wavelength (nm) | Max Power (mW) | Max Current (mA) | Intensity at 910 nm | Price (ZAR) |
|-------------|----------------------|----------------|------------------|---------------------|-------------|
| L-71135F7BT | 850                  | 80             | 50               | 0.2                 | 11.39       |
| L-53F3C     | 940                  | 100            | 50               | 0.5                 | 3.62        |

The NPN BJT, U1, must operate with a  $30mA$  collector current and have fast switching capabilities. [Table 4.7](#) presents a comparison of two NPN BJTs considered for the LED switching application. Since the required collector current is only  $30mA$ , well below the maximum current rating of both devices, the 2N3904 is selected. It offers sufficient switching speed with a transition frequency of 300, identical to the 2N2222A, while being significantly more cost-effective. As the application does not demand high current handling, the cheaper 2N3904 is the appropriate choice.

Table 4.7: Comparison of NPN BJTs

| Transistor Model | Max Collector Current ( $I_C$ ) | Transition Frequency ( $f_T$ ) | Price (ZAR) |
|------------------|---------------------------------|--------------------------------|-------------|
| 2N2222A          | 800 mA                          | 300 MHz                        | 12.28       |
| 2N3904           | 200 mA                          | 300 MHz                        | 0.62        |

### Calculations for Resistor Values

To safely operate the LEDs and ensure proper switching through the NPN BJT, resistor values were calculated for both the collector and base paths. The red LED and infrared LED has a maximum current rating of  $30mA$  and  $50mA$ . A drive current of  $30mA$  was chosen for both LEDs. At this current, both LEDs exhibit sufficient luminous intensity (1.5 units) and the forward voltage drop  $V_D$  across the LEDs is  $1.3V$ .

The collector-emitter saturation voltage of the NPN BJT ( $V_{CE}$ ) is  $1V$ . The collector resistor  $R_1$  is used to limit the current through the LED. Applying Kirchhoff's Voltage Law around the collector loop gives:

$$3.3 - V_{CE} - V_D - I_{LED}R_1 = 0 \quad (4.1)$$

Substituting the values,  $R_1 = \frac{1.0}{0.03} = 33.3\Omega$ . Thus, an E12 resistor value of  $33\Omega$  is chosen for  $R_1$ .

The base current should be approximately ten times less than the collector current. Hence, a base current of  $3mA$  is chosen. The GPIO outputs a  $3.3V$  PWM signal and the base-emitter voltage drop  $V_{BE}$  is  $0.5V$ , the base resistor  $R_2$  is calculated using Kirchhoff's Voltage Law around the base-emitter loop gives:

$$3.3 - I_B R_2 - V_{BE} = 0 \quad (4.2)$$

Substituting the values,  $R_2 = \frac{3.3-0.5}{0.003} = 933.3\Omega$ . Thus, an E12 standard resistor value of  $1000\Omega$  is chosen for  $R_2$ .

#### 4.3.2 Light Detection and Amplification Module

##### Circuit Design

The photodiode must detect the reflected light from the penguins wing to produce a current. This current must be converted to a volatage in the range  $0V$  to  $3.3V$  to be read by the micro-controllers ADC. The Light Detection and Amplification Module has access to the micro-controllers the  $3.3V$  output pin and  $5V$  output pin. The  $5V$  output pin was used for this module to ensure high amplification.

Option 1: In the Resistor-Based Detection configuration, a voltage supply is connected to a photodiode in series with a resistor, and the output is measured across the resistor. When light strikes the photodiode, it generates a photocurrent, which flows through the resistor and produces a voltage proportional to the light intensity.

Option 2: In the Photovoltaic Transimpedance Amplifier configuration, the photodiode operates in photovoltaic mode without an external bias, producing a small current in response to incident light. This current is converted into a voltage using an op-amp configured as a transimpedance amplifier with a negative gain. The resulting negative voltage is then inverted using a second amplifier stage to produce a positive signal. A voltage divider is used to ensure the final output is within the acceptable range for the ADC. A transimpedance amplifier has negative gain but the micro-controller cannot

provide a negative voltage. Therefore, a false ground of 2.5V is created through a voltage divider from the 5V supply. The op-amp is supplied with 5V and 0V and biased around the false ground so that the values 0V to 2.5V act as the negative range.

Option 3: In the Photocurrent Transimpedance Amplifier configuration, the photodiode is reverse-biased to widen the depletion region, thereby improving speed and linearity. The photocurrent generated in response to light is fed into the inverting input of an op-amp set up as a transimpedance amplifier, producing a negative voltage at the output. This is then passed through a second inverting amplifier to flip the signal to a positive voltage, which is scaled using a voltage divider to match the input range of the microcontroller's ADC. Similar to Option 2, a false ground of 2.5V is created.

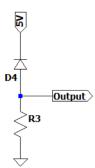


Figure 4.4: Circuit diagram for Module 2 Option 1

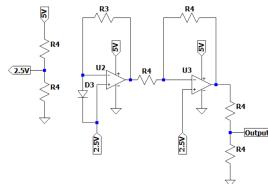


Figure 4.5: Circuit diagram for Module 2 Option 2

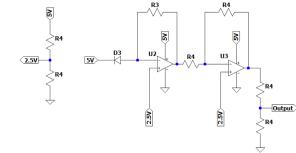


Figure 4.6: Circuit diagram for Module 2 Option 3

Option 1, which uses a simple resistor for signal detection, is linear but is highly sensitive to temperature and noise. Option 2, the photovoltaic transimpedance amplifier has poor linearity. Option 3, the photocurrent transimpedance amplifier, is more robust than Option 1 and is linear for varying signal intensities. Therefore, Option 3 was chosen as the final design for the Light Detection and Amplification Module.

### Component Choice

The photodiode, D3, must be able to detect light at wavelengths 660nm and 940nm and produce a reliable current which represents the light intensity detected. While the BPW21r performs well in the visible range, its wavelength spectrum only extends to 675nm, making it unsuitable for detecting the 940nm infrared signal essential to pulse oximetry. In contrast, the BPW34 covers a the spectrum from 430nm to 1100nm, effectively detecting both the 660nm and 940nm wavelengths. Additionally, it produces a significantly higher reverse light current of 100 $\mu$ A at 1000Ix, indicating better sensitivity. Its wider angle of half sensitivity (65) further enhances light collection in reflective configurations. Based on the comparison, the BPW34 photodiode is the appropriate choice.

Table 4.8: Comparison of Photodiodes LEDs

| Photodiode Model | Wavelength Spectrum (nm) | Angle of Half Sensitivity (degrees) | Reverse Light Current at 1000 Illuminance (uA) | Price (ZAR) |
|------------------|--------------------------|-------------------------------------|--|-------------|
| BPW21r           | 420 - 675                | 50                                  | 10   | -           |
| BPW34            | 430 - 1100               | 65                                  | 100  | 13.66       |

The operational amplifiers, U3 and U4, must be powered by the micro-controller's 5V output pin. It must handle the fast change in input from the detected alternating red and infrared light. The MC33078

is selected due to its lower cost and identical slew rate performance compared to the UPC4570C. Importantly, it operates within the microcontroller's 5V supply.

Table 4.9: Comparison of Op-amps LEDs

| Op-amp Model | Minimum Supply Voltage (V) | Slew Rate (V/us) | Price (ZAR) |
|--------------|----------------------------|------------------|-------------|
| UPC4570C     | 8                          | 7                | 28.75       |
| MC33078      | 5                          | 7                | 4.77        |

### Calculations for Resistor Values

The 5V supply from the microcontroller must be divided in half to generate a reference voltage of 2.5V. This is achieved using a voltage divider circuit, which consists of two resistors connected in series between the 5V supply and ground. The output voltage is taken from the junction of the two resistors. The voltage divider equation is given by:

$$V_{\text{out}} = V_{\text{in}} \cdot \frac{R_B}{R_T + R_B} \quad (4.3)$$

To achieve an output of 2.5V, the resistors must be equal in value. Therefore, both resistors are chosen to be 1000Ω

The transimpedance gain is defined as the ratio of output voltage to input current, the gain is:

$$V_{\text{out}} = -I_{\text{in}} \cdot R_F \quad (4.4)$$

The negative sign indicates that the op-amp is operating in an inverting configuration. The design uses a 56000Ω feedback resistor for large trans-amplification of the small current.

Following the transimpedance stage, an inverting amplifier is used with a gain of -1 to restore the signal to a positive polarity. For this, the feedback resistor and the input resistor are set to equal values:

$$\text{Gain} = -\frac{R_F}{R_{\text{in}}} = -1 \Rightarrow R_F = R_{\text{in}} \quad (4.5)$$

Therefore, both resistors are chosen to be 1000Ω

Finally, the output of this amplifier is passed through another voltage divider to shift the signal into the 0V to 3.3V range appropriate for the ADC. Again, equal resistor values are used to halve the signal, ensuring it fits within the input range of the microcontroller's ADC. Therefore, both resistors are chosen to be 1000Ω.

#### 4.3.3 Final Design

The final design of both the Light Emission Module and the Light Detection and Amplification Module is shown in [Figure 4.7](#). In the light emission circuit, red and infrared LEDs are controlled via PWM

signals from the microcontroller through an NPN BJT, enabling timed light pulses for measurement. In the detection and amplification circuit, a reverse-biased photodiode is used in a photocurrent transimpedance amplifier configuration to convert light intensity into a voltage signal. The signal, initially negative, is inverted and scaled using a second amplifier and voltage divider to fit within the 0 to 3.3V ADC input range of the microcontroller. A virtual ground at 2.5 V is established to allow bidirectional signal swing while using a single-supply op-amp powered by 5V and 0V.

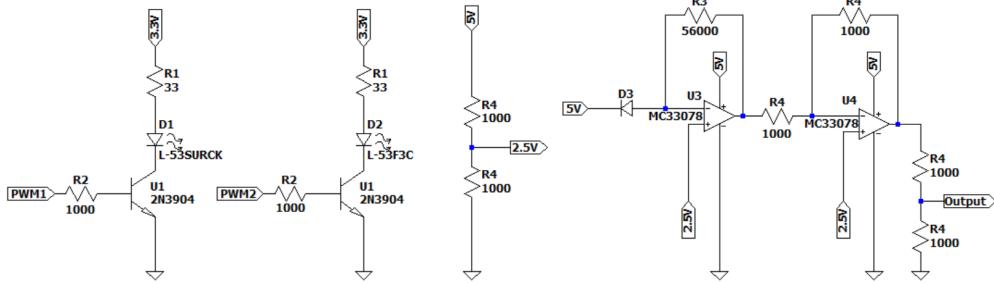


Figure 4.7: Final Schematic of Subsystem 1

The circuit components for both modules were arranged and soldered onto vero-boards to ensure reliable and compact integration. As shown in Figure 4.8 and Figure 4.9. The photodiode was placed centrally between the red and infrared LEDs on the light emission circuit to achieve even and maximum light detection but it has no electrical connection to Module 1.

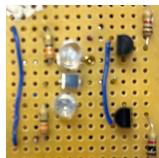


Figure 4.8: Vero-board for  
Module 1

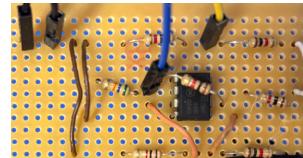


Figure 4.9: Vero-board for  
Module 2

#### 4.3.4 System Integration and Interfacing

The interfacing diagram shown in Figure 4.10 clearly illustrates the input and output electrical connections between Subsystem 1 and Subsystem 2 and Subsystem 1's physical placement inside Subsystem 3.

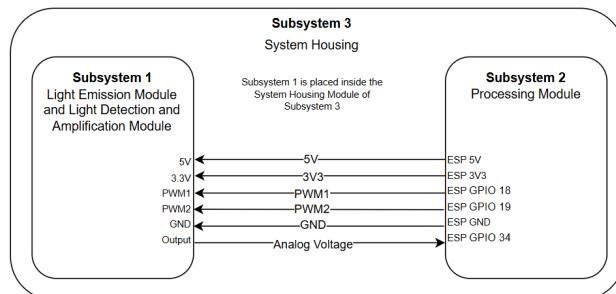


Figure 4.10: Interfacing Diagram of Subsystem 1

## 4.4 Results

### 4.4.1 Tests

The acceptance tests are fully explained in [Table 4.10](#). Initial setup: The emitter circuit is powered by a 3.3V power supply and PWM1 and PWM2 are set to a constant 3.3V. The detection and amplification circuit is powered by a 5V power supply.

Table 4.10: Acceptance Tests for Subsystem 1

| Test ID | Description  | Procedure   | Pass Criteria  |
|---------|--|---|--|
| AT01    | Measure the voltage over the circuits.   | Connect a multi-meter over the light emitter circuit. Record this voltage as $V_1$ . Connect a multi-meter over the light detection and amplification circuit. Record this voltage as $V_2$ .   | $V_1$ must equal $3.3 + 5\%V$ and $V_2$ must equal $5 + 5\%V$ .  |
| AT02    | Measure the current through the circuits.  | Connect a multi-meter through the light emitter circuit. Record this current as $I_1$ . Connect a multi-meter through the light detection and amplification circuit. Record this current as $I_2$ . Add $I_1$ and $I_2$ , and record this as $I$ .  | $I$ must be less than $400mA$  |
| AT03    | Simultaneously measure the voltage over the infrared LED and the voltage over the red LED. | Use a signal generator to create two distinct square waveforms with $100Hz$ frequency and 50% duty cycle with a half period offset between them. Connect these to the PWM1 and PWM2 inputs. Connect probe 1 of the oscilloscope over the red LED and connect probe 2 of the oscilloscope over the infrared LED.   | The oscilloscope display screen must show two distinct square waveforms with $100Hz$ frequency and 50% duty cycle with a half a period offset between them.                                    |
| AT04    | Measure the voltage output signal.   | Assume the same setup as in AT03. In addition, place the light emission vero-board on the top of the wing of the penguin (for this project a human hand was used). Connect probe 1 of the oscilloscope to the output of the detection and amplification circuit.  | The oscilloscope display screen must show a single square waveform. The average of the square waveform must increase and decrease periodically. All voltages must be between $0V$ and $3.3V$ . |
| AT05    | Measure the length, width and height of the vero-boards.                                   | Disconnect the circuits from all power sources. Use a standard $30cm$ ruler to measure the length, width and height of the light emitter vero-board. Record these values as $L_1$ , $W_1$ and $H_1$ . Use a standard $30cm$ ruler to measure the length, width and height of the light detection and amplification vero-board. Record these values as $L_2$ , $W_2$ and $H_2$ . | $L_1$ , $W_1$ and $H_1$ must equal exactly $33mm \times 26mm \times 8mm$ . $L_2$ , $W_2$ and $H_2$ must be less than $90mm \times 40mm \times 10mm$ .  |
| AT06    | Add the cost of each component.  | The cost of each component used to build Subsystem 1 is added and recorded as $T$ .   | $T$ must be less than $R500$ .   |

### 4.4.2 Results Analysis

#### Results Analysis AT01

[Figure 4.11](#) shows the voltage measured over the light emitter circuit ( $V_1$ ), while [Figure 4.12](#) shows the voltage over the light detection and amplification circuit ( $V_2$ ). The measured values were  $V_1 = 3.3017V$  and  $V_2 = 5.0474V$ , both of which fall within the required  $\pm 5\%$  tolerance range. Therefore, AT01 is passed.



