**Automated Pulse Sensing Tool**



**國立東華大學資訊工程系**

**National Dong Hwa University**

**110學年度大學部畢業專題研究報告**

110 CSIE Undergraduate Project Report

Advisor: 顏士淨 (Shi-Jim Yen)

Member: Tyler Edwardo Eck

**National Dong Hwa University**

**110 Department of**

*Computer Science and Information Engineering*

**3rd year Project Final Report**

**~ Automated Pulse Sensing Tool ~**

**Advising Professor:**

顏士淨 (Shi-Jim Yen)

**Team Member:**

* Tyler Edwardo Eck 410821337

**Project Name:** Automated Pulse Sensing Tool

**Project Client:**          Medical Professionals

中 華 民 國　　111 　年　　05　　月 　20 　 日

**國立東華大學資訊工程學系**

**專題報告原創性聲明**

**National Dong Hwa University**

**Department of Computer Science and Information Engineering**

**Statement of Originality**

I hereby affirm that the submitted project report is the result of research under the supervision of my advisor. Except where due references are made, the report contains no material previously published or written by another person or group. All significant facilitators to the project have been mentioned explicitly. Should any part of the statement be breached, I am subject to the punishment enforced by the University and any legal responsibility incurred.



Tyler Eck

410821337

**ACKNOWLEDGEMENTS:**

In the process of conducting this project, I have received an abundance of support.

Firstly, I would like to thank my Advisor, Professor 顏士淨 (Shi-Jim Yen), for the assistance and resources needed to conduct such a complex project. The guidance that he gave me helped to overcome many problems that I found myself stuck with. Moreover, his expertise and knowledge in this field gave me insight into the paths that I had taken to complete this project.

I would also like to thank邱顯棟 (Jelly) and Serkan Kavak, two employees of the AI Lab, whose advice and one-on-one help pushed me to overcome the many programming obstacles that stood in my way.

With the help of the ⼨關尺辨 識 team and permission from my advisor, I was able to implement their pulse point detection algorithm into my GUI. Without this, my project would not have been as engaging and practical.

Lastly, I would like to thank my parents, family, and friends whose value of support was immeasurable. Without this, I wouldn’t have pushed myself hard enough to make this project the best that it could be. They helped keep me grounded and were there to lend a helping hand if needed.

**ABSTRACT:**

Due to the rise in popularity of technology, especially of the internet, people have sought to use creative and functional methods of implementing it in their daily life. Automation, for instance, can have a huge impact in the amount of time saved doing remedial tasks. As seen in the manufacturing sector, both throughput and speed are pushed to the limit with industrial robots in the place of working human beings. This exponential improvement that automation brings should be applied to something more important such as the medical field.

Combining automation with medical tools creates a crossroad for the tool proposed in this project. With an automated pulse sensor that can remotely record a person’s pulse faster, medical checkups for the elderly living in remote areas becomes easier and more efficient. By using a cloud-based platform, this tool can also give doctors the ability to remotely observe multiple people’s pulse recordings at any given time. Limitations on location and time would be eradicated as a result of the implementation of this tool.

**Keywords:** automation, medical, pulse, cloud-based

**CONTENTS:**

Title Page …………………………………………………………….…..1

Statement of Originality …………………………………………………2

Acknowledgments ……………………………………………………….3

Abstract …………………………………………………………………...4

Introduction ……………………………………………………………….6

Motivation and Research Problem ……………………….…………....8

Research Method ………………………………………………………..9

Program Execution ……………………………………………………..16

**INTRODUCTION**

In this evolving technological era, it is common to find ourselves dependent on automation technology whether it may be for a simple smart home device or something more complex such as a self-driving car (autonomous vehicle). Regardless, it is important to note that this fundamental use of automation can have a drastic impact on our seemingly difficult life. As popularity and demand grow, I believe that the next logical step is to implement automation in more important roles such as for medical purposes. By doing this, more time will be saved which may potentially save more lives.

