July 6, 2019

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
V5A 1S6

RE: ENSC405W/440 Design Specification: Pulse Tracer

Dear Dr. Rawicz,

Please see attached the Design Specifications for Pulse Tracer, a patient monitoring system created by LumoAnalytics. Pulse Tracer uses photoplethysmography (PPG) to remotely measure the heart and respiratory rates of primarily immobile elderly patients who are often left unattended. Our goal for this device is to provide a way to accurately measure the heart and respiratory rates of patients and to detect significant variations from the average values in an attempt to alert caregivers of possible emerging health conditions. This technology will provide patients the ability to live in their own homes while still having a professional caregiver monitor their health.

The attached document will provide detailed specifications for the designs we have developed for the various components of Pulse Tracer. It will first provide an overview of the various components of Pulse Tracer and then provide a detailed design of the individual components, namely the hardware, image processing, and data analysis components. This document also includes test plans for the proof of concept prototype, as well as a design of the user interface as appendices.

LumoAnalytics is comprised of a team of five engineering students: Winsey Chui, Wenpei Li, Huy Thong Bui, Corey Myrdal, and Brittany Hewitson. Our team members come from a range of engineering concentrations including electronics, biomedical, and computer engineering. We believe each member brings a particular skill set to the team that will contribute to the success of Pulse Tracer.

Thank you in advance for taking the time to review our design specifications document. If you have any questions or concerns, please feel free to email our Chief Communications Officer, Huy Thong Bui, by email at htbui@sfu.ca

Sincerely,

Brittany Hewitson
Chief Executive Officer
LumoAnalytics

LumoAnalytics

Design Specifications for
Pulse Tracer

Project Team:

Brittany Hewitson
Corey Myrdal
Huy Thong Bui
Wenpei Li
Winsey Chui

Chief Communications Officer:

Huy Thong Bui
htbhui@sfu.ca

Submitted to:

Dr. Andrew Rawicz - ENSC 440
Dr. Craig Scratchley - ENSC 405W
School of Engineering Science
Simon Fraser University

Published Date:

July 6, 2019
Version 1.0

Abstract

Continuous improvement in healthcare technology in the past several years [insert reference] has led to longer life expectancies in Canada. With this, comes the increasing need to provide health care services for the elderly and those unable to care for themselves. This places a strain on health care facilities and nursing homes that are unable to take so many patients, as well as causing irritation in patients who wish to continue independent living without constant supervision from caregivers.

With this in mind, LumoAnalytics introduces Pulse Tracer, a remote patient monitoring system that utilizes photoplethysmographic technology for the purpose of tracking heart and respiratory rates in elderly, immobile patients. This document specifies and defines the design specifications for Pulse Tracer, specifically outlining the three main subsystems:

- **Hardware System**
  The hardware system is responsible for remotely capturing data. This system contains near-infrared light sources and a camera for capturing the reflected waves in the form of videos, which are sent to a microcontroller.
- **Software System**
  The software system is responsible for converting raw data into results which the user can understand. Region of interests are automatically selected from the video input produced by the camera before going through processing and analysis to extract heart rate and respiratory rate data.
- **User Interface**
  The user interface displays real time and historical heart rate and respiratory rate data and allows patient and caregiver interaction, with different settings depending on the user.

The design of this device is split into three categories: proof-of-concept (PoC), prototype and final product, which will be clearly specified. For the purpose of this document, the proof-of-concept will be the main focus. Additional features will be integrated in future designs.

This document will begin with a brief system overview before explaining design requirements for each of the subsystems in further detail. In each of these subsections, the functionality of each component and their necessary design specifications will be discussed. The document will finish with appendices detailing the test plan for Pulse Tracer, as well as an evaluation of the design of its user interface.

Table of Contents

List of Tables

List of Figures

Glossary

PoC: proof-of-concept

PPG: Photoplethysmography, is a simple and low-cost optical technique that can be used to detect blood volume changes in the microvascular bed of tissue

NIR: Near-infrared, electromagnetic spectrum (from 780 nm to 2500 nm)

LED: light-emitting diode

Raspberry Pi: A low cost, credit-card sized computer

HDMI: High-Definition Multimedia Interface, supports the connection between a device such as a Blu-ray player or cable box and a flat-screen HDTV or projector

RR: Respiratory Rate

HR: Heart Rate

GPIO: General Purpose Input Output, is an uncommitted digital signal pin on an integrated circuit or electronic circuit board

PCB: Printed circuit board

Python: A programming language.

CSI: Camera Serial Interface, it defines an interface between a camera and a host processor

NoIR: No InfraRed

OpenCV: Open source computer vision, is a library of programming functions mainly aimed at real-time computer vision

NumPy: A library for the Python programming language

Dlib: A general purpose cross-platform software library written in the programming language C++

ROI: Region Of Interest

VM: Virtual Machine

Haar Cascade: A classifier which is used to detect the object for which it has been trained for, from the source

XML: Extensible Markup Language, is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable

RGB: Red-Green-Blue

YCbCr: It is a family of color spaces used as a part of the color image pipeline in video

HSI: (hue, saturation, lightness), Alternative representations of the RGB color model

ICA: Independent Component Analysis

JADE: Joint Approximate Diagonalization of Eigen-matrices, an algorithm for independent component analysis that separates observed mixed signals into latent source signals by exploiting fourth order moments

RIFV: Respiratory Induced Frequency Variation, is a periodic change in the pulse rate which is caused by an autonomous nervous system response

RIIV: Respiratory Induced Intensity Variation, is a change in the baseline signal that is caused by a variation of perfusion due to intra-thoracic pressure variations

RIAV: Respiratory Induced Amplitude Variation, is a change in pulse strength that is caused by a decrease in cardiac output due to reduced ventricular lling during inspiration

RDS: Relational Database Service, is a web service that makes it easier to set up, operate, and scale a relational database in the cloud

AWS: Amazon Web Services

MQTT: Message Queuing Telemetry Transport, is an ISO standard publish-subscribe-based messaging protocol

IoT: The Internet of things, is the extension of Internet connectivity into physical devices and everyday objects.

Django: A high-level Python Web framework that encourages rapid development and clean, pragmatic design

PostgreSQL: A general-purpose object-relational database management system

HTML: Hypertext Markup Language, is the standard markup language for documents designed to be displayed in a web browser

1 Introduction

1.1 Background

As healthcare technologies continue to improve, the life expectancy in Canada is on the rise [1]. However, with this increase comes the need to provide sufficient health care services for the aging population. This can be a challenge as health care facilities can often become crowded and understaffed, leaving the elderly on wait lists or to live at home with little to no regular supervision from caregivers.

LumoAnalytics hopes to use Pulse Tracer to provide patients the ability to live independently in their own homes while still ensuring a professional caregiver can monitor their health. Pulse Tracer accomplishes this with the use of photoplethysmography (PPG) to remotely monitor the heart and respiratory rates of patients [2]. Using this data, Pulse Tracer is able to determine baseline heart and respiratory rates and detect significant changes in these values. Professional caregivers are then able to view this data and identify if the patient may have any emerging health conditions, as well as to monitor patients living with conditions such as sleep apnea and arrhythmias [3]. Pulse Tracer can also be used to detect behavioural changes in patients, such as predicting when a patient is encountering periods of stress that could potentially lead to medical emergencies such as a heart attack [4].

1.2 Scope

This document will provide a system overview for Pulse Tracer and identify the design specifications of the device. It will detail the components involved in Pulse Tracer, and outline how the components work together to form the final product. The specifications will focus primarily on the proof of concept design for Pulse Tracer, but will also include details regarding the engineering prototype and final product. This document also includes an appendix relating to the design and usability of the user interface for pulse tracer, as well as a test plan for the functionality of the device.

1.3 Intended Audience

This document is intended to be reviewed by the members of LumoAnalytics, Dr. Craig Scratchley, Dr. Andrew Rawicz, the teaching assistants for ENSC 405W/440, and any potential customers and partners interested in the device.

1.4 Design Classification

The design classification used in this document will be as follows:
Des {section code}.{subsection code}.{design specification number}-{phase number}

The section and subsection codes map the design specification to its corresponding category, while the phase number indicates which prototype the design specification relates to. Table 1.1 below shows the prototypes in this project.

The proof of concept stage will be completed at the end of ENSC 405W, the engineering prototype will be completed in ENSC 440, and the final product represents the functions of the product once its in production.

Phase Number	Description
I	Proof of Concept
II	Engineering Prototype
III	Final Product

Table 1.1: Design Stages

1.5 System Overview

Pulse Tracer is remote patient monitoring device designed to monitor the heart and respiratory rates of immobile elderly patients. Through the use of near-infrared light, information is captured from the patient and subsequently processed by an analysis pipeline to determine heart and respiratory rates. The device can be divided into three major components: a hardware component for the collection of data, a software component for processing images and extracting necessary data, and an app for users to view the processed data. The following image shows the relationship between the components involved in Pulse Tracer.

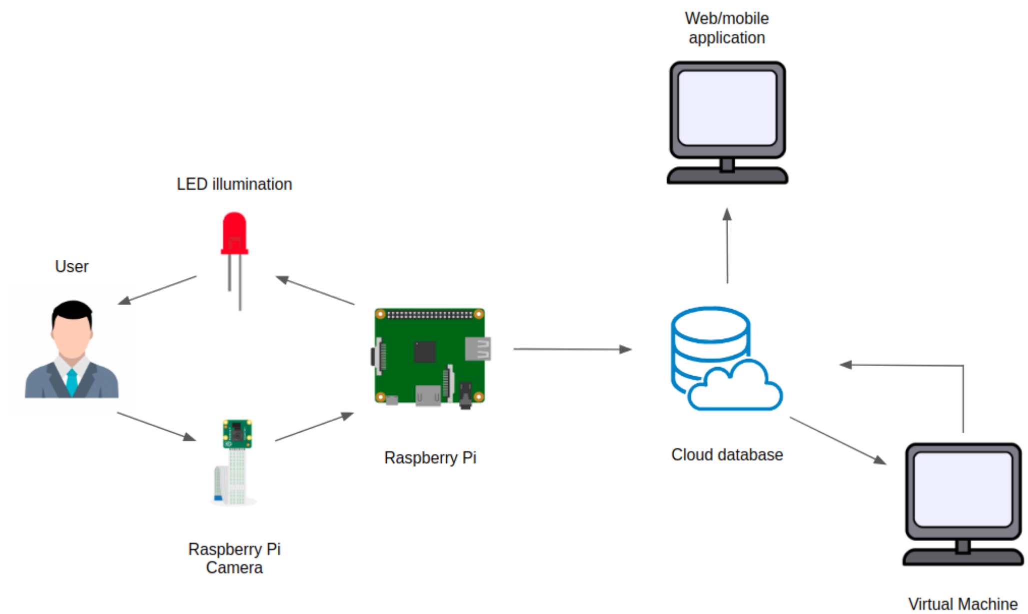


Figure 1.1: System overview for Pulse Tracer

The device works by shining near-infrared light on an area of the patients exposed skin. A camera connected to the microcontroller collects video footage of this event and an algorithm processes the stream in real time to extract a specified region of interest in each frame. An algorithm to compare the region of interest for each frame is then used to extract heart and respiratory rates from the data. These rates are then sent to a database via WiFi to be displayed on the web application. Users are able to turn Pulse Tracer on and off by using a switch on the device.

2 Hardware Design Specifications

The hardware component is the main physical presence of Pulse Tracer within the users day to day life, and plays the essential role of data collection. The focus of the proof-of-concept prototype is to use simple components in the form of a breadboard, LEDs and Raspberry Pi to ensure the functionality of our device by collecting clean, accurate signals, as seen in the figure below, while Phase II will begin to focus on increasing the metrics collected, the user interface and minimizing its spatial occupancy. The following sections will go into further details about the general design, optics, circuitry and microprocessor of Pulse Tracer.

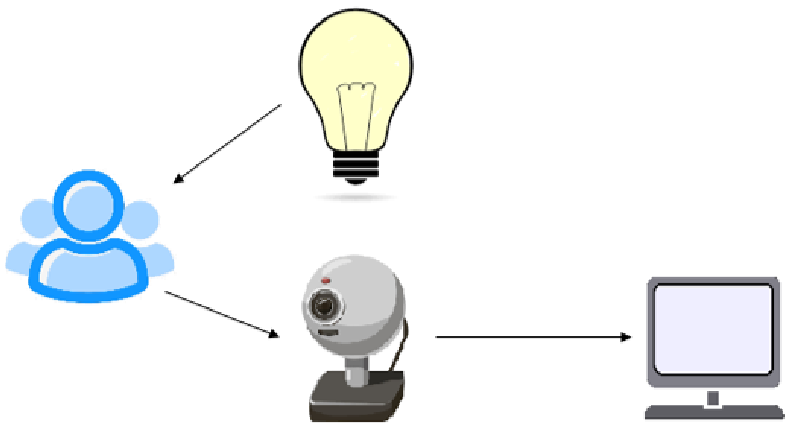


Figure 2.1: Conceptual Diagram of Pulse Tracer

2.1 General Design Specification

Given the concept diagram shown above which details the breakdown of the hardware portion of Pulse Tracer, it is important to briefly go over the general hardware design specifications. As noted above, Phase I of Pulse Tracer is defined by ensuring the functionality and accuracy of the device. Because of this, the device will not be enclosed and will be powered by using a wall outlet. Data will be viewed on a monitor attached directly to the Raspberry Pi.

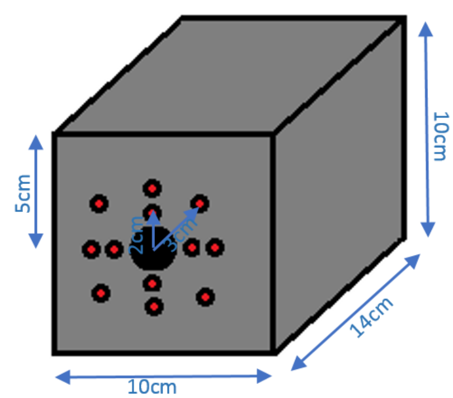


Figure 2.2: Pulse Tracer Phase II Enclosure

In comparison, during Phase II, Pulse Tracer shifts its power source from a wall outlet to a rechargeable battery, and is placed into a custom-build enclosed space that can then be relocated to the users desired location. At this point, the data will also no longer be viewed on a laptop, but on a web application that will be described in further detail in the software portion of this document.

