Group 11

Shamila Thennakoon, Lihini Hewage, Upeksha Liyanage

Pulse Wave

Project Report

First-year Hardware Project

School of ICT

Metropolia University of Applied Sciences

12 May 2024

Abstract

The aim of the Pulse Wave project was to design and develop an innovative device for recording and analyzing heart rate using photoplethysmography (PPG). The device provides the user with accurate real-time monitoring of heart rate and calculates its variability. This device can be developed as an accessible tool for cardiovascular health management in today’s hectic lifestyle.

For the project, a Raspberry Pi Pico W microcontroller board was used along with components, the OLED display and the CrowTail Pulse Sensor v2.0 to develop a user-friendly device that can analyze heart rate in real time. Other components of the protoboard were also used, such as a rotary encoder, an OLED display and LEDs. MicroPython was chosen as the programming language and Thonny was the project's IDE. Heart rate variability analysis algorithms were also implemented, which were processed locally and retrieved from the Kubios cloud service based on the peak-to-peak intervals captured during the measurements.

The project faced challenges due to the complexity of the coding. Despite the issues with the encoder and thresholds, the team was able to overcome these through effective teamwork and commitment. However, limiting the accuracy and scaling the system to improve accuracy and applicability can be addressed as part of future investments.

Version history

Ver	Description	Date	Author(s)
1.0	Created structure for the project report. Added instructions for what should be included in the different parts of the document.	13.3.2023	Saana Vallius
1.1	Added Instructions (REMOVE WHEN READY), minor editions.	14.3.2023	Sakari Lukkarinen
1.2	Highlighted all instructions with red font.	19.3.2023	Sakari Lukkarinen
1.3	Updated Appendix 2 based on comments. Updated table of contents.	22.3.2023	Saana Vallius
2.0	Added a section for group work summary. Changed the highlighted instructions to an easier color to handle.	22.10.2023	Saana Vallius
3.0	Introduction	07.04.2024	Lihini Hewage
3.1	Theoretical background	10.04.2024	Shamila Tennakoon
3.2	Methods and Material	10.04.2024	Upeksha Liyanage
3.3	Midway Summary	11.04.2024	Lihini, Shamila, Upeksha.
3.4	Implementation	11.05.2024	Upeksha Liyanage
3.5	Final Summary	10.05.2024	Lihini, Shamila, Upeksha
3.6	Abstract	11 .05.2024	Shamila Tennakoon
3.7	Conclusion	11.05.2024	Lihini Hewage

Contents

1 Introduction

Pulse Wave is a heart rate detection and analysis device implemented during Hardware 2 at Metropolia University of Applied Sciences. This device analyzes the heart rate and its variability (HRV) by using photoplethysmography (PPG). This heart rate detection device combines advanced technology with user-friendly design to bring the accurate and convenient heart rate monitoring capabilities.

The aim of the device is to calculate heart rate variability (HRV) analysis with heart rate detection using a Raspberry Pi Pico microcontroller board and additional components. The product encourages users to monitor their heart rate and stress level more accurately and conveniently for their health and wellbeing.

The motivation behind this project is the increase in the demand for accessible tools that empower individuals to track and manage their cardiovascular health effectively due to prevalence of sedentary lifestyles and stress-related health issues. By developing the Pulse Wave device, it will contribute to this evolving landscape by providing users with a user-friendly and accurate means of monitoring their heart rate and stress levels.

This documentation describes the purpose and the structure of the project, details the necessary technical features for successful implementation, and provides an insight to the challenges and the outcome of the project.

2 Theoretical Background

The heart rate measures the number of contractions or beats of the heart per minute [1]. It is measured in beats per minute (bpm). It is one of the vital signs that indicate the state of health of the human body. According to the American Heart Association, the normal heart rate at rest should be between 60 and 100 beats per minute [2]. Heart rate can be defined as follows. The normal heart rate varies from person to person. Age, activity level, physical fitness, medication and other factors can affect a person’s resting heart rate [1]. According to Guinness World Records, the lowest heart rate is 27 beats per minute [3], and the predicted maximum heart rate is 220 beats per minute at maximum exertion. A person’s maximum heart rate can be calculated by subtracting the person’s age from 220 [4]. Based on these values, the range of heart rate measurement for this project is set at 20-220 beats per minute.

