Team member: Nhi Dinh, Trung Vu, Mahnoor Fatima

Smart Heart

Project Report

First-year Hardware Project

School of ICT

Metropolia University of Applied Sciences

05 May 2024

Abstract

Advancements in the IT domain are driving transformative changes in the field of healthcare technology. The project was a noble endeavour in regard that it aimed to improve human well-being by monitoring heart rates in real time. It encourages people to take proactive health measures and sustain their fitness levels.

The project aimed to develop an efficient non-invasive heart rate measuring device. The device aspired to provide the user with advanced health indicators among other options available. Another goal of the project was to design a device that could be used by a professional or someone having limited IT knowledge with similar ease.

The project utilised Photoplethysmography (PPG) signals that measure the light reflected by skin tissues as the blood volume changes. The reflected light is converted into a pulsatile waveform by a diode. The wave's peak-to-peak interval, calculated using advanced algorithms, represents heart rate. With the help of micropython libraries users can measure their heart rate, access Kubios advanced heart rate parameters and link additional devices using MQTT.

The project concluded with the achievement of a fully functional heart rate monitoring device that enables the user to select among different options of basic HR analysis, Kubios or History. The device was experimentally validated by a series of tests and trials conducted on a diverse sample population. The results were within the allowed limits and demonstrated the accuracy of the project's heart rate measurement system.

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
2.1	Updated the Midway Summary section	23.03.2024	Nhi Dinh
2.2	Updated the methods and material section	11.04.2024	Trung Vu
2.3	Updated the theoretical background section	12.04.2024	Mahnoor Fatima
2.4	Updated the introduction	12.04.2024	Nhi Dinh
2.5	Updated the theoretical background section	16.04.2024	Mahnoor Fatima
2.6	Updated the abstract	26.04.2024	Mahnoor Fatima
2.7	Updated the implementation section	27.04.2024	Mahnoor Fatima
2.8	Updated the implementation section	28.04.2024	Nhi Dinh
2.9	Updated the conclusion	28.04.2024	Trung Vu
2.10	Updated the final summary section	28.04.2024	Nhi Dinh
2.11	Updated the final summary section	29.04.2024	Mahnoor Fatima

Contents

1 Introduction

Health technology is experiencing unprecedented growth as global healthcare demands continue to rise. Innovations in this sector are extremely important as they contribute significantly to improving public healthcare. With the growing role of technology in health management, projects that integrate advanced technical solutions with healthcare applications are becoming more popular. Our team's project, Smart Heart, aligns with this trend.

As a crucial component of the Hardware 2 course at Metropolia University of Applied Sciences, the Smart Heart – Heart Rate Detector project aims to equip first-year ICT students with fundamental hardware engineering skills and practical experience in developing a heart rate monitoring device for use by individuals, patients, and healthcare professionals. The primary objective of this project is to create a comprehensive heart rate detection and analysis system. It includes developing a heart rate detection algorithm, ensuring it operates effectively in real-time, calculating heart rate variability (HRV), integrating with Kubios Cloud service for in-depth HRV analysis, and displaying stress and recovery metrics on the device to provide users with real-time feedback on their physiological health.

This project is not merely an academic exercise. It strives to equip students with the necessary skills for practical application in health technology in particular and embedded systems development in general. By combining theoretical knowledge with practical application, the project encourages students to explore innovative solutions in the growing field of health informatics, laying the foundation for future advancements.

2 Theoretical Background

The document provides a framework for understanding the basic idea of heart rate and the theory of Heart Rate Variability (HRV), explaining HRV dependence on heart rate. The theoretical part of the report delves into the core concepts of heart rate and its variability to establish a solid foundation for the subsequent discussion and conclusion in later chapters.

2.1 Heart Rate and Heart Rate Variability

Heart rate is the number of cardiac muscle contractions per minute. It is recorded in bpm (beats per minute) [1]. Heart rate in human beings ranges from 40 bpm to 200 bpm, nonetheless, higher values are recorded in athletes. However, the heart rate may fluctuate from the norm due to several factors, including regular physical activity, a health condition, stress and use of medications. [2.] Heart rate can be measured from an artery close to the skin; at the wrist, the side of the neck, the backside of the knee, the front side of the foot and the forehead [3].

HRV is the fluctuations in the time intervals amidst each heartbeat [4]. A healthy adult has an average HRV of 42 milliseconds, with a normal range from 19 to 75 milliseconds. A well trained athlete can observe high HRV values of approximately 120 milliseconds. [5.] The HRV is regulated by a core part of the nervous system called the Autonomic Nervous System (ANS). The brain is actively processing the heart rate variations in the hypothalamus area. The ANS delivers signals to the hypothalamus, which aligns the body either to react actively or to relax various body functions altering HRV. However, with constant agitators such as stress, pressure, poor sleep or lack of physical activity, ANS equilibrium can be disturbed, leading to higher HRV fluctuations. [4.]

HRV reflects many physiological factors for example, an interplay between the “stress or alert” and “calm or relax” nervous systems by the ANS activity. These factors are regulated by two parts of ANS: the sympathetic and parasympathetic nervous systems. [4; 6.]

When the body senses a potential threat, it shifts to the “fight or flight” mode, it speeds up breathing and decreases HRV, this is controlled by the sympathetic nervous system (SNS). SNS provides the power and means needed to survive such a critical situation. The parasympathetic nervous system (PNS) is the “state of relaxation”, it slows down the heart rate and increases HRV while helping the body get back to baseline condition. Therefore, heart rate is lowest and HRV is highest when we are at rest and fully recovered. A significant time difference between heartbeats, or a high HRV score, indicates that ANS can quickly activate the SNS in stressful situations and then switch to the PNS when the threat is gone. [6.]