Table 2.1 below summarizes the general hardware design specifications.

Design ID	Design Description
Des 2.1.1 - I	Data will be viewed on a monitor with HDMI input
Des 2.1.2 - I	LEDs will provide a light source
Des 2.1.3 - I	The LEDs will provide continuous illumination for continuous data collection.
Des 2.1.4 - I	Data will be captured by a Raspberry Pi Camera module
Des 2.1.5 - I	The Raspberry Pi will be connected to the monitor through an HDMI
Des 2.1.6 - II	Data will be viewed on a web application
Des 2.1.7 - II	The hardware component of Pulse Tracer will be in an enclosed container with dimensions at least 10x10x14cm
Des 2.1.8 - II	The enclosed container will have openings for the lens, light source, switch and charging station
Des 2.1.9 - II	There will be LEDs to indicate charging status
Des 2.1.10 - II	There will be a LED to indicate whether device is in operation
Des 2.1.11 - II	There will be a charging port
Des 2.1.12 - II	The enclosed container will be a switch
Des 2.1.13 - II	The charging port and switch will be placed on opposite sides of the enclosure with their respective LEDs for ease of use
Des 2.1.14 - II	The enclosed container should be an unobtrusive as possible
Des 2.1.15 - II	The enclosed container must completely hide the Raspberry Pi from view to ensure no tampering can be done
Des 2.1.16 - II	The enclosed container must be easy to set up for users of all ages and technical backgrounds

Table 2.1: General Hardware Design Specifications

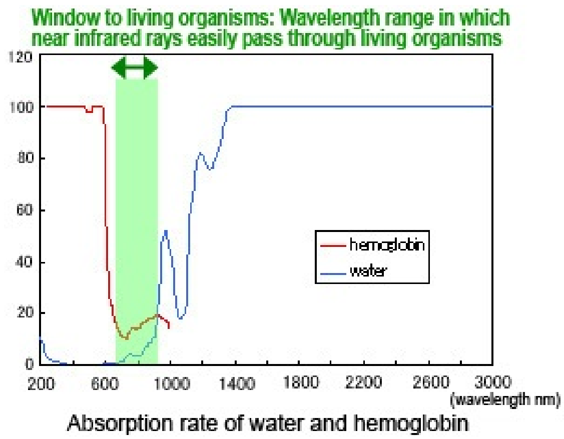
2.2 Optical Design Specifications

2.2.1 Blood Absorption Principles

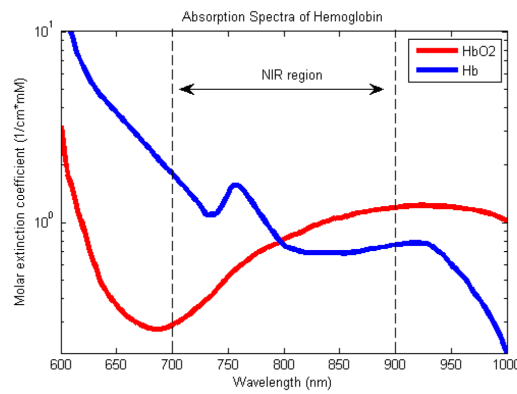
This section details the light source of Pulse Tracer the LEDs. According to principles of biophotonics used most commonly in pulse oximetry, light at wavelength of 700nm to 900nm is absorbed by hemoglobin within the blood, both oxygenated and deoxygenated, and is absorbed, scattered, and reflected [5]. At these wavelengths, the absorption of blood is much higher in comparison to water, as seen in Figure 2.3a. In the adjacent figure (Figure 2.3b), within these set of wavelengths, it is clear that there is a boundary between the absorption levels for deoxygenated and oxygenated blood, where oxygenated blood has a greater absorption from approximately 800nm to 900n, and deoxygenated blood on the lower end of the usable wavelengths.

Because the goal of Pulse Tracer is to unobtrusively gather data from patients in relation to their heart rate and respiratory rate, by taking all these factors into consideration, the hardware design team has settled on using a wavelength in the near-infrared range of 850nm. Light at 850nm is in the near-infrared range, and thus outside of the visible light spectrum, allowing for data collection even in the dark and without external interference from ambient light. This wavelength is the only light source used during Phase I of prototyping Pulse Tracer.

During Phase II, additional metrics are added to Pulse Tracer, including measurement of blood oxygenation levels. In order to do this, we include a wavelength at 790nm, just past the understood boundary of visible light, but still with a greater absorption in deoxygenated bloody [6].



(a) Absorption of Light in Human Tissue



(b) Hemoglobin Absorption Spectra

Figure 2.3: Absorption of Light in Human Tissue vs Hemoglobin Absorption Spectra

2.2.2 Illumination Configuration

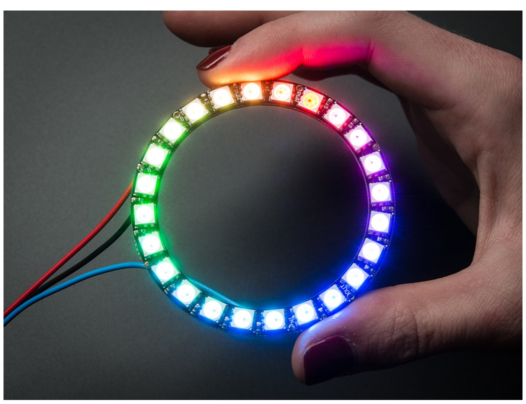


Figure 2.4: Example LED Ring [7]

While it is impossible to achieve completely even illumination due to the complexities of the human face, ensure the entire face is illuminated will allow for different selections of regions of interest. Because of this, the configuration of the LEDs in Pulse Tracer becomes important. The LEDs will be placed in a ring configuration (left) to ensure even illumination around the camera where it will be placed, and mostly even illumination, as recorded in [8].

Given that the average size of a human head is approximately 23.9cm [9], we have increased this value to 30cm, and thus a 15cm radius and a surface area of 707cm^2 for the purpose of our calculations. For Phase I of Pulse Tracer, we use a distance of 1m between the light source and subject for our calculations. Given that the size of the Raspberry Pi camera module around which the LEDs will be placed is 23mmx25mm [10], and a necessity for minimal overlap in light on the face due to safety concerns, during Phase I, the LED ring has a minimum radius of 5.24cm, as calculated in Figure 2.5 below. In additional to the first ring, which contains 4 LEDs placed at 5.25cm away from the center of the lens, the secondary ring contains 8 LEDs and is placed at a distance of 12.6cm from the 'origin'. This ensures minimal overlap in the center of the face, as seen in Figure 2.7. With these calculations, LED configuration for the proof-of-concept prototype is as seen in Figure 2.6a.

In the prototype phase of Pulse Tracer, we aim to minimize the physical size of the hardware device. In order to do this, we minimize the radius of the LED rings by placing all the LEDs at angles. By reducing the inner radius to 2cm, and the outer ring to 3cm, we can decrease pulse tracer to having a having a minimum radius of 3.5cm, given that the radius of the LED is 0.25cm[11]. The LED configuration for Phase II can be seen in Figure 2.6b.

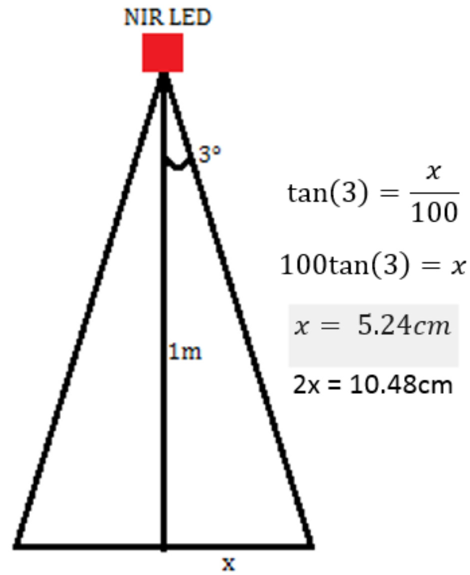
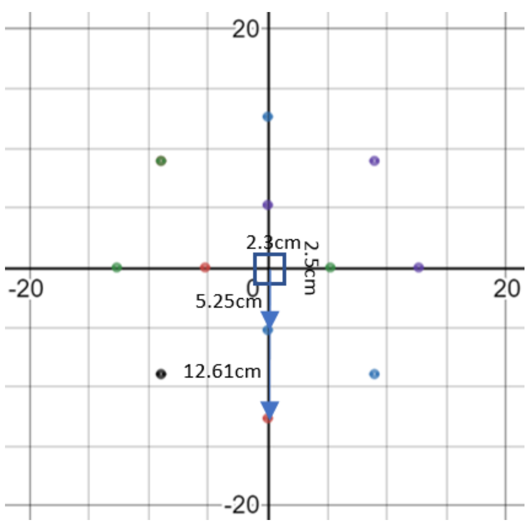
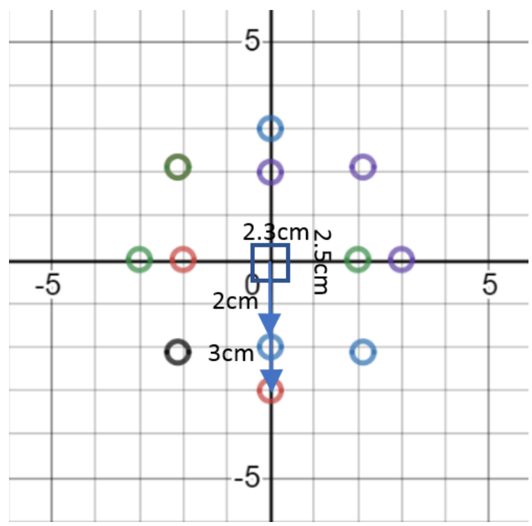


Figure 2.5: Calculation of Light Dispersion



(a) LED and camera lens placement in centimeters (Phase I)



(b) LED and camera lens placement in centimeters (Phase II)

Figure 2.6: LED and camera lens placement: phase I vs phase II

Both of these configurations will provide the illumination seen in Figure 2.7, where the smaller circles are the fields of illumination provided by each LED, and the large circle with radius 30cm is the upscaled size of an average human head.

The angles for the LED rings in Phase II can be calculated as shown in Figure 2.8 below.

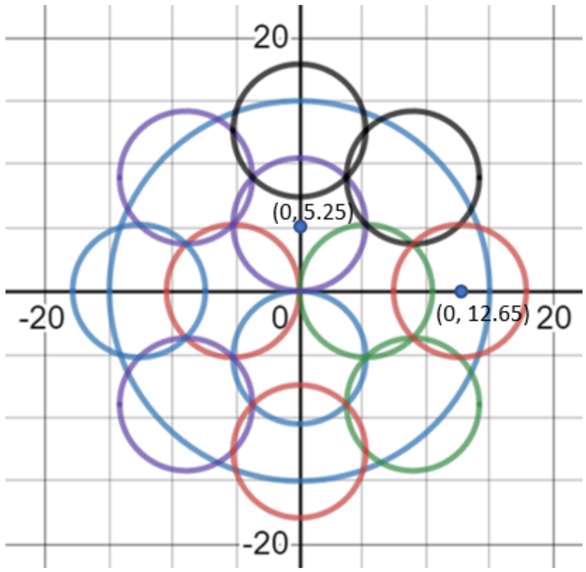


Figure 2.7: Illumination of LED rings on subject.

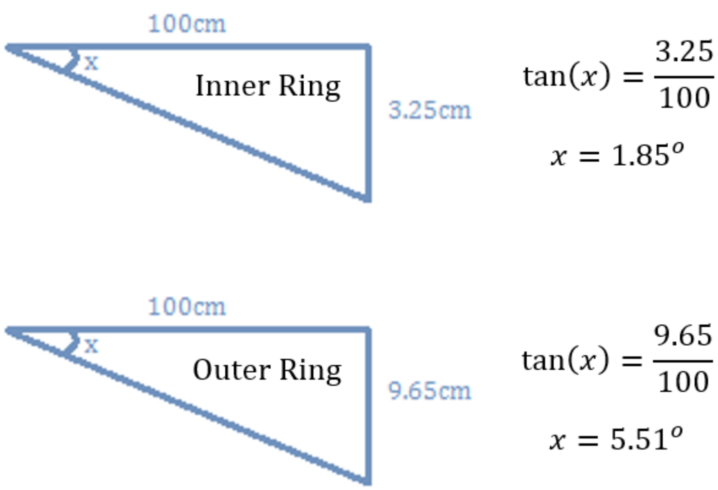


Figure 2.8: Illumination of LED rings on subject.

2.2.3 Power

As mentioned previously, the powering of the LEDs is important because of potential photoaging in the tissue caused by overexposure. Because of this, the maximum possible exposure for light at 850nm on skin is 0.1W/cm^2[12]. However, because data collection happens continuously, it is possible that light at this wavelength may hit the eye as well. Because of this, the irradiance used is the maximum permissible exposure for the cornea, which is approximately 1mW/cm^2 [13].

As shown in Figure 2.7 above, which details the illumination on the face from the LED ring, there is overlap in illumination in certain areas. Because of this, to avoid over illumination, we decrease the maximum permissible exposure to 0.5mW/cm^2. From this, we can calculate the irradiance desired at the output of the LEDs from a distance of 1cm.

$$\frac{I_1}{I_2} = \frac{d_2^2}{d_1^2}$$

$$\frac{I_1}{0.001W/cm^2} = \frac{(100cm)^2}{(1cm)^2}$$

$$I_1 = 10W/cm^2$$

Design ID	Design Description
Des 2.2.1 - I	LEDs operate at wavelengths of 850nm
Des 2.2.2 - I	LEDs operate in a ring configuration to illuminate the entire face
Des 2.2.3 - I	LEDs operate with an irradiance of 0.5mW/cm2 at a distance of 1m.
Des 2.2.4 - I	The ring configuration for the LEDs will have a minimum radius of 16cm.
Des 2.2.5 - I	The ring configuration will provide enough illumination to collect data for HR and RR extraction.
Des 2.2.6 - I	The ring configuration must be placed at the same distance from the subject as the Raspberry Pi Camera module.
Des 2.2.7 - II	LEDs of wavelengths 850nm and 780nm will be used
Des 2.2.8 - II	The ring configuration will have an inner radius of 1.5cm and outer radius of 3.5cm.