Due to its popularity here in Eastern Taiwan, the specific medical procedure that I want to automate is pulse diagnosis in Traditional Chinese Medicine (TCM). As opposed to pulse diagnosis in western medicine that retrieves little information from a single point, TCM pulse diagnosis focuses on three locations, cun, guan and chi, on a person’s wrist to get the condition of the heart, liver, and kidney respectively. Although it is argued that TCM pulse diagnosis results are biased and only dependent on the practitioner’s experience and intuition, there have been many studies that aim to quantify the methods and assessments that can be drawn from the diagnosis. Moreover, most of the Taiwanese population, especially the elderly, rely on TCM assessments and consultations when they’re sick.

Therefore, with the advantages that automation brings and the popularity of Traditional Chinese Medicine in Taiwan, I propose a fully automated pulse sensing tool that will allow for users to measure their pulse with ease.

With the simple push of a button, this tool will detect the pulse location on a person’s wrist, move the pulse sensor to that location, and then finally make a digital recording of the pulse reading that will automatically be uploaded to the cloud.

As a result, the pulse reading can be analyzed by a doctor anywhere in the world making it more convenient for patients to see the doctors and for doctors to analyze and assess pulse diagnosis.

**MOTIVATION AND RESARCH PROBLEM**

Technology has solved many of our proximity problems whether it may be remote schooling due to COVID-19 or online conferences for company chains in different locations. The value of being able to connect with each other regardless of our location is quite essential in this modern age. Although we take it for granted, this idea of connection must be pushed to its limit. Therefore, the motivation of this project is to extend this connection to people who are physically unable to go for frequent doctor visits as a result of living in remote areas.

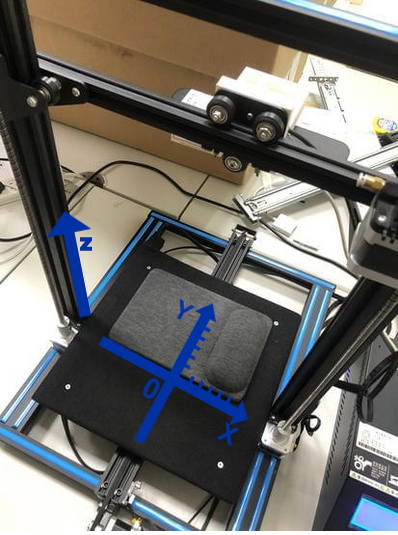
Small villages, especially in the east of Taiwan, lack the efficient number of doctors and special experts that are needed due to high numbers of elderly population living in those areas. Because of this, a lot of people need to travel far distances just to get a routine check-up. This poses a problem when they are physically unable to go due to health conditions, lack of transport, or the dynamic COVID-19 restrictions. The proposed tool allows for convenience as the goal is to make it accessible for people to send their information remotely to the doctor for examination.

**RESEARCH METHOD**

The project’s goal was to make an automated system that remotely transmits data. It needed both the necessary hardware to move motors and measure the pulse and the necessary software for detection of the wrist location, GUI for users, and the transmission of data. Therefore, I have split the project into two main parts as shown below: hardware and software

1. **Hardware**

The three main hardware components necessary to complete the project were the step motors to move the sensor, the camera as digital input for detection, and the pulse sensor to measure the pulse. Fortunately, I was provided with a Creality CR10s 3D printer that had all the necessary motors with its stop switches connected in an X,Y,Z plane shown below in Figure 1.



**Figure 1: 3D printer with dimensions**

The repurposing of this 3D printer required reprogramming the stepper motors that were connected to the motherboard. The motherboard, which is housed separately with the battery, is an Arduino2560 microcontroller and can therefore be controlled using the Arduino IDE. After the reprogramming was done, the motors could move in accordance with the X, Y, and Z Cartesian plane based on coordinates that was supplied to it through the serial.