Figure 4.11: Voltage across light emitter circuit (V1)



Figure 4.12: Voltage across light detection circuit (V2)

### Results Analysis AT02

Figure 4.13 and Figure 4.14 show the current measured through the light emitter ( $I_1$ ) and light detection and amplification circuit ( $I_2$ ), respectively. The values were  $I_1 = 67.050mA$  and  $I_2 = 69.750mA$ , with a total current  $I = 136.80mA$ . As  $I < 400mA$ , the system meets the current requirements and AT02 is passed.



Figure 4.13: Current through light emitter circuit ( $I_1$ )



Figure 4.14: Current through light detection circuit ( $I_2$ )

### Results Analysis AT03

Figure 4.15 shows the oscilloscope output for the red and infrared LEDs when two square waveforms were applied with a  $100Hz$  frequency and 50% duty cycle, offset by half a period. The waveform demonstrates clear switching between the two LEDs with appropriate timing, confirming that the LEDs are driven correctly. Therefore, AT03 is passed.

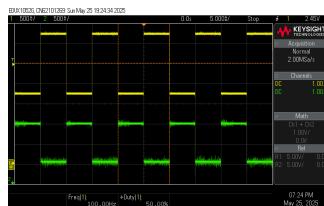


Figure 4.15: PWM signals for red and infrared LEDs

### Results Analysis AT04

The oscilloscope display showed the output signal from the detection and amplification circuit when the emitter board was placed against human skin. The signal is a distinct square waveform. This confirms that the system is capable of detecting blood oxygen saturation levels from the reflected light. It falls within the  $0V$  to  $3.3V$  range so it can be read by the ADC. The serial plotter in the Arduino IDE, as seen in Figure 4.16, represents what the micro-controller receives from the output. However, the average value does not increase and decrease periodically to indicate pulse. AT04 has failed. Solutions to this issue will be discussed in subsection 4.5.1.

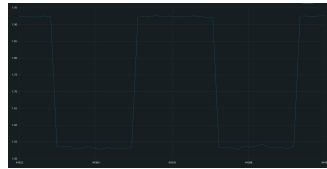


Figure 4.16: Output signal from detection circuit

### Results Analysis AT05

The dimensions of the light emitter board were measured as  $33mm \times 26mm \times 8mm$ , and the light detection and amplification board as  $85mm \times 40mm \times 9mm$ . These values meet the specifications. Thus, AT05 is passed.

### Results Analysis AT06

The total cost of all components used in Subsystem 1 was calculated as  $R158.19$ , which is below the  $R500$  threshold. See [Appendix A](#) for calculations. Therefore, AT06 is passed.

Table 4.11: Acceptance Test Results for Subsystem 1

| Test ID | Pass Criteria  | Result |
|---------|--|--------|
| AT01    | $V1$ must equal $3.3 + 5\%V$ and $V2$ must equal $5 + 5\%V$ .  | Pass   |
| AT02    | $I$ must be less than $400mA$  | Pass   |
| AT03    | The oscilloscope display screen must show two distinct square waveforms with $100Hz$ frequency and 50% duty cycle with a half a period offset between them.                                    | Pass   |
| AT04    | The oscilloscope display screen must show a single square waveform. The average of the square waveform must increase and decrease periodically. All voltages must be between $0V$ and $3.3V$ . | Fail   |
| AT05    | $L1$ , $W1$ and $H1$ must equal exactly $33mm \times 26mm \times 8mm$ . $L2$ , $W2$ and $H2$ must be less than $90mm \times 40mm \times 10mm$ .  | Pass   |
| AT06    | $T$ must be less than $R500$ .   | Pass   |

## 4.5 Conclusion

Subsystem 1 successfully developed the hardware for receiving the data to calculate the blood oxygen saturation of a penguin. However, the pulse rate could not be determined with this design. The next iteration will consider the following recommendations.

### 4.5.1 Recommendations

AT04 failed because the pulse could not be determined from the output. In the next iteration, the light emitter and detector placement on the penguin's wing should be optimized to ensure consistent skin contact and minimize ambient light interference. The LEDs must be surface mount. The LEDs and photodiode should be placed on either side of a body extremity (the thinnest part of the wing) because light passes through arterial blood and tissues, allowing clearer detection of the pulsatile flow associated with heartbeats. Additionally, the amplification circuit should be reviewed to enhance the AC component of the signal, which corresponds to the pulse. Implementing a bandpass filter centred around the expected heart rate frequency (typically 0.5–4 Hz for birds) may help isolate the pulsatile component.

# Chapter 5

## Subsystem 2

Author: Priya Moodley (MDLPRI040)

### 5.1 Introduction

African penguin populations are under increasing threat, yet current methods for monitoring their health conditions remain limited. To address this challenge, this project proposes a remote pulse oximeter capable of tracking vital signs and location data in real time. This chapter focuses on Subsystem 2, which involves power provision, location tracking and signals processing.

#### 5.1.1 Context

Subsystem 2 is the bridge between Subsystem 1 and Subsystem 3. It provides the Light Emission and Detection Module of Subsystem 1 with both power and PWM signals. The Amplification Module of Subsystem 1 also receives power and in turn, sends the output voltage signal to the processing module to be sampled and sent to the Graphical User Interface of Subsystem 3. This interaction with other Subsystems is shown in the context diagram, [Figure 5.1](#).

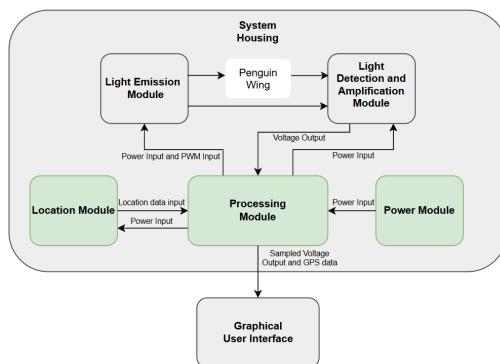


Figure 5.1: Context diagram that features the inputs and outputs for various modules of Subsystem 2

#### 5.1.2 Scope and Limitations

Subsystem 2 can be divided into three modules. Module 1, the Power Module, is responsible for delivering stable, efficient power to all circuits across Subsystems 1 and 2. Module 2, the Location Module, enables GPS-based monitoring of penguin movement. Finally, Module 3, the Processing Module, utilises an ESP32 as a hub for signals and data transmission. The circuitry for light emission and detection, as well as the system housing and graphical user interface is not within the scope for Subsystem 2.

The limitations considered are:

- Technical Limitation 1: The battery life will be limited for continuous operation (trade-off will be made with power consumption vs battery capacity).
- Technical limitation 2: GPS Accuracy may be affected underwater or in areas with dense foliage.
- Time Limitation: The final project (comprising all Subsystems) must be completed by the demonstration date: 26/05/2025.
- Cost Limitation: There is a cost restriction of R750 for Subsystem 2.

## 5.2 Requirements Analysis

### 5.2.1 Requirements

The Requirements[R] for Module 1 [M1], Module 2 [M2] and Module 3 [M3] are outlined in [Table 5.1](#).

Table 5.1: Requirements [R] of Modules of Subsystem 2

| Module | Requirement ID | Description   |
|--------|----------------|---|
| M1     | R01            | The power module must provide a stable voltage to the micro-controller.     |
| M1     | R02            | There must be protection circuitry for the battery.                         |
| M1     | R03            | The battery must be rechargeable.   |
| M2     | R04            | The position co-ordinates sent to the micro-controller must be accurate.    |
| M2     | R05            | There should be a short cold start time.                                    |
| M3     | R06            | The micro-controller must provide stable voltages to the required circuits. |
| M3     | R07            | Stable PWM signals must be generated.                                       |
| M3     | R08            | The micro-controller must be able to accurately sample the output voltage.  |
| M3     | R09            | The micro-controller must be able to process and transmit the sampled data. |
| -      | R10            | The entire subsystem design must not exceed the budget.                     |

### 5.2.2 Specifications

Specifications of Subsystem 2 are as follows, each defined for a specific module/ all modules:

Table 5.2: Specifications [SP] of Subsystem 2

| Module | Specification ID | Description   |
|--------|------------------|---|
| M1     | SP01             | The power module must provide $5V \pm 0.20V$ input to the micro-controller.   |
| M1     | SP02             | There must be over-charge and over-discharge protection.  |
| M1     | SP03             | The module must support micro-USB / USB-C charging.   |
| M2     | SP04             | With good atmospheric conditions, accuracy $< 5$ meters is expected.  |
| M2     | SP05             | The cold start time should be less than 2 minutes.  |
| M3     | SP06-1           | The micro-controller must provide a $5V \pm 0.2V$ input to the GPS Module.  |
| M3     | SP06-2           | The micro-controller must provide a $5V \pm 0.2V$ input to the Light Detection and Amplification module in Subsystem 1.                   |
| M3     | SP06-3           | The micro-controller must provide a $3.3V \pm 0.2V$ input to the Light Emission Module in Subsystem 1.                                    |
| M3     | SP07             | The micro-controller must be able to generate two inverse PWM signals (0-3.3V) between the range of 100 Hz-10kHz and a duty cycle of 50%. |
| M3     | SP08             | The ADC used to sample analogue signals should have 12-bit resolution.  |
| M3     | SP09             | The micro-controller must be able to process data and append information on a cloud back-end.   |
| -      | SP10             | The combined cost for all modules must be below R750.   |

### 5.2.3 Testing Procedure

The testing procedures outlined in section 5.4.1 have been summarised into the following table:

Table 5.3: Summary of Acceptance Testing [AT] of Subsystem 2

| Module | AT ID  | Description   |
|--------|--------|---|
| M1     | AT01   | Measure the micro-controller's input voltage.                                     |
| M1     | AT02   | Confirm the existence of over-charge and over-voltage protection.                 |
| M1     | AT03   | Verify that the battery is charged with a micro-USB/USB-C cable.                  |
| M2     | AT04   | Confirm that test GPS co-ordinates match recorded GPS values.                     |
| M2     | AT05   | Measure the time taken for the first GPS Satellite reading.                       |
| M3     | AT06-1 | Measure the voltage provided to the GPS from the microcontroller.                 |
| M3     | AT06-2 | Measure the voltage sent to the Light Detection and Amplification Module.         |
| M3     | AT06-3 | Measure the voltage provided to Light Emission Module.                            |
| M3     | AT07   | Confirm that the correct PWM signals are generated                                |
| M3     | AT08   | Verify the ADC operation is correct and accurate.                                 |
| M3     | AT09-1 | Ensure that GPS input data is processed correctly.                                |
| M3     | AT09-2 | Ensure that voltage input data is processed into ratio values correctly.          |
| M3     | AT09-3 | Ensure that the voltage input data is processed into pulse rate values correctly. |
| M3     | AT09-4 | Ensure that output data is transmitted to the cloud back-end correctly.           |
| -      | AT10   | Calculate the total cost.   |

### 5.2.4 Traceability Analysis

The acceptance tests are outlined in section 5.4.1.

Table 5.4: Requirements Traceability Matrix of Subsystem 2

| #  | Module | Requirements | Specifications         | Acceptance Tests            |
|----|--------|--------------|------------------------|-----------------------------|
| 1  | M1     | R01          | SP01                   | AT01                        |
| 2  | M1     | R02          | SP02                   | AT02                        |
| 3  | M1     | R03          | SP03                   | AT03                        |
| 4  | M2     | R04          | SP04                   | AT04                        |
| 5  | M2     | R05          | SP05                   | AT05                        |
| 6  | M3     | R06          | SP06-1, SP06-2, SP06-3 | AT06                        |
| 7  | M3     | R07          | SP07                   | AT07                        |
| 8  | M3     | R08          | SP08                   | AT08                        |
| 9  | M3     | R09          | SP09                   | AT09-1,AT09-2,AT09-3,AT09-4 |
| 10 | -      | R10          | SP10                   | AT10                        |

**Traceability Analysis 1 :** The need for stable microcontroller power from the battery (R01) is detailed in SP01, which states that a stable 5V supply is required. To test this, AT01 suggests voltage measurement to verify that the expected value is received.

**Traceability Analysis 2 :** Battery protection requirements (R02) are addressed with overcharge/discharge circuits (SP02) and confirmed via AT02, which is component verification on data sheets.

**Traceability Analysis 3 :** Recharge-ability (R03) is enabled by either micro-USB or USB-C cable support (SP03) and tested with charging validation (AT03).

**Traceability Analysis 4 :** Position accuracy (R04) is detailed by the specification SP04 that states that the accuracy of the GPS Module must be less than 5m matching to the true value and validated against those true co-ordinates (AT04).

**Traceability Analysis 5 :** Cold start time (R05) is limited to less than 2 minutes (SP05) and measured during initialization (AT05).

**Traceability Analysis 6 :** Voltage provisioning (R06) is split into 5V and 3.3V outputs (SP06-1/2/3) and tested per rail (AT06-1/2/3).

**Traceability Analysis 7 :** PWM stability (R07) is achieved with 100Hz–10kHz signals with 8-12 bit resolution (SP07) and verified via oscilloscope readings (AT07).

**Traceability Analysis 8 :** Voltage sampling (R08) uses a 12-bit ADC (SP08), confirmed through resolution tests (AT08).

**Traceability Analysis 9 :** Remote processing and transmission (R09) is achieved through the use of a cloud back-end (SP09) and verified by sending and receiving test data (AT09-1/2/3/4).

**Traceability Analysis 10 :** Budget compliance (R10) is enforced via a R750 maximum (SP10) and audited through cost summation (AT10).

## 5.3 Design

### 5.3.1 Power Module Component Selection

Several power solutions were evaluated for the penguin monitoring system. Due to the time limitation, it was decided that designing a PCB would have too long a lead time, as testing could not occur when waiting for the PCB to be delivered. The alternative is to buy components from local stores and create a module that not only meets the requirements of the Subsystem's Modules, but can be connected and tested for a longer period of time. This led to the choice of a suitable battery and converter board to fulfil all specifications set out for the power module.

To select the most suitable combination of battery and converter board, multiple options were considered. A lead-acid battery (6V, 2000mAh) was rejected due to excessive weight and incompatibility with USB charging, while a 1200mAh LiPo offered marginal runtime gains but required bulky protective casing. For voltage regulation, a linear regulator (LM7805) was dismissed for its inefficiency ( $>60\%$  power loss) and input voltage constraints, and an advanced buck-boost converter (TPS63060) was deemed unnecessary for the fixed  $3.7V \rightarrow 5V$  conversion. Ultimately, a  $3.7V$ , 800mAh LiPo battery paired with a compact dc/dc boost converter that specifically converts  $3.7V$  to  $5V$  was selected. The battery has its own protection circuit with a DW01 battery protection integrated circuit that has over-current and short-circuit protection amongst others. The boost module supported 5V USB output and a micro-USB charging functionality with overcharge and over-discharge protection. This combination has the optimal balance of cost ( $R58 + R126.3 = R184.3$  total), lightweight design, and compliance with all specifications.