Table 2.2: Optics Design Specifications

2.3 Circuitry

In this section, we will demonstrate the functionalities of each sub-system and the correlation between them. The general circuitry of the proof-of-concept Pulse Tracer prototype was created as shown in Figure 2.3.1. The hardware system is based on the microcontroller. In this particular design, a Raspberry Pi 3B+ provides the most important platform for generating near infrared illuminance, transmitting data, and performing calculation.

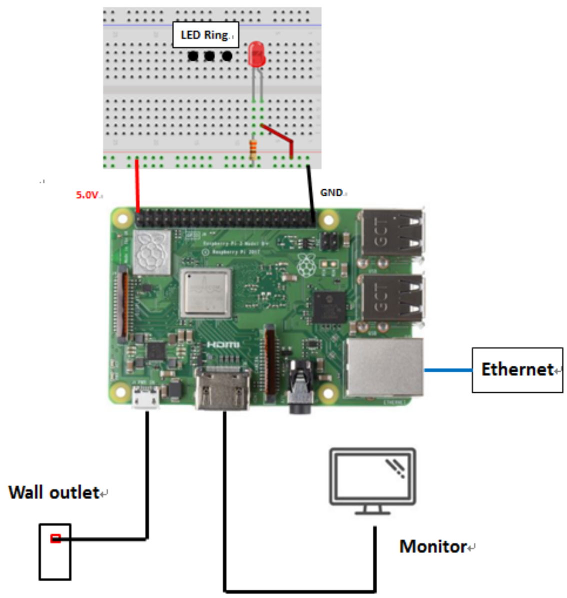


Figure 2.9: The general circuitry of the proof-of-concept prototype

The overall circuitry can be divided into four components. It includes power supply system, light emitting system, network system, as well as image capture and display system. In addition, a protection circuit will also be added to ensure the safety of the users.

2.3.1 Power Supply System

The power supply system consists of a power plug socket and an 110V/5V transformer. The Raspberry Pi can obtain the power by simply connecting the socket to the wall outlet. In phase II, we will promote this stationary power supply system to a rechargeable and portable lithium-ion battery. An 1820 mAh lithium-ion battery provides 4-6 hours charge, and the electricity can support the Raspberry Pi operating for 24 hours or even longer.[14]

The Pulse Tracer will keep running for at least 72 hours, so the battery should maintain its power during this 72 hours period. According to the data collected by other Raspberry Pi researchers, we need a minimum 6000 mAh battery to support the board. Since the pi is calculating and transmitting data all the time, both the workload of the microprocessor and the energy consumption will be higher than our expectation.

2.3.2 NIR Light Emitting System

According to section 2.2, a LED ring circuitry with 12 SFH4550 LEDs that can generate over 5.25cm radius of near infrared illuminance is required in the proof-of-concept prototype design. The rated power of SFH4550 LED is 180mW, however, we will power it to 70mW to avoid over exposure. The figure only shows four LEDs for a better illustration.[15]

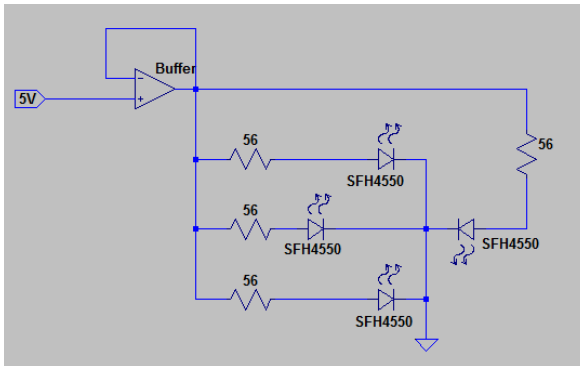


Figure 2.10: The NIR Illuminance Generation Circuit

The data sheet of SFH4550 NIR light emitting diode also provides the information we need for calculating the power. In addition, in proof-of-concept phase, the 5V voltage input is provided by GPIO pin 2 of the Raspberry Pi, and the ground terminal of the circuitry is connected to GPIO pin 39. The GPIO Pinout will be presented later. The buffer is used for maximizing the input impedance and minimizing the output impedance, it is an essential part in phase II design.

The LED ring circuitry will be built on a bread board firstly because of its convenience for testing. After that, a customized PCB will be invented to fit for the Raspberry Pi 3B+. A customized PCB will have a LED ring circuit on it, and it will be powered by rechargeable and portable batteries instead. The PCB and the Raspberry Pi board will be soldered together at the end of the phase II design procedure.

2.3.3 The Operating and Controlled System

The microcontroller used for Pulse Tracer is a Raspberry Pi 3+B due to the ease of image processing in Python. In relation to the hardware, the Raspberry Pi must contain an input for the camera module, as well as at least a singular analog output channel, which has an operating voltage of 5V to power the LEDs.

In the proof-of-concept phase of Pulse Tracer, the microcontroller is turned on and off through a switch included in the purchasing of the Raspberry Pi, and is powered by connecting with a wall socket. However, in the prototype phase, power sources switch from a wall socket to a rechargeable battery. For ease of use, the battery is stored within the enclosure, and can be recharged by attaching a USB cable, similar to a mobile phone, with an output voltage of 5V. A detailed layout of the Raspberry Pi and its GPIO pins are shown in the figures below

Currently, we only used the GPIO pins to generate power and provide ground terminal, but more pins might be used depends on the software requirement in the future.

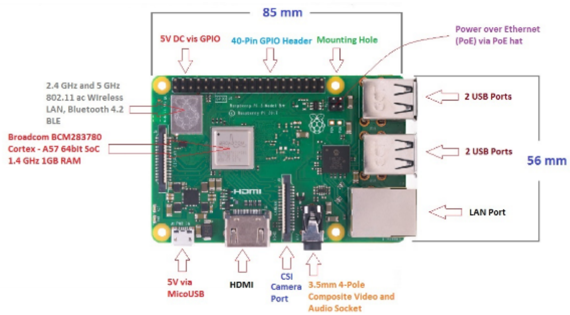


Figure 2.11: The Raspberry Pi Layout

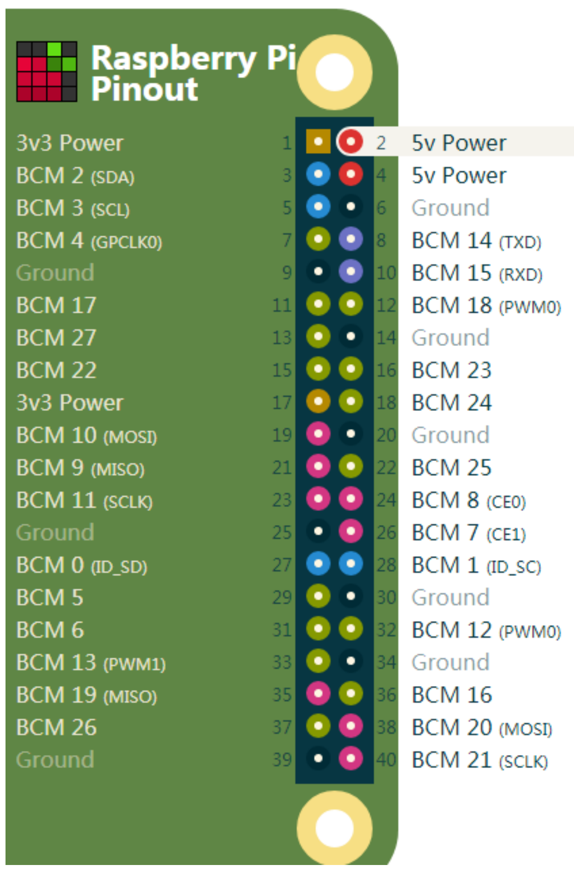


Figure 2.12: The Raspberry Pi Pinout [16]

2.3.4 Image Capture System

To capture the images, we will use a NoIR Raspberry Pi camera as shown in the figure below. This camera module is designed for Raspberry Pi that does not have an IR cut filter. There is a connection cable attached in the camera, the other side of the cable could be connected the Raspberry Pi easily. The camera uses CSI interface, which has the ability of handling extremely high data rates, and it exclusively carries pixel data.

In order to make a video to collect data, we have to investigate the camera environment of the Raspberry Pi. The overall procedure includes downloading software, activating the requirements of OpenCV, enabling the camera function in the boost setting, and running a simple Python script.

Figure 2.13: Raspberry Pi Camera Board 5MP V1.3 [17]

2.3.5 Display System

As we mentioned previously, the Raspberry Pi should be connected to a monitor. It can be obtained by either a laptop or a computer. A cable will be connected to the HDMI port of the monitor to display any necessary information we need to see from the Raspberry Pi such as the source codes and raw videos.

The monitor is only used during the development process. The actual product itself will not contain the monitor. At the end of Phase II, we can access data though a phone app instead of the monitor.

2.3.6 Network System

An Ethernet cable that allows the Raspberry Pi to access the Internet is expected in the proof-of-concept design process. So the OpenCV can be downloaded and all necessary libraries can be installed and updated through the Internet. The processed data of the images can be transmitted to the database further more. In phase II, the Ethernet cable will be replaced by the wireless network (Wi-Fi). The promotion in the network environment is expected to be done so that the Pulse Tracer can be completely controlled by the user remotely.

2.3.7 Power Surge Protection System

There are many risks we need to consider about. The most important one is how to handle the power surge that might occur in the hardware system. As a result, a protection circuit must be invented at the end of phase II. This circuit behaves like an automatic switch, it can shut down the whole system once the current exceed the threshold.

There are many kinds of protection circuit can be applied to our system but we only need focus on two of them. The first one is to avoid the over charge of the battery, the second one is to deal with the short circuit that might appears in the light emitting system. Both of them will be considered as an essential part of the hardware design in phase II.

2.3.8 Maintenance System

The maintenance system is simply done by applying a mechanical key to the whole system. The development group has the priority to open the device and fix the problems by using this key. This system will not be applied until phase III when the 3D printed box is created.

Design ID	Design Description
Des 2.3.1 - I	The Raspberry Pi must be powered by wall outlet
Des 2.3.2 - I	A breadboard level LED ring circuit that can produce a reasonable illumination must be designed
Des 2.3.3 - I	Th LED ring circuit must be powered and grounded by GPIOs of the Raspberry Pi
Des 2.3.4 - I	The camera must work with the Raspberry Pi properly and all function should be activated
Des 2.3.5 - I	There must be a monitor to display all information we need from the Raspberry Pi
Des 2.3.6 - I	There must be a Ethernet cable connected to the Raspberry Pi
Des 2.3.7 - II	A minimum 6000 mAh Lithum battery must be provided to support the Raspberry Pi
Des 2.3.8 - II	A customized PCB must be invented to replace the breadboard
Des 2.3.9 - II	The Raspberry Pi should be able to be connected to the Wi-Fi and all procedures should be able to be done remotely
Des 2.3.10 - II	An overcurrent protection circuit must be invented to avoid the short circuit
Des 2.3.11 - II	An over charge protection circuit must be invented to avoid the over charge of the battery
Des 2.3.12 - III	A mechanical key must be designed for the maintenance purpose

Table 2.3: Circuitry Design Specifications

3 Software Design Specifications

The software component of Pulse Tracer will include three major sub-components, namely image processing, data extraction, and a user interface application. The image processing component will be responsible for preparing the data to be used by the extraction algorithms. This will include facial detection of the patient and automatically selecting regions of interest for each image frame captured by the hardware system. The data extraction component will involve algorithms for determining heart and respiratory rates of the patient, as well as any future additional metrics. Finally, the user interface application will provide the users a way of interacting the data. This will include a database to hold all patient data, as well as data transmission from Pulse Tracer devices to the database. The design of the user interface application will be evaluated in Appendix A of this document.

3.1 General Software Design Specifications

All software components for Pulse Tracer will be written primarily in Python. Using this language will allow for rapid development of Pulse Tracer software by making use of existing Python libraries such as NumPy, OpenCV, and dlib. The software will also be easier to read when compared to other languages such as Java and C/C++, allowing future developers to more easily modify the software.

Figure 3.1 below shows how the software components of Pulse Tracer will be connected to form a single pipeline. Due to the limited processing power of the Raspberry Pi, the image processing and data transmission stages of the pipeline will be the only components that perform on the device. After the ROI data is sent to the cloud database, the remaining stages will be performed on a virtual machine (VM) that accesses the database.

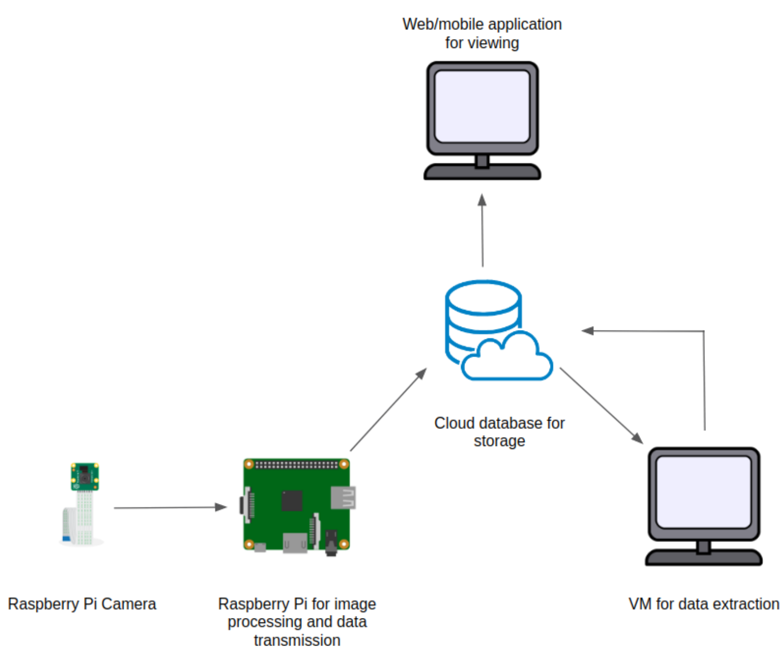


Figure 3.1: Data analysis pipeline

The first stage of the pipeline will be image processing, where a script will read the video stream from the Raspberry Pi and perform facial detection to identify regions of interest. After this, the Raspberry Pi will send the identified regions of interest for each frame to be stored on the cloud database. Next, a VM will use the ROI to extract heart rate information, and subsequently the respiratory rate. Finally, the heart rate and respiratory rates will be stored in the database and then posted to the web application for the users to view.