Heart rate variability is a fluctuation between the timing of heartbeats [5], measured in milliseconds (ms) as the variation in time intervals between successive heartbeats, and serves as an indicator of psychological factors such as recovery and stress levels [6]. Heart rate variability (HRV) shows a significant correlation with average heart rate (HR) and thus provides insight into both HR and its fluctuations. Determining which factor - heart rate or its variability - is more relevant to the clinical value of HRV remains a challenge. Essentially, the question is to what extent HR influences the clinical significance of HRV. Furthermore, the link between HRV and HR involves both physiological and mathematical aspects. Physiologically, the dependence of HRV on heart rate is due to the activity of the autonomic nervous system, while mathematically it's influenced by the nonlinear relationship between RR intervals and heart rate [7].

The heart rate can depend on several factors. The most important factor and the main focus of this project are emotions. Emotions such as stress, fear and anger can lead to an increase in heart rate. [8]. Heart rate variability is also influenced by a person’s autonomic nervous system, which is responsible for regulating many involuntary bodily functions, including heart rate. A person's better health and fitness can ensure a higher HRV. A lower HRV can be a sign of stress, anxiety or other health problems. Age, fitness level, stress and illness are the main factors that influence lower HRV [9]. The typical range of resting heart rate in adults is usually between 60 and 100 beats per minute (bpm). The lowest value is seen in some well-trained athletes, whose resting heart rate can be as low as 40 bpm. The maximum heart rate can exceed 200 beats per minute during intensive training. The range of variability varies depending on factors such as age, fitness level and general health [10].

The heart rate can be determined using electrical signals in various ways. Electrocardiography (ECG) is considered the gold standard and is widely used in both clinical and non-clinical settings [2]. However, a different optical detection technique, photoplethysmography (PPG), was chosen for heart rate monitoring in this project. Photoplethysmography (PPG) measures changes in microvascular tissue indirectly via light-emitting diodes (LEDs). The photodiodes then record the amount and direction of light reflected by the arteries.Figure 1 shows how to take a PPG signal.

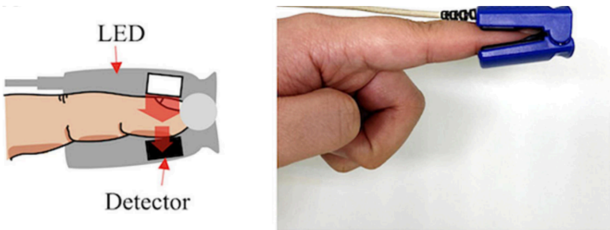

Figure 1: Configuration for photoplethysmography measurement: transmissive type [11].

With PPG signals, the intervals between the beats (IBI) or the intervals between the peak values (PPI) are measured. The following formula can be used to calculate the heart rate (HR).

$$HR = \frac{60}{IBI} \times 1000$$

The main difference between the PPG signal and the ECG signal is shown in Figure 2. The difference results from the location where the pulse is detected. In the PPG signal, the IBI is detected between the red dots, while the ECG measures the R-to-R distance between the red dots [12]

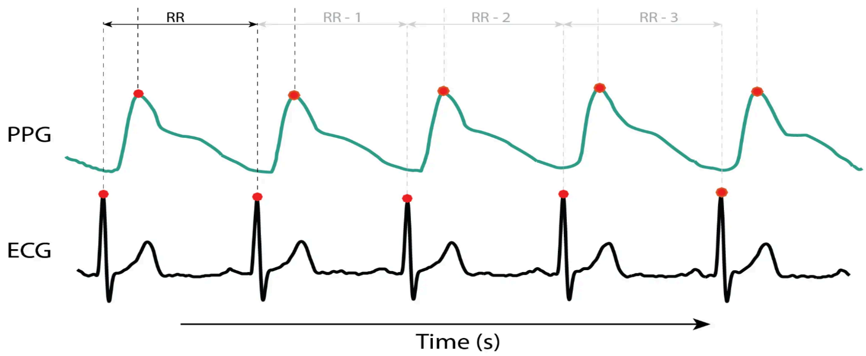

Figure2: Beat-to-beat analysis from R-R intervals (RRI) and peak-to-peak intervals (PPI) [12].