The PNS index also known as recovery index (cf. Figure 1) is typically between -2 and +2 for a normal person. The index is derived from Kubios HRV software grounded on the following parameters: Mean RR that is measure of heart beat and parasympathetic cardiac stimulation, Root mean square of successive RR interval differences (RMSSD), Poincaré plot index SD1 (short-term HRV). [7.]

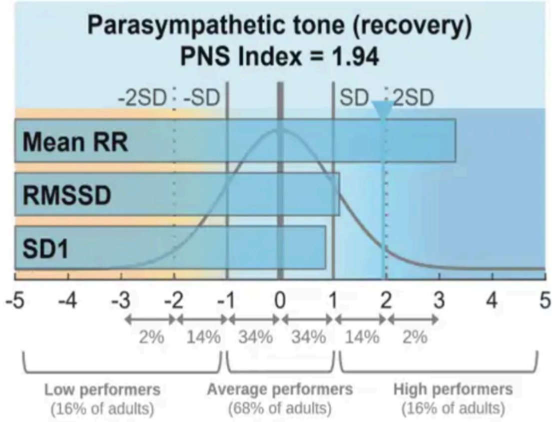

Figure 1. PNS Index [7].

The SNS index or stress index (cf. Figure 2) lies between -2 and +2 for a normal person at rest. The SNS index is computed using three parameters: Mean RR that is a measure of heart beat and parasympathetic cardiac stimulation, Root mean square of successive RR interval differences (RMSSD), Poincaré plot index SD2 (long-term HRV). [7.]

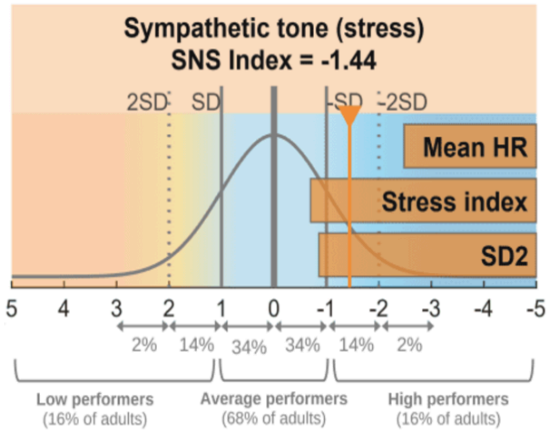

Figure 2. SNS Index [7].

2.2 Signal Conditioning using PPG:

In the project, photoplethysmography (PPG) is used to measure the heart rate by generating a pulsatile waveform. PPG is a valuable tool, offering a non-invasive surface level optical approach to determine HRV. It uses optics to measure the blood circulation changes in the capillaries of the body. A thin skin surface that is enriched with blood capillaries is exposed to light by a light-emitting diode (LED) which can be either an infra-red rays or red LED [8; 9.]

The light reflected by the skin tissues is captured by a photodiode depicting the blood volume changes. The diode is placed either in parallel or on the opposite side with the light source side to collect the transmitted or reflected light by the skin. The project uses Crowtail Pulse sensor v2.0 by placing it on the same side as the light source (cf. Figure 3) to capture the reflected light intensity. [10.] It is recommended to place the sensor at the earlobe, wrist or finger for accurate readings [11].

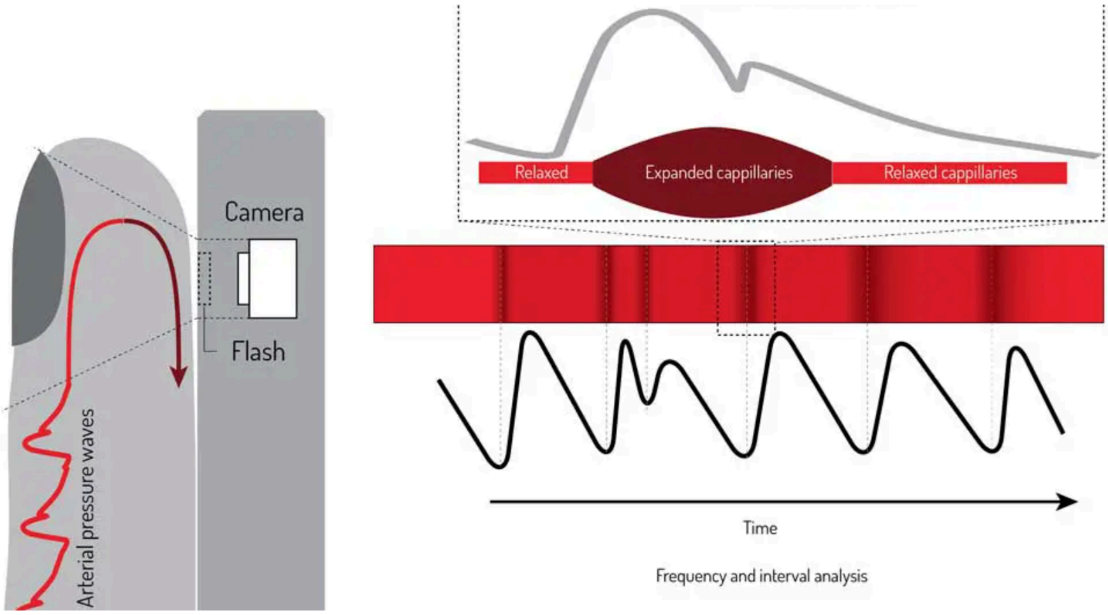

Figure 3. PPG Signal System [11].