The pulse sensor and camera were both provided by the lab as well. The pulse sensor is a HK-2019 Cylindrical Pulse Sensor. This 200Hz pulse sensor, as shown in Figure 2 (left), measures both pulse and pressure. The camera is a StereoLabs ZED dual camera as shown in Figure 2 (right). Because the website specifically advertises it for autonomous robotics due to its high frame rate and long-range 3D sensing, it was selected for the project.

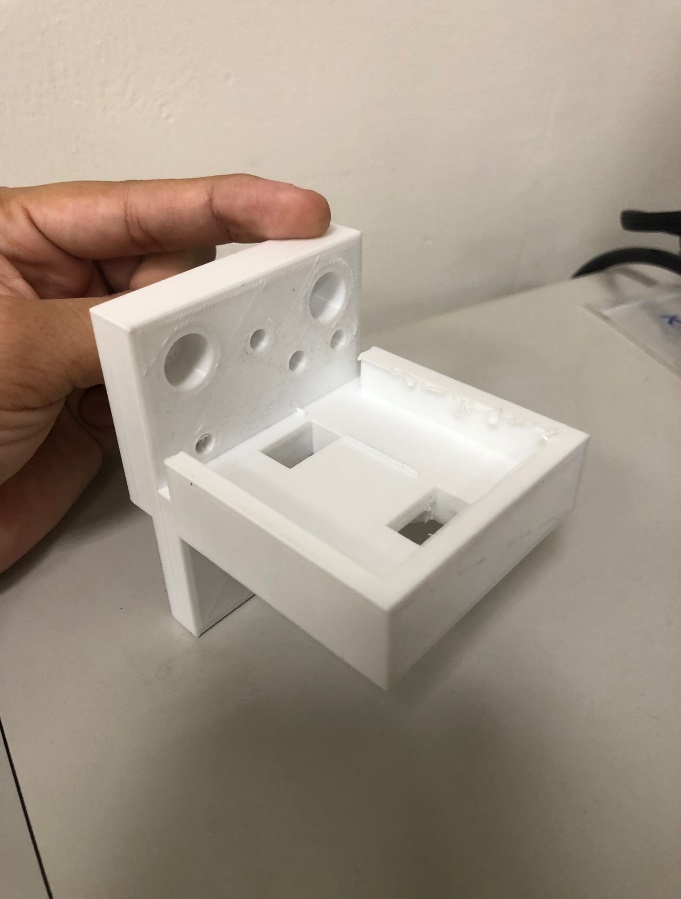
A close-up of a headphone

Description automatically generated with low confidence

**Figure 2: Pulse Sensor and ZED Dual Camera**

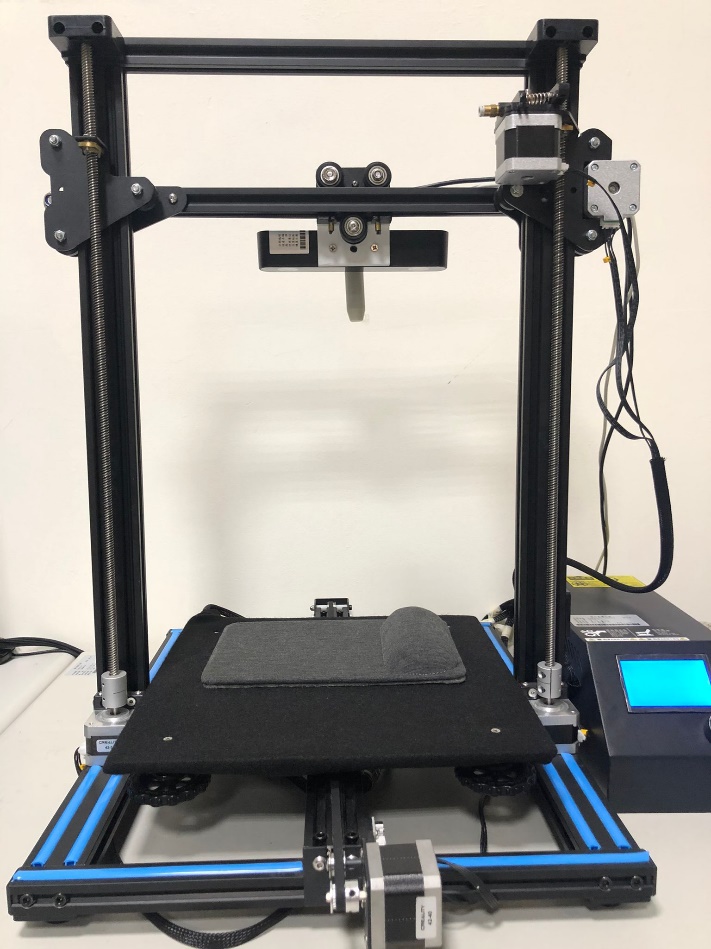
Lastly, to mount both the camera and the sensor together for the motors to move, I had to learn how to design a 3D model and how to print using a 3D printer. Fortunately, the lab had a functioning 3D printer to my disposal, so I designed a 3D-printed part shown below in Figure 3. (On the left shows the design of the part in the software and the right shows the printed piece).