The Crowtail Pulse Sensor v2.0 is a special device for measuring the human heart rate. Its hardware components include an optical heart rate sensor with a light-emitting diode (LED) and a photodiode, an analog amplifier, a noise suppression circuit and an analog signal output. The sensor side of the hardware, shown in Figure 3, shows the placement of the LED in the center and the photodiode directly below it. For optimal measurement results, it is recommended to position the sensor and photodiode primarily on the earlobe or finger so that the sensor side faces the skin. However, measurements can also be taken on other parts of the body such as the wrist, palm, arm, chest, cheek and forehead.

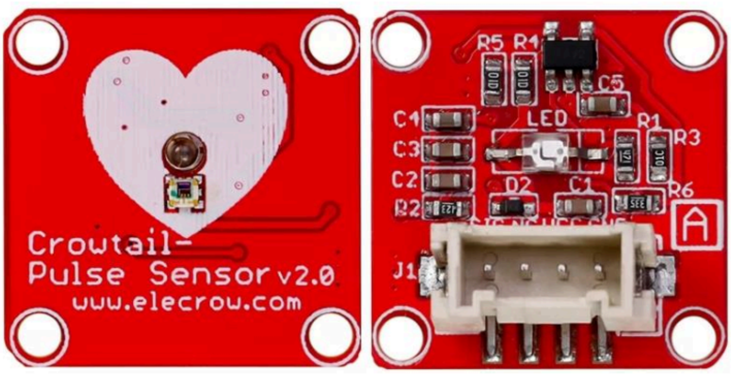

Figure 3: Crowtail Pulse Sensor v2.0 from the sensor (left) and component (right) side [13]

3 Methods and Material

In this section, the methodology and materials utilized in the project will be outlined.

Hardware used in the project

Raspberry Pi Pico W

For the project, the Raspberry Pi Pico was chosen as the microcontroller board, depicted in Figure 4. This choice was made because it supports a variety of peripherals and features a 2.4 GHz wireless LAN connection. Additionally, it includes onboard LEDs and a BOOTSEL button for programming purposes. [14.]

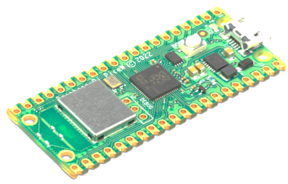

Figure 4: Raspberry Pi Pico W model [15]

SSD1306 OLED display

In our setup, we use a SSD1306 OLED display to show output information to the user. This display, as seen in Figure 5, is a simple 128x64 pixel monochrome digital screen that communicates via I2C. [16.]

Figure 5: SSD1306 OLED display [16]

Rotary encoder

Figure 6 shows a generic rotary encoder. It includes a metal encoder with three functions: Rot Switch, ROT A, and ROT B.

Figure 6: Rotary encoder [17]

The Rot Switch works as a standard push button, while ROT A and ROT B allow clockwise and anticlockwise rotation.

Crowtail Pulse Sensor

Figure 7 shows the Crowtail Pulse Sensor v2.0 to measure human heart rate. This sensor is specifically designed for this purpose.

Figure 7: Crowtail Pulse Sensor [18]

It includes components like an optical heart rate sensor with LED and photodiode, an analog amplifier, noise cancellation circuitry, and provides an analog signal output.[18]

3.1 Software used in this project

To program the Raspberry Pi Pico W, we use Thonny IDE with MicroPython. **Thonny IDE** is used as the integrated development environment (IDE) for writing and running MicroPython code on the Raspberry Pi Pico. **MicroPython** is a streamlined and efficient implementation of Python designed specifically to operate on microcontrollers and embedded systems like the Raspberry Pi Pico. **Kubios HRV** software is a highly detailed tool for analyzing HRV, trusted and widely used by researchers and professionals globally. [19].

3.2 System Testing and Verification.

To conduct testing and verification of the system's accuracy in measuring heart rate using the Raspberry Pi Pico and Crowtail sensor, a comparative analysis was performed against data collected from a pulse oximeter. The goal was to ensure that the system's readings fell within an acceptable range and exhibited accurate heart rate measurements. The results confirmed the system's performance and provided assurance of its suitability for real-world applications requiring reliable heart rate monitoring. Ongoing testing and refinement may further enhance the system's accuracy and robustness.

4 Implementation

This chapter explains the details of implementation of the heart rate detection device. Here, the primary focus is on the completed system, algorithms, and the processing and handling of data.

4.1 System

Figure 8 shows the implemented system utilizes a Raspberry Pi Pico microcontroller to capture data from a Crowtail sensor, which collects user-specific information like heart rate and other physiological metrics.