The intensity of the light is converted into a voltage signal by photodiode. The amplitude of the signal depends on the light absorbed by the blood capillaries (cf. Figure 3), the absorbing power is directly proportional to the flow of blood cells through the tissues. Heart rate corresponds to the periodicity of this PPG signal, which can be used towards estimating HRV. In the project, the PPG signal is processed in time-domain; the peaks are determined by an advanced algorithm, providing live heart rate value and pulse signal [10;11.] Heart rate is the peak to peak interval of the signal and by further processing it, the device displays advanced heart rate metrics [11].

3 Methods and Material

This section discusses the methodology and materials used in the project. The first part covers the hardware requirements, followed by the implementation of software programs. The last part provides the operational principles of the device and outlines methods for accuracy testing.

3.1 Hardware requirements

Raspberry Pi Pico W (see Figure 4, part 1) is a microcontroller board that operates with the Raspberry Pi RP2040 microcontroller chip. The board acts as the main control system of the device due to its wide range of functionalities. [12.]

The board manages the process of capturing data from the Crowtail Pulse Sensor 2.0 (see Figure 4, part 2) for heart rate detection and works with software programs to analyse heart rate variability [13.] Then, the processed data is displayed on a 128x64 monochrome OLED screen, using I2C method to provide real time heart rate readings [14.] Next to the OLED screen is a rotary knob (see Figure 4, part 3) that acts as a button for navigation, enabling scrolling and selection of device features on the screen [15.] Additionally, the board is equipped for internet connection through 2.4GHz band wireless interface, which is used to transmit heart rate data to the Kubios cloud for further analysis [16.] The board also comes with a USB B port (see Figure 4, part 4) that connects the device to a power source.

The router (see Figure 4, part 7) provides internet connection on the 2.4GHz band, synchronised with the Raspberry Pi Pico. To protect the sensitive heart rate data of users, access to the internet is restricted exclusively to authorised users.

The broker (see Figure 4, part 5) and its accompanying power plug (see Figure 4, part 6) enables the transmission of heart rate data to subscribed clients' laptops. This structure combines the router connection with the MQTT protocol to ensure safe data transfer. [17.]

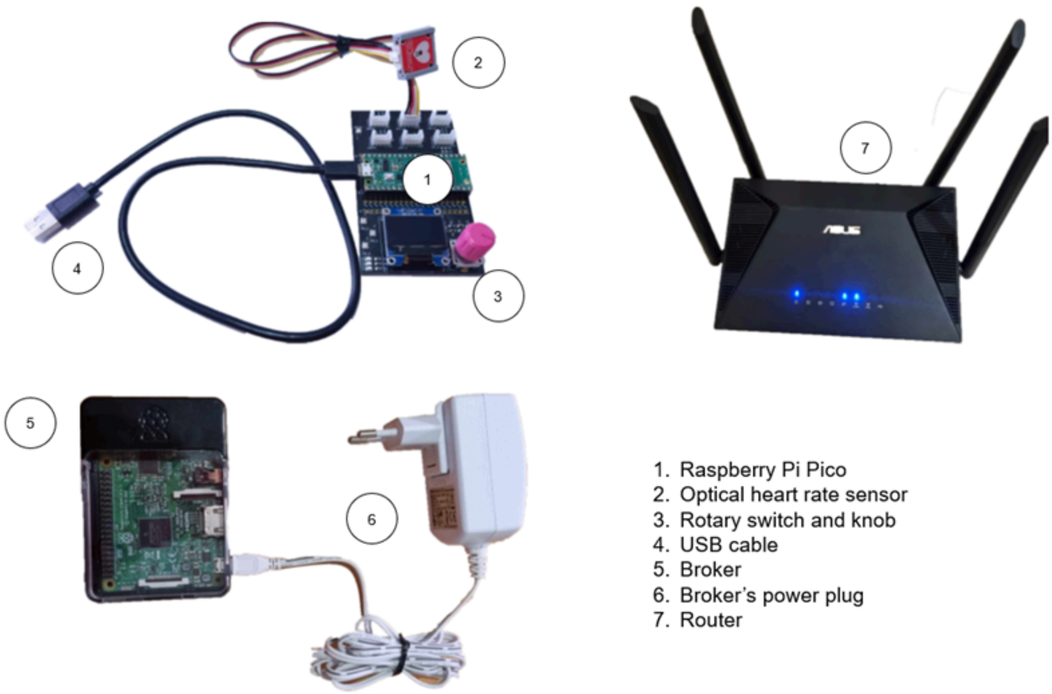

Figure 4. The system components.

3.2 Software requirements

The device's program is written using the MicroPython programming language and Thonny IDE programming environment.

Thonny IDE is an open-source platform which is well-known for its compatibility and versatility with embedded devices. Users can download and install Raspberry Pi Pico W firmware into Thonny, where embedded functions can be directly programmed and easily tested. [18.]
MicroPython is a programming language suitable for embedded devices as it provides various libraries for the system's functionalities [19]. For example, "umqtt" module handles the transmission of MQTT messages to subscribed devices, "ssd1306" module controls the interface and display of OLED screens, and "machine" module offers connection to Raspberry Pi Pico W's different pins and led lights, enabling control functionalities such as the rotary knob [20.]

In addition, classes like Fifo, Piotimer, and interrupt functions are provided by project lecturers and to be utilised in the development of our project. Fifo (First In First Out) is a buffer used to allocate data temporarily and extract data in a safe way so that there are no contradictions in between program runs, this method should be implemented in a way that the speed of data input is the same with the speed of data extracting to prevent the Fifo from getting full. This Fifo class works in harmony with interrupt functions, designed to handle user interactions responsively, such as button pushes and button rotates. This interrupt function is executed based entirely on hardware, and their response runs independently from the main program. Piotimer is another type of hard interrupts which is based on a timer. The events triggered by Piotimer will be handled immediately through each cycle of time, which is necessary for applications where accurate timing is essential [21.]

