Elderly Homes Health Tracking System

ECE 445 Project Proposal

Team 6

Sanjana Pingali (pingali4)

Aishee Mondal (aisheem2)

Jeep Kaewla (ckaewla2)

TA: Akshatkumar Sanatbhai Sanghvi

Professor: Olga Mironenko

Date: 02-07-202

Table of Contents

1. Introduction

1.1 Problem

Many elderly persons may live in elderly homes or retirement homes and have many health-related problems. It is difficult for the staff to keep track of the health of all the individuals and reach them in time in case assistance is needed. It might be hard for them to track problems immediately due to difficulty communicating, and the caretakers would only know when their health is deteriorating once their condition becomes severe. In elderly homes or retirement centers, it might be hard to keep track of many elderly patients at once, especially when technology like Apple Watches or Fitbits is not intuitive for older people, too expensive, and not customized to their needs. It is not feasible to ask the staff to monitor each individualized Fitbit and track it separately without a more centralized system.

1.2 Solution

In this situation, a cost-effective device keeps track of various health data such as heart rate, temperature, blood oxygen, and an emergency like fall detection. In addition, if the elderly person is lost, we would have GPS tracking enabled to determine their location and assist them as needed. Our connecting web application would allow the elderly home caretakers to monitor multiple elderly people simultaneously, track individual health irregularities, and communicate them to the doctors. A notification would be sent on the app when an irregular critical heart rate or breathing activity for a particular person is observed, and an alarm on the person would be triggered. The alarm will also be triggered when a fall is detected. We could also store past health data points in a database and monitor for any irregularities, or doctors can use this during checkups.

The way the process takes place is described below:

A person wears a belt with sensors such as a pulse sensor, blood oxygen, fall detection system, accelerometer, temperature sensor, and GPS tracking. These sensors will measure data and send it to a microcontroller that will then use a wifi module to update a database and reflect changes in our web application. If any data from these sensors is outside normal parameters, then the microcontroller will send an alert to a beeper and a notification to the web application.

1.3 Visual Aid

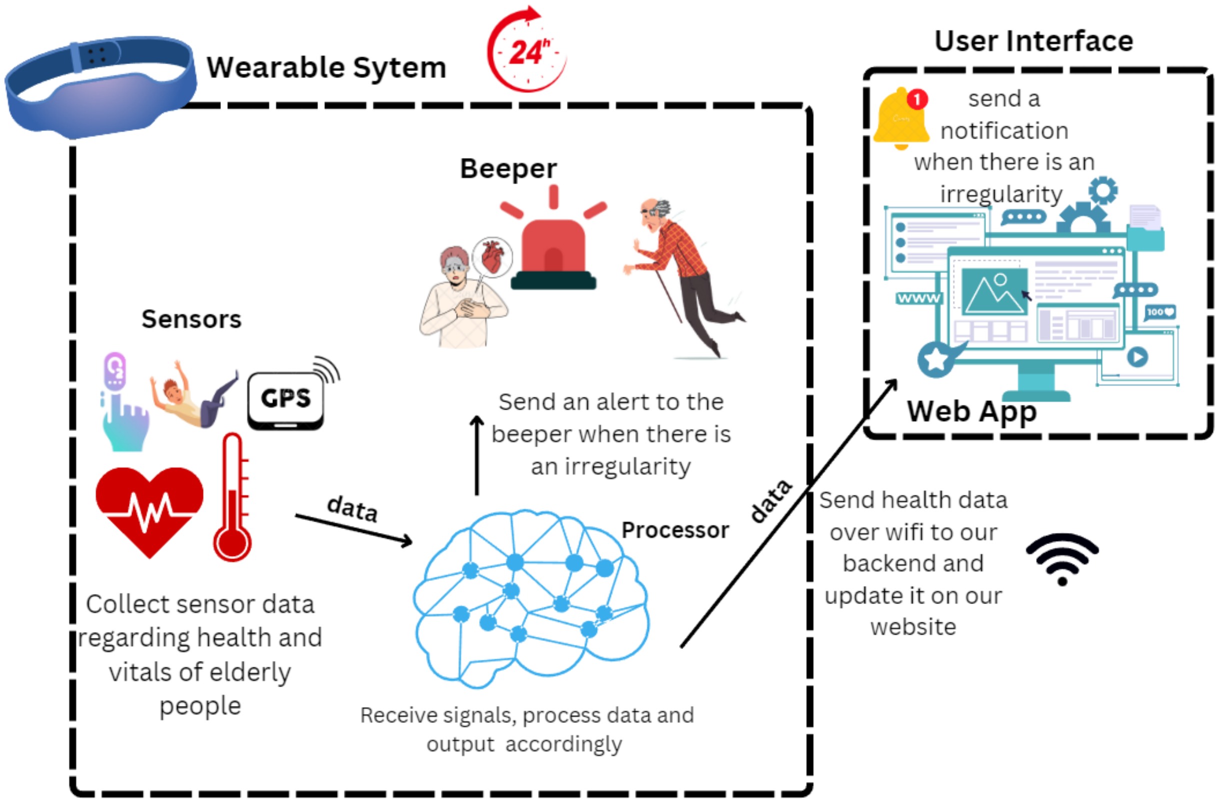

Figure 1. A visual representation of our project consisting of the three different subsystems

1.4 High-level requirements list

- A notification is sent to the web app within 2 mins ± 30 secs when there is an irregular heart rate, temperature, or fall detection.
- The staff can monitor close to real-time heart rate, temperature, GPS location, and fall detection of the person within 2 minutes ± 30 seconds.
- The beeper on the belt emits an alert within 30 seconds of an irregularity being observed to attract immediate attention.

2. Design

2.1 Block Diagram