A picture containing indoor, electronics, jack

Description automatically generated 

**Figure 3: 3D-printed part**

After all the necessary hardware configurations were done, the right USB drivers were installed, and USB extensions were added to each hardware component, the final hardware product, as shown below in Figure 4, was completed.

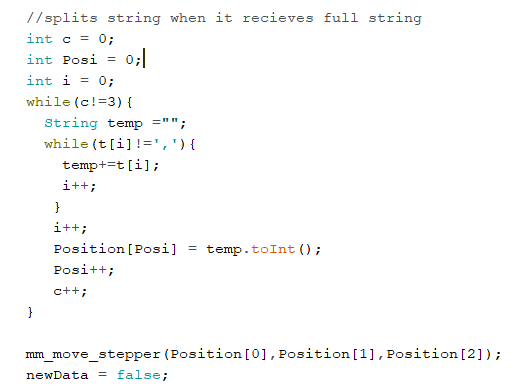
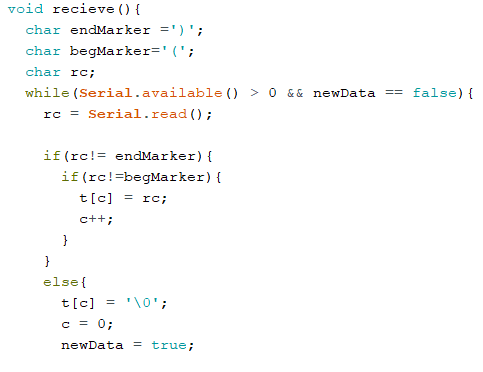
****

**Figure 4: Completed Hardware**

1. **Software**

There were two main portions of the project that consisted of software: The reprogramming of the Arduino and the GUI for the doctors/patients to use. These two programs will communicate with each other via the serial to have the motors move based on the specifications made in the GUI, thus successfully connecting the hardware and the software together.

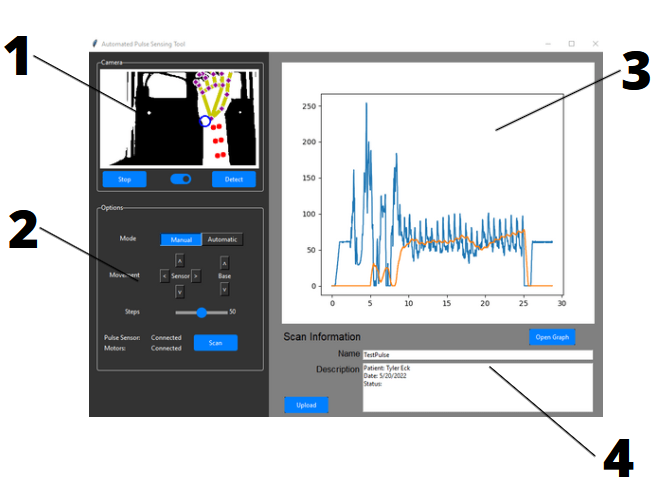
The Arduino language, which is a language that uses C++, is a fairly simple language used for the wiring and programming of Arduino microcontroller boards. As stated in the Hardware section, the motherboard used in the Creality 3D printer is an Arduino Mega 2560 running Marlin 2.0, the firmware used for all Creality printers. However, with simple programming done in the Arduino IDE, I successfully wrote a function that would constantly listen for coordinates through the serial. With these coordinates, I moved the X,Y, and Z coordinate motors with the help of an official Arduino library called SpeedyStepper. Shown below is a snippet of the Arduino code that constantly loops to listen for coordinates.