3.3 Operations and testings

The device’s high-level system starts from the left of Figure 5, where the user places their finger on the heart rate sensor. The microcontroller board then captures the data and processes it into heart rate variability. Next, the heart rate data is sent to the Kubios cloud for deeper health analysis via Wi-Fi router. The results are ultimately transferred back to the client's laptop via the MQTT method and simultaneously displayed on the device's OLED screen.

In order to assess the accuracy of the device measuring capabilities, experiments are conducted using different sensors and involving many test users. Testing locations of the users are also considered to assess the sensitivity of the sensor. The recorded heart rates are consistently compared to normal human heart rate of 50-100 beats per minute to disregard inaccurate readings and ensure the reading’s validity. The result of this experiment is used to gain further understanding of the sensor’s functionality and necessary data to enhance accuracy and record potential improvements.

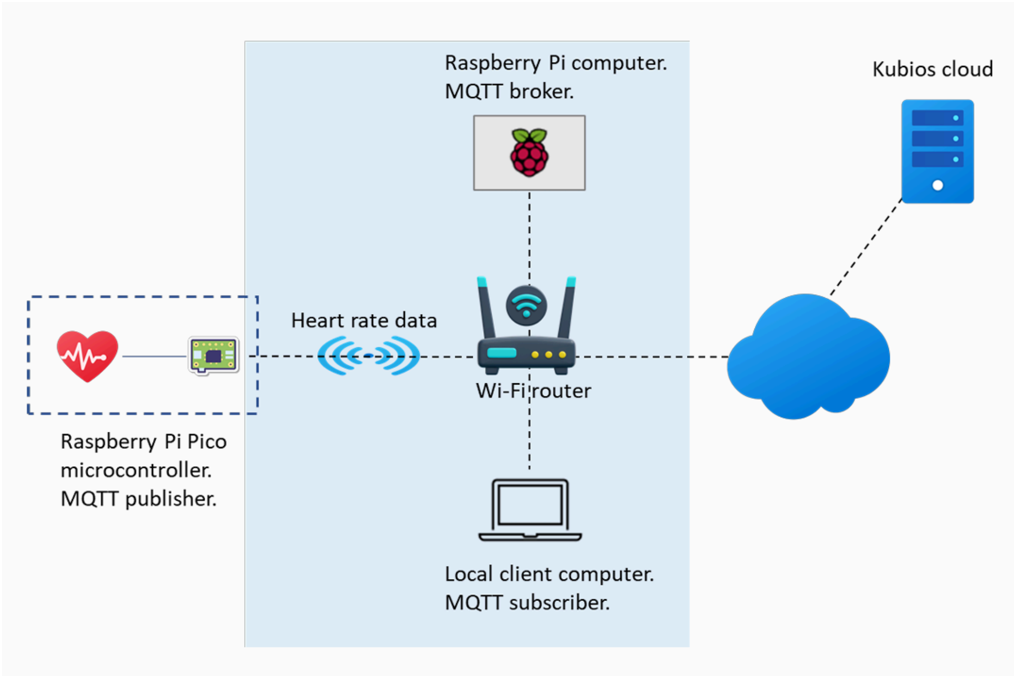

Figure 5. High-level system diagram of our device [22].

4 Implementation

This chapter focuses on the comprehensive details of the development of our heart rate detection system. The first subchapter provides the overview of our final end-to-end system including all the devices used, their connections, and their roles in the system. The second chapter will cover the processes of data collection and data handling, which are crucial to ensuring that the system operates both efficiently and accurately. In the final subchapter, the algorithms developed specifically for this project by our group to calculate heart rate and heart rate variability will be introduced.

4.1 System

The project is a sophisticated amalgamation of its hardware and software components required for an accurate heart rate monitoring system. The essential components are shown below in Figure 6.