The following table outlines the general design specifications for all software used in Pulse Tracer.

Design ID	Design Description
Des 3.1.1 - I	The software components shall be written in Python
Des 3.1.2 - I	The version of Python used shall be 3.5 or greater
Des 3.1.3 - I	All software components involved in Pulse Tracer shall be organized into a single pipeline
Des 3.1.4 - I	The pipeline shall include the reading, analysis, and posting of data from the input camera
Des 3.1.5 - I	The only processes performed on the Raspberry Pi shall be the processing of raw image data and transmission of selected regions of interest to the cloud database
Des 3.1.6 - II	The extraction of heart and respiratory rates from transmitted regions of interest shall take place on a remote machine

Table 3.1: General Software Design Specifications

3.2 Image Processing

Determining the heart and respiratory rates of the patient requires video footage of only a small portion of the patient's exposed skin to be compared over time [2]. Pulse Tracer targets regions of exposed skin on the patient's face, specifically larger areas such as the cheeks, forehead, and nose. In order to identify these regions of interest, the Haar Cascades classifier is used for facial detection. This algorithm is based on the principle of training a classifier using positive and negative images. In this case, the positive images will be those that include a face, whereas negative images will not contain a face. Haar features are then used to extract objects from the positive images. The following figure shows an example of Haar features.

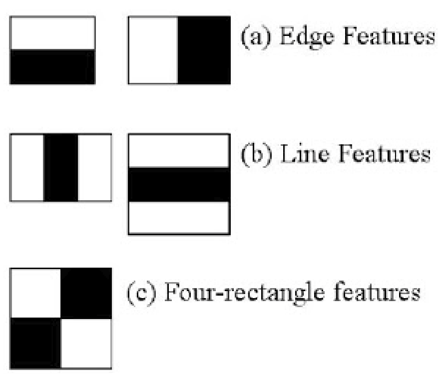


Figure 3.2: Haar feature examples [18]

These features are used as a convolution kernel and are multiplied and subsequently added with the image at every pixel location. This introduces a large number of calculations to be performed, potentially slowing down the algorithm. Therefore, the Haar Cascade classifier uses an integral image to increase the speed of the calculations. The integral image is a summed-area table, which can be thought of as an intermediate representation of the image. It is used to quickly calculate the sum of all pixel intensities in a specified region, namely the region covered by the Haar feature at each pixel location.

The Haar features used in the classifier are also varied in size, increasing the total number of possible features. For example, a 24x24 feature window will produce a set of more than 180,000 features [19]. However, not all of these features are relevant. The figure below demonstrates the use of these features in an image. The vertical line feature is able to accurately identify the nose in the image, whereas the horizontal feature can identify the eyes. However, if the two windows are applied to the cheeks where the there is a constant pixel intensity throughout, they would be irrelevant. For this reason, the classifier uses Adaboost to eliminate irrelevant features and therefore improve efficiency of the algorithm.

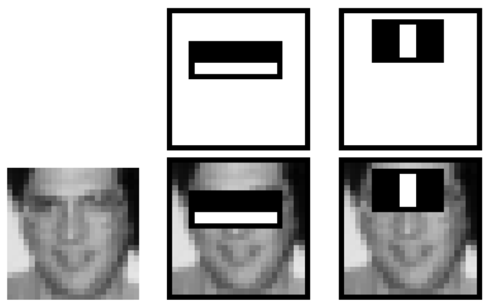


Figure 3.3: Demonstration of Haar features on an image [19]

OpenCV is used to implement the Haar Cascade classifier in this application. The Python library comes with pre-trained classifiers for the face, which are used by downloading the XML files and including them as the classifier. Pulse Tracer uses a combination of three pre-trained classifiers for frontal face detection. This includes the default frontal face detector, followed by the two alternative frontal face detectors if the default returns no result.

During Phase 2, the above algorithm will be optimized for speed and detection accuracy, which may include replacing the Haar Cascade Classifier with a newer and faster facial detection algorithm. This may require training a new model instead of using a pre-trained classifier.

Once the face has been detected in the image, Pulse Tracer then identifies facial landmarks using the pre-trained facial landmark detector from Python's `dlib` library. This detector takes the coordinates of the box bounding the detected face and estimates the location of 68 x-y coordinates which map to the facial landmarks. This detector can identify the mouth, left and right eyebrows, left and right eyes, the nose, and the jaw. The figure below shows a mapping of the 68 x-y coordinate to their corresponding landmarks.

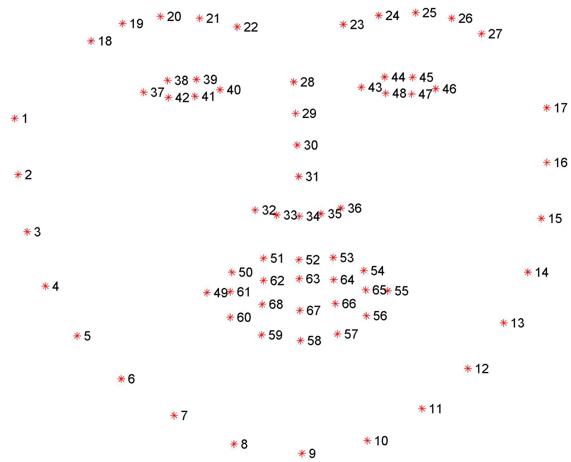


Figure 3.4: Map of 68 coordinates to facial landmarks [20]

After identifying the coordinates of the facial landmarks, regions of interest for the forehead, nose, and cheeks are selected. This is accomplished by creating a square around the detected coordinate for each landmark. The size of the square is dependent on the size of the face detected, and each should be approximately 0.5% of the total area of the face. The small size of the ROIs will allow the algorithm to consider multiple regions of the face when extracting the heart and respiratory rates. The algorithm was tested on two videos collected with faces at different distances from the camera. The following figures show the ROIs identified by the algorithm.

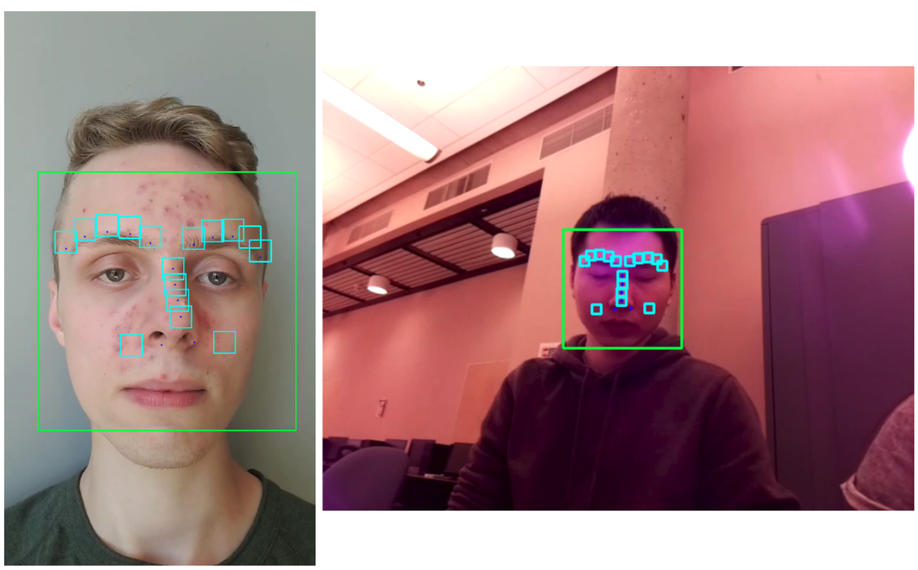


Figure 3.5: ROI as detected by the algorithm

The design specifications for the image processing involved in Pulse Tracer are detailed in the following table.

Design ID	Design Description
Des 3.2.1 - I	The ROI selection shall detect region of interest on the patient's nose, forehead, and cheeks
Des 3.2.2 - I	For each frame where a face is detected, the image processing algorithm shall produce at least two viable regions of interest to be used by downstream analysis
Des 3.2.3 - I	The selected regions of interest shall each have an area that is approximately 0.5-1.0% of the total area of the detected face
Des 3.2.4 - I	The facial detection algorithm shall accurately detect faces at various distances from the camera
Des 3.2.5 - II	The image processing algorithm shall be able to process the video stream as close to real time as possible
Des 3.2.6 - II	The facial detection algorithm shall accurately detect faces that include obstructions such as facial hair, eye glasses, and headwear
Des 3.2.7 - II	The facial detection algorithm shall accurately detect faces in varying levels of illumination
Des 3.2.8 - II	The image processing algorithm shall raise an alert if no face is detected for more than 2 seconds
Des 3.2.9 - II	The facial detection algorithm shall work for both frontal views and profile views of the face
Des 3.2.10 - II	The facial detection algorithm shall be optimized to improve speed and accuracy

Table 3.2: Image Processing Design Specifications

3.3 Data Extraction

3.3.1 Heart Rate Data

Pulse Tracer makes use of remote photoplethysmography (PPG) to monitor vital signs from the incoming video stream. PPG itself is an optical technique for measuring changes in blood volume in peripheral vasculature [21]. The principle behind remote PPG is that hemoglobin in the blood absorb light in a way that is proportional to the hemoglobin concentration in the blood. Because hemoglobin concentration varies with the change in blood volume in a vessel, the absorption of light can be used to remotely identify the pulse [22]. This persists as a slight colour change on the surface of the skin which is invisible to the human eye, but can be detected by a camera.

The change in colour of the patient's skin corresponds to a change in the intensity of the pixel values detected by the camera. Therefore, the pixel intensity captured by a digital camera can be decomposed into the illumination intensity and the reflectance of the skin [23] to produce a PPG signal. There are several proposed approaches to relate the pixel value detected by the camera to the corresponding PPG waveform.

The first step is to decide which colour channel to select for the algorithm. As with most digital cameras, the Raspberry Pi NoIR camera stores digital images according to the Red-Green-blue (RGB) colour model. In this model, each image is represented by three m-by-n matrices. Each of these matrices corresponds to one colour, either red, green, or blue. Although there are other colour models that could represent the images, namely YCbCr and HSI [24], the RGB colour model will be used in this algorithm because it is the most common model and the default for storing digital images [25]. Figure 3.6 below shows a visual representation of the RGB colour model.

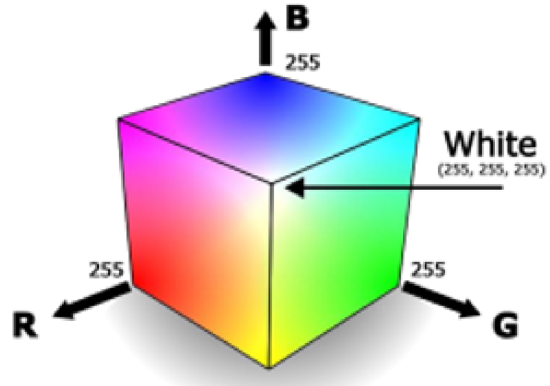


Figure 3.6: Red-Green-Blue model [24]

Because Pulse Tracer is intended for use during the day and night, both ambient and NIR light will be used as light sources. When ambient light is used for illumination, the image will appear to be normal and will be in full colour. However, when NIR light is used for illumination, ambient light will be blocked out. This will result in a gray-scale representation of the image. Although the green channel has historically been used for remote PPG image processing [26], preliminary tests have shown the red channel is most sensitive to changes in pixel intensity when using NIR. For this reason, Pulse Tracer will use the red colour channel to extract the heart rate.

Figure 3.7 below demonstrates the extraction of the raw RGB traces from a video.

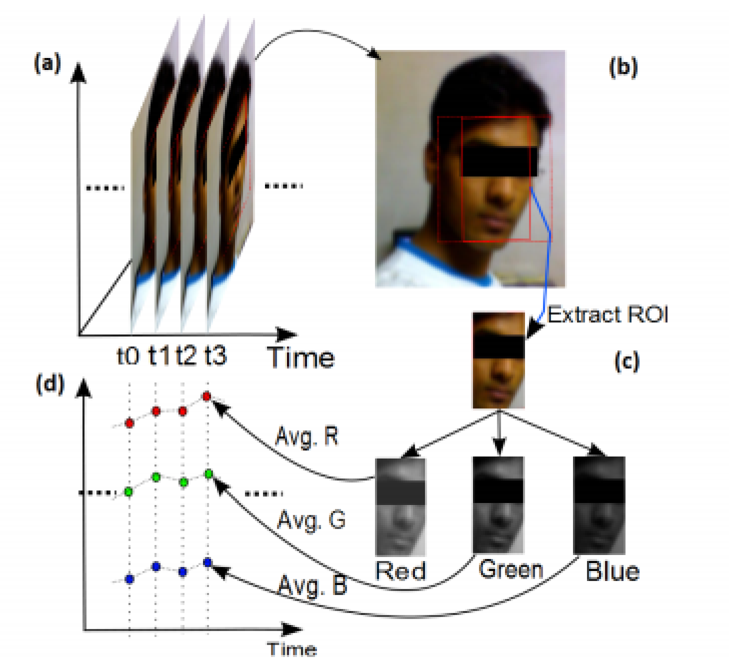


Figure 3.7: Extraction of raw RGB trace from a video sequence [27]

After extracting the ROI from the video frame, the mean of the pixels in this ROI will be calculated across each colour channel. This will provide three signals, xR(t), xG(t), and xB(t), corresponding to the average of the red, green, and blue pixels as a function of time, respectively. After calculating the average, the acquired signal will be represented as x(t) = (xR(t), xG(t), XB(t)). However, represents the observed signal, which is a combination of the source signal, s(t), and noise. Therefore, the observed signal for each of the three-colour traces needs to be separated into a source signal, s(t), and a noise signal, n(t).