**Figure 5: Arduino Code**

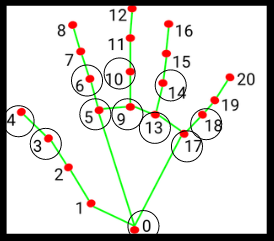
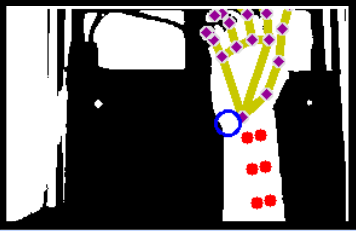
Based on previous knowledge of GUI designing and implementation, I decided to use Tkinter, a package in the standard Python interface GUI toolkit, to construct a user-friendly interface for the doctors and patients to use. As shown below in Figure 6, the GUI consists of 4 parts:

1. Camera – the section on the top left to show users a live feed of the camera and its detection of the pulse point
2. Options – the section on the bottom left to switch between the two main modes: Manual and Automatic, control the motors manually with arrows and step selection, and start the scan when needed. The section also gives connection information regarding the pulse sensor, motors, and camera.
3. Graph – the section on the top right to display, examine, and manipulate the graph taken.
4. Information – the section on the bottom right to give additional information of the graph including the name, date, and a description (if needed). After all necessary information is inputted, an upload button allows for the user to send the data via Firebase to the cloud for evaluation.

****

**Figure 6: Automated Pulse Sensing Tool GUI**

Lastly, the detection algorithm used to detect the pulse point was developed by the ⼨關尺辨 識 group and was implemented into this project. The algorithm calculates the point by using a Python package called Mediapipe. As shown below, Mediapipe can detect a hand and identify 21 uniquely marked points on the hand known as landmarks. It is with 11 of these 21 landmarks that the algorithm calculates and marks the points in the Camera section.

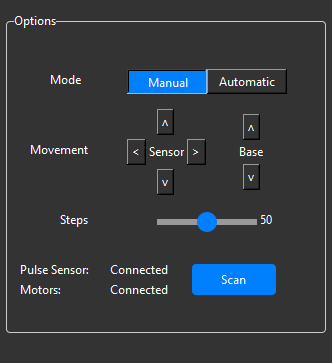
 

**Figure 7: Automated Pulse Sensing Tool GUI**

**PROGRAM EXECUTION**

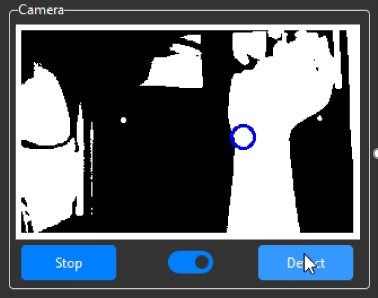
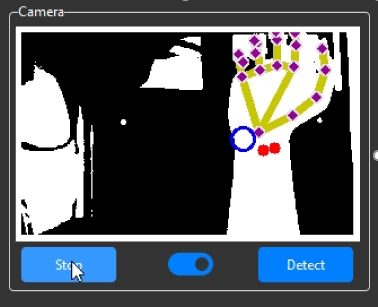
The main flow of the program starts with the detection and recognition of all connected devices, including the motors from the Arduino, the pulse sensor, and the camera that are all connected via USB. After everything is confirmed to be connected, the user must select one of the two modes to run: Manual or Automatic