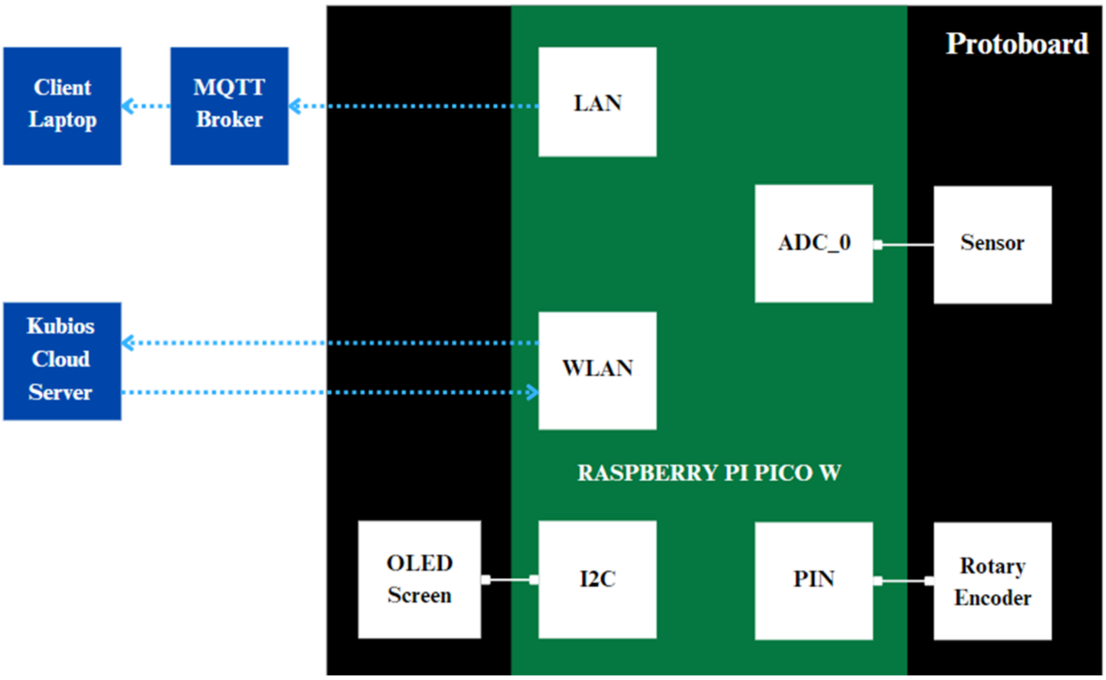

Figure 6. The system

As discussed in section 2.2, the core of the system is its PPG sensor (Crowtail Pulse sensor v2.0) that is strategically positioned on the user's skin to detect the reflected light. As shown in Figure 6, the sensor is connected to Raspberry pico through an ADC pin (ADC_0), it interfaces with an electric diode to form the basis of signal acquisition.

The signal processing unit performs advanced algorithms aimed to calculate heart rate from the captured PPG signals. The system incorporates a user interface with the help of rotary encoder. The rotary encoder is integrated with a pico board (cf. Figure 6) and facilitates the user to navigate and select different options.

The I2C interface on the Pico board interacts with the OLED display (cf. Figure 6) to show the heart rate on the OLED screen. Furthermore, the OLED screen displays signal quality by generating a live pulse of a fluctuating heart rate signal as well as other options like Kubios and MQTT.

To increase system versatility, pico board is equipped with the required libraries for WLAN connectivity (cf. Figure 6). The Pico board establishes a TCP/IP socket connection with the Kubios server using WLAN connection. The real time heart rate data is sent to Kubios software, a crucial element in deep analysis of heart rate.

Additionally, MQTT protocol connection with LAN (cf. Figure 6) facilitates fluid data communication between client laptops and the system. The system publishes heart rate measurements for a specific topic to a designated broker allowing subscribed clients to receive the data as needed.

4.2 Data collection

In this project, analog PPG signals are collected through the Crowtail Pulse Sensor v2.0, with detailed information about the sensor and PPG signal provided in Chapter 2.2. The measurement process initiates when a human subject comes into direct contact with the sensor. This optical sensor detects the heartbeats as analog signals, which are then transmitted to the Raspberry Pi Pico. These signals are converted into digital ADC values by the microcontroller's AD converter.

The Piotimer library is employed specifically to sample these digital ADC values, ensuring that the sampling frequency remains consistent and independent of other system components. This hardware timer guarantees that data collection occurs at precisely defined intervals. In this system, the sampling frequency is set at 250 Hz, meaning that 250 samples are collected every second. To implement this method, the function designated to store data in an instance of the Fifo class must be assigned as the handler for the timer interrupt.

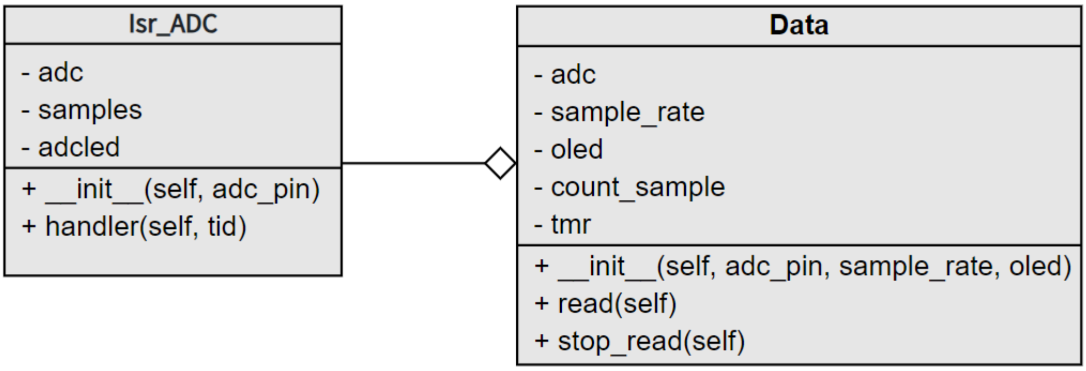

Figure 7. Heart rate signal data collection program's class diagram.

As illustrated in Figure 7, the Isr_ADC class is defined with a method to collect data and store them in an instance of the Fifo class named samples. An object of the Isr_ADC class is instantiated and used as an object parameter within the Data class, thereby establishing a connection between these two classes. Additionally, the methods for starting and stopping data collection are implemented as the object initializer and deactivator of the Piotimer class, using the Isr_ADC class object as a callback function.

For HRV analysis and Kubios services, it is essential to capture at least 30 seconds of data at a frequency of 250 Hz. In this system, data collection can continue for up to 60 seconds, but the user has the option to stop the collection at any point during the measurement by pressing the Rotary Encoder button. This design ensures that sufficient data is collected for both local HRV analysis and HRV analysis via the Kubios cloud service.