There are several methods to extract the independent source signals from the observed signals, such as Green, CHROM, and ICA. Pulse Tracer will utilize Independent Component Analysis (ICA) however, as it is most widely used and has a relatively higher accuracy when compared to other methods [28]. Pulse Tracer will specifically make use of the Joint Approximate Diagonalization of Eigen-matrices (JADE) algorithm, as it is commonly used for heart rate detection due to its numerical efficiency.

The decomposed source signal will then be filtered with a band pass filter, only selecting frequencies between 0.7 and 4 Hz. This range corresponds to the physiological heart rate range of 45 to 240 bpm. At this point, the PPG signal will be analyzed by an additional algorithm to extract the respiratory rate. To further analyze the heart rate however, the filtered source signal will be converted to the frequency domain. In this domain, the frequency with the range corresponding to the peak with the highest magnitude will be considered as the heart rate value [29].

The frequency domain will also keep track of historical data of pulse frequency to reject noise by fixing a threshold for maximum fluctuation in pulse rate between consecutive measurements. If the change between the current pulse rate and the previous acquired value exceeds a specific threshold, the algorithm will reject the current pulse rate and search the frequency range for the peak with the next highest magnitude that meets this restriction. If there is no value that satisfies this criteria, the algorithm will keep the last computed pulse frequency [27]

Figure 3.8 below outlines the steps involved in extracting the heart rate from the video input.

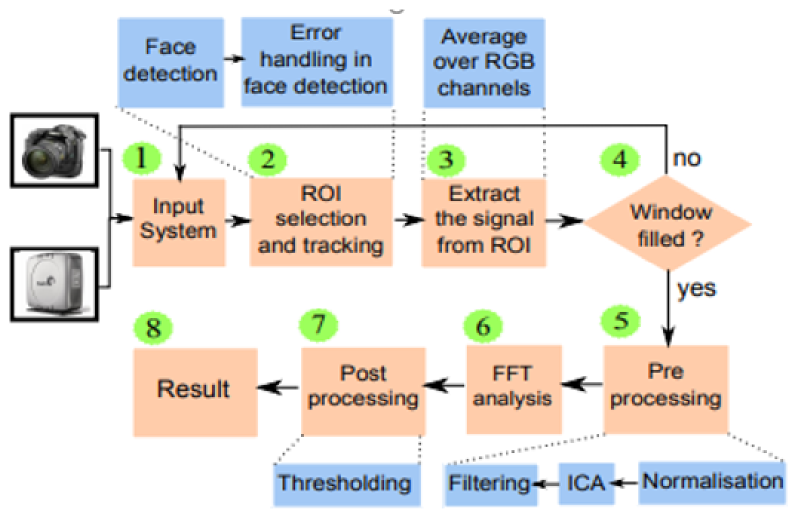


Figure 3.8: Summary of Heart Rate Extraction process [27]

??? The performance of the techniques for extracting the heart rate from the PPG signal depends on the length of the PPG signal used for computation. As a result, the majority of methods in theory are assessed over a period of at least 30 seconds [30]. Therefore, Pulse Tracer will analyze the video received from the input camera in chunks 30 seconds in duration. The processed signal will be normalized across a 30 second time range window

The performance of the PPG techniques for extracting heart rate also depends on the data length of the PPG signal which is used for computing heart beats; as a result, the majority of methods in theory are assessed over the segment of minimum 30 s duration [30]. For our design, the processed signal will be normalized across a 30 second time range window with a 1 second range; thus, we expect to received the heart rate for every 1 second. ???

Design ID	Design Description
Des 3.3.1 - I	The heart rate will be computed based on the intensity variations of of pixels of 1 region of interest(ROI).
Des 3.3.2 - I	Red-Green-Blue (RGB) is the color method which is applied for analysis thanks to its robustness
Des 3.3.3 - I	The heart rate extraction algorithm will calculate the average of the pixels in ROI across RGB color channel; for near-infrared illumination, only green channel is used.
Des 3.3.4 - I	The processed signals will be normalized to acquire heart rate for every second.
Des 3.3.5 - I	The RBG processed signals with be separated using Independent Component Analysis method
Des 3.3.6 - I	The isolated signal will be convert to frequency domain using Fast Fourier Transform.
Des 3.3.7 - I	The heart rate will be extracted from power spectrum with band frequencies from 0.7 to 4 Hz corresponding with 45 to 240 beats/minutes.
Des 3.3.8 - II	Noise reduction by applying threshold.
Des 3.3.9 - II	Heart rate will be calculating from multiple ROI and find the average.
Des 3.3.10 - II	Heart rate calculation algorithm will consider various situation such as skin colors, movements.

Table 3.3: Heart Rate Extraction

3.3.2 Respiratory Rate Data

The respiratory rate of a patient can be extracted from PPG data utilizing a number of different algorithms. These algorithms vary in their effectiveness and complexity but all rely on the fundamental way the respiratory rate affects the PPG data, specifically how breathing affects a person's heart rate. There are three main effects that respiration has on the heart rate. Respiratory Induced Frequency Variation (RIFV) is a periodic change in the pulse rate which is caused by an autonomous nervous system response [31]. Respiratory Induced Intensity Variation (RIIV) is a change in the baseline signal that is caused by a variation of perfusion due to intra-thoracic pressure variations [32]. Finally, Respiratory Induced Amplitude Variation (RIAV) is a change in pulse strength that is caused by a decrease in cardiac output due to reduced ventricular filling during inspiration [33]. These three effects can be seen affecting a sample PPG signal in the figure below.

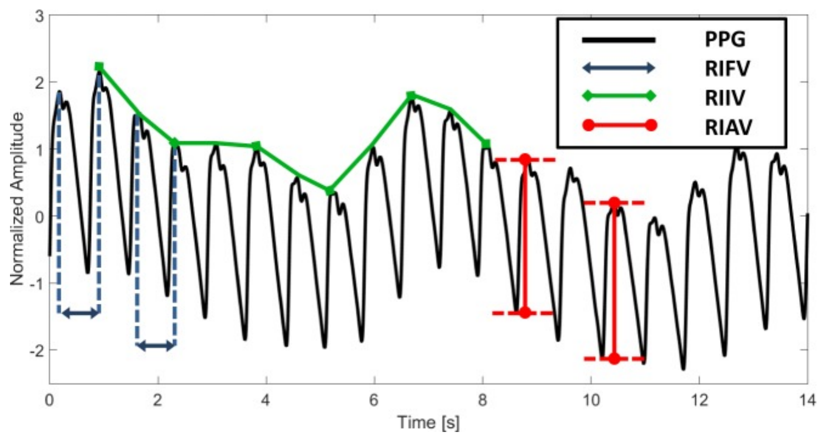


Figure 3.9: Effect on PPG signal of Respiration [31]

Since there is fundamental respiratory data within the PPG signal, various algorithms can be used to extract it. For this project, some different algorithms will need to be investigated but the general structure is as follows [34, 32, 33, 35, 36, 37]. The first step is to extract the respiratory signal from the PPG signal. In order to achieve this, low frequencies need to be eliminated using a high-pass filter with a cutoff frequency around 4 breaths per minute or 0.067 Hz. The next step can follow one of two paths, either using a filter or feature based extraction method.

Filtering methods can be as simple as just a band pass filter between 4 and 60 breaths per minute or between 0.067 and 1 Hz. However, a more accurate method would be to take the maximum amplitude of the continuous wavelet transform within reasonable heart rate frequencies between 30-220 beats per minute [34]. This filters out the cardiac signal leaving the portion of the signal related to respiration behind.

A feature based method analyzes the shape of the waveform looking for differences caused by the aforementioned RIFV, RIIV and RIAV [31]. The signal is first filtered to remove high frequencies above about 35 Hz. Then a combination of three analyses can be done. One method is to calculate the difference between the amplitudes of troughs and peaks [34]. Another is to compare the time intervals between peaks [36]. Then the last is to compare the mean signal value between consecutive troughs [37]. The described methods will be implemented and compared to find out which offers the most accurate respiratory signal using the system. After the signal is extracted it looks like the following example from literature, which follows closely to an ECG.

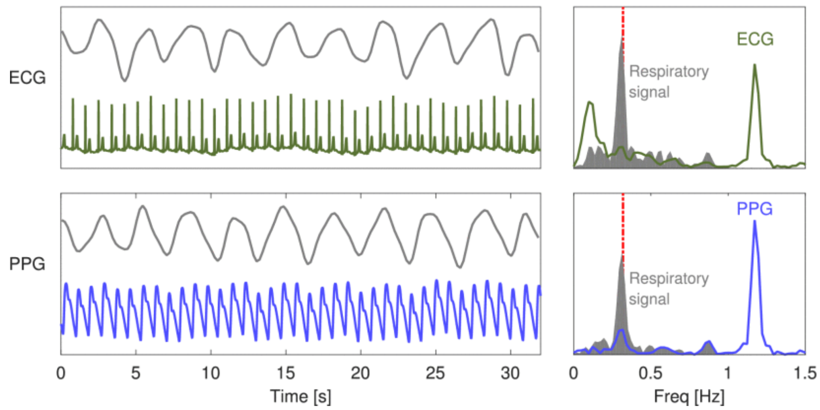


Figure 3.10: Extracted Respiratory Signal [33]

Once the respiratory signal has been extracted, the respiratory rate can be estimated. The most simple method is to detect the peaks in the extracted signal to count how many occur in a certain period of time and then generate the breaths per minute from that count [34]. A more accurate method filters the peaks and troughs in the signal by eliminating peaks less than the mean and troughs greater than the mean as well as eliminating peaks or troughs that occur closely together [34]. The algorithm will be tweaked to find the peaks and troughs most accurately representing the respiratory rate of the patient.

A final but optional step can be undertaken to improve the accuracy of the respiratory rate estimates. In literature it is called the fusion of respiratory rates and it refers to methods that combine the results from multiple respiratory rate estimation algorithms to output a more accurate result [34]. A smart fusion of respiratory rates can be done where their quality is assessed by means of standard deviation analysis. Also, temporal smoothing can be performed on the estimated respiratory rates to give an aggregated result. Combining the three main stages of this algorithm (signal extraction, rate estimation and fusion) will allow the respiratory rate of the patient to be determined from the PPG data.

3.3.3 Additional Metrics

Heart rate variability (HRV) refers to the variation in the time interval between heartbeats, measured by the variation in the beat-to-beat interval. It has been shown that reduced HRV is a predictor of mortality after myocardial infarction [38]. Reduced HRV is also associated with congestive heart failure, diabetic neuropathy, post-cardiac-transplant depression and poor survival in premature babies [39]. Therefore, detecting HRV would be beneficial. Using remote PPG the HRV can be determined by examining the spectrogram however a more accurate and simple method is to perform peak detection on the PPG signal [40]. One such algorithm for HRV can be seen in the below figure. The ability to detect HRV can be implemented in future revisions of the software however there are challenges and potential hurdles to overcome, such as the HRV requiring a more samples or more frames per second in order to be accurately determined.

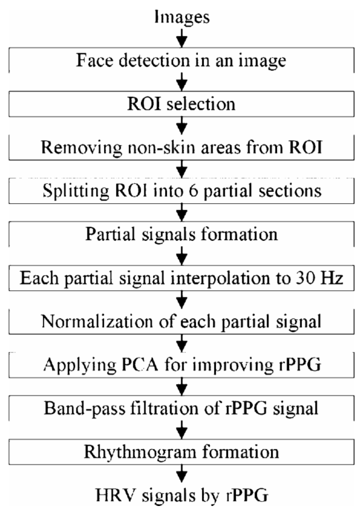


Figure 3.11: HRV Algorithm from Remote PPG [40]

Blood oxygen saturation is a common health metric indicative of the amount of oxygen the patient is metabolizing. Pulse oximetry is one common method for determining the amount of oxygenated hemoglobin in the blood and it helps warn clinicians of patients with hypoxemia (low levels of oxygen) which can lead to serious health complications [41]. Pulse oximetry works by illuminating the skin in close contact with two wavelengths of light at 660 nm and 940 nm [41]. The two wavelengths are chosen as they have different absorption by oxygenated and deoxygenated hemoglobin as shown in the figure below.