While the 800mAh capacity limits runtime under continuous load, it is an appropriate choice for the

first iteration of the design and testing process. If the power module was to be implemented in the field, attached to the penguin and monitoring vital signs for a fixed period of time, a higher capacity battery would be required. Additionally, the micro-controller can be set to low-power and deep sleep modes when not in operation to reduce power consumption.

### 5.3.2 Location Module Component Selection

For the location module, multiple GPS modules were considered. As discussed previously, it was decided to use components bought from local stores to comply with the time limitations, so all options detailed below are available in local stores. Note that the price listed below is inclusive of VAT.

Table 5.5: Comparison of GPS Modules considered for the Location Module

| Name         | Price | Accuracy | Cold start time | Antenna          | Comment                   |
|--------------|-------|----------|-----------------|------------------|---------------------------|
| Adafruit GPS | R547  | 3m       | 40s             | Internal Patch   | Too expensive             |
| SIM800L      | R209  | 5m       | 60s             | External Cable   | Supply voltage: 3.4V-4.4V |
| NEO-6M       | R100  | 2m       | 38s             | External Ceramic | Out of stock              |
| NEO-M8N      | R295  | 0.9m     | 30s             | External Ceramic | A suitable choice         |

Table 5.5 explains the characteristics of each GPS Module considered as well as the reason for it being rejected or selected. The Adafruit GPS price was not only high when compared to the other prices, the characteristics of the module did not justify the increased price, and for this reason it was rejected. The SIM800L required a voltage that was not easily provided (the intended voltages for power were decided to be 3.3V from a micro-controller and 5V from the power module). The NEO-6M would have been an ideal choice as the price is much lower than the others and the characteristics are within the specifications, however it was out of stock at the time of purchase. The NEO-M8N was selected, despite its high cost, as the budget allowed it, and it was well within the bounds of performance set by the specifications.

### 5.3.3 Processing Module

#### Micro-controller Selection

When deciding on a micro-controller to use for the processing module, the STM32 and the ESP32 were both considered. While the STM32 has up to 32 bit resolution and more UART/SPI ports, it requires an external programmer (ST-LINK programmer) and does not have built-in wireless communication functionality like the ESP32.

The ESP32 (development kit version 1) offers a solution that is easier to manufacture and test as it does not require external peripherals, it can be programmed with a micro-USB cable and has Wi-Fi capabilities allowing for wireless transmission of data. The pinout It has a VIN pin which allows 5V to be sent to the required circuits. It has 0V-3.3V pins to provide the PWM signal and stable power to the required circuits. The ADC can be set to 12 bit resolution. Additionally, the cost is R117, which falls within the budget. Thus, it fulfils all requirements and the associated specifications. The pinout for the ESP32 devkit v1 can be found here [23].

#### Development environment selection

Various development environments were considered for use in this Subsystem. These are compared in [Table 5.6](#). For Maintenance Cost, Arduino IDE is seen to have a low cost as many community-driven solutions and online resources are available. PlatformIO and ESP-IDF will have higher costs associated as more expertise is required.

Table 5.6: Comparison of development environments considered for the Processing Module

| Name        | Technical Maturity | Testing Simplicity      | Maintenance Cost | Library Support                |
|-------------|--------------------|-------------------------|------------------|--------------------------------|
| Arduino IDE | High               | Serial Monitor Built-in | Low              | 3000+ Arduino libraries        |
| PlatformIO  | Medium             | Advanced unit testing   | Medium           | Arduino and ESP-IDF hybrids    |
| ESP-IDF     | High               | Requires JTAG debugger  | High             | Limited (Hardware level focus) |

From [Table 5.6](#), it can be seen that Arduino IDE is the most suitable choice for this application.

### 5.3.4 Final Design

After component selection, each module had to be configured. The details are outlined below.

#### GPS Module Connection

The GPS Module is connected to the ESP32 using UART1 on GPIO16 and GPIO17. The default Baud rate is 9600. To interpret and process the incoming data from the GPS Module, the library TinyGPSPlus [\[24\]](#) is used. A TinyGPS object is made and the latitude and longitude information obtained from the GPS Module is saved on the ESP32 to be sent to the cloud back-end.

#### PWM Signal Generation

The Light Emission Module of Subsystem 1 requires two inverse PWM signals with a duty cycle of 50%. This means that when one PWM is high, the other is low, and vice versa. This is achieved through the use of the LED Control (LEDC) library from Espressif Systems [\[25\]](#), which is primarily used to control the intensity of LEDs, but can be used to generate PWM signals. The GPIO Pins 18 and 19 are selected for PWM generation and the specified PWM signals are set using the ledcAttach, ledcWrite and ledcOutputInvert functions. This ensures that the signals are inverted as required by the specific circuits in Subsystem 1. The frequency of these signals is selected as 100Hz, with a 10-bit resolution (therefore the full cycle will be  $2^{10} = 1024$ ), thus the 50% duty is set by assigning the value of 512 (half of the full resolution value).

#### ADC Sampling

The ESP32 has a 12-bit ADC, therefore there are 4096 discrete analogue levels. In this project, Wi-Fi will be enabled, so ADC1 pins are used as ADC2 pins cannot be used when Wi-Fi is enabled. The GPIO Pin 34 is selected for the ADC to be configured. Since the voltage signal coming into the ADC will be a square wave with a frequency of 100Hz (period of 10ms), the square wave will be high for 5ms and low for 5ms. This leads to the conclusion that the ADC pin must be read every 5ms to get a reading for both the high and low values of the square wave. However, 5ms is the bare minimum and a value of 2.5ms is selected as the delay between readings - allowing for 2 readings per high and low value of the square wave. The ADC takes 2000 samples, each with a delay of 2.5ms, therefore it samples for 5 seconds.

### Blood Oxygen Saturation Ratios

The result of the ADC sampling is an array with 2000 values. Since this should be 1000 high values and 1000 low values, they are arranged into ascending order and an average value is found for both the high values and the low values. This results in the two ratio values that are appended to the Google Sheets. Those values are then used as inputs to Subsystem 3, where it is processed into information.

### Pulse Rate Calculation

As mentioned previously, the result of the ADC sampling is an array with 2000 values. This is from sampling for 5 seconds. A function is created to locate the local maximums (distinct peaks in data). The voltage that forms the output square waveform will ideally have a number of distinct peaks (the value will be slightly higher than the other periods of the waveform before and after it). The number of distinct peaks in the waveform for the 5 seconds can be multiplied by 12 to translate the value into the number of distinct peaks in the waveform for 60 seconds. Thus the number of pulses for 1 minute can be extrapolated from the data.

### Calibration function

The value received on the ADC pin will be influenced by the amount of light present. This is described in detail in the Light Detection and Amplification Module of Subsystem 2. To obtain accurate values, the PWM signals are set for the 5 seconds then both are switched off and another measurement is obtained: a calibration value. This value is obtained by sampling the ADC pin value 20 times and calculating the average value. This is the amount of ambient light present, as the PWM pins that provide power to the LEDs (IR and red) are off. This calibration value is appended to the Google Sheets along with the sampled ADC values, to be interpreted and displayed by the next subsystem, Subsystem 3.

### Wi-Fi Connection

The ESP32 is able to connect to Wi-Fi without any additional peripherals. Since the chosen development environment is Arduino IDE, there are many online resources that detail exactly how to create a Wi-Fi connection. The specific file used is ‘esp\_wpa2.h’ from the ESP32-eduroam[26] library, which allows connection to Enterprise Wi-Fi networks. The identity and password variables can be adjusted to allow personal log-ins where required.

### Google Sheets Connection

The ESP32 processing module transmits penguin biometric and GPS data to Google Sheets via HTTP POST requests to a deployed Google Apps Script web app. The script acts as a middleware API, parsing incoming JSON payloads (containing the ratio values, GPS co-ordinates and pulse rate) and dynamically appending records to the spreadsheet while optionally updating status cells. This architecture uses Google’s infrastructure for reliable data logging without requiring a dedicated back-end, with timestamps generated server-side to maintain accuracy even during ESP32 sleep cycles.

### Final Code

The final code is uploaded to the github. It includes a `setUp()` function that is run once to begin serial communication and set up the Wi-Fi connection and a `main loop()` that is run continuously. The `loop()` will reset the `Calibration` variable at every iteration to ensure the value is accurate for each sampling process. The PWM pins are then set as described in the earlier sections, using the `ledc`

library. The ADC on pin 34 then reads the values using `analogRead()` and stores the values in an array `adcReadings`. The array `adcReadings` is processed to provide `ratio1` and `ratio2` and the pulse rate. The PWM signals are then switched off (to save power) and the calibration value is read and stored. Once the readings from Subsystem 1 is complete, the latitude and longitude values from the GPS Module is obtained. Once all the data has been recorded, the Wi-Fi connection established earlier is used to append and update the values in the Google Sheets.

### 5.3.5 System Integration and Interfacing

The connections between Subsystem 1 and the other Subsystems are detailed in [Figure 5.2](#).

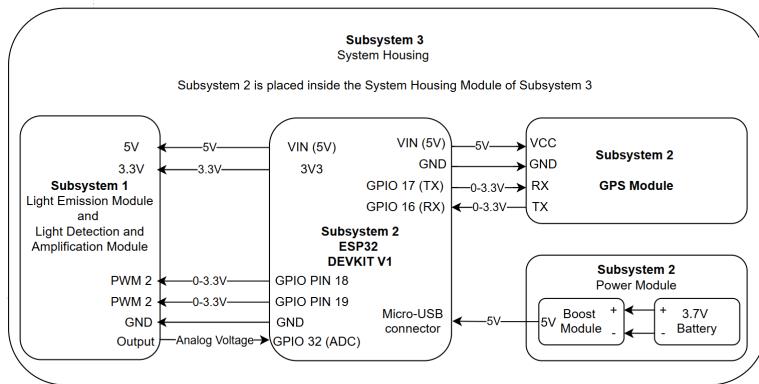


Figure 5.2: Subsystem 2 Interfacing Diagram with other Modules and other Subsystems

## 5.4 Results

### 5.4.1 Tests

The acceptance tests are explained in detail in [Table 5.7](#). Any connections must be done as instructed by the system interfacing diagram: no further explanation will be described in the ‘Procedure’ column. When power is needed and the Power Module is not being tested, a power supply (with a micro-USB output that feeds into the ESP32 micro-USB power input port) will be used to provide power to the other systems. This is to test each module independently.

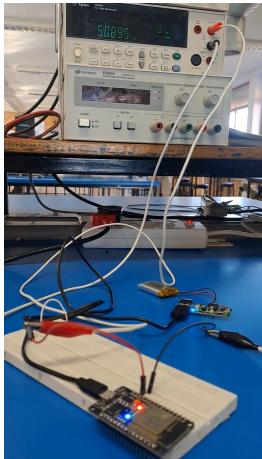
Table 5.7: Acceptance Tests for Subsystem 2

| Test ID | Description   | Procedure  | Pass Criteria   |
|---------|---|--|---|
| AT01    | Measure the micro-controller’s input voltage.                     | Connect the Power module to the ESP32. Connect a multimeter across the VIN pin and GND of the microcontroller and measure the input voltage of the micro-controller.   | The voltage must be within the acceptable range $5V \pm 0.2V$ .                           |
| AT02    | Confirm the existence of over-charge and over-voltage protection. | Verify the components on the data sheet of the selected modules and confirm their presence on the module itself  | The data sheets must specify protection circuitry that can be seen on the physical model. |
| AT03    | Verify that the battery is charged with a micro-USB cable.        | Connect the battery to the booster board. Plug the micro-USB end of the cable into the charging port of the booster module. Use an oscilloscope or multimeter to monitor the voltage, ensuring that the USB end of the cable is plugged into a power supply. | The voltage increases over time until it reaches the maximum 4.2V.                        |

|        |  |  |  |
|--------|--|--|--|
| AT04   | Confirm that test co-ordinates match recorded GPS values.  | Connect the ESP32 to a 5V power supply and the GPS Module. Obtain the true location co-ordinates of the location of the GPS.   | The longitude and latitude obtained by the GPS Module match the true location co-ordinates within an acceptable range. |
| AT05   | Measure the time taken for the first GPS reading.  | Connect the ESP32 to a 5V power supply and GPS Module. Ensure the GPS Antenna is facing the open sky. Measure the time taken for a first fix with a stopwatch.   | The time taken must be less than 2 minutes.  |
| AT06-1 | Measure the voltage provided to the GPS from the micro-controller                                      | Connect the ESP32 to a 5V power supply and the GPS Module. Use a multimeter across the VCC and GND pins of the GPS Module.   | The measured voltage must be 5V.   |
| AT06-2 | Measure the voltage provided to the Light Detection and Amplification Module from the micro-controller | Connect the ESP32 to a 5V power supply and the Light Detection and Amplification Module. Use a multi-meter to confirm that the voltage provided is 5V.   | The multimeter value reads $5V \pm 0.2V$ .   |
| AT06-3 | Measure the voltage provided to the Light Emission Module from the micro-controller                    | Connect the ESP32 to a 5V power supply and the Light Emission Module. Use a multi-meter to confirm that the voltage provided is 3.3V.  | The multimeter value reads $3.3V \pm 0.2V$ .   |
| AT07   | Confirm that the correct PWM signals are generated.  | Connect the ESP32 to a 5V power supply. Flash the temporary testing file that just sets the PWM signals and reads the values. Use an oscilloscope to monitor the PWM signals.  | The PWM signals must be 100Hz and move from 0-3.3V. The signals must be inverted (when one is high the other is low).  |
| AT08   | Verify that the ADC operation is correct and accurate.   | Connect the ESP32 to a power supply. Use a signal generator to create a square waveform with a frequency of 100Hz and duty cycle of 50%. Monitor the output on the ADC pin 34 using the Serial Plotter on Arduino IDE. | The oscilloscope must produce the waveform that has been provided.   |
| AT09-1 | Ensure that GPS input data is processed correctly.   | Connect the ESP32 to a power supply. Flash GPSv1.ino from the code source files to the ESP32. Monitor the latitude and longitude values being processed.   | The location co-ordinates must be in the correct format.   |
| AT09-2 | Ensure that voltage input data is processed into ratio values correctly.                               | Connect the ESP32 to a power supply. Generate a test square wave signal of 1.5V to 2V to be sampled.   | Ensure that the ratios in the Serial Monitor match the input.  |
| AT09-3 | Ensure that voltage input data is processed into pulse rate correctly.                                 | Connect the ESP32 to a power supply. Use a limited 100Hz sine wave with 7 distinct peaks to test the peak counting function.   | The pulse rate matches 84 beats per min.   |
| AT09-4 | Ensure that the output data is transmitted to the cloud back end correctly.                            | Connect the ESP32 to a power supply. Flash Test.ino of the code files as that has manually added values to the ESP32 and is simply checking the upload functionality.  | The values in the ESP32 code must be appended to the Google Sheets.  |
| AT10   | Calculate the total cost.  | The cost of each component used to build Subsystem 2 is summed.  | The total value must be less than $R750$ .   |

#### 5.4.2 Results Analysis

##### Results Analysis: AT01



This test follows R01 and SP01 - the battery must provide a  $5V \pm 0.20V$  input to the micro-controller. In order to pass this test, the power module must be configured by connecting the battery to the boost converter. Then a USB output from the boost converter must be connected to the micro-USB input to the ESP32. The voltage across VIN and GND must be measured with a multi-meter. This setup is shown to the left. Since the measured voltage across the input pins of the ESP32 is seen to be 5.0895V, this test is passed.