4.3 HR and HRV detection algorithm

As described in Chapter 2.2, PPG signals generate a pulsatile waveform characterised by periodic peaks, each corresponding to a heartbeat. The frequency of these peaks within a minute determines the heart rate. Utilising this foundational theory, a specialised algorithm is developed to detect the positive peaks of the PPG signal. It then calculates the peak-to-peak interval (PPI), from which the heart rate is derived by taking the inverse of the PPI. All successfully identified PPIs are compiled into a list, which is used to calculate the average heart rate (HR) and basic heart rate variability (HRV) metrics, such as RMSSD and SDNN. The algorithm faces two major challenges in accurately identifying the true peaks of the PPG signal. Firstly, the signal received from the sensor is often contaminated with substantial noise from various sources, such as ambient light during PPG measurements, electromagnetic interference from other electronic devices, and movement artefacts. Secondly, the amplitude of the PPG signal can vary significantly, further complicating the accurate detection of peaks.

Based on extensive research on heart rate (HR) and heart rate variability (HRV) mentioned in chapter 2, our team's algorithms effectively address these challenges. The peak detection function is enhanced by applying a dynamic thresholding mechanism. Only data samples that exceed this threshold are considered potential peaks. The threshold is adjustable to accommodate varying signal magnitudes. Furthermore, empirical observations of the actual signal indicate that the values of two consecutive true peaks typically do not differ by more than 80%. This observation serves as a basis for a filter designed to eliminate false peaks resulting from noise. The peak-to-peak intervals detected by this program are then compared to the typical maximum and minimum PPI values observed in humans. If the PPI meets these criteria, it undergoes one final check: the difference between two consecutive PPIs should not exceed 120 milliseconds. This limit is an exceptionally high HRV value, typically observed in athletes, particularly those who are well-trained in endurance sports. This algorithm is detailed in Figure 8.

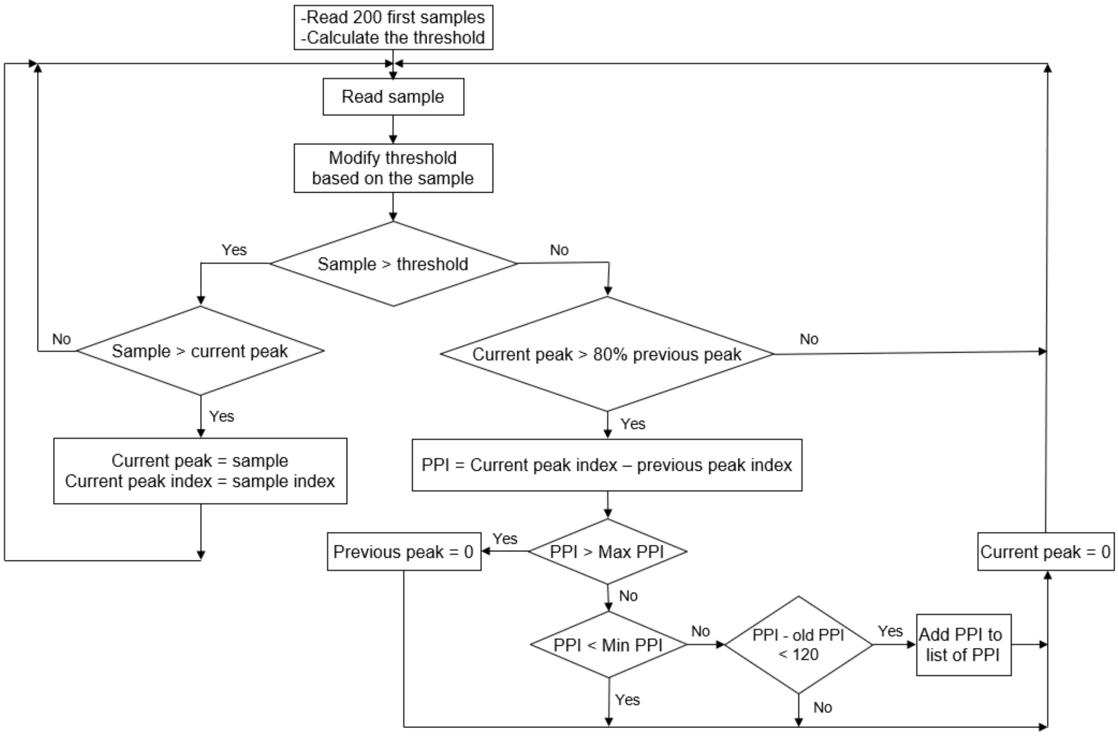

Figure 8. The flowchart of the heart peak and PPI detection algorithm.

As illustrated in Figure 8, the first 200 samples from the fifo called samples are used to calculate an initial average that serves as a threshold. Following this, the peak and PPI detection loop begins by reading a new sample and recalculating the threshold based on this sample. The sample is compared to the threshold; if it exceeds the threshold, it is then compared to the current peak. Should the sample exceed the current peak, it is designated as the new current peak, and its index is saved as the current peak index. This comparison continues until the sample value falls below the threshold. This method identifies the maximum point within an interval of samples that exceeds the threshold, with the highest point being recognized as the signal's peak for that period. To further enhance accuracy, this peak is verified to ensure it is at least 80% of the previous peak's value, thus helping to eliminate false peaks caused by noise.

After two peaks are detected, the difference between them represents the peak-to-peak interval (PPI). Subsequently, various filters are applied to further validate and refine the PPI values, discarding any that may be inaccurate due to signal errors. If a PPI exceeds the typical human range, specifically more than 1500 milliseconds, the previously detected peak is discarded. A PPI value is only retained if it meets all the following criteria: it falls within the normal human range (between 300 milliseconds and 1500 milliseconds), and the variation from the previous PPI does not exceed 120 milliseconds. Only then is the PPI value added to the list of valid PPIs.