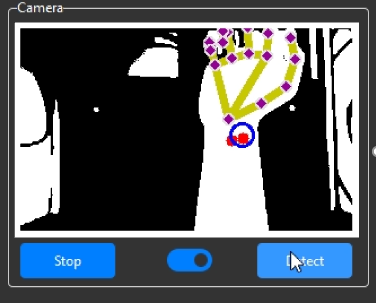
The default mode, manual mode, allows for a more interactive and hands-on experience which would most likely be done by a doctor. Movement of the motors can simply be done by selecting the number of centimeters each press should take and then pressing the Up, Down, Left, and Right buttons located in the Options section as shown in Figure 8.



**Figure 8: Manual Mode: Selection of Cm per Step**

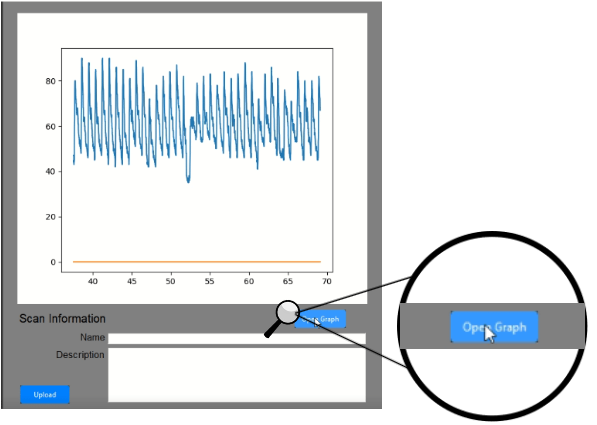
To aid the doctor in reaching the proper x and y coordinate of the pulse point, the detect button should be pressed to activate the detection algorithm. The live camera feed will calculate and mark the pulse point on the screen. Due to its fast-dynamic calculations, the algorithm’s marking may change abruptly. To combat this, the Stop button should be pressed to stop the frame of the camera. From this point, the doctor can press where the pulse sensor, which is marked on the screen as a blue circle, should go and the motors will move to that location. Shown below in Figure 9 is the flow of this operation.

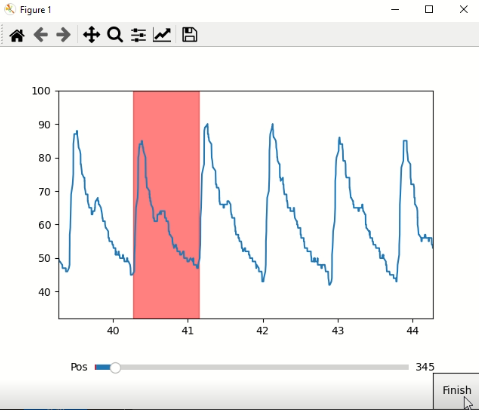


**Figure 9: Manual Mode: Detection of Pulse Location**

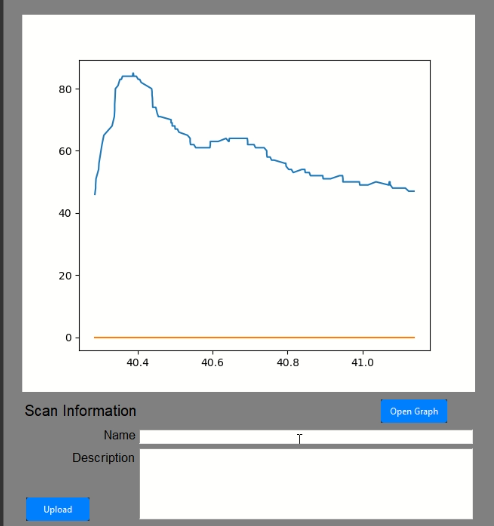
After the x and y coordinates have been reached, the doctor can then manually move the Z-axis, sensor motors in Figure 7, to the necessary height. When the desired location is reached, the scan button can then be pressed to graph the pulse and pressure taken from the pulse sensor and the stop button can be pressed to finish scanning and display the graph in the Graph section. If the doctor chooses to edit the graph, a button called “Open Graph”, located below the graph, can be pressed. The Open Graph window, shown in Figure 10, will display the graph and allow the doctor to select a snippet by left clicking for the start position and right clicking for the end position. When finished, the selected snippet, indicated by a red hue, will display as the graph on the main application.