The system offers various operational modes based on user selection, including:

Heart Rate Display: Real-time presentation of the user's heart rate.

HRV Analysis: Examination of Heart Rate Variability for insights into autonomic nervous system activity.

Kubios Analysis: Advanced analysis of heart rate data using Kubios algorithms.

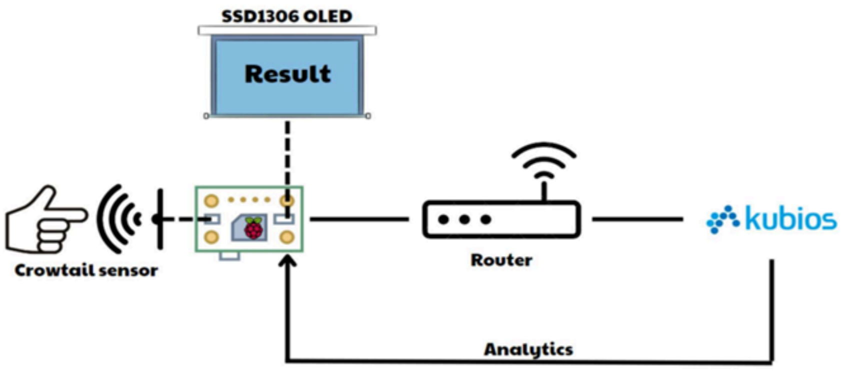

Figure 8: The implemented system

Upon selection of a mode, the Raspberry Pi Pico processes the collected data accordingly. This processing involves specific computations and algorithms tailored to each mode's requirements.

The results of this processing are conveyed to two primary display mediums:

OLED Display: Immediate visualization of processed data on the OLED screen for user interaction and feedback.

Client Computer via MQTT: The processed and analyzed data is transmitted over MQTT (Message Queuing Telemetry Transport) to a client computer. This allows remote monitoring and further analysis on the client's end.

4.2 Algorithms for calculating heart rate and heart rate variability

The main program running on the Raspberry Pi Pico initiates automatic connection to the designated Wi-Fi network upon startup. It then activates the sensor module to begin reading Analog-to-Digital Converter (ADC) values at a fixed sampling rate of 20Hz, corresponding to 20 samples per second. Each sampled ADC value is immediately compared against a predefined threshold. If the value is below the threshold, indicating background noise or non-heartbeat signals, the sensor continues reading without logging the sample. Conversely, if the sampled value exceeds the threshold, it signifies a potential heartbeat signal and is recorded in a sample list for further processing.

The recorded samples, representing intervals between detected heartbeats, are utilized for subsequent analysis. The primary analysis involves calculating peak-to-peak intervals (PPI) from the recorded heartbeat signals. Peak-to-peak intervals are determined by measuring the time difference between consecutive detected heartbeat peaks in the sample list. In addition to offline HRV analysis, the collected PPI data undergoes advanced analysis using Kubios software algorithms.

Threshold level analysis:

To establish an adaptive threshold for filtering ADC values, a scaling formula in List 1 is utilized. This formula dynamically adjusts the threshold based on the current sensor readings, optimizing sensitivity to heartbeat signals while minimizing noise interference. As ADC values are sequentially collected, they are added to a rolling list containing up to 750 samples.

Once the sample list reaches its maximum capacity, the system extracts the maximum and minimum values from this dataset. These extremum values are then organized into a processed value list, which serves as the foundational dataset for subsequent calculations in the analysis pipeline.

```
#calculate threshold
High_thrsld = (minimum_val + maximum_val * 3) // 4
Low_thrsld = (minimum_val + maximum_val) // 2
Range_of_values = maximum_val - minimum_val
```

List 1 . Threshold levels calculations code snippet.

Calculation of Beats Per Minute (BPM)

After obtaining the processed PPI values from the recorded heartbeat signals, the BPM is calculated 60,000 divided by the PPI value. List 2 illustrates how to calculate mean PPI and BPM. This calculation converts the time interval between consecutive heartbeats into beats per minute, providing a measure of heart rate. In a typical human adult, the normal BPM range spans from 30 to 240 beats per minute. Specifically, the BPM values are confined within a narrower range of 40 to 240 beats per minute. This restriction helps to mitigate erroneous readings that may arise from noise or artifacts in the sensor data, ensuring that the reported BPM values remain clinically meaningful and within physiologically plausible limits for heart rate.

```
#calculate mean ppi then calculate bpm
if 250 < ppi < 1500: #limit range of bpm from 40-240 bpm
    sumppi += ppi
    pulseCount += 1
    #calculate mean ppi after pulseCount
    if pulseCount == 1: #modify this value to get better result
        pulseCount = 0
        avrppi = sumppi #modify this value to get better result
        bpm = 60000 // avrppi
        intervals.append(avrppi)
        intervalsCount += 1
        sumppi = 0
```

List 2 . Mean PPI and BPM calculations code snippet.