Figure 5.3: Acceptance Test 01 Setup and Measurement

### Results Analysis AT02

This test follows the requirement R01, which states that there must be protection circuitry for the power module, and SP01 which states the over-charge and over-voltage protection must be present. The chosen battery has a DW01 IC and a small PCB containing its protection circuitry, and in addition to the battery itself being protected, the dc/dc boost converter has its own protection circuitry. Thus, the test is passed.

### Results Analysis AT03

This test follows R03, which requires the battery to be rechargeable, and SP03, which states that the power module must support micro-USB charging. This is conducted by plugging in a cable supported by the Power Module and monitoring the voltage across the battery. The result of the test is that the battery does charge until the green LED on the boost module switches on, indicating that the battery is fully charged. Thus, the test is passed.

### Results Analysis AT04

This test follows the first requirement for the Location Module, R04, which is asking for accurate readings to be obtained. The readings must be within 5 meters, according to SP04. When the NEO-M8N was connected, all Baud rates resulted in garbled outputs. Despite multiple efforts at trouble shooting, a solution was not found with the NEO-M8N. However, at this point, a NEO-6M became available. When the NEO-6M was used, the NMEA data was received in the correct format. However, the accuracy of the NEO-6M was not as good as indicated. Two tests were performed where the location was known and used to measure the accuracy of the GPS Module. The straight line distance tool on Google Maps was used to track how far off the GPS Module co-ordinates were. The first test resulted in the co-ordinates being 470.88m away from the true test location and the seconds test was 148.29m away from the true location. Since the specification is to have it within 5m, this test is failed.

### Results Analysis AT05

This test follows the requirement R05, which states that the cold start time should be less than 2 minutes (SP05). This specification is under ideal conditions (a clear view of the sky is required). A stopwatch measured a cold start time of 31.38 seconds before the first measurement was obtained. Thus, the test is passed.

### Results Analysis AT06-1/2/3

This test follows the first requirement for the Processing Module, R06, which says that stable voltages (Voltage  $\pm 0.2V$  - SP06-1/2/3) must be provided to the necessary circuits and components. The GPS Module and the Light Detection and Amplification Module require 5V and the Light Emission Module requires 3.3V. These are all shown with their setups in the figures below. The measured voltages are 4.9063V, 5.0522V and 3.2859V for the GPS Module, Light Detection and Amplification Module and the Light Emission Module respectively. Since the measured voltages are within the acceptable ranges, these tests are passed.

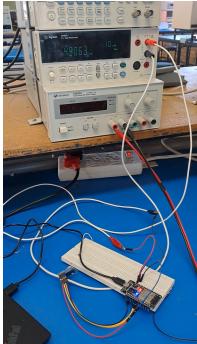


Figure 5.4: Measured Voltage (4.9063V) supplied to GPS Module

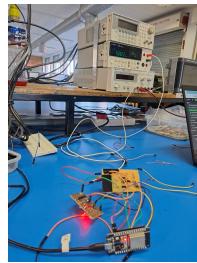


Figure 5.5: Measured Voltage (5.0522V) supplied to Light Detection and Amplification Module

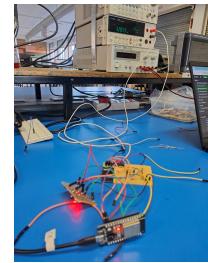


Figure 5.6: Measured Voltage (3.2859V) supplied to Light Emission Module

### Results Analysis AT07

This test follows R07, which requires two stable 100Hz, 0-3.3V, 50% duty cycle PWM signals that are inverted with respect to the other (SP07). This test is conducted by using an oscilloscope to view the signals generated by the ESP32. Since they have a measured frequency of 100Hz and duty cycles of 50% and are inverse signals, seen by [Figure 5.7](#) the test is passed. Additionally they were both 0-3.3V, although this cannot be seen in the figure below.

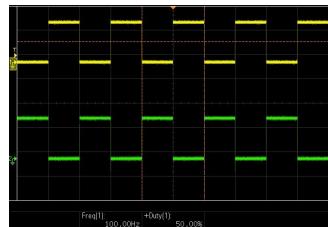


Figure 5.7: PWM signals generated by ESP32

### Results Analysis AT08

This test follows the ADC requirement, which is that 12-bit resolution should be used to accurately sample the output voltage (SP08). A PWM signal is generated by a signal generator and passed through the ADC Pin and displayed on the Serial Monitor of Arduino IDE. The output can be seen in the figure below, and since the signal can be accurately reconstructed and the 12-bit resolution is verified, the test is passed.

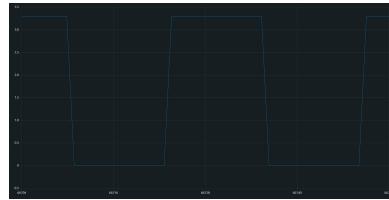


Figure 5.8: ADC Sampling reconstruction of square wave

### Results Analysis AT09

These tests follow the last requirement for the processing module, R09, which is regarding the processing and transmission of the sampled data. AT09-1 tests that the GPS data is processed correctly using the TinyGPSPlus library. The NMEA sentences are parsed into latitude and longitude which confirm the location of the tracker, thus passing the test. AT09-2 is the connection of a test waveform that is passed into the ADC to be sampled. The values in the Serial Monitor matched 1.5V and 2V with an accuracy of 95%, thus passing the test. AT09-3 was tested with a limited sine wave and 7 peaks were counted in the 5 seconds. The function counted 7 distinct peaks, and multiplied that by 12 to give a pulse rate of 84. Thus, the test is passed. AT09-4 tested the transmission of set data. This can be run once with sample data, as is found in version1.ino of the code. Since version1 is able to transmit data successfully, the test is passed.

### Results Analysis AT10

This test follows the Subsystem requirement R10 for the overall cost to be within the budget of R750, as set out by SP10. As seen in the Bill of Materials, Subsystem 2 section in the appendix, the total cost of Subsystem 2 is R358.69, which is less than R750, thus passing the test.

## 5.5 Conclusion

Acceptance testing shows that Subsystem 2 meets most of its requirements and performs reliably. Of the ten tests (AT01–AT10), nine were passed, confirming stable voltage delivery, proper protection circuitry, and effective micro-USB charging. The ESP32 produced accurate PWM signals, the ADC correctly sampled signals at 12-bit resolution, and data was successfully transmitted to Google Sheets. The total cost of R358.69 is well below the R750 budget, making the design cost-effective.

However, the GPS module failed to meet the 5-meter accuracy requirement. The NEO-M8N could not be used due to communication issues, and the backup NEO-6M, while functional, produced errors of 470.88 m and 148.29 m. Despite meeting cold start time requirements, its inaccuracy limits its use for precise applications.

It is recommended to replace or upgrade the GPS module, either by revisiting the NEO-M8N or selecting a higher-accuracy alternative like the NEO-M8T or a multi-constellation receiver. Future iterations could also benefit from a higher-capacity battery to support longer operation, especially if additional components are added.

# Chapter 6

## Subsystem 3

Author: Kashan Pillay (PLLKAS005)

### 6.1 Introduction

The greater project is a remote penguin pulse oximeter. This is in essence a device that is capable of measuring the oxygen saturation levels and heart rate of a penguin. This particular subsystem involves two tasks, the design and manufacture of a robust and suitable housing and the design and implementation of an aesthetically pleasing front-end user interface for the system. The housing will be responsible for securing the first two subsystems from various environmental factors and securing the system to a penguin flipper and the user interface will be responsible for allowing the user to access, filter and monitor the various health metrics remotely, so as not to ensure minimal handling of the penguin.

#### 6.1.1 Context

Subsystem 3 is the final link in the overarching remote pulse oximeter system. Subsystem 2 provides the processed signals as well as GPS data to the cloud, from which the GUI is able to analyse, manipulate and display as per the user's requirement. Additionally, the System Housing is used as a means to secure the delicate electronic equipment from user and environmental harm as well as ensure appropriate positioning on the penguin.

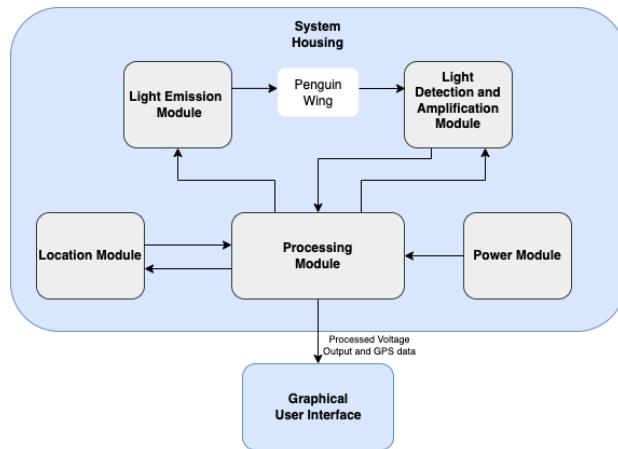


Figure 6.1: A diagram indicating how Subsystem 3 interacts and fits in with the system as a whole.

#### 6.1.2 Scope and Limitations

The scope of this subsystem is to design, manufacture and implement a suitable system housing structure to protect Subsystem 1 and Subsystem 2 as well as an easy to use graphical user interface for the stakeholder to access the health metrics. Subsystem 1 is responsible for the design and manufacture of the analogue electronic circuitry which supplies Subsystem 2 with signals, from which Subsystem 2

processes will provide this subsystem with the necessary data. Additionally, the following objectives are also to be implemented:

- Designing for reliability.
- Ensuring an easily accessible and user friendly experience.
- Ensuring a minimally invasive but secure housing for delicate electronics.

There are also several limitations to consider:

- Design Limitation: The system housing should be a suitable size for African Penguins with the necessary security straps.
- Time Limitation: The final complete and working overall system (Subsystem 1, 2 and 3) is to be demonstrated on 26/05/2025.
- Cost Limitation: There is a cost restriction of R250 for this subsystem.
- Environmental Limitation: The system housing should be able to protect the electronic components from various environmental and user hazards.

## 6.2 Requirements Analysis

### 6.2.1 Requirements

The requirements for subsystem 3, system housing and graphical user interface are described in [Table 6.1](#)

Table 6.1: Requirements[R] of the sensing subsystem design.

| Requirement ID | Description   |
|----------------|---|
| R01            | The system housing must be an appropriate size to fit a penguin flipper.                      |
| R02            | The system housing must be minimally invasive.  |
| R03            | The system housing must be easily attachable to the penguin flipper.                          |
| R04            | The system housing and front-end development must cost within the specified budget.           |
| R05            | The GUI of the front-end system must be user friendly.  |
| R06            | The front-end system must be able to work remotely.   |
| R07            | The front-end system must display the data gathered by the sensors.                           |
| R08            | The GUI of the front-end system must be aesthetically pleasing.                               |
| R09            | The front-end system must be modular to ensure easy scalability, development and maintenance. |

### 6.2.2 Specifications and Testing Procedure

The specifications, refined from the requirements in [Table 6.1](#), for the micro-mouse sensing module are described below in [Table 6.3](#). Additionally, a summary of the acceptance testing [AT] procedures detailed in [section 6.4](#) is given below in [Table 6.4](#)

Reference Key:

'AT' - Acceptance Testing; 'UT' - Unit Testing; 'UAT' - User Acceptance Testing; 'BAT' - Business Acceptance Testing.

Table 6.2: Specifications[SP] of the sensing subsystem derived from the requirements in [Table 6.1](#).

| Specification ID | Description  |
|------------------|--|
| SP01             | The system housing must fit a wing span of 12cm to 17cm.   |
| SP02             | The system housing must weight less than 5% of a penguins body weight and not increase drag in the water.  |
| SP03             | The system housing must be retrievable without recapture and attach with temporary harnesses or leg bands. |
| SP04             | The design and manufacture of the entirety of subsystem 3 must not cost more than R250.                    |
| SP05-1           | The front-end system GUI must have clear navigation elements and error handling.                           |
| SP05-2           | The front-end system GUI must have a font larger than 12pt and respond to a user action within 2 seconds.  |

Table 6.3: Specifications[SP] of the sensing subsystem derived from the requirements in [Table 6.1](#).

|        |   |
|--------|---|
| SP06   | The front-end system must be on a device that supports any wireless network connection to connect to the cloud back end.      |
| SP07-1 | The front-end system must display current and historical health and tracking data labelled descriptively.                     |
| SP07-2 | The front-end system must filter data by a time-range and have threshold colour-coded indicator alerts.                       |
| SP08-1 | The front-end system GUI must use a consistent colours, fonts as well as a grid layout alignment structure and hover effects. |
| SP08-2 | The front-end system GUI must adapt to screen sizes without loss of alignment.  |
| SP09-1 | The front-end system development must be done on a component based framework.   |
| SP09-2 | The front-end system development must ensure use of object orientated programming.  |

Table 6.4: Acceptance Testing [AT] descriptions derived from the table of requirements in [Table 6.1](#)

| AT ID | AT Type | Description  |
|-------|---------|--|
| AT01  | UAT     | Utilise model design measurements to ensure fitment on a wingspan of 12cm to 17cm.                 |
| AT02  | UAT     | Weigh the fully assembled system and ensure that it is less than 175g.                             |
| AT03  | UAT     | Ensure that the attachment mechanism is secure but fast and easy to attach and detach.             |
| AT04  | BAT     | Review the Bill of Materials.  |
| AT05  | UAT     | Conduct user test sessions to complete a certain task.   |
| AT06  | UT      | Attempt to access data from the cloud back-end with a device connected to various networks.        |
| AT07  | UAT     | Load the GUI and navigate to the necessary page to verify that the required data is presented.     |
| AT08  | UAT     | Load the GUI and conduct a user feedback survey on colour, clarity, attractiveness and impression. |
| AT09  | BAT     | Conduct an inspection of the source code to ensure encapsulation in reusable modules.              |

### 6.2.3 Traceability Analysis

To show how the requirements ([Table 6.1](#)), specifications ([Table 6.3](#)) and testing procedures ([Table 6.4](#)) all connect, [Table 6.5](#) is provided.