**Figure 10.1: Manual Mode: Edit the Graph**

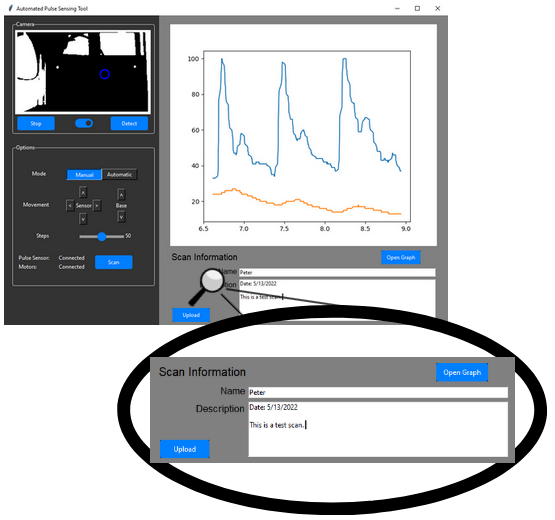


**Figure 10.2: Manual Mode: Edit the Graph**

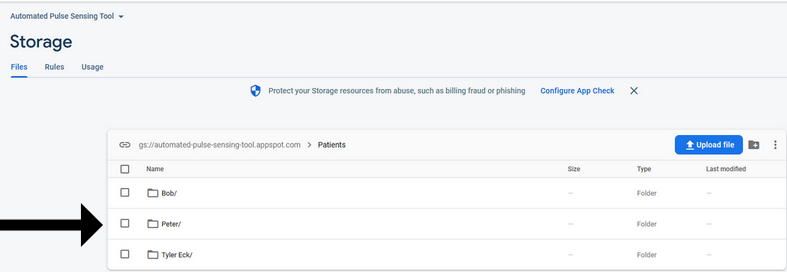


**Figure 10.3: Manual Mode: Edit the Graph**

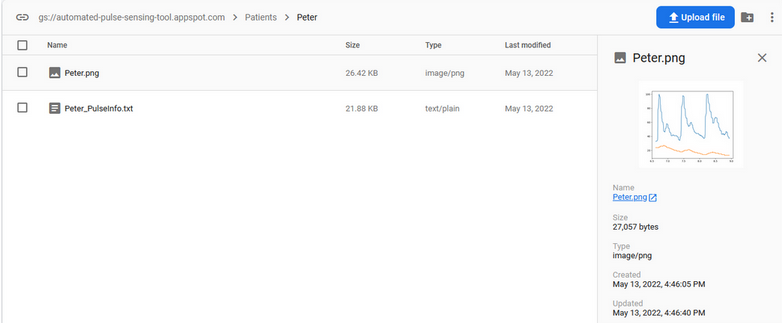
Lastly, the name and description areas are for the doctors to fill in necessary information regarding the pulse scan. The Upload Button can then be pressed to upload files to Firebase Storage. As shown below in Figure 11, the two files that are uploaded, the picture of the reading and the information text file, are both inserted into folders named after the patient.



**Figure 11.1: Manual Mode: Upload Graph**



**Figure 11.2: Manual Mode: Upload Graph**



**Figure 11.3: Manual Mode: Upload Graph**

Automatic mode does the detection of the pulse point via the camera, the movement of the motors based on that pulse point, and the scanning of the pulse automatically. As a result, the patient would just need to enter their name and other information and then upload the scan. Because this mode is intended for the general user (patients), when the Automatic button is pressed, a series of instructions are displayed. The user will need to read these instructions before the automation occurs. Shown below are the instruction pages.