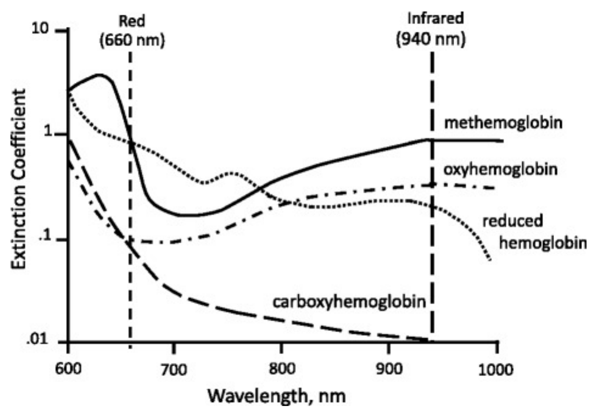


Figure 3.12: Light Absorbance Spectra of Hemoglobin [41]

The standard method of pulse oximetry usually requires contact with a patient which can be uncomfortable and easy to fall off especially with elderly patients. Remote PPG is capable of measuring the blood oxygenation from a distance, avoiding these issues. Using a NIR light source or ambient light can provide the illumination to detect changes in oxygenation. Two different color channels such as red and blue can be chosen from the camera and compared using the equation in the below figure [42]. Using this equation, the oxygen saturation can be determined after some filtering is done to the PPG signal for both the red and blue color channels [42]. This metric can be added into the software during Phase 2 of the Pulse Tracer.

$$SpO2 = A - B\frac{(Iac/Idc)\lambda\text{red}}{(Iac/Idc)\lambda\text{infrared}}$$

Figure 3.13: Oxygen Saturation Equation [42]

Design ID	Design Description
Des 3.3.1 - I	The respiratory rate algorithm shall filter out low and high frequency signals from the PPG signal that are not associated with respiration
Des 3.3.1 - I	The respiratory rate algorithm shall detect features such as RIFV, RIIV and RIAV in the PPG signal
Des 3.3.1 - I	The respiratory rate algorithm shall estimate the respiratory rate from the extracted respiratory signal
Des 3.3.1 - II	The respiratory rate algorithm shall combine results from multiple algorithms to get a more accurate respiratory rate
Des 3.3.1 - II	Heart rate variability analysis shall be implemented on the PPG signal
Des 3.3.1 - II	Pulse oximetry using two color channels shall be implemented to provide blood oxygenation data
Des 3.3.1 - III	The respiratory rate algorithm shall be made efficient to be able to be run on less powerful hardware

Table 3.4: Data Extraction Design Specifications

3.4 User Interface Application

3.4.1 Data Storage and Transmission

All data collected and produced by Pulse Tracer devices will be stored on a database hosted by a cloud service, and subsequently processed by VMs in the same cloud service. Cloud computing offers many benefits when compared to traditional on premise computing, such as ease of access and improved security. In particular, Pulse Tracer will make use of Amazon Web Services (AWS) platform, primarily the Elastic Compute Cloud (EC2) and Relational Database Service (RDS).

Making use of the AWS cloud platform will require that the data collected by Pulse Tracer devices be wirelessly transmitted to the cloud instances. This will involve the following steps.

- **Step 1:** Create the AWS IoT Policy to allow the Raspberry Pi to perform operations on AWS
- **Step 2:** Create a thing in AW IoT to represent the Raspberry Pi. All of the devices connected to AWS IoT are represented as "things" in the AWS IoT registry, which allows for a record of all devices connected to the AWS account
- **Step 3:** Data will be sent in JSON format from the Pulse Tracer device and subsequently stored in AWS IoT
- **Step 4:** Lambda actions will capture the event and save the information to the database

Figure 3.14 below shows how the data will communicate with AWS.

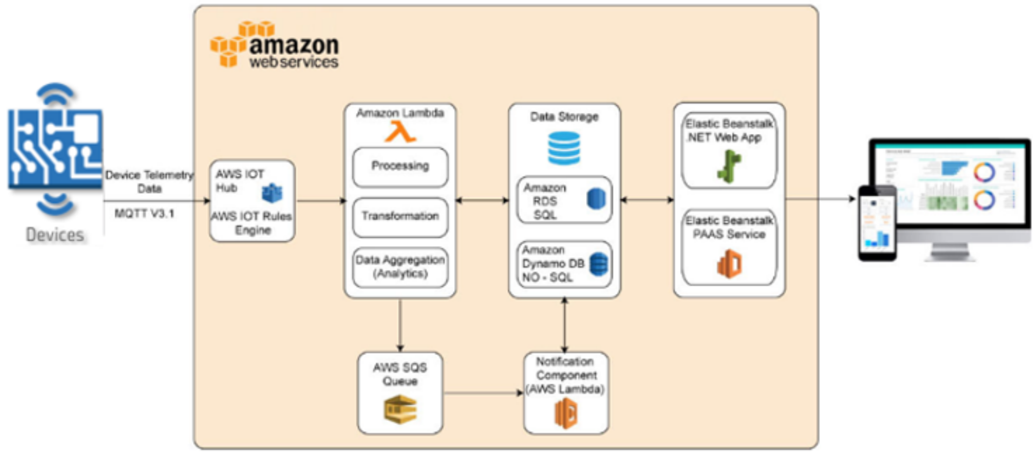


Figure 3.14: Overview of data flow [43]

Design ID	Design Description
Des 3.4.1 - I	Amazon Web Services (AWS) IoT Policy will be created, so the microcontroller can connect to the AWS.
Des 3.4.2 - I	The microcontroller will be registered on AWS IoT, and AWS can keep the record corresponding with registered account.
Des 3.4.2 - I	Data will be transfered from microcontroller to AWS using JSON format.
Des 3.4.3 - II	Lambda action will forward the receiving data from AWS IoT to th relational database.

Table 3.5: Data Transmission and Storage

3.4.2 Web Application and Database

The software interface for Pulse Tracer will consist of a web application in Phase 2 and will extend to a mobile application in Phase 3. The backend will be built using Django, which is a high-level Python web framework. Because Django is a high-level framework, it allows for rapid development while still providing sufficient features. The front end of the web application will make use of Bootstrap and Javascript alongside Django's HTML templates.

The Django backend will be connected to a PostgreSQL database, which will be hosted in the cloud as an AWS Relational Database Service (RDS). This will allow many Pulse Tracer devices to remotely send ROI data to the database. A virtual machine (VM) will also access this database to process the ROI data into heart rate and respiratory rates, and then store the results. The database itself will hold metadata about the patient and caregiver, raw ROI data for the patient, and the processed heart rate and respiratory rate data. The following diagram shows the schema for the database.

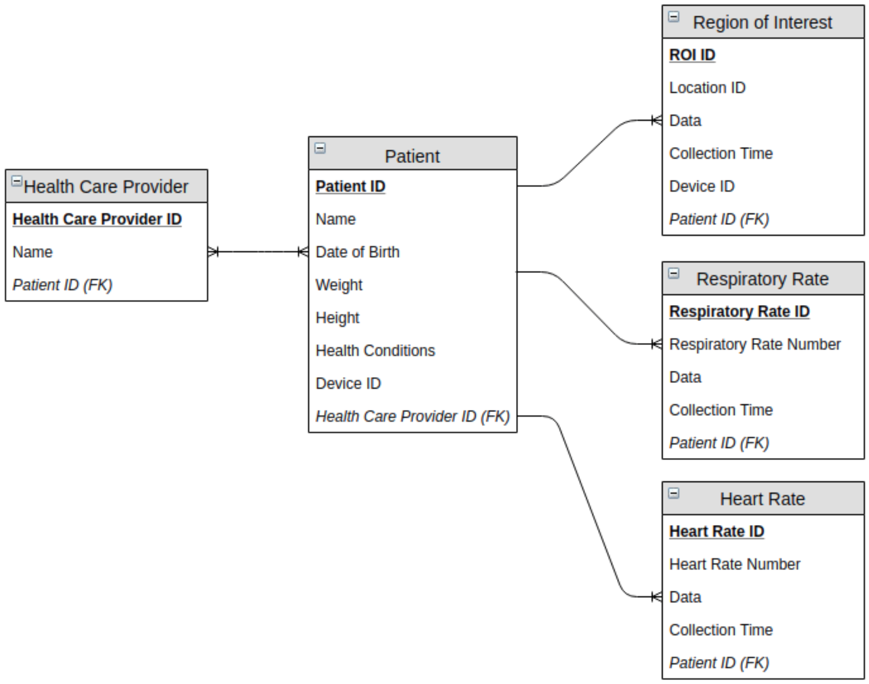


Figure 3.15: Schema for PostgreSQL Database

Data within the database will be compressed such as to limit the size of the database while still preserving the required information for both health care providers and patients. The ROI data is not needed once the heart rate waveform has been extracted, therefore only approximately 10 minutes of ROI data will be stored in the database at any given time. This will allow the VM sufficient time to process ROI data into the heart rate waveform. Additionally, the database will reduce the number of data points stored over time. For example, the heart rate waveform will be stored for all points every day for the last 90 days. After this point, only the average heart rate for every 30 minutes throughout the day will be stored.

The design specifications for the web application and database used in Pulse Tracer are detailed in the following table.

Design ID	Design Description
Des 3.4.1 - II	The web application shall have a backend built from the Django web framework
Des 3.4.2 - II	The wed application shall have a front end built from Bootstrap, Javascript, and Django templates
Des 3.4.3 - II	The database shall be hosted in a cloud service such as AWS RDS
Des 3.4.4 - II	The database must store metadata about each patient and health care provider
Des 3.4.5 - II	The database must store heart rate and respiratory rate data for each patient
Des 3.4.6 - II	The database must store approximately 10 minutes of ROI data at any given time
Des 3.4.7 - II	The web application shall allow users to view their heart rate and respiratory rate data by selecting dates and periods of time
Des 3.4.8 - III	The software interface shall be extended to include both a web application and a mobile application
Des 3.4.9 - III	The mobile application must work on Android, Windows, and iOS devices
Des 3.4.10 - III	Both web and mobile applications shall include a layer of security to only allow patients and their health care providers access to the patient's data
Des 3.4.11 - III	The database must include a compression algorithm to compress heart rate and respiratory rate data over time

Table 3.6: Web Application and Database Design Specifications

Conclusion

Pulse Tracer is a patient monitoring system designed to offer patients the opportunity to remain independent in their own home while still ensuring their health can be monitored by caregivers. This document outlines detailed design plans for Pulse Tracer broken down into its hardware and software components. Pulse Tracer combines inexpensive hardware with innovative software to meet the design goals outlined in this document.

Pulse Tracer is comprised of the following subsystems as described in the document:

1. **Hardware System**
   - Raspberry Pi powers the LEDs, records video from camera, performs image processing and facial recognition and sends the data to a database
   - NoIR camera captures video of a face in the NIR as well as the visible range
   - NIR LEDs illuminate the subject
2. **Software System**
   - Image processing and facial recognition acquire ROIs on the face that can be used to extract heart rate and respiratory rate
   - Heart rate extraction takes the ROIs and extracts the PPG signal to determine heart rate
   - Respiratory rate extraction takes the heart rate signal and applies additional analysis to extract the respiratory rate
3. **User Interface**
   - The main UI displays real time and historical data for the heart rate and respiratory rate
   - Different views allow both patients and caregivers to access data
   - The database stores data as well as provides hardware for additional processing

This design specifications document will function as a guide and a reference for the LumoAnalytics as the team continues prototyping the Pulse Tracer. These designs may be subject to change as the device goes through development and engineering iteration.

References

[1] “Life expectancy at birth, total (years).” [Online]. Available: https://data.worldbank.org/indicator/SP.DYN.LE00.IN?locations=CA. [Accessed: 2019-06-09].

[2] W. Verkruysse, L. O. Svaasand, and J. S. Nelson, “Remote plethysmographic imaging using ambient light,” *Optics Express*, 2009.

[3] A. L. Bui and G. C. Fonarow, “Home monitoring for heart failure management,” *Journal of the American College of Cardiology*, 2013.

[4] H. Kim *et al.*, “Stress and heart rate variability: A meta-analysis and review of the literature,” *Psychiatry Investigation*, 2018.

[5] “About nirs (principle of operation and how it works).” [Online]. Available: https://www.shimadzu.eu.com/about-nirs-principle-operation-and-how-it-works. [Accessed: 2019-06-27].

[6] “Tabulated molar extinction coefficient for hemoglobin in water.” [Online]. Available: https://omlc.org/spectra/hemoglobin/moaveni.html. [Accessed: 2019-06-27].

[7] “Neopixel ring - 24 x 5050 rgb led with integrated drivers.” [Online]. Available: https:https://www.adafruit.com/product/1586. [Accessed: 2019-06-27].

[8] A. S. L. Travis W Sawyer and S. E. Bohndiek, “Evaluation of illumination system uniformity for wide-field biomedical hyperspectral imaging,” *Journal of Optics*, 2016.

[9] “Head anthropometry.” [Online]. Available: https://upload.wikimedia.org/wikipedia/commons/6/61/HeadAnthropo [Accessed: 2019-07-04].

[10] “Raspberry pi camera module v2.” [Online]. Available: https://www.sparkfun.com/products/14028l. [Accessed: 2019-06-27].

[11] “High power infrared emitter (850 nm).” [Online]. Available: http://www.farnell.com/datasheets/1441891.pdfl. [Accessed: 2019-06-29].

[12] C. Soraghan *et al.*, “Optical safety assessment of a near-infrared brain-computer interface,” *International Social Science Council*, 2008.

[13] “Laser safety.” [Online]. Available: https://en.wikipedia.org/wiki/Laser_safety. [Accessed: 2019-06-30].

[14] “3 raspberry pi battery packs for portable projects.” [Online]. Available: https://www.makeuseof.com/tag/pi-go-x-ways-powering-raspberry-pi-portable-projects/ [Accessed: 2019-07-04].

[15] “Sfh4550 data sheet.” [Online]. Available: https://www.osram.com/media/resource/hires/osram-dam-2496286/SFH204550.pdf [Accessed: 2019-07-04].

[16] “Raspberry pi pinout.” [Online]. Available: https://pinout.xyz [Accessed: 2019-07-04].

[17] “Raspberry pi camera board 5mp v1.3.” [Online]. Available: https://leeselectronic.com/en/product/15236.html [Accessed: 2019-07-04].

[18] “Face detection using haar cascades.” [Online]. Available: https://docs.opencv.org/3.4.1/d7/d8b/tutorial_py_face_detection.html. [Accessed: 2019-06-27].

[19] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” 2001. [Online]. Available: https://www.cs.cmu.edu/ efros/courses/LBMV07/Papers/viola-cvpr-01.pdf. [Accessed: 2019-06-27].

[20] “300 faces in-the-wild challenge (300-w), iccv 2013.” [Online]. Available: https://ibug.doc.ic.ac.uk/resources/300-W/. [Accessed: 2019-06-27].

[21] J. Allen, “Photoplethysmography and its application in clinical physiological measurement.,” *Physiological Measurement*, 2007.

[22] L.-M. Po *et al.*, “Block-based adaptive roi for remote photoplethysmography,” *Multimedia Tools and Applications*, 2017.

[23] D. F. N. S. M. A. Hassan, A. S. Malik and F. Meriaudeau, “Novel health monitoring method using an rgb camera,” *Biomedical Optics Express*, 2017.