Table 6.5: Requirements Traceability Matrix

|   | Requirements | Specifications | Acceptance Test |
|---|--------------|----------------|-----------------|
| 1 | R01          | SP01           | AT01            |
| 2 | R02          | SP02           | AT02            |
| 3 | R03          | SP03           | AT03            |
| 4 | R04          | SP04           | AT04            |
| 5 | R05          | SP05-1, SP05-2 | AT05            |
| 6 | R06          | SP06           | AT06            |
| 7 | R07          | SP07-1, SP07-2 | AT07            |
| 8 | R08          | SP08-1, SP08-2 | AT08            |
| 9 | R09          | SP09-1, SP09-2 | AT09            |

**Traceability Analysis 1:** The system housing must fit a penguin flipper with a wingspan of 12cm–17cm (SP01). This ensures minimal interference with movement. AT01 verifies fitment using physical measurements or a model, confirming compliance with R01. A key consideration is that non-compliance with the above could hinder movement or decrease swimming efficiency.

**Traceability Analysis 2:** The housing must weigh less than 175g (5% of the body weight of an African penguin) to minimize drag (SP02). AT02 checks weight compliance, ensuring the requirement (R02) of minimal invasiveness is met. Similarly to the above, non-compliance with respect to this requirement could hinder movement or decrease swimming efficiency.

**Traceability Analysis 3:** The housing must use temporary harnesses/leg bands for attachment to reduce the need for recapture (SP03). AT03 tests the mechanism’s ease of use, validating R03. A key consideration surrounding the above analysis is that poor attachment could result in premature detachment or injury.

**Traceability Analysis 4:** The total cost must be less than R250 (SP04). AT04 calls for the review the Bill of Materials (BOM) to ensure budget adherence. Thus validating R04.

**Traceability Analysis 5:** The GUI must have clear navigation, a font size greater than 12pt, and less than 2 seconds response time (SP05-1 and SP05-2). AT05 involves user testing to confirm intuitiveness, and so validating R05. A key consideration here would be that a non-user centric design could make data retrieval difficult for the stakeholder.

**Traceability Analysis 6:** The system must be on a device that supports WiFi/cellular cloud connectivity (SP06). AT06 tests remote data access across networks, validating R06.

**Traceability Analysis 7:** The GUI must show current and historical data with time filters and colour-coded alerts (SP07-1 and SP07-2). AT07 verifies data visibility to the stakeholder, verifying R07. A key consideration here is that mislabelled or misidentified data could result in the cause of poor ecological conclusions.

**Traceability Analysis 8:** The GUI must use consistent colours, a grid layout, and responsive design (SP08-1 and SP08-2). AT08 collects user feedback on aesthetics, validating R08.

**Traceability Analysis 9:** The front-end must use a component-based framework and Object Orientated Programming (SP09-1 and SP09-2). AT09 inspects code modularity, verifying R09. A key consideration in this case would be that poorly formatted code can severely impact further development.

## 6.3 Subsystem Design

### 6.3.1 Design Decisions

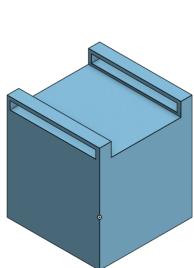
In this section, some of the most crucial aspects of the design process of this subsystem are going to be explored. From which, many design decisions are to be made.

#### Submodule 1: System Housing

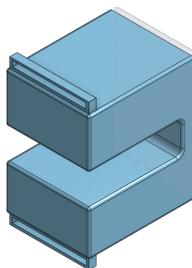
Three main considerations when it came to the design of the system housing are going to be considered:

- Structural Design:

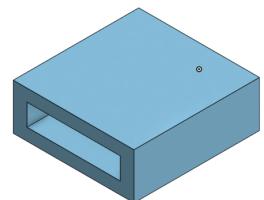
Three different conceptual designs were considered for the system housing - Model 1, Model 2 and Model 3. The purpose of this housing unit is to ensure that the delicate electronic equipment is secured from damage due to user actions or environmental conditions. Additionally, the unit should remain secured to the user's extremity while being minimally invasive.



(a) Structural Design Model 1



(b) Structural Design Model 2



(c) Structural Design Model 3

Figure 6.2: Isometric views of the three considered designs.

Table 6.6: Pros and Cons of the considered system housing designs.

|     | Model 1  | Model 2   | Model 3  |
|-----|--|---|--|
| Pro | -Small and lightweight.<br>-Lower cost.<br>-Secure fastening brackets. | -Hydrodynamic shape.<br>-Robust (low profile of sides).<br>-Secure fastening brackets | -Robust due to low profile.<br>-Lower cost.  |
| Con | -High drag profile.<br>-Less robust due to height.                     | -Higher cost.   | -Unable to securely fasten.<br>-Minimal room for electrical components.<br>-High drag profile. |

## Selected Design Decision:

Model 2 was selected as the ideal structural design due to its hydrodynamic shape and secure fastening brackets. These qualities allow for the housing to be minimally invasive while ensuring that the system remains attached and the electronics inside, secure. While there is a higher cost associated with the design, it provides for the best compromise between scientific enquiry and not impeding on the natural behaviour of the animals.

- 3D Printing Material:

Three different materials were considered for the system housing - Polyethylene Terephthalate Glycol (PETG), Thermoplastic Polyurethane (TPU) and Acrylonitrile Butadiene Styrene (ABS). The reason for a good choice of material is that of ensuring adequate protection of delicate electronic components while keeping low cost and light weight requirements satisfied.

Table 6.7: Pros and Cons of the considered system housing materials.

|     | PETG  | TPU   | ABS   |
|-----|---|---|---|
| Pro | -Lightweight.<br>-Easy to print.<br>-Water resistant. | -Flexible.<br>-Impact resistant.  | -Strong<br>-Heat resistant.<br>-R299 per kg.                            |
| Con | -R349 per kg.   | -Increased drag.<br>-Complexities with precision printing.<br>-R499 per kg. | -Easily warped.<br>-Made from harmful chemicals.<br>-High drag profile. |

## Selected Design Decision:

PETG is selected as the most suitable material for the system housing, fulfilling environmental, structural, and manufacturing requirements essential for minimally invasive deployment. It is lightweight, water-resistant, strong, and easier to print than ABS or TPU, with minimal warping and high dimensional stability. Its semi-rigid nature also ensures adequate protection of internal components without compromising on hydrodynamics.

**Submodule 2: Front-End and Graphical User Interface Development**

The main considerations when it came to development of the front-end and graphical user interface are going to be considered:

- Development Software and IDE:

Two different languages, IDE's and frameworks were considered for the front-end and GUI development. In terms of language and IDE's, the options considered were Python/PyCharm and C++/Visual Studios.

Table 6.8: Pros and Cons of the considered programming languages and IDEs.

|     | Python/PyCharm  | C++/VS  |
|-----|---|---|
| Pro | -Simple syntax.<br>-Version Control.<br>-IDE offers GUI plug-in tools.<br>-Native plotting libraries. | -Good for high frequency sensors.   |
| Con | -No native mobile app support.  | -More suited for Windows applications.<br>-No native plotting libraries.<br>-More complex syntax. |

## Selected Design Decision:

While a preference in coding language is highly subjective, Python and Pycharm were selected primarily for its ability to allow for development speed, cross-platform compatibility and vast plotting and graphical libraries. This significantly reduces the cognitive overhead that would have been required should c# have been used, thus accelerating the development and integration processes. Additionally, Pycharm offered intelligent code suggestions and integrated version control. Ultimately, Python and PyCharm offer the technical agility and toolchain simplicity required to prioritize experimentation and system integration —making them the most appropriate choice for this front-end development project.

- GUI Extension:

Two GUI frameworks were considered for the development of the Graphical User Interface, PyQt and Tkinter.

Table 6.9: Pros and Cons of the considered GUI Frameworks.

|     | PyQt   | Tkinter   |
|-----|--|---|
| Pro | -Modern.<br>-Large widget set.<br>-Powerful component based framework.<br>-Cross platform development. | - Built-in to Python.<br>-Very simple syntax.<br>-Small memory footprint. |
| Con | -More complex syntax.<br>-Greater average execution time.  | -Outdated appearance.<br>-Minimal styling customisation.                  |

## Selected Design Decision:

PyQt was selected as the GUI framework due to its flexibility in design capabilities and component based structure. This allows for long term maintenance as well as easy encapsulation of different aspects of the GUI. Compared to Tkinter, PyQt offers a comprehensive cohort of native widgets, layout managers and styling and colour tools all easily integrated with Pycharm. Additionally, there are no licensing costs and good cross platform consistency (able to maintain behaviour across Windows, macOS and Linux). Given the need for a modular, intuitive, and user-friendly GUI in this project, PyQt presents the best technical and practical choice, outperforming Tkinter in both capabilities and alignment with engineering standards.

- Cloud Back-End:

Two different options were considered for usage a cloud back-end for this system. The first option was Google Sheets, and the second option was FireBase.

Table 6.10: Pros and Cons of the considered cloud back-ends.

|     | Google Sheets  | FibreBase  |
|-----|--|--|
| Pro | -No cost.<br>-Very simple to use.<br>-Version control.<br>-Human readable. | -Easily scalable.<br>-Caching Function.  |
| Con | -Latency.<br>-Complex security (OAuth).                                    | -Cost (usage quota applies).<br>-Data is not readable.<br>-Complex data structuring. |

## Selected Design Decision:

Google sheets was selected for this task due to its alignment with key engineering considerations and requirements surrounding cost-efficiency, implementation simplicity and ease of maintenance. The free usage tier is more than sufficient for the purpose of this project. Additionally, Google Sheets offers widespread support and is a well-established and stable platform on which to base our cloud back end. A distinct advantage in this decision is that Google Sheets offers readable data, which allows for manual verification of data. There is also very little maintenance (no server set up or database schema updating) and no constant back-end monitoring required. Therefore, based on cost, ease of implementation, testability, and long-term maintainability, Google Sheets is the more appropriate and efficient back-end choice for this project.

- Wireless Transmission: Both Wifi and Cellular connectivity were considered for the wireless connection of the unit and front-end system to the cloud back-end.

Table 6.11: Pros and Cons of the considered transmission networks.

|     | WiFi  | Cellular  |
|-----|---|---|
| Pro | -Low operating cost and power consumption.<br>-Low maintenance cost and latency.<br>-Easy testing on a local network. | -Wide area coverage.  |
| Con | -Speed is link dependent.<br>-Coverage is limited to WiFi range.  | -High operating cost.<br>-High power consumption.<br>-Speed is network dependent.<br>-Complexity due to carrier compatibility |

## Selected Design Decision:

WiFi was selected as the communication method for the system's front-end due to its superior cost-efficiency, simplicity of integration, and power efficiency. When compared to cellular, WiFi modules are cheaper, require no recurring data charges, and avoid the complexity that arises from SIM management or network registration. From a power perspective, WiFi consumes significantly less energy, making it ideal for battery-powered systems where longevity and lightweight design are paramount. In environments where WiFi access points can be installed or are installed, WiFi offers sufficient range, consistent throughput, and minimal latency for both real-time and batch data transmission.

- Colour Palette:

Two different colour palettes are considered for use in the graphical user interface - a pink and blue tone and a green tone.

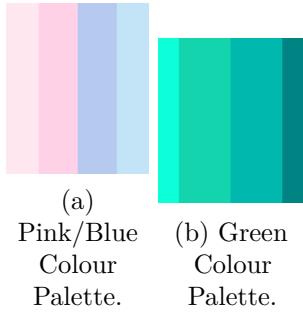


Figure 6.3: Considered colour palettes.

Table 6.12: Pros and Cons of the considered colour palettes.

|     | Blue/Pink   | Green   |
|-----|---|---|
| Pro | -Gentle and modern tone. -Colours are associated with calmness and trust. -Works well with light and dark text. | -Strong contrast.   |
| Con | -Could appear washed out on devices with poor screen quality.   | -Very harsh tone. -Could feel very clinical. -Works best with light text. |

#### Selected Design Decision:

While choice of colour is extremely subjective, the soft pink to soft blue gradient was chosen because it balances emotional appeal, usability, and accessibility, making it ideal for a GUI used in research and animal welfare applications. From a user interface perspective, pastel gradients reduce visual strain while preserving aesthetic clarity, which is essential for dashboards viewed over long periods. Furthermore, the colours were selected based on their association with calmness and trust, vital in systems that communicate critical health or location data without inducing user stress. Technically, the gradient supports responsive rendering, minimal colour banding, and maintains good contrast against typical text and icon overlays. Its lightweight implementation in PyQt ensures efficient rendering with very little of a performance trade-off. Compared to darker, more saturated palettes, this pastel scheme delivers a friendly, non-invasive visual identity that aligns well with conservation and wildlife tracking, where professional clarity is important.

#### 6.3.2 Final Design

The following design was selected based off of the above discussed (subsection 6.3.1) unique design decisions. The system housing was based on Model 2 (6.2b) and the front-end system and GUI was developed using Python in PyCharm with PyQt5 and Matplotlib with a Google Sheets cloud based back-end with a soft pink and blue colour scheme.

##### Submodule 1: System Housing

The prototype developed for the system housing was built for testing on humans and as a result has differing measurements than that required for use on a penguin and is 3D printed with PLA filament rather than PETG.

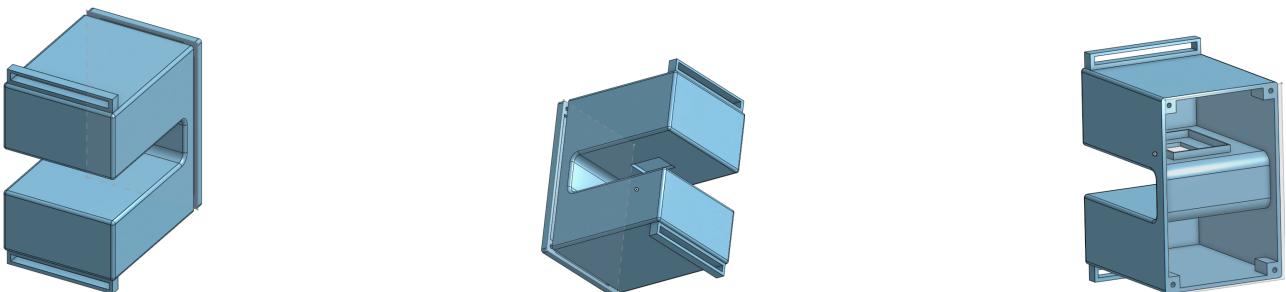


Figure 6.4: Orthographic views of the final design.

### 6.3. Subsystem Design

From Figure 6.4 and Figure 6.5 of the final design, it can be seen that requirements R01, R02 and R03 were taken into consideration and were the primary points around which the design was completed. While the prototype simply aims to provide proof of concept, the concept fully abides by the set out requirements. The design is hydrodynamic, minimally invasive (when manufactured to scale for a penguin) and is easily attachable and detachable. The penguin flipper can fit securely between the upper and lower parts of casing, allowing for optimal readings while the strap brackets ensure a secure fit. Additionally, the removable back allows for simple and easy replacement of components should there be any damage.