HRV Analysis & MQTT

Once the program has accumulated a sufficient number of peak-to-peak interval (PPI) samples, the Heart Rate Variability (HRV) analysis is initiated. The collected PPI list undergoes local computation using various mathematical calculations and algorithms tailored to HRV assessment.

The HRV analysis yields several key metrics:

Mean PPI: This metric represents the average peak-to-peak interval, providing insights into overall heart rate variability.

Average Heart Rate: Calculated as the reciprocal of the mean PPI (60,000 divided by mean PPI), this metric indicates the average beats per minute (BPM) over the analyzed period.

SDNN (Standard Deviation of NN intervals): This measure quantifies the overall variability in heart rate, reflecting both sympathetic and parasympathetic influences on cardiac rhythm.

RMSSD (Root Mean Square of Successive Differences): This metric assesses short-term variations in heart rate, reflecting parasympathetic nervous system activity.

The results of the HRV analysis are displayed on the OLED screen in real-time, presenting the calculated mean PPI, average heart rate, SDNN, and RMSSD for immediate user feedback and monitoring. Concurrently, the Raspberry Pi Pico transmits the same analysis results to the client's computer via MQTT (Message Queuing Telemetry Transport) for remote monitoring and visualization.By leveraging MQTT communication, the system enables seamless integration with external devices or applications, allowing users to access and analyze HRV metrics remotely.

4.3 Data collection and processing

When the user places their fingertip on the Crowtail sensor, the data collection for heart rate measurement begins. The Raspberry Pi Pico employs an algorithm designed to detect and gather relevant heart rate data while filtering out irrelevant or noisy signals. Only the necessary data that meets specific requirements is sampled and processed for further analysis. Depending on the user's selection through the OLED user interface, the Raspberry Pi Pico initiates different modes of data collection and processing:

Heart Rate Display Mode: If the user selects for displaying only heart rate, the Raspberry Pi Pico gathers sufficient data and proceeds with basic heart rate variability (HRV) analysis. The resulting heart rate is then displayed on the OLED screen for immediate feedback.

HRV Analysis Mode: In this mode, the Raspberry Pi Pico collects a larger dataset to conduct comprehensive HRV analysis. This includes calculating metrics such as mean heart rate (HR), standard deviation of peak-to-peak intervals (SDNN), and root mean square of successive differences (RMSSD). The calculated HRV metrics are displayed on the OLED screen.

Kubios Analysis Mode: If the user selects Kubios Analysis mode, the collected samples are transmitted securely to the Kubios Cloud using the group's Wi-Fi credentials. The Kubios Cloud performs sophisticated HRV analysis on the transmitted data within seconds. The Raspberry Pi Pico then receives the analyzed results back from the Kubios Cloud. Subsequently, the OLED screen displays key HRV metrics including mean peak-to-peak interval (PPI), mean heart rate (HR), SDNN, RMSSD, as well as sympathetic nervous system (SNS) and parasympathetic nervous system (PNS) indexes derived from the Kubios analysis.

5 Group Work Summary

Our group project focuses on Heart pulse detector, and teamwork plays a crucial role in our approach. Each team member brings valuable skills, allowing us to work together effectively towards our project objectives.

5.1 Midway Summary

Shamila: During the first half of the project, I took part in researching OLED coding techniques, reviewing the project's technical background. I made significant contributions to the report writing, particularly by compiling the theoretical background section based on relevant sources. Additionally, I actively organized team meetings, keeping us on schedule and focused on our objectives.

Lihini: So far, I have focused on the OLED and other related coding parts and conducting preliminary analyses based on the research questions we outlined. I diligently collected details about the hardware structure of the final product, enriching our understanding of the technical requirements. I have also started drafting the report of our project and took part in writing the introduction of the report. I took part in reviewing the project’s technical background.

