IoT-Based Patient Monitoring Device

Matthew Lowe, Matthew Baldwin, Tyler Dieter
Department of Mechanical Engineering
Northern Illinios University
Dekalb, IL United States

***Abstract*—the IoT is the Internet of Things which promotes increased informational utility by increasing the connectivity of devices either by Bluetooth or hardwired connection. These principals were applied to an integrated system which is capable of running three independent functions (data collection and processing, database updates, GUI and display) across two distinct processors (Raspberry Pi and Arduino Uno) with the ability to include additional hardware modules in the future.**

Keywords- IoT; GUI; Health Metrics; Database; Sensor

I. Introduction

With the rise of computing technology becoming widely available, transparency and accessibility to information is expected to increase. The importance of accessibility in health information and services has inspired the design of a IoT-based health monitoring system. The conception was to allow an individual of any level of technological comfortability – hereafter referred to as the user – to interact with a high degree of transparency with their own biological measurements. Specifically, this would ensure either safety during recovery in situations like physical therapy or could also allow for maximizing the efficiency of a workout to allow for greater results or monitor with precision the degree to which a physical activity exerts the body. This data would be made transparent to the user via the graphical user interface (GUI) to display real-time data of a verity of different metrics. The initial conception includes an array of sensors which would measure the heart's beats per minute (BPM), the blood's oxygen percentage level (O2 level), the user's temperature, and the blood pressure. The GUI would display these three values as often as an update is received in three locations that can be viewed simultaneously as well as the duration of the session. Whenever the GUI is updated, the program would also log the values of the health metrics to a database for storage until a bulk transmission would be made. Once the session has been completed by the user, the data would have the option to be sent to a physician or logged likewise via the internet.

II. Design features

The device uses a pulse sensor that is connected an Arduino Uno that is then connected via a wire to a Raspberry Pi. An LCD screen is connected to the Raspberry Pi and it displays a GUI and a graph of the pulse sensor data. The system is powered by a Lipo rechargeable battery that is integrated through a battery board. The Raspberry Pi, LCD display, battery board, and battery are all enclosed within a 3D printed case.

A. Raspberry Pi

The Raspberry Pi is the main hub of the device, it reads from the serial of the Arduino Uno to collect the pulse sensor data, and then displays the information in the GUI. The pi stores all of the necessary files and all the information either flows into or out of it.

B. Arduino Uno

The Arduino Uno is a microcontroller board that is flexible and has numerous libraries and pins that are used to connect to the pulse sensor. It has a C based integrated development environment (IDE) that collects data from the sensor and pushes it to the serial plotter using the Pulse Sensor Playground library. See Figure 1 below.

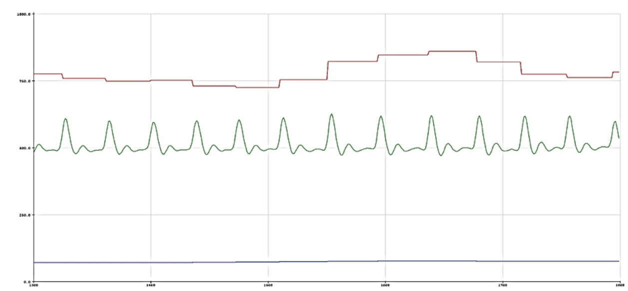

Figure 1: Serial Plotter of Pulse Sensor Data

C. Pulse Sensor

The pulse sensor is used as the main health monitoring sensor for our project. This sensor is able to read beats per minute (BPM), amplitude wavelength, and the inter-beat interval (IBI). The sensor data is read by the Arduino Uno.

D. Power Supply

The battery that powers the whole system is a lithium polymer ion 10,000 mAh rechargeable battery. The size of the battery needed was determined through empirical tests of how long a pair of 3600 mAh batteries lasted under multiple loading conditions. Then this data was linearly interpolated to allow for an expected 8 hours of battery life.

E. Housing

The major mechanical process within the project was the case design. The case was designed with ease of maintenance. The case houses the Raspberry Pi, the LCD screen, and the battery/battery board. The switch on the outside of the case allows the user to turn the device on and off without opening up the case. The cord protruding from the side of the device allows for the battery to be charged easily as well. The ports are left accessible so that the

Arduino can be connected to the raspberry pi without any obstruction. This same slot is also used as a means for providing ventilation to the Rpi. The physical and virtual prototype design for the case can be seen in Figures 2 and 3 below.

Figures 2 and 3: Physical and Virtual Case Designs

F. GUI

The GUI was built using a library that allowed for the insertion of buttons and display the data collection when the aforementioned buttons are pressed. Once the data collection is complete, the stop button is pressed which then sends the collected data to a file which is then handled by the database process. The GUI is displayed in Figure 4 below.

Figure 4: Graphic User Interface

G. Database

Once the data is collected and sent to a file, it is then emailed. The email process utilizes a library to do so, and it is connected to a Gmail account specifically for the device. The email with the data file attached can be sent to any other email address required, including the user for their own records.

III. Communication

Certain limitations reduced the efficacy of the wireless communication being the sole mediator between the main processor and the sensor array peripheral. The complexity of Bluetooth integration directly to the peripherals suggested more reliable results could be obtained by using wired connections from the peripherals to a external processor which would act as a hub for the data and a potential node for the Bluetooth connection specifically acting as the transmitter for the data. This unit was successfully integrated into the prototype and serves as a collection point and early data processing unit. The communication method has the remote processor – the Arduino Uno in the prototype – wired directly to the main processor – the Raspberry Pi – such that communication happens via this cable and the digital information is written to the remote processor's serial port and is read from there by the main processor. However, the digital infrastructure has been left in a modular setting such that future iteration can easily enhance the design by allowing for an upgrade to Bluetooth communication if found to be a viable procedure. However, the Bluetooth protocol structures had unexpected levels of complexity that were prohibitive to the progress of the project within the framework that would require such an inclusion be made.

IV. Conclusion

The design connects the health metric sensors to the main processor which is able to log and display the data for the user and optionally submit the data to a medical professional for review. The user can view the data in real time and begin and end sessions at any time. Ending a session will trigger the submission of the data – if so desired – to an email address that they may designate. Iteration may be performed to increase the mobility of the unit by upgrading the physical connection to a Bluetooth version which would require notable computer science proficiency to accomplish. Like with many electrical components, reducing the excess volume is an area of great interest. To accomplish this would require reducing the size of the housing unit by increasing the packing efficiency in a way that still permits the circulation to properly cool the processor. Another way to improve the device would be allow the display of data over time, likely in the form a graph in the GUI; this can currently be done in the Arduino IDE, but isn't yet implemented into the GUI.

Acknowledgment

The authors would like to acknowledge the College of Engineering and Engineering Technology for providing our group with the means and support to complete this project. We would also like to thank both Dr. Ji-Chul Ryu and Dr. Pradip Majumdar for their support, resources, and guidance during the entire duration.

References

[1] Industries, A. (n.d.). Adafruit. Retrieved April 09, 2021, from https://www.adafruit.com/?gclid=CjwKCAjw9r-DBhBxEiwA9qYUpUvd2y4lgabJYX0VgJUA2VPbIL_VKN_42UWeDVGyAg7o7aok9cGfRxoCcy4QAvD_BwE

[2] Python Package Index. (n.d.). Find, install and publish python packages with the python package index. Retrieved April 09, 2021, from https://pypi.org/

[3] Raspberry Pi. (n.d.). Teach, learn, and make with Raspberry Pi. Retrieved April 09, 2021, from https://www.raspberrypi.org/