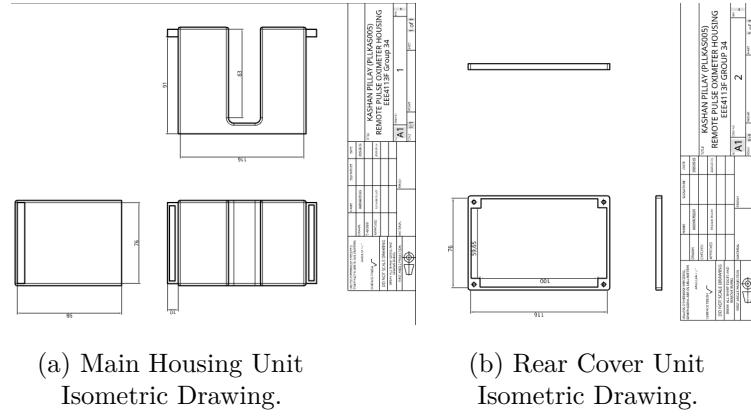


Figure 6.5: Final System Housing Design.

**Submodule 2: Front-end System and Graphical User Interface** As seen below in Figure C.1, Figure C.2 and Figure C.3 the front end GUI follows the guidelines discussed above to ensure an aesthetically pleasing and user friendly experience. There are clear navigation elements on each page and the font is clear and bold.

#### - Home Page:

The home page is simply the navigational centre that links all the different aspects of the front-end system. There are several navigational methods, a drop down menu (as seen on the top left of every page) as well as large buttons (as seen on Figure C.1a), that the user can utilise to navigate to their desired page. Additionally, there are instructions (as seen on Figure C.1b) on how the system works as well as a break down of features that the system offers (as seen on Figure C.1a). Both further the user-centric design of the front-end system.

#### - Data Page:

The data page acts as a point of retrieval for the health metrics (of the penguin) for the user. On this page, the user can view the pulse and oxygen saturation in two summary blocks that also act as warning or notification bars which change to red based off of a threshold violation. Furthermore, the user may choose to view historical data which can be done by viewing the oxygen saturation graph. To plot trends, display and filter the data as well as provide a professional user experience the Matplotlib Library is utilised. As seen in Figure C.2b, the current and historical data is displayed and labelled descriptively. Graph Initialization is done using a GraphCard Class which creates a resizeable, high-DPI plot area inside a Qt widget. PyQt's canvas.draw() is used to sync the plot with the event loop. The filtering is done by using the built in DateFormatter. The calculation of the equation (Y)

used to plot the function  $y = a + b * \text{ror}$  is done before hand and acts as a calibration method.  $y$  is the oxygen saturation,  $\text{ror}$  is the ratio of the reflection of red light to infrared light and  $a$  and  $b$  are linear mapping constants. Using a clinical or tested pulse oximeter, we get known  $y$  values (oxygen saturation) and simultaneously record the outputs of our system. From there, we solve simultaneously to arrive at our mapping function which is then used to plot the oxygen saturation graph as seen in [Figure C.2b](#).

#### - Tracking Page:

The tracking page, as seen in [Figure C.3](#), is where the user may go to view the real time location of the penguin. The system utilizes Google Maps embedded within a PyQt5 QWebEngineView widget to visualize live penguin coordinates retrieved from a Google Sheets back-end, discussed below. The map is dynamically rendered via an HTML iframe, with the source URL formatted to center on the latest latitude and longitude values ( $q=\text{lat},\text{lon}$ ) at a fixed zoom level. Real-time updates are achieved through a dedicated thread that polls the Google Sheets API, refreshing the map only when new coordinates are detected. To ensure usability, a QTimer forcibly resets the map view every 10 seconds, preventing user actions from obscuring the penguin's location. The iframe blocks unauthorized navigation.

#### - Cloud Back-End:

As mentioned above, Google Sheets was selected to act as our cloud back-end. [Figure C.4](#) shows how the data is formatted on the google sheet (populated directly from Subsystem 2). This data is then utilised on the front-end system for graphical display to the user. The system employs Google Sheets API through the gspread library to retrieve both sensor data and GPS coordinates for monitoring. For authentication, OAuth2 service account credentials (stored in apd.json) are used, granting restricted access to specified Google Sheets documents. The `get_sheet_data()` function fetches all records from the 'test\_data' sheet as a list of dictionaries, enabling the Data Page to process and display vital metrics (e.g., heart rate, SpO) by iterating through each row and calculating derived values (e.g., oxygen saturation from raw ratios). Concurrently, the `get_coordinates()` function extracts latitude and longitude from cells F2 and F3, combining them into a string format (lat,lon) for the Tracking Page, which embeds these coordinates into a Google Maps iframe. Both functions implement error handling to log exceptions and return fallback values, ensuring graceful degradation. This approach allows seamless synchronization between cloud-stored data and the front-end view without the complexity of managing a full database, making it an efficient solution for real-time applications where low latency and ease of maintenance are key.

### 6.3.3 Failure Management

[Table 6.13](#) summarises the precautions taken to account for system errors/flaws.

Table 6.13: Failure Management Measures

| Measure        | Description  |
|----------------|--|
| Error Message  | Error messages have been used as a failure management measure across the front-end system. They are used for errors when fetching data from the cloud back-end, they are used when there is no data on the cloud back-end and they are used when filtering the data should the user select a date that has no attached data. Additionally, they are used as a hardware fault notification mechanism. |
| Default Values | These have been used when calculating the ratio for the oxygen saturation plot in order to avoid dividing by zero. Additionally, when the GPS module has disconnected from the rest of Subsystem 2, default coordinates are appended to the cloud back-end.  |

## 6.4 Results

### 6.4.1 Tests

Table 6.14: Subsystem 3 Acceptance Tests

| AT ID | Description  | Testing Procedure  | Pass/Fail Criteria  |
|-------|--|--|---|
| AT01  | (User Acceptance Testing) Utilise model measurements to ensure fitment on a wingspan of 12cm to 17cm.  | Measure the length of the opening where the flipper is expected to sit.  | PASS: There is significant space for the flipper to sit comfortably and securely. FAIL: The flipper does not fit in the designated space.   |
| AT02  | (User Acceptance Testing) Weigh the system and ensure it is less than 175g.  | Place the assembled housing on a digital scale.  | PASS: The total weight being less than 175g. FAIL: The total weight exceeding 175g.   |
| AT03  | (User Acceptance Testing) Ensure that the attachment mechanism is secure, but fast and easy to attach and detach.  | Place a substitute in the designated space and attach and detach the system.   | PASS: The system is able to be attached and detached in less than 5 seconds. FAIL: The system takes longer than 5 seconds to attach and detach.   |
| AT04  | (Business Acceptance Testing) Review the Bill of Materials for this subsystem.   | Looking at the cost of 3D printing and the cost of strap and screws.   | PASS: The cost being less than R250. FAIL: The cost being more than R250.   |
| AT05  | (User Acceptance Testing) Conduct a user test session to complete certain tasks on the GUI.  | Allow a user who is unfamiliar with the front-end system to navigate to the data page and open the Oxygen Saturation graph.  | PASS: The user is able to perform the task in less than 3 actions. FAIL: The user takes more than 3 actions to perform the task.  |
| AT06  | (Unit Testing) Attempt to access data from the cloud back-end with a device connected to various networks.   | Attempt to access the oxygen saturation graphs when connected to various networks (different to that of Subsystem 2)   | PASS: The data is able to load successfully. FAIL: The data is unable to load.  |
| AT07  | (User Acceptance Testing) Load the GUI and navigate to the necessary page to ensure that the required data is presented.   | i. Ensure there is substantial data in the cloud back-end. With the GUI loaded, navigate to the data page and attempt to load the oxygen saturation graph set to 'ALL DATA'.<br>ii. Manually insert desired coordinates in the required cells of the GoogleSheet. With the GUI loaded, navigate to the tracking page to see tracking of GPS coordinates is accurate.<br>iii. Manually alter the sensor values (GoogleSheet) such that the oxygen saturation calculation will produce a level less than 90%. Click the fetch data button. | PASS: Successfully loads all data from all times/dates stored on the GoogleSheet.<br>FAIL: Only loads some or none of the data.<br><br>PASS: The tracking page successfully loads the updated coordinates.<br>FAIL: The tracking page does not update or updates to the incorrect location.<br><br>PASS: The Oxygen Saturation Block illuminates red.<br>FAIL: The Oxygen Saturation block does not change. |
| AT08  | (User Acceptance Testing) Load the GUI and conduct a user feedback survey on colour, clarity and impression while ensuring screen size adjustment and hover effects are working. | Load the GUI and ask the user to rate their impression on a scale from 1 to 10. While the GUI is loaded, minimise and maximise the window size and hover over various buttons.   | PASS: The user rates their impression greater than 8, the program is able to withstand a varying window size and the tested button has a hover shadow.  |
| AT09  | (Business Acceptance Testing) Conduct an inspection of the source code to ensure encapsulation in reusable modules.  | Open the source code, and ensure the use of '__' or '__init__' for encapsulation and ensure logical separation between main GUI components and functions.  | PASS: presence of '__' or '__init__' initialisation for classes, and division of data page, tracking page and home page.<br>FAIL: No encapsulation of classes or logical separation of components.  |

## 6.4.2 Analysis of Testing and Results

### Analysis of AT01

This user acceptance testing procedure was to measure the length of the opening where the flipper is expected to sit. This ATP follows R01 and SP01 and in order to pass this ATP, there needed to be enough space for the system to fit comfortably and securely. While the height of the prototype discussed in [subsection 6.3.2](#) was intended for human use, the allotted space was similar to that of the requirements for a penguin with a flipper spanning the range of 12cm to 17cm. It can be seen from [Figure 6.5](#) that the available space allows for the penguin flipper to comfortable fit while leaving enough leverage for the strap to secure the system. Therefore, this acceptance test was **PASSED**.

### Analysis of AT02

This user acceptance testing procedure was to ensure that the weight of the system was less than 175g. This ATP follows that of R02 and SP02 which require a minimally invasive system housing which requires that the system housing weight less than 5% of the mass of an African penguin. As seen in [Figure C.14](#), the weight of the system housing comes in at 164.7g, thus verifying that this acceptance test was **PASSED**.

### Analysis of AT03

This user acceptance testing procedure was to test that the attachment mechanism is secure, fast and easy to attach and detach. This ATP follows R03 and SP03 which require that the system housing be easily attachable and specify that the housing be attachable and detachable by a temporary harness or leg band. After several user tests with the harness below all being able to attach and detach within 5 seconds, it is evident that this test was **PASSED**. The key reason for the validation of this acceptance test is to ensure that the attachment and detachment process is has minimally invasive as possible and does not require recapture of the penguin. The strap can be seen in [Figure C.5](#).

### Analysis of AT04

This business acceptance testing procedure was to review the BOM. This ATP follows R04 and SP04 which require and specify that the total cost of this subsystem be less than R250. As is evident by the Bill of Materials for Subsystem 3 and the costs outlines below, this test was **PASSED**. The total cost of this subsystem was R157,86 ([Table A.1](#)).

### Analysis of AT05

This user acceptance testing procedure was to allow a user who is unfamiliar with the front-end system to navigate to the data page and open the oxygen saturation graph. This ATP follows R05 and SP05-1/SP05-2 which requires that the GUI be user friendly and have clear navigational elements, error handling and a bold font. After handing the loaded GUI to two people unfamiliar with the system, they were both able to load the oxygen saturation graph in 2 user actions, thus verifying that this acceptance test was **PASSED**.

### Analysis of AT06

This unit testing procedure was to attempt to access data from the cloud back-end when connected to at least two different networks. This ATP follows R06 and SP06 which require that the front-end system work remotely and specify that a wireless connection the cloud back-end be established. As seen in [Figure C.6](#), data retrieval when connected to different networks was successful and so this acceptance test was **PASSED**.

### Analysis of AT07

This user acceptance testing procedure was to navigate to the data page and attempt to plot all the historical data for oxygen saturation as well as to navigate to the tracking page and ensure that it accurately pin points the location. Additionally, this testing procedure required some form of indication or warning illumination to be validated. This ATP follows R07 and SP07-1/SP07-2 which require that all the necessary data be displayed to the user in an appropriate manner. As per the screenshots in [Figure C.7](#), [Figure C.8](#), [Figure C.9](#) and [Figure C.10](#) - it is evident that the front-end system can display both current and historic health and tracking data as well as an illumination in the event of a threshold violation (low oxygen saturation in this case). Therefore, this acceptance test was **PASSED**.

### Analysis of AT08

This user acceptance testing procedure was to get feedback on the impression the front-end GUI makes on a test user, additionally tests of the effect of a varying window size as well as the hover function were to be conducted. This ATP follows R08 and SP08-1/SP08-2 which require that the GUI be aesthetically pleasing. It can be seen from [Figure C.11](#) and [Figure C.13](#) that there is neither distortion for a varied window size and that the hover effect is in play. Additionally, upon showing the GUI to a test user, they indicated that the soft colours were easy on the eyes and allows for good contrast between the plot and the background. Thus, it can be concluded that this acceptance test was **PASSED**.

### Analysis of AT09

This business acceptance testing procedure was to ensure that the source code made sufficient use of encapsulation and took an object orientated approach to the development of the front-end system. This is in keeping with the R09 and SP09-1/SP09-2 which require that the front end system be modular, scalable and easy to maintain. As per the code in the branch Subsystem 3 of the [GitHub Repository](#), proper encapsulation of code and good logical separation between the different pages as well as data retrieval and data plotting/displaying was utilised. Therefore, this acceptance was **PASSED**.

## 6.5 Conclusion and Recommendations

The design and implementation of Subsystem 3, comprising the system housing and front-end GUI, successfully met all specified requirements (R01–R09) and acceptance tests (AT01–AT09). The system housing, modelled hydrodynamically, adhered to size (12–17cm wingspan), weight (175g), and attachment specifications (strap) while remaining cost-effective (R157.86). The PyQt5-based GUI demonstrated robust functionality, including real-time data visualization (oxygen saturation, heart rate), GPS tracking via Google Maps, and threshold alerts, all while maintaining a user-friendly and aesthetically pleasing interface. Modular OOP design ensured scalability, and cloud integration via Google Sheets provided reliable remote connectivity. Some recommendations in order to improve upon this subsystem would definitely be a less invasive and more streamlined housing unit with good weather and waterproofing qualities. Additionally, a more comprehensive user interface with the ability to download historical data or include unique penguin identifiers on the tracking page would provide more geological information and better trends to the stakeholder.