[24] B. J. Kolkur Kalbande, Shimpi, “Human skin detection using rgb, hsv and ycbcr color models,” 2017.

[25] T. P. Chen Wang and G. Chane, “A comparative survey of methods for remote heart rate detection from frontal face video,” 2018.

[26] S. S. G. L. QI ZHANG, YIMIN ZHOU and HAIYANG, “Heart rate extraction based on near-infrared camera: Towards driver state monitoring,” 2018.

[27] “Improving motion robustness of contact-less monitoring of heart rate using video analysis.” [Online]. Available: https://pure.tue.nl/ws/portalfiles/portal/47024614/719463-1.pdf. [Accessed: 2019-07-03].

[28] S. K. A. PRAKASH and C. S. TUCKER, “Bounded kalman filter method for motionrobust, non-contact heart rate estimation,” 2018.

[29] “Measuring heart rate from video.” [Online]. Available: https://web.stanford.edu/class/cs231a/prev_projects_2016/finalReport.pdf. [Accessed: 2019-07-03].

[30] H. Sharma, “Heart rate extraction from ppg signals using variational mode decomposition,” 2018.

[31] S. S. Mark van Gastel and G. de Haan, “Robust respiration detection from remote photoplethysmography,” *Biomed Opt Express*, 2016.

[32] J. M. A. Walter Karlen, Srinivas Raman and G. A. Dumont, “Multiparameter respiratory rate estimation from the photoplethysmogram,” *IEEE transactions on biomedical engineering*, 2013.

[33] P. H. C. et al, “Breathing rate estimation from the electrocardiogram and photoplethysmogram: A review,” *IEEE Reviews in Biomedical Engineering*, 2017.

[34] T. L. C. D. A. B. R. Charlton P. H., Bonnici T. and W. P. J., “An assessment of algorithms to estimate respiratory rate from the electrocardiogram and photoplethysmogram,” *Physiological Measurement*, 2016.

[35] M. R and S. Prabhu, "A novel algorithm to obtain respiratory rate from the ppg signal," *International Journal of Computer Applications*, 2015.

[36] V. M. F. B. Leila Mirmohamadsadeghi, Sibylle Fallet and J.-M. Vesin, "Real-time respiratory rate estimation using imaging photoplethysmography inter-beat intervals," *Computing in cardiology*, 2016.

[37] B. M. J. M. A. Parastoo Dehkordi, Ainara Garde and G. A. Dumont, "Extracting instantaneous respiratory rate from multiple photoplethysmogram respiratory-induced variations," *Frontiers in Physiology*, 2018.

[38] R. C. S. L. M. R. R. E. K. J. Thomas Bigger, Joseph L. Fleiss and J. N. Rottman, "Frequency domain measures of heart period variability and mortality after myocardial infarction," *Circulation*, 1991.

[39] K. J. and C. AJ., "Clinical relevance of heart rate variability," *Clinical Cardiology*, 1997.

[40] A. P. K. Purtov, V. Kublanov and T. Petrenko, "Remote photoplethysmography application to the analysis of time-frequency changes of human heart rate variability," *2016 18th Conference of Open Innovations Association and Seminar on Information Security and Protection of Information Technology (FRUCT-ISPIT)*, 2016.

[41] A. Jubran, "Pulse oximetry," *Critical Care*, 2015.

[42] U. Bal, "Non-contact estimation of heart rate and oxygen saturation using ambient light," *Biomed Opt Express*, 2014.

[43] "Building an industrial iot platform using aws iot." [Online]. Available: https://www.saviantconsulting.com/blog/aws-iot-manufacturing.aspx. [Accessed: 2019-07-03].

[44] V. Claes *et al.*, "Attitudes and perceptions of adults of 60 years and older towards in-home monitoring of the activities of daily living with contactless sensors: An explorative study," *International Journal of Nursing Studies*, 2015.

[45] D. Norman, "The design of everyday things," 2013.

[46] "Ieee standard for user interface elements in power control of electronic devices employed in office/consumer environments." [Online]. Available: https://standards.ieee.org/standard/1621-2004.html. [Accessed: 2019-07-03].

[47] "Ieee standard for system and software verification and validation." [Online]. Available: https://standards.ieee.org/standard/1012-2012.html. [Accessed: 2019-07-03].

[48] "Ieee standard for information technology - standardized application environment profile (aep) - posix(r) realtime and embedded application support." [Online]. Available: https://standards.ieee.org/standard/1003_13-2003.html. [Accessed: 2019-07-03].

[49] "General requirements for battery-powered appliances." [Online]. Available: https://www.scc.ca/en/standardsdb/standards/28121. [Accessed: 2019-07-03].

[50] "Ieee recommended practice for information technology– telecommunications and information exchange between systems– local and metropolitan area networks– specific requirements part 15.5: Mesh topology capability in wireless personal area networks (wpans)." [Online]. Available: https://standards.ieee.org/standard/802_15_5-2009.html. [Accessed: 2019-07-03].

[51] "Ieee standard for biometric open protocol." [Online]. Available: https://standards.ieee.org/standard/2410-2015.html. [Accessed: 2019-07-03].

[52] "Medical device software – part 1: Guidance on the application of iso 14971 to medical device software." [Online]. Available: https://www.iso.org/standard/54146.html. [Accessed: 2019-07-03].

[53] "Ieee standard health informatics–point-of-care medical device communication." [Online]. Available: https://standards.ieee.org/standard/11073-10101a-2015.html. [Accessed: 2019-07-03].

[54] "pplication of risk management for it-networks incorporating medical devices – part 1: Roles, responsibilities and activities." [Online]. Available: https://www.iso.org/standard/44863.html. [Accessed: 2019-07-03].

A User Interface and Appearance

A.1 Introduction

Pulse Tracer is a remote monitoring system intended to be used by elderly patients and their care-givers. In order to encourage the adoption of Pulse Tracer by the elderly population, LumoAnalytics must ensure the user interface of the system is designed to be simple and easy to use, as well as to address the concerns of the intended users. This includes developing a device that is contact-less such as to not interfere with the patient or cause discomfort, as well as ensuring that patient data is secure and only accessible by a professional health care provider [44].

The User Interface components found in Pulse Tracer include an on/off switch, an LED indicating whether the system is recording or not, and a web application for viewing patient data.

A.1.1 Purpose

The purpose of this document is to provide an overview of the user interface design for Pulse Tracer and evaluate the usability of the system.

A.1.2 Scope

This document will outline the user interface of Pulse Tracer throughout the entire design cycle, including the proof-of-concept prototype, engineering prototype, and final product. It will begin by providing a user analysis, outlining the knowledge and restrictions the intended users will be expected to have. Next, a technical analysis will be performed which will investigate Don Norman's "Seven Elements of UI Interaction" and demonstrate how the hardware and software user interfaces of Pulse Tracer will address these concepts. The engineering standards associated with the user interfaces will then be discussed, followed by a detailed description of the analytical and empirical usability tests that will be performed on the final Pulse Tracer device.

A.2 User Analysis

According to our research and investigation, the users of Pulse Tracer can be classified into three major categories: caregiver, family member, and others. Other factors such as age, gender, occupancy, and race of the user can also affect the efficiency and correctness of using the Pulse Tracer. In addition, the education background of the users plays another important role in the user analysis.

In order to use the Pulse Tracer correctly and efficiently without learning too much things, the users should have the ability to use the modern technologies as well as some basic medical cognition. Those skills include: capable of using phone app to check data regularly, awareness of reasonable range of the respiratory rate and heart rate, and be able to find a correct solution when things become abnormal.

Next, we will analyze each factor to get the most accurate and reliable result.

First, if we look at the three major categories, we can tell that the caregivers have abundant time and they are often well trained because of their responsibilities which also closely related to their occupancy. Thus, the caregiver has a higher chance than others to find the unusual situation on time. The family members do not has too much time to focus on caring elders, but the phone app can still remind them if something really happened. Besides that, the family members have the strongest relationship with the patients so that they have the strongest will to find the correct solution too.

By using common sense, we can easily conclude that younger people is more likely to understand the procedure of using the Pulse Tracer. The gender does not affect our analysis but the women will lead the market because most of the caregivers are women. The race will only affect the data analysis because of the different skin colors. Lastly, it is worth to tell that the person with high education background and working on a medical related company can use the Pulse Tracer easily.

A.3 Technical Analysis

LumoAnalytics is dedicated to providing an intuitive system that is easy and simple for users of all ages and technical backgrounds. In order to accomplish this, our team has taken into account the Discoverability, Feedback, Conceptual Model, Affordance, Signifiers, Mapping and Constraints of Pulse Tracer, as detailed in Don Norman's "Seven Elements of UI Interaction" from his book *The Design of Everyday Things*[45]. The following sections will discuss these elements in further detail.

A.3.1 Discoverability

Discoverability emphasizes the intuitive design of the device how intuitive is this device to use and control. For Pulse Tracers prototype, this can be separated into two components the physical presence of Pulse Tracer and the app.

Pulse Tracer the device in Phase II of production is placed within an enclosure, with holes for the camera, the LED ring, lights to indicate whether the device is operating and whether it requires charging, a charging outlet and a switch.

To determine the discoverability for the hardware portion of the prototype, we have compiled a series of actions the user will be required to perform:

1. Turn on device with power switch
2. Charge Pulse Tracer when light indicates a necessity

In order to achieve these tasks, the following factors have been taken into account for Pulse Tracers design:

1. The power switch will be labeled clearly and placed on the left of the device for easy understanding and access
2. The charging station and corresponding light to indicate power status are clearly labelled and placed on the right side of the power switch to avoid confusion
3. The light that indicates whether the device is in operation clearly labelled and placed on the left side of the device is close proximity to the power switch
4. The camera and LED ring are placed at the front of the device

The software portion of Pulse Trace is available as an app. The app will contain two different settings for the user and caregivers respectively. In the user setting, the patient will be able to access their current and historical heart rate and respiratory rate data, along with any additional metrics, in the form of graphical data. The user will also be able to flag data and input personal information such as body weight, age and height. The caregiver setting will allow for viewing of multiple patients and will contain the same information as well as the ability to flag data, but can only view rather than change patient information.

To determine the discoverability for the software portion of the prototype, we have compiled a series of actions the user will be required to perform:

1. Access the patient setting and complete the following:
   (a) Open each metric individually and view graphical data
   (b) Flag data
   (c) Input personal information
2. Access the caregiver setting and complete the following:
   (a) Open each metric individually and view graphical data

(b) Flag data
(c) View multiple patients
(d) View patient data

In order to achieve these tasks, the following factors have been taken into account for Pulse Tracers design:

1. The application is clearly labelled with a unique icon
2. The initial set up will allow the user to select if they are a patient or a caregiver
3. The app will allow for a main menu to select which vital sign to view
4. There will be a separate page to input and view patient information
5. There will be a button to flag data

A.3.2 Feedback

Feedback in Pulse Tracer is used to inform the user of the power and battery status of Pulse Tracer, as well as possible changes in the application. For the prototype model, the following is used:

Battery LEDs:

- Red to indicate power is low and to attach charging cord
- Blue to indicate device is currently charging
- Green to show device is fully charged

Power LED

- Red to show device is on
- No color to show off

Software Application

- When metrics are selected in the main menu, graphs and average value is displayed
- When metrics are selected in the main menu, graphs and average value is displayed
- When flagging data has been selected, a pop up will ask to confirm flag data is desired
- After flagging data, an indicator on the graph will be located where data was flagged
- On the caregiver setting, when each patient is selected, their information will appear

A.3.3 Conceptual Model

A conceptual model is designed to show the user how the system works and allow better understanding of the functions of the device. The physical portion of Pulse Tracer is intended to mimic a wireless webcam, with the lens on display and LEDs to indicate the power and charging of the device. The Pulse Tracer app is modeled after typical smart phone apps with additional designs specified for elderly use, such as easy navigation and clear text. This allows for a very intuitive design that can easily be taught and/or used by anyone with a familiarity with apps.

A.3.4 Affordance

Affordance is defined by how the device should be used within given constraints. Pulse Tracer uses a simple app to display graphical data and average data, with simple navigational tools that show what the app is capable of displaying and nothing else. The hardware application in the prototype stages has a clearly labelled switch and power source as its only sources of interaction, placed in different locations and with different shapes to provide ease of use.

A.3.5 Signifiers

Signifiers are used to discover what affordances are possible. Pulse Tracers signifiers are similar to the feedback provided, where LEDs are turned on and off to indicate the power and charging status of the device. In the application side, metrics and averages are displayed in real time values, and flagged data is shown with an indicator at the location of the chosen data.

A.3.6 Mapping

Mapping is used to indicate the relationship between two sets of things. To ensure good mapping in Pulse Tracers prototype stage, all LEDs and the switch are labelled, with the power and charging LEDs placed on the left and right side respectively to encourage intuition. By position them on opposite sides, this ensures there is no accidental misunderstanding of what each side means, as there are three charging LEDs on the right side as opposed to the singular LED for power on the left side.

A.3.7 Constraints

Constraints are the limits of the user interface. While the constraints on the hardware component are limited to needing an outlet from which to plug in the charger, remembering to charge the device and not being color blind in order to differentiate the different lights used as signifiers, the software portion contains several more constraints. First, the user must own a computer in order to use the web application and have access to internet information cannot be displayed without a screen. From this, the user also requires a keyboard and mouse for the purpose of selecting and flagging data. Secondly, because the data is intended to be secure, the user must be able to remember a login and password in order to access their information, or the information of their patients. Finally, the user needs to the intuition to recognize that clicking certain words or symbols may offer more information.