The list of valid PPIs is utilised to calculate the mean HR and the basic time-domain HRV metrics, such as RMSSD and SDNN. The formulas for calculating these values are illustrated in Figure 9.

$$\overline{IBI} = \frac{1}{N}\sum_{n=1}^{N} IBI_n, \qquad \overline{HR} = \frac{60}{\overline{IBI}} \cdot 1000$$

$$SDNN = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}(IBI_n - \overline{IBI})^2}$$

$$RMSSD = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N-1}(IBI_{n+1} - IBI_n)^2}$$

Figure 9. The formulas for calculating HR and the basic time-domain HRV metrics.

5 Group Work Summary

5.1 Midway Summary

The table below describes the work each member has done to complete the first stage of the project from the first to the fourth week. It also includes the plans for the following weeks.

Table 1. Individual contributions of group members in the first stage.

	Nhi	Mahnoor	Trung
Week 1	- Researched the OLED display - Solved the first week assignments	- Analysed notes, researched OLED documentation. - Finished week 1 tasks.	- Studied the OLED functions. - Completed week 1 assignments.
	-Researching the pulse detection algorithm and fifo class		
Week 2	- Researched the GPIO interrupts, the pulse detection algorithm, Fifo and Filefifo class - Solved the second week assignments	- Went through GPIO lecture notes, fifo library and its impact on code. - Finished week 2 tasks.	- Studied pulse detections and GPIO interrupts. - Completed week 2 assignments.
	-Researching the Rotary encoder		
Week 3	- Researched the Rotary encoder, Timer and ADC value reading - Solved the third week assignments	- Studied notes and internet for Rotary Knob coding. - Finished week 3 tasks.	- Studied FIFO class and rotary encoder. - Completed week 3 assignments.
	-Researching the Timer and ADC value reading		
Week 4	- Researched the Timer and ADC value reading - Solved the fourth week assignments - Implemented the system flowchart	- Investigated about timers and why it's important. - Wrote Theoretical background part of the project.	- Studied timer and ADC. - Completed the methods and material part of the project.

	- Completed the Introduction part of the project report		
	-Implementing the pulse detection algorithm for real data taken from the sensor	-Implementing Kubios part of the project. -Implement MQTT part of project	- Implemented the history feature to the device.

5.2 Final Summary

The table below describes the work each member has done to complete the second stage of the project from the fifth week to seventh week. It also includes the plans for the following weeks.

Table 2. Individual contributions of group members in the second stage.

	Nhi	Mahnoor	Trung
Week 5	-Implemented the pulse detection algorithm -Completed the OOP design for the system -Implemented the Data class utilised to measure, process and display data in real time system -Implemented the States class which is defined all the states of the system -Implemented the main program	- Implemented Kubios class and its methods for simpler integration with main logic. -Crafted code for MQTT class for local connection. -Understanding project logic and its implementation with my colleagues.	- Implemented the History class. - Applied the OOP design system from Nhi for easier integration. -Integrated the History class into the States class and the main program
	-Writing the implementation section of the document	-Drafting the Theoretical Background and reference section of the report.	-Writing the conclusion section for the project report

Week 6	-Completed the implementation section of the document -Tested the final system and solved errors	-Integrated code with main logic and pushed it on Git lab. - Completed abstract and system section of the report.	-Tested and solved errors - Wrote the conclusion for the project report. - Made presentation material.
	-Preparing for the presentation	-Preparing for the presentation.	-Preparing for the presentation
Week 7	-Tested the final system and solved errors -Prepared for the final presentation	-Finalising the whole documentation and checking it on Turnitin. -Testing the device with my group members.	- Tested the device and Prepared for the presentation.

Throughout the project, our group maintained strong collaboration and effective communication, working independently to build essential knowledge while having regular meetings to exchange information and solve problems together. This approach fostered diverse perspectives, enhancing our problem-solving capabilities. In the second stage, tasks were allocated based on individual strengths, ensuring optimal project outcomes and coherent integration of the system’s components. We also held detailed discussions on each member's contributions to deepen our collective understanding and facilitate skill development. Ultimately, we successfully developed a highly accurate heart rate detection system. Overall, the project not only achieved its set goals, leading to significant learning and skill development among team members but also highlighted the importance of continuous improvement in teamwork practices for future projects.

6 Conclusions

The project is an implementation of various technologies to yield a final device that can capture heart rate accurately and provide users with thorough analysis and information about their health. In comparison with the project goals, the final device has displayed its capabilities and functionalities that satisfy the need of reliable heart rate measuring. While possessing many useful features, the device still maintains a non-invasive and user-friendly interface that is suitable for both individual and professional use.

In the process of achieving the final product, several challenges were encountered and dealt with. The data collecting algorithm has to take in account external interferences and continuous tests in various conditions so that the device can capture accurate readings. With the Kubios feature, the device requires a stable internet connection and also safe error handling to not confuse users when an internet connection is unstable.

With the device's accomplishment, there are also limitations that need to be further developed in the future. The heart rate sensor is sensitive to the outside environment and needs careful storage and maintenance. Some of the device features also heavily rely on internet connection, and cannot function without the internet.

For future developments, there are ideas to implement an algorithm that assesses the user's emotions based on heart rate variability. There was already a similar system using HRV to detect basic emotion with an 83.8% accuracy [23.] This feature would be a useful addition to our device because it can be utilised in the healthcare field where there is a need to understand a patient's emotions without intruding on the patient.

The journey to achieve the project goal has offered valuable experiences and knowledge towards both hardware and software utilisation in health technology applications. With the advancements of technology, the need for product improvement is always present and the applications for this device can be further analysed.