Upeksha: In the first half of the project, I played a crucial role in coding and understanding the technical background required for implementation. I effectively applied theoretical concepts to practical coding tasks, ensuring the development of functional solutions. I contributed significantly to the project report by detailing the methods and materials used in our project. My contributions added depth and clarity to our project documentation.

As depicted in Table 1 you can get an overview of our future plan for this project.

Table 1: Structured plan for the upcoming weeks.

WEEK NO	PLAN
Week 5	● Do coding with OLED for the menu with four options. ● Finding more about hardware implementation. ● Continuing complete project report with implementations.
Week 6	● Concentrate on measurements and review about coding part and background. ● Review of project outcomes needs to be finalized and pending. ● Continuing complete project report with implementations.
Week 7	● Do implementations to connect the device with Wi-fi and do updating with Kubios Cloud. ● Review of project outcomes needs to be finalized and pending. ● Continuing complete project report with implementations
Week 8	● Finalize the project with correct coding and other stuff related to it. ● Completion of hardware part of project. ● Finalize the project report and peer review. ● Ready for the presentation and demo.

Midway through our device implementation project, we encountered significant challenges related to hardware compatibility and coding complexity. Integrating diverse hardware components like the OLED display and Raspberry Pi Pico presented issues due to compatibility issues and technical limitations. Additionally, coding for the OLED display in MicroPython proved more complex than anticipated, resulting in unexpected errors that required extensive testing and debugging efforts. Addressing these challenges effectively is crucial for ensuring the successful completion of our project. We are actively working to overcome these hurdles to deliver a functional and reliable device implementation.

5.2 Final Summary

Shamila: After doing research on the project's technical background, I took part in coding for the OLED display, the welcoming, the beginning of the programme, and the PPG signal wave. I was involved in testing and validation of the final programme. I made significant contributions to the report writing by completing the theoretical background and abstract sections by keeping the records of relevant references.I am also responsible for the theoretical background part in the presentation.I actively organized team meetings, keeping us on schedule and focused on our objectives.

Lihini:I have conducted research on the calculations and completed the coding part for the functions, including HRV calculation, mean PPI, RMSSD, SDNN, and integrated all the codes to make the final program functional. I have also contributed in drafting the report and took part in writing the introduction, final summary and conclusion. I was responsible for the Introduction and the implementation part in the presentation. Also, I have taken part in reviewing the project's technical background and also done testing and validation.

Upeksha: I was responsible for the coding parts for the OLED menu and other displays, encoder, MQTT and Kubios. I have effectively applied theoretical concepts which I have researched, in the coding tasks ensuring the development of functional solutions. I contributed significantly to the project report by detailing methods and materials used in the project and the

implementation part with relevant references. Also in the presentation I have done the methods and materials, conclusion, references and the final tune up.
I was also involved in testing and validating the project.

While doing the project, we faced so many challenges. Some of them are finding the correct threshold, reading the correct heart rate, and handling the encoder button press. Finally, as a dedicated, hardworking team we were able to find solutions for each and every problem and finish the project successfully.

6 Conclusions

In conclusion, the project has been a journey marked by both achievements and challenges. Overall, the project progressed, with significant improvement made towards achieving the original goals. Successfully developed and implemented a functioning heart rate detection device that provides accurate heart rate readings, advancing our objectives.

While progress was notable, encountered several challenges along the way. Faced issues with the encoder, determining the correct threshold level, and obtaining accurate heart rate readings. However, through effective use of theory, techniques, and teamwork, we navigated these obstacles with resilience and determination. Unfortunately, we were unable to implement the history part due to time limitations.

Despite successes, it's important to acknowledge the limitations inherent in the prototype measurement system. The main issues with these restrictions are accuracy and scalability. Moving forward, it is crucial to address these limitations as a priority to enhance the system's accuracy and applicability across diverse scenarios.

By avoiding the limitations and expanding the compatibility with different hardware configurations could significantly elevate the project's impact. Additionally, continued research and development could extend the functionality of the system, opening doors to new applications and opportunities for innovation.

In summary, while the project journey had its share of challenges, it made substantial progress towards the project’s goals. By addressing limitations and building upon the achievements, pave the way for further advancements and the realization of the project's full potential in the future.