A.4 Engineering Standards

IEEE 1621: Standards for User Interface Elements in Power Control of Electronic Devices Employed in Office/Consumer Environments[46]

IEEE 1012: Standards for System and Software Verification and Validation [47]

IEEE Std 1003.13: Standards for portable Real-time and Embedded Applications [48]

C22.2 NO. 0.23-15: General requirements for battery-powered appliances[49]

IEEE 802.15.5: Standard for telecommunications and information exchange between systems.This standard ensures security of component communications using wireless networks such as cloud server, mobile application [50]

IEEE 2410: Standard for Biometric Open Protocol, it ensures security regarding cloud communication with devices [51]

IEC/TR 80002-1: Standard for medical device software [52]

IEEE Std 11073-10101a: Standard Health informatics -Point-of-care medical device communication [53]

IEC 80001-1:2010: Standard of risk management for IT-networks incorporating medical devices [54]

A.5 Analytical Usability

The analytical usability testing will focus on any inefficiencies or annoyances with the user interface that could lead to the user becoming frustrated or unable to complete their desired task. A series of tests will be performed that are designed to test the major portions of the user interface, typical user tasks and perceived areas of difficulty.

Task	Success/Fail	Notes
The camera is in a secure position facing the patient when it is setup		
A light turns on to indicate the camera is on and recording		
A light turns off to indicate the camera is off and not recording		
A light stops flashing to indicate the camera is connected to the internet and the user application		
A light turns green when charging		
A number of battery LEDs indicate how much power is left in the battery		
A message is sent to the user application when the camera is recording		

Table A.1: Hardware Analytical Usability

Task	Success/Fail	Notes
The application successfully launches and shows the home page		
An account can be set up by a user showing their username and password		
When a patient or caretaker logs into the application, the proper application mode is displayed, showing either a patient or caretaker view		
The current heart rate of a patient is shown in the heart rate display		
The historical heart rate of a patient is shown in the historical heart rate graph		
The current respiratory rate is shown in the respiratory rate display		
The historical respiratory rate is shown in the historical respiratory rate graph		
Each page in the application is navigable from the home page		
An abnormality in heart or respiratory rate sends an alert		
There is a confirmation box for when the user flags data		
Multiple patients can be viewed in the caretaker view mode		

Table A.2: Software Analytical Usability

A.6 Empirical Usability

Empirical usability testing will play a crucial role in evaluating the usability of both the software and hardware components of Pulse Tracer. This stage of usability testing will require that approximately 5 to 10 potential users perform a series of tasks provided by the LumoAnalytics team. The tasks will evaluate both the hardware and software user interfaces of Pulse Tracer.

The following table identifies the tasks required to evaluate the hardware interface for Pulse Tracer. This includes the button to turn the device on and the interface involved with the rechargeable battery.

Task	Success/Fail	Notes
Turn the device on		
Determine whether the system is recording or not		
Determine whether the device needs to be charged		
Charge the device		
Determine whether the device is finished charging		

Table A.3: Hardware Tasks

The following table identifies the tasks required to evaluate the software interface for Pulse Tracer. This includes the web and phone applications for updating patient information, viewing calculated heart and respiratory rate data, and switching between patients for caregivers.

Task	Success/Fail	Notes
Login to your account		
Edit your profile information (i.e. change username, age, height, etc.)		
View your heart rate data from Tuesday of last week		
View the trend in your heart rate over the past week		
Flag a data entry as an event that should not raise concern		

Table A.4: Software Tasks

The goal of this test is to evaluate the system rather than the users; as such, the tasks are aimed towards evaluating how intuitive the user interface of Pulse Tracer is. The members of the LumoAnalytics team who will be conducting the test will record observations of each user. They will specifically be looking for whether the user was successful in completing the task or not, the time spent on each task, any errors involved with working towards completing the task, and any breakdowns or workarounds required for completion. Any recorded problems encountered by the users will be compared to the chart shown in the figure below.

		Damage	
		Low	High
Frequency	Low	Not severe - can ignore the issue	Severe - the issue should be addressed
	High	Annoyance - the issue should be addressed	Catastrophic - the issue must be addressed

Figure A.1: Severity chart of issues from the empirical usability testing

Any catastrophic incidents that have a high frequency and a high negative impact will be addressed immediately in the user interface. Additionally, incidents with a large negative impact that rarely occur or incidents that have a small negative impact that occur frequently should be addressed but are not of great concern. Finally, incidents with a small negative impact that rarely occur will not be addressed.

Following the task-based usability test, LumoAnalytics will provide each user with the following questionnaire regarding their experience with Pulse Tracer.

Questions	Response
Would you use this device in the future?	
Was the device was easy to use?	
Did the device frustrate you?	
What was the greatest strength of the device?	
What could be improved about the device?	
What was your overall impression of the device?	
What features would you like to see added to this device?	
What features would you like to see removed from this device?	
Would you recommend this device to another person? Why or why not?	

Table A.5: Post-Test Questionnaire

With these results in mind, LumoAnalytics will modify the Pulse Tracer system to fix any significant issues in an attempt to ensure users are comfortable and able to efficiently use the device.

B Supporting Test Plan

The following table identifies the category numbers.

Category Number	Category Description
Test 2.1	General Hardware Design Specifications
Test 2.2	Optical Design Specifications
Test 2.3	Circuitry
Test 3.1	General Software Design Specifications
Test 3.2	Image Processing
Test 3.3	Data Extraction
Test 3.4	User Interface Application

Table B.1: Category Numbers

Once the system has test the passed, a check mark can be added to the last column of the table.

The following tables outline various tests to verify and validate the proof of concept,appearance and engineering prototype. They are numbered according to their related category and test number as follows:

Test.{section code}.{subsection code}.{test number}-{phase number}

ID	Category	Description	Acceptance Condition	P/F
Test 2.1.1-I	General Hardware	Ensure all data can be viewed on a monitor with HDMI input	The monitor displays all data correctly	
Test 2.1.2-I	General Hardware	Ensure LEDs can provide contin-ues illumination	The illumination provided by LEDs is reasonable	
Test 2.1.3-I	General Hardware	Ensure the Raspberry Pi can work fine with the camera mod-ules	Videos can be captured by the Raspberry Pi camera correctly	
Test 2.1.4-II	General Hardware	Ensure the data can be viewed on a web application	Users can access data though the web	
Test 2.1.5-II	General Hardware	Ensure the Pulse Tracer will be in an enclosed container with di-mensions at least 10x10x14cm	The size of the container is perfectly match the device	
Test 2.1.6-II	General Hardware	Ensure the enclosed container will have openings for the lens, light source, switch and charging station	Users can easily check and operate the device from the outside	
Test 2.1.7-II	General Hardware	Ensure there will be LEDs to indicate the current switch-ing,charging,and operating sta-tus	Users can obtain the sta-tus of the system by ob-serving the LEDs	
Test 2.1.8-II	General Hardware	Ensure the enclosed container is unobtrusive and easy to set up	Users feels relax and com-fortable when using the device	
Test 2.2.1-I	Optics	Ensure LEDs operates at wave-length of 850nm	Videos are able to show provide acceptable data for performing algorithms	
Test 2.2.2-I	Optics	Ensure LEDs operate in a ring configuration to illuminate the entire face	Videos can display a com-pletely illuminated face	
Test 2.2.3-I	Optics	Ensure LEDs operate with an ir-radiance of 0.5mW/cm2 at a dis-tance of 1m	The irradiance meter shows that the irradiance of LEDs is within desired range	
Test 2.2.4-I	Optics	Ensure the ring configuration for the LEDs will have a minimum radius of 16cm	The size of the ring mea-sured by users is correct	
Test 2.2.5-I	Optics	Ensure the ring configuration must be placed at the same dis-tance from the subject as the Raspberry Pi Camera module	The Raspberry Pi camera and the LED ring circuit are in the same vertical level	
Test 2.2.6-II	Optics	Ensure the ring configuration will have an inner radius of 1.5cm and outer radius of3.5cm.	The size of inner ring and outer ring measured by users are correct	

Table B.2: Supporting Test Plan

ID	Category	Description	Acceptance Condition	P/F
Test 2.3.1-I	Circuitry	Ensure the LED ring circuit can work properly	LED ring circuit generates reasonable illumination	
Test 2.3.2-I	Circuitry	Ensure the LED ring circuit is powered and grounded by GPIOs of the Raspberry Pi	The LED ring circuit powered and grounded by the Raspberry Pi works properly	
Test 2.3.3-I	Circuitry	Ensure the camera can work with the Raspberry Pi properly and all function are activated	Camera can be accessed though Raspberry Pi without error	
Test 2.3.4-I	Circuitry	Ensure that there is a Ethernet cable connected to the Raspberry Pi	The Raspberry Pi can access the Internet though the Ethernet cable	
Test 2.3.5-II	Circuitry	Ensure that a minimum 6000 mAh Lithum battery is provided to support the Raspberry Pi	The Raspberry Pi can keep working for at least 72 hours without charging	
Test 2.3.6-II	Circuitry	Ensure that a customized PCB is invented to replace the breadboard	The customized PCB has the same functionality as the breadboard	
Test 2.3.7-II	Circuitry	Ensure the Raspberry Pi can be able to be connected to the Wi-Fi and all procedures can be able to be done remotely	Developers can access the files though Wi-Fi	
Test 2.3.8-II	Circuitry	Ensure that an overcurrent protection circuit is invented to avoid the short circuit	The system automatically shut down once the current exceeds the threshold	
Test 2.3.9-II	Circuitry	Ensure that an over charge protection circuit is invented to avoid the over charge of the battery	The charging circuit is cut-off once the battery is fully charged	
Test 2.3.10-III	Circuitry	Ensure that a mechanical key is designed for the maintenance purpose	Developers have the ability to open the enclosed container at any time	
Test 3.1.1-I	General Software	Ensure that all software components involved in Pulse Tracer shall be organized into a single pipeline	No conflicts occur between software components	
Test 3.1.2-I	General Software	Ensure the pipeline include reading, analysis, and posting of data from the input camera	The pipeline is unobstructed	
Test 3.1.3-II	General Software	Ensure the extraction of heart and respiratory rates from transmitted regions of interest can take place on a remote machine	Developers can edit the HR and HR extraction algorithm remotely without error	
Test 3.2.1-I	Image Processing	Ensure the ROI selection can detect region of interest on the patient's nose, forehead, and cheeks	The nose,forehead, and cheeks can be successfully detected	

Table B.3: Supporting Test Plan

ID	Category	Description	Acceptance Condition	P/F
Test 3.2.2-I	Image Processing	Ensure that for each frame where a face is detected, the image processing algorithm shall produce at least two viable regions of interest to be used by downstream analysis	Developers can see at least two ROI after performing the algorithm	
Test 3.2.3-I	Image Processing	Ensure the selected regions of interest shall each have an area that is approximately 0.5-1.0 percentage of the total area of the detected face	The ROI surface area measured by developers is desirable	
Test 3.2.4-I	Image Processing	Ensure the facial detection algorithm can accurately detect faces at various distances from the camera	The results will not be disturbed by changing the distance between camera and face	
Test 3.2.5-II	Image Processing	Ensure the image processing algorithm is real-time based	The algorithm does not cause propagation delay to the system	
Test 3.2.6-II	Image Processing	Ensure the facial detection algorithm can accurately detect faces that include obstruc- tions such as facial hair, eye glasses, and headwear	Users with facial accessories can be detected correctly	
Test 3.2.7-II	Image Processing	Ensure the facial detection algorithm can accurately detect faces in varying levels of illumination	Users under varies levels of illumination can be detected correctly	
Test 3.2.8-II	Image Processing	Ensure the image processing algorithm can raise an alert if no face is detected for more than 2 seconds	Users are able to know what happened if alert raises	
Test 3.2.9-II	Image Processing	Ensure the facial detection algorithm can work for both frontal views and profile views of the face	Users viewed from different angles can be detected correctly	
Test 3.2.10-II	Image Processing	Ensure the facial detection algorithm can be optimized to improve speed and accuracy	The workload of microprocessor reduces significantly	
Test 3.3.1-I	Data Extraction	Ensure the respiratory rate algorithm can filter out low and high frequency signals from the PPG signal that are not associated with respiration	Signals filtered by the algorithm are all in the desired frequency range	
Test 3.3.2-I	Data Extraction	Ensure the respiratory rate algorithm can detect features such as RIFV, RIIV and RIAV in the PPG signal	All features are observed by developers	

Table B.4: Supporting Test Plan

ID	Category	Description	Acceptance Condition	P/F
Test 3.3.3-I	Data Extraction	Ensure the respiratory rate algorithm can estimate the respiratory rate from the extracted respiratory signal	The extracted respiratory rate matches the real RR of the tester	
Test 3.3.4-II	Data Extraction	Ensure the respiratory rate algorithm can combine results from multiple algorithms to get a more accurate respiratory rate	The accuracy of the RR result is significantly improved	
Test 3.3.5-II	Data Extraction	Ensure the Pulse oximetry using two color channels can be implemented to provide blood oxygenation data	Developers can view the blood oxygenation data without errors	
Test 3.3.6-III	Data Extraction	Ensure the HR and RR algorithm is efficient and can be able to be run on less powerful hardware	The workload of the micro processor is significantly reduced	
Test 3.4.1-II	UI Application	Ensure the web application have a back end built from the Django web framework	Web function can be logically controlled by developers through the Django framework	
Test 3.4.2-II	UI Application	Ensure the wed application have a front end built from Bootstrap, Javascript, and Django templates	Web shows correct information to users	
Test 3.4.3-II	UI Application	Ensure the database can be hosted in a cloud service	Developers can access the cloud without error	
Test 3.4.4-II	UI Application	Ensure the database can store 10 mins of ROI data,RR,and HR of the patients	Users can view RR and HR data, and developers can view all the data though web	
Test 3.4.5-II	UI Application	Ensure the web application is user friendly	Users can view specific RR and HR data by entering date/time to a filter	
Test 3.4.6-III	UI Application	Ensure the mobile application can work on Android, Windows, and iOS devices	Users can view HR and RR on any mobile device	
Test 3.4.7-III	UI Application	Ensure both web and mobile applications include a layer of security to only allow patients and their health care providers access to the patient's data	Other people without the permission can not access the data in any circumstance	
Test 3.4.8-III	UI Application	Ensure the database include a compression algorithm to compress heart rate and respiratory rate data over time	Users can store more data into the database after compression	

Table B.5: Supporting Test Plan