References

1 British Heart Foundation. What is a Normal Pulse Rate? [Internet]. Bhf.org.uk. British Heart Foundation; 2021 [Cited 11th April 2024]. Available from: https://www.bhf.org.uk/informationsupport/heart-matters-magazine/medical/ask-the-experts/pulse-rate

2 Hipp D. Normal Resting Heart Rate by Age (Chart) [Internet]. Forbes Health. 2022 [Cited 11th April 2024]. Available from: https://www.forbes.com/health/wellness/normal-heart-rate-by-age/

3 National Cancer Institute. NCI Dictionary of Cancer Terms. National Cancer Institute. 2011 [Cited 11th April 2024]. Available from: https://www.cancer.gov/publications/dictionaries/cancer-terms/def/heart-rate

4 Campos M. Heart rate variability: A New Way to Track Well-being - Harvard Health Blog [Internet]. Harvard Health Blog. 2017 [Cited 11th April 2024]. Available from: https://www.health.harvard.edu/blog/heart-rate-variability-new-way-track-well-2017112212789

5 Tiwari R, Kumar R, Malik S, Raj T, Kumar P. Analysis of Heart Rate Variability and Implication of Different Factors on Heart Rate Variability. Current Cardiology Reviews. 2020 Dec 31;17(5).

6 Tech P. Why We Should Care About HRV (Heart rate variability) [Internet]. Pulsetto. [Cited 11th April 2024]. Available from: https://pulsetto.tech/blogs/blog/why-we-should-care-about-hrv-heart-rate-variability?gc_id=20758129114&gad_source=1&gclid=CjwKCAjw8diwBh

AbEiwA7i_sJUtoetyKFnGfhxGVqdVVDAfhOYiT9Te0TrNSym2ywnh4n5lgu8ESbBoC6kEQAvD_BwE

7 HRV and Autonomic Nervous System Function [Internet]. Kubios. [Cited 11th April 2024]. Available from: https://www.kubios.com/hrv-ans-function/

8 Cheriyedath S. Photoplethysmography (PPG) [Internet]. News-Medical.net. 2016 [Cited 11th April 2024]. Available from: https://www.news-medical.net/health/Photoplethysmography-(PPG).aspx

9 Nuno Cercas Pinheiro, Couceiro R, Henriques J, Jens Muehlsteff, Quintal I, Gonçalves L, Carvalho P. Can PPG be Used for HRV Analysis? PubMed. August 2016:2945-2949

10 Allen J. (PDF) Photoplethysmography and its Application in Clinical Physiological Measurement [Internet]. ResearchGate. April 2007:28(3):R1-39

11 FibriCheck Beat-To-Beat Accuracy Compared with Wearable ECG in Broad Dynamic Range. FibriCheck, 20 June 2019. [Cited 9th May 2024]. Available from: https://www.fibricheck.com/fibricheck-beat-to-beat-accuracy-compared-with-wearable-ecg-in-broad-dynamic-range/

12 Raspberry Pi Ltd. Raspberry Pi Pico W Datasheet. [online]. 2024. [Cited 11th April 2024]. Available from: https://datasheets.raspberrypi.com/picow/pico-w-datasheet.pdf

13 Williams B. Pulse Sensor on a Pi Pico. [online]. Elektor Magazine. 2022. [Cited 11th April 2024]. Available from: https://www.elektormagazine.com/news/pulse-sensor-on-a-pi-pico

14 Santos R. Guide for I2C OLED Display with Arduino. [online]. Random Nerd Tutorials. 2016. [Cited 11th April 2024]. Available from: https://randomnerdtutorials.com/guide-for-oled-display-with-arduino/

15 The Pi Hut. How to Use a Rotary Encoder with the Raspberry Pi. [online]. 2014. [Cited 11th April 2024]. Available from: https://thepihut.com/blogs/raspberry-pi-tutorials/how-to-use-a-rotary-encoder-with-the-raspberry-pi

16 Tarvainen M, Niskanen J, Lipponen J, Ranta-aho P, Karjalainen P. Kubios HRV – Heart Rate Variability Analysis Software. [online]. 2013:113(1):210-20. [Cited 11th April 2024]. Available from: https://www.sciencedirect.com/science/article/abs/pii/S0169260713002599?via%3Dihub

17 Padron E. Applications with Raspberry Pi and Arduino Using MQTT. [online]. Medium. 2023. [Cited 11th April 2024]. Available from: https://fulldataalchemist.medium.com/applications-with-raspberry-pi-and-arduino-using-mqtt-b6594c29aed8

18 Annamaa A. Introducing Thonny, a Python IDE for Learning Programming. [online]. 2015. [Cited 11th April 2024]. Available from: https://dl.acm.org/doi/10.1145/2828959.2828969

19 Madhumita V. Micro Python: More Powerful than Arduino, Simpler than the Raspberry Pi. [online]. Wired. 2013. [Cited 11th April 2024]. Available from: https://www.wired.com/story/micro-python/

20 MicroPython. Python for Microcontrollers. [online]. [Cited 11th April 2024]. Available from: https://micropython.org/

21 Lansk. Pico test. [online]. 2024. [Cited 11th April 2024]. Available from: https://gitlab.metropolia.fi/lansk/pico-test

22 Uusitalo M. Network Final Assignment. [online]. [Cited 11th April 2024]. Available from: https://moodle.metropolia.fi/

23 Rakshit R, Reddy V, Deshpande P. Emotion Detection and Recognition using HRV Features Derived from Photoplethysmogram Signals.[online]. 2016. [Cited 28th April 2024]. Available from: https://dl.acm.org/doi/10.1145/3009960.3009962