References

[1] MacGill, Markus. 2023. What should my heart rate be? [online]. Medically reviewed by Carissa Stephens, Medical News Today. 30 November 2023. Accessed: 07 April 2024. https://www.medicalnewstoday.com/articles/235710

[2] American Heart Association. 2024. All about heart rate [online]. Accessed: 07 April 2024. https://www.heart.org/en/health-topics/high-blood-pressure/the-facts-about-high-blood-pressure/all-about-heart-rate-pulse

[3] Wood, Robert. 2008. Heart Rate Measurement Records [online]. Topend Sports Website, first published 2008. Accessed: 9 April 2024. https://www.topendsports.com/testing/records/heart-rate.htm#SnippetTab

[4] Medicine Net. 1996. What Is the maximum heart rate? [online]. Medical Reviewed by Sachdev, Poonam. 2021. Accessed: 07 April 2024. https://www.medicinenet.com/highest_heart_rate_you_can_have_without_dying/article.htm

[5] Bailey, Aubrey: PT: DPT: CHT. 2022. What is heart rate variability & What can it tell you? Medically reviewed by Jeffrey S. Lander, MD [online] Accessed: 7 April 2024. https://www.verywellhealth.com/heart-rate-variability-5215411

[6] Schäfer, Axel: Vagedes, Jan. 2012. How accurate is pulse rate variability as an estimate of heart rate variability? - International Journal of Cardiology. Accessed: 9 April 2024. https://www.internationaljournalofcardiology.com/article/S0167-5273(12)00326-9/abstract

[7] Sacha Jerzy (Ann Noninvasive Electrocardiol). 2014. Interaction between heart rate and heart rate variability [online]. Accessed:9 April 2024.
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6932101/

[8] Summer, Jay: Singh, Abhinav. 2024. What is a normal sleeping heart rate[online]. Accessed: 9 April 2024.
https://www.sleepfoundation.org/physical-health/sleeping-heart-rate

[9] Lindsay, Mark. 2023. Everything You Need to Know About Heart Rate Variability (HRV) [online]. Accessed: 9 April 2024.
https://www.polar.com/blog/heart-rate-variability-hrv/

[10] Healthline. 2023. Why Do Athletes Have a Lower Resting Heart Rate? [online]. Accessed: 11 May 2024.
https://wiki.elecrow.com/crowtail--pulse-sensor.html

[11] Park, Junhyung: Seok, Hyeon Seok: Kim, Sang-Su: Shin, Hangsik. 2022. Photoplethysmogram analysis and applications: An integrative review. Sec. Vascular Physiology [online]. Accessed: 9 April 2024.
https://www.frontiersin.org/journals/physiology/articles/10.3389/fphys.2021.808451/full

[12] Belgium, Corda Campus. 2019. FibriCheck beat-to-beat accuracy compared with wearable ECG in broad dynamic range. FibriCheck. (n.d.) [online]. Accessed: 9 April 2024.
https://www.fibricheck.com/fibricheck-beat-to-beat-accuracy-compared-with-wearable-ecg-in-broad-dynamic-range/

[13] Elecrow Bazaar. 2022. Crowtail - Pulse Sensor [online]. Assessed: 10 April 2024
https://wiki.elecrow.com/crowtail--pulse-sensor.html

[14] Raspberry Pi Pico W Datasheet. 2024. RP2040 A microcontroller by Raspberry pi. Accessed: 07 April 2024. https://datasheets.raspberrypi.com/picow/pico-w-datasheet.pdf

[15] RaspberryPi.dk.Jkollerup.dk ApS,Denmark. Accessed: 07 April 2024. https://raspberrypi.dk/en/product/raspberry-pi-pico-w/

[16] NettiGo, OLED Display 0.96" I2C 128x64 SSD1306 white. Accessed: 07 April 2024. https://nettigo.eu/products/oled-display-0-96-i2c-128x64-ssd1306-white

[17] AEAK official store.360 Degrees Rotary Encoder Module Brick Sensor Switch Development KY-040 for Arduino. https://www.aliexpress.com/item/1005002895119611.html

[18] Crowtail- Pulse Sensor 2.0. Accessed: 07 April 2024. https://www.elecrow.com/crowtail-pulse-sensor-p-1673.html

[19] Tarvainen MP, Lipponen JA, Niskanen J-P, and Ranta-aho PO.2021. Heart Rate Variability (HRV) Software. Accessed: 07 April 2024. https://www.kubios.com/