# Chapter 7

## Conclusions

The aim of this project was to remotely monitor African penguins' health during molting, without disturbing them, to support ecological research and conservation. As seen in [chapter 3](#), a literature review on avian health metrics guided the choice to develop a lightweight pulse oximeter to track pulse and blood oxygen saturation throughout the molt. The bulk of the work for this project followed next, in [chapter 4](#), [chapter 5](#) and [chapter 6](#).

In [chapter 4](#), the design of Subsystem 1 can be seen. It focuses on the hardware design for light emission and detection to calculate pulse and blood oxygen saturation. The blood oxygen saturation measurements were successful and the output signal from Subsystem 1 was able to be further processed and displayed for research and analysis by Subsystem 2 and Subsystem 3. However, the subsystem could not get meaningful data to calculate the pulse. Recommendations on how to solve this issue is discussed in [subsection 4.5.1](#).

In [chapter 5](#), Subsystem 2 handles power delivery, location tracking, and signal processing. It supplies power, generates PWM signals, tracks location, and processes the output from Subsystem 1 for transmission to the user interface. The Power Module was able to handle the system requirements, however if the model was to be implemented in the field, a different battery with a higher capacity will have to be selected. The location module was able to provide some information on the co-ordinates, however the performance did not match what was desired in terms of accuracy. A different GPS module that has increased capabilities is recommended. These recommendations are explained in [section 5.5](#). The Processing Module is able to perform the processing on the sampled data and transmit it wirelessly to the Google Sheets via Wi-Fi connection.

In [chapter 6](#), subsystem 3 is outlined. It forms the bridge between the sensing components and user interface, integrating seamlessly with Subsystems 1 and 2. The hydrodynamic housing unit safely encloses and protects the sensitive electronics from Subsystems 1 and 2 while meeting strict size and weight requirements to minimize invasiveness, with its snap mechanism enabling ethical deployment. The GUI retrieves and displays processed physiological data (oxygen saturation and heart rate from the back-end) alongside GPS tracking information through an intuitive dashboard, complete with threshold alerts and historical trend visualization. Utilising Google Sheets allows for the system to maintain cost-effectiveness while ensuring all components interface well. Future enhancements could be focused on a more comprehensive user interface as well as a more streamlined system housing design. More detail on this can be found in [section 6.5](#).

In summation, this project achieved several key objectives by designing a device capable of emitting and detecting light signals related to blood oxygen saturation (Subsystem 1), powering the system, processing signals, and providing rough location tracking (Subsystem 2), and displaying the health data remotely via a user-friendly interface all packaged into a penguin-centric housing unit (Subsystem

3). While the system successfully measured blood oxygen saturation and delivered basic location data, it was not able to reliably detect the pulse with the current design. This approach addresses the core challenges of wildlife monitoring: providing reliable scientific data collection while prioritizing animal welfare through minimal handling, all packaged in a user-friendly system accessible to researchers with varying technical expertise.

## 7.1 Recommendations

Recommendations for Subsystem 1, Subsystem 2 and Subsystem 3 are explained in [subsection 4.5.1](#), [section 5.5](#) and [section 6.5](#), respectively. In addition to these, Subsystem 1 and Subsystem 2 should be implemented on PCBs to utilise the space more efficiently and provide a more lightweight solution.

# Bibliography

- [1] M. Giese, R. Handsworth, and R. Stephenson. Measuring resting heart rates in penguins using an artificial egg. (accessed: 14.03.2025). [Online]. Available: <https://www-jstor-org.ezproxy.uct.ac.za/stable/4514381>
- [2] M. A. Ottinger, T. Maness, J. K. Grace, R. R. Wilson, and P. G. Jodice. Gomann strategic bird monitoring guidelines: Avian health. [Online]. Available: <https://gomann.org/strategic-bird-monitoring-guidelines#:~:text=The%20Strategic%20Bird%20Monitoring%20Guidelines,actions%20and%20underlying%20ecological%20processes>.
- [3] A. McInnes, “Discussion on monitoring the african penguins during molt,” Personal interview, 2025, interview conducted on March 5, 2025.
- [4] M. Lichtenberger. Determination of indirect blood pressure in the companion bird. (accessed: 14.03.2025). [Online]. Available: [https://www.sciencedirect.com/science/article/pii/S1055937X0500023X#:~:text=All%20indirect%20blood%20pressure%20measurement,in%20ducks%20\(Anas%20platyrhynchos\)](https://www.sciencedirect.com/science/article/pii/S1055937X0500023X#:~:text=All%20indirect%20blood%20pressure%20measurement,in%20ducks%20(Anas%20platyrhynchos)).
- [5] R. Schnellbacher, A. da Cunha, E. E. Olson, and J. Mayer. Arterial catheterization, interpretation, and treatment of arterial blood pressures and blood gases in birds. (accessed: 14.03.2025). [Online]. Available: <https://www-sciencedirect-com.ezproxy.uct.ac.za/science/article/pii/S1557506314000494>
- [6] A. Zehnder, M. Hawkins, and P. Pascoe. Evaluation of indirect blood pressure monitoring in awake and anesthetized red-tailed hawks (*buteo jamaicensis*). (accessed: 14.03.2025). [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/19709051/>
- [7] M. J. Acierno, A. da Cunha, J. Smith, T. T. Jr, D. S.-M. Guzman, and V. Serra. Agreement between direct and indirect blood pressure measurements obtained from anesthetized hispaniolan amazon parrots. (accessed: 14.03.2025). [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/19014291/>
- [8] C. L. Rzucidlo, E. Curry, and M. R. Shero. Non-invasive measurements of respiration and heart rate across wildlife species using eulerian video magnification of infrared thermal imagery. (accessed: 14.03.2025). [Online]. Available: <https://doi.org/10.1186/s12915-023-01555-9>
- [9] F. Bonadonna, S. P. Caro1, S. Belle1, and A. G. Torrente. Blue petrel electrocardiograms measured through a dummy egg reveal a slow heart rate during egg incubation. (accessed: 21.03.2025). [Online]. Available: <https://go-gale-com.ezproxy.uct.ac.za/ps/i.do?p=AONE&u=unict&id=GALE%7CA802562229&v=2.1&it=r>
- [10] V. A. Viblanc, P. Bize, F. ois Criscuolo, M. L. Vaillant, C. Saraux, S. Pardonnet, B. Gineste, M. Kauffmann, O. sime Prud'homme, Y. Handrich, S. Massemartin, R. Groscolas, and J.-P. Robin. Body girth as an alternative to body mass for establishing condition indexes in field studies: A validation in the king penguin. (accessed: 21.03.2025). [Online]. Available: <https://www-jstor-org.ezproxy.uct.ac.za/stable/4514381>

- //www.journals.uchicago.edu/doi/full/10.1086/667540?casa\_token=3GgrVQqmLxAAAAAA%3AlhTkli8EJjD9FyIvDFauu4eWOCp0A60YZGyVTMrOmKbgqSAzzUzLfNnrVJSXV5w1lGwBFNHJJdNa
- [11] Anonymous. Automated penguin monitoring systems. (accessed: 21.03.2025). [Online]. Available: <https://www.birdlife.org.za/african-penguin-conservation/>
  - [12] K. KERRY, J. CLARKE, and G. ELSE. The use of an automated weighing and recording system for the study of the biology of adelie penguins. (accessed: 21.03.2025). [Online]. Available: <https://nopr.repo.nii.ac.jp/records/5186>
  - [13] V. Afanasyev, S. V. Buldyrev, M. J. Dunn, J. Robst, M. Preston, S. F. Bremner, D. R. Briggs, R. Brown, S. Adlard, and H. J. Peat1. Increasing accuracy: A new design and algorithm for automatically measuring weights, travel direction and radio frequency identification (rfid) of penguins. (accessed: 21.03.2025). [Online]. Available: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0126292>
  - [14] H. Heidari, E. Bonizzoni, U. Gatti, and F. Maloberti. Acmoscurrent-mode magnetic hall sensor with integrated front-end. (accessed: 21.03.2025). [Online]. Available: [https://ieeexplore.ieee.org/abstract/document/7097117?casa\\_token=UY50SuNJQBEAAAAAsan\\_S1o1npxrL1wofbWjcrohju0kSbUZb2V-V1z7gyXLPklOOcX-ESy-aUUsSfcyKEkx9pJv4dc](https://ieeexplore.ieee.org/abstract/document/7097117?casa_token=UY50SuNJQBEAAAAAsan_S1o1npxrL1wofbWjcrohju0kSbUZb2V-V1z7gyXLPklOOcX-ESy-aUUsSfcyKEkx9pJv4dc)
  - [15] S. Welman, J. A. Green, P. G. Ryan, N. J. Parsons, and L. Pichegru. Body temperature and thermoregulatory behaviour in the endangered african penguin spheniscus demersus. [Online]. Available: <https://www.cambridge.org/core/journals/bird-conservation-international/article/body-temperature-and-thermoregulatory-behaviour-in-the-endangered-african-penguin-spheniscus-demersus/F53EDBF0A828E31AC064CB9F13BF2B10>
  - [16] L. Gauchet, A. Jaeger, and D. Grémillet. Using facial infrared thermography to infer avian body temperatures in the wild. [Online]. Available: <https://link.springer.com/article/10.1007/s00227-022-04041-y>
  - [17] J. K. R. Tabh, G. Burness, O. H. Wearing, G. J. Tattersall, and G. F. Mastromonaco. Infrared thermography as a technique to measure physiological stress in birds: Body region and image angle matter. [Online]. Available: <https://physoc.onlinelibrary.wiley.com/doi/full/10.14814/phy2.14865>
  - [18] M. S. Hill, G. B. Zytlewski, J. D. Zytlewski, and J. M. Gasvoda. Development and evaluation of portable pit tag detection units: Pitpacks. [Online]. Available: [https://www.sciencedirect.com/science/article/pii/S0165783605002195?casa\\_token=bKZ8ldbW7bwAAAAAeOKd61kzC0JfPyGOsK35wrRSXQ3aeDKMNhXYPjA31ABdUNC0p6W83VcOoqmWg6HP9YFK\\_pX\\_uTA](https://www.sciencedirect.com/science/article/pii/S0165783605002195?casa_token=bKZ8ldbW7bwAAAAAeOKd61kzC0JfPyGOsK35wrRSXQ3aeDKMNhXYPjA31ABdUNC0p6W83VcOoqmWg6HP9YFK_pX_uTA)
  - [19] S. E. Orosz and M. LichtenBerger. Avian respiratory distress: Etiology, diagnosis, and treatment. (accessed: 21.03.2025). [Online]. Available: [https://www.vetexotic.theclinics.com/article/S1094-9194\(11\)00004-1/abstract](https://www.vetexotic.theclinics.com/article/S1094-9194(11)00004-1/abstract)
  - [20] E. K. MOREY. Development of a remote pulse oximeter. (accessed: 21.03.2025). [Online]. Available: <https://repository.arizona.edu/handle/10150/156896>

- [21] P. M. Schmitt, T. Göbel, and E. Trautvetter. Evaluation of pulse oximetry as a monitoring method in avian anesthesia. (accessed: 21.03.2025). [Online]. Available: <https://repository.arizona.edu/handle/10150/156896>
- [22] R. Wilson, A. Steinfurth, Y. Ropert-Coudert, A. Kato, and M. Kurita. Lip-reading in remote subjects: an attempt to quantify and separate ingestion, breathing and vocalisation in free-living animals using penguins as a model. (accessed: 21.03.2025). [Online]. Available: <https://link-springer-com.ezproxy.uct.ac.za/article/10.1007/s002270100659>
- [23] L. M. Engineers. Esp32 pinout reference. [Online]. Available: <https://lastminuteengineers.com/esp32-pinout-reference/>
- [24] M. Hart. Tinygpsplus. [Online]. Available: <https://github.com/mikalhart/TinyGPSPlus>
- [25] E. Systems. Led control (ledc). [Online]. Available: <https://docs.espressif.com/projects/arduino-esp32/en/latest/api/ledc.html>
- [26] Martinius. Esp32-eduroam. [Online]. Available: <https://github.com/martinius96/ESP32-eduroam/blob/master/2022/eduroam/eduroam.ino>

# Appendix A

## Bill of Materials

Table A.1: Bill of Materials

| Item                        | Quantity | Unit Cost (ZAR) | Total Cost (ZAR) |
|-----------------------------|----------|-----------------|------------------|
| <b>Subsystem 1</b>          |          |                 |                  |
| L-53SURCK                   | 1        | 2.19            | 2.19             |
| L-53F3C                     | 1        | 3.62            | 3.62             |
| 2N3904                      | 2        | 0.62            | 1.24             |
| BPW34                       | 1        | 13.66           | 13.66            |
| MC33078                     | 2        | 4.77            | 9.54             |
| Resistors                   | 11       | 5.66            | 62.26            |
| Male to Male Connectors     | 2        | 0.75            | 1.50             |
| Vero-boards 100mm × 200mm   | 2        | 32.09           | 64.18            |
| Total                       |          |                 | 158.19           |
| <b>Subsystem 2</b>          |          |                 |                  |
| 3.7V 800mAh LiPo Battery    | 1        | 58              | 58               |
| ESP32 Devkit v1 Board       | 1        | 117             | 117              |
| GPS Module NEO-6M + Antenna | 1        | 100             | 100              |
| USB to Micro-USB Cable      | 1        | 49              | 49               |
| Male to Female Connectors   | 11       | 0.75            | 8.29             |
| Total                       |          |                 | 358.69           |
| <b>Subsystem 3</b>          |          |                 |                  |
| PLA 3D Printing Filament    | 211,54g  | R299,99 per kg  | R63,45           |
| Strap/Harness               | 1        | R79,95          | R79,95           |
| Machine Screws              | 4        | R0,71           | R2,84            |
| Hook-snap                   | 1        | R10,99          | R10,99           |
| Brass Eyelets               | 3        | R0,21           | R0,63            |
| Total                       |          |                 | R157,86          |
| System Total                |          |                 | 674.74           |

# Appendix B

## GA Requirements

Table B.1: Graduate Attribute Mapping - Student: Liyana Singh (SNGLIY001)

| Graduate Attribute                                      | Where Met  |
|---|--|
| GA 3: Engineering Design                                | In <a href="#">section 4.3</a> (page 15 - 20) the full design of my subsystem can be seen. This included comparing multiple circuit configurations, choosing appropriate components and the technical calculations.                                  |
| GA 7: Sustainability and Impact of Engineering Activity | <a href="#">chapter 1</a> Introduction (page 1) to the topic of wildlife conservation and our role in that and <a href="#">chapter 2</a> D-School (page 2 - 3) which highlights the issue and guides us on how we can contribute to positive change. |
| GA 8: Individual, Team and Multi-disciplinary Working   | See Teams group for meeting schedules, meeting minutes, file sharing and general communication.  |
| GA 10: Engineering Professionalism                      | All submission activities met including final report and presentation.   |

Table B.2: Graduate Attribute Mapping - **Student: Priya Moodley (MDLPRI040)**

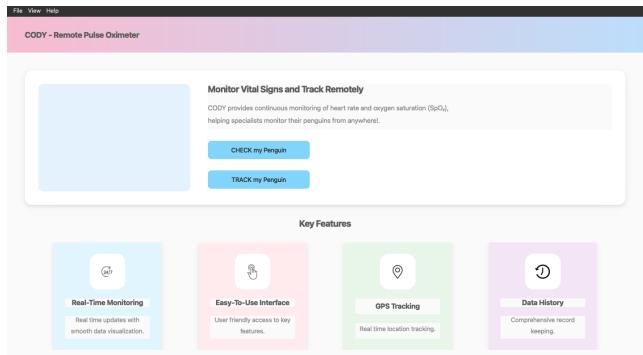
| <b>Graduate Attribute</b>                               | <b>Where Met</b>   |
|---|--|
| GA 3: Engineering Design                                | In <a href="#">section 5.3</a> , the design steps of the entire Subsystem are outlined. The component selection process (pages 28-30), where various options are compared and evaluated, are detailed for each of the three Modules within Subsystem 2. Additionally, in pages 30 to 32, the C++ code written in Arduino IDE is fully explained. |
| GA 7: Sustainability and Impact of Engineering Activity | Sustainability is addressed in the Problem Analysis section <a href="#">chapter 2</a> , D-school and the Introduction. Subsystem 2 incorporated power saving when sampling and Wireless transmission to allow for minimal attachments to the penguin itself.   |
| GA 8: Individual, Team and Multi-disciplinary Working   | See Teams group for meeting schedules, meeting minutes, file sharing and general communication.  |
| GA 10: Engineering Professionalism                      | All submission activities met including final report and presentation.   |

Table B.3: Graduate Attribute Mapping - **Student: Kashan Pillay (PLLKAS005)**

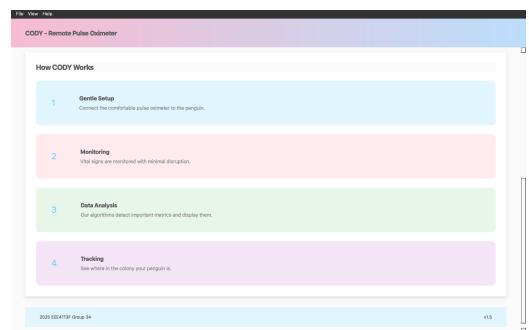
| <b>Graduate Attribute</b>                               | <b>Where Met</b>   |
|---|--|
| GA 3: Engineering Design                                | Subsystem Design - <a href="#">section 6.3</a> , pg 39-45: Engineering Design attributes can be seen in the mechanical design of the housing which was subsequently 3D printed. Additionally, further evidence of engineering design can be seen in the development of the front-end GUI development and the use of encapsulation, PyQt5, Matplotlib, Object Orientated Programming and many other complex components - as seen in the GitHub Repository.  |
| GA 7: Sustainability and Impact of Engineering Activity | Problem Analysis - <a href="#">chapter 2</a> , D-School, pg 2-3 and Introduction - <a href="#">chapter 1</a> , pg 1: These sections raise the topic of wildlife conservation and our role in its progression. Additionally, the concepts of positive change and knowing the impact of our engineering decisions can be seen in the D-School sessions (as well as previous social studies courses). Requirements Analysis - <a href="#">section 6.2</a> , pg 38-39: Here, evidence of the idea of knowing the impact of engineering activity can be further emphasized in that the designing of the system housing focused on minimal invasiveness (penguin-centric), low impact as well as low cost with the front-end system was designed to be user-centric. |
| GA 8: Individual, Team and Multi-disciplinary Working   | See Teams group for meeting schedules, meeting minutes, file sharing and general communication.  |
| GA 10: Engineering Professionalism                      | All submission activities met including final report and presentation.   |

# Appendix C

## Subsystem 3: Appendices

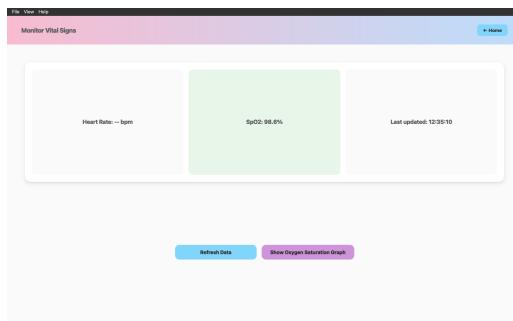


(a) The top half of the home page GUI.

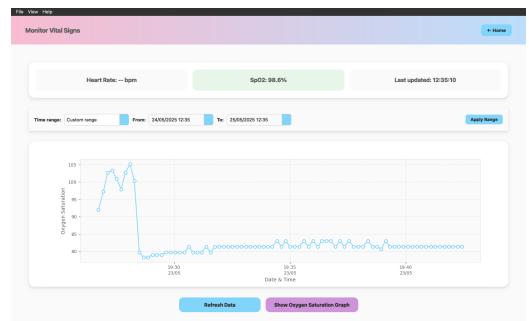


(b) The bottom half of the home page GUI.

Figure C.1: The Home Page



(a) The data page prior to the loading of any data.



(b) The data page displaying the data to the user.

Figure C.2: The Data Page

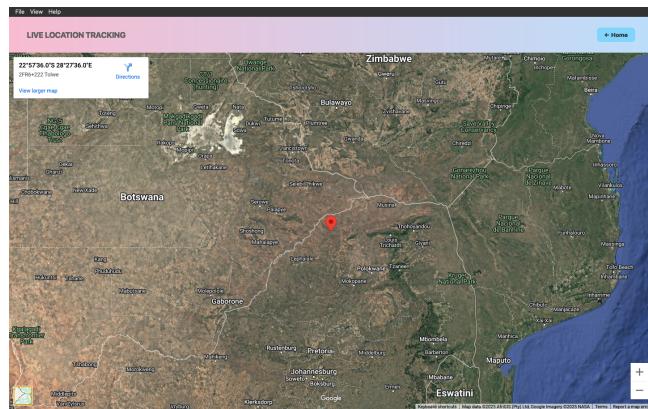


Figure C.3: The Tracking Page

|    | A          | B        | C       | D       | E        | F      | G     |
|----|------------|----------|---------|---------|----------|--------|-------|
| 1  | Date       | Time     | Ratio 1 | Ratio 2 | DC Value | GPS    | Pulse |
| 2  |            |          |         |         |          | -22.96 | 120   |
| 3  |            |          |         |         |          | 28.46  |       |
| 4  | 23/05/2025 | 19:26:44 | 1.54    | 2       | 0.28     |        |       |
| 5  | 23/05/2025 | 19:26:57 | 1.54    | 1.97    | 0.28     |        |       |
| 6  | 23/05/2025 | 19:27:08 | 1.57    | 1.98    | 0.28     |        |       |
| 7  | 23/05/2025 | 19:27:20 | 1.58    | 1.99    | 0.28     |        |       |
| 8  | 23/05/2025 | 19:27:31 | 1.58    | 2       | 0.29     |        |       |
| 9  | 23/05/2025 | 19:27:43 | 1.56    | 1.99    | 0.29     |        |       |
| 10 | 23/05/2025 | 19:27:54 | 1.58    | 1.99    | 0.29     |        |       |
| 11 | 23/05/2025 | 19:28:06 | 1.59    | 1.99    | 0.29     |        |       |
| 12 | 23/05/2025 | 19:28:17 | 1.56    | 1.98    | 0.28     |        |       |
| 13 | 23/05/2025 | 19:28:30 | 1.52    | 2.04    | 0.29     |        |       |
| 14 | 23/05/2025 | 19:28:41 | 1.5     | 2.02    | 0.29     |        |       |
| 15 | 23/05/2025 | 19:28:52 | 1.5     | 2.02    | 0.29     |        |       |

Figure C.4: The cloud back end with random test data in it.



Figure C.5: The strap/harness utilised in section 6.4.2

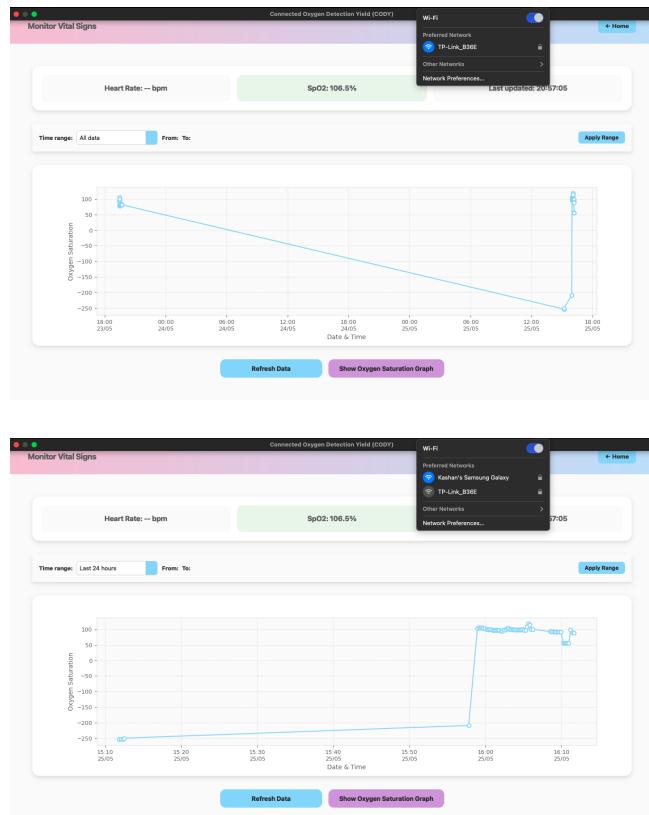


Figure C.6: The retrieval of data on different networks as in section 6.4.2



Figure C.7: The display of all historical data as required in section 6.4.2.

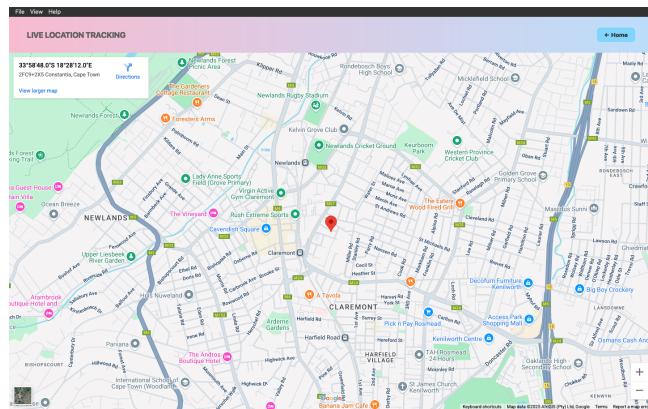


Figure C.8: The accurate tracking of the gps coordinates (Figure C.9) as required in section 6.4.2.



Figure C.9: The GPS coordinates required as per section 6.4.2.

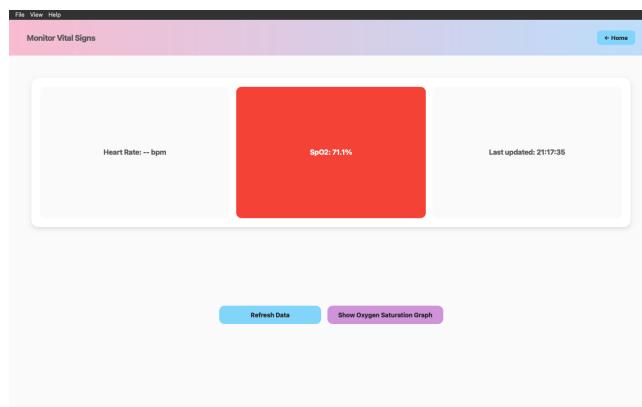


Figure C.10: The illuminated warning panel as required in section 6.4.2.

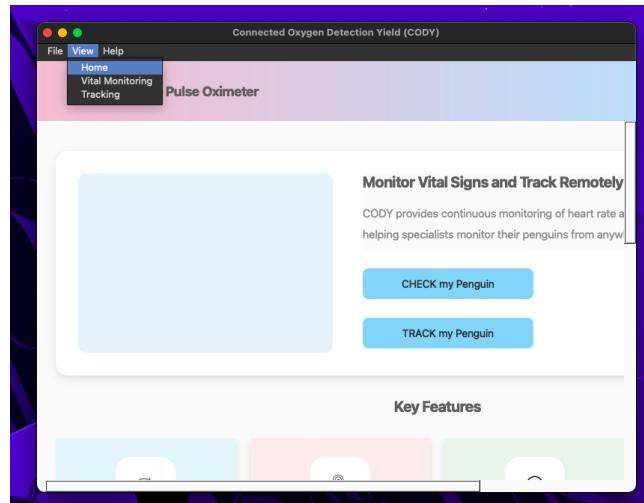


Figure C.11: The minimized window without any distortion as required in section 6.4.2.

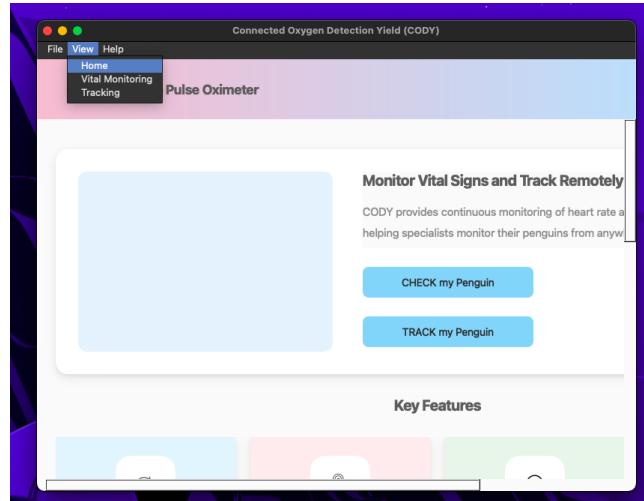


Figure C.12: The hover feature as required in section 6.4.2.

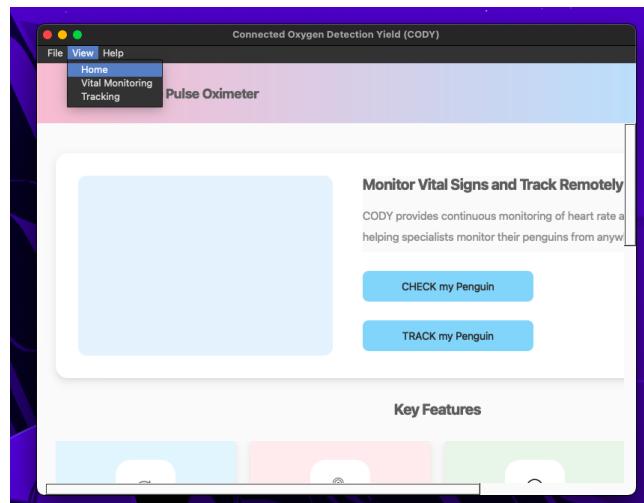


Figure C.13: The hover feature as required in section 6.4.2.



Figure C.14: The weight of the system housing as per [section 6.4.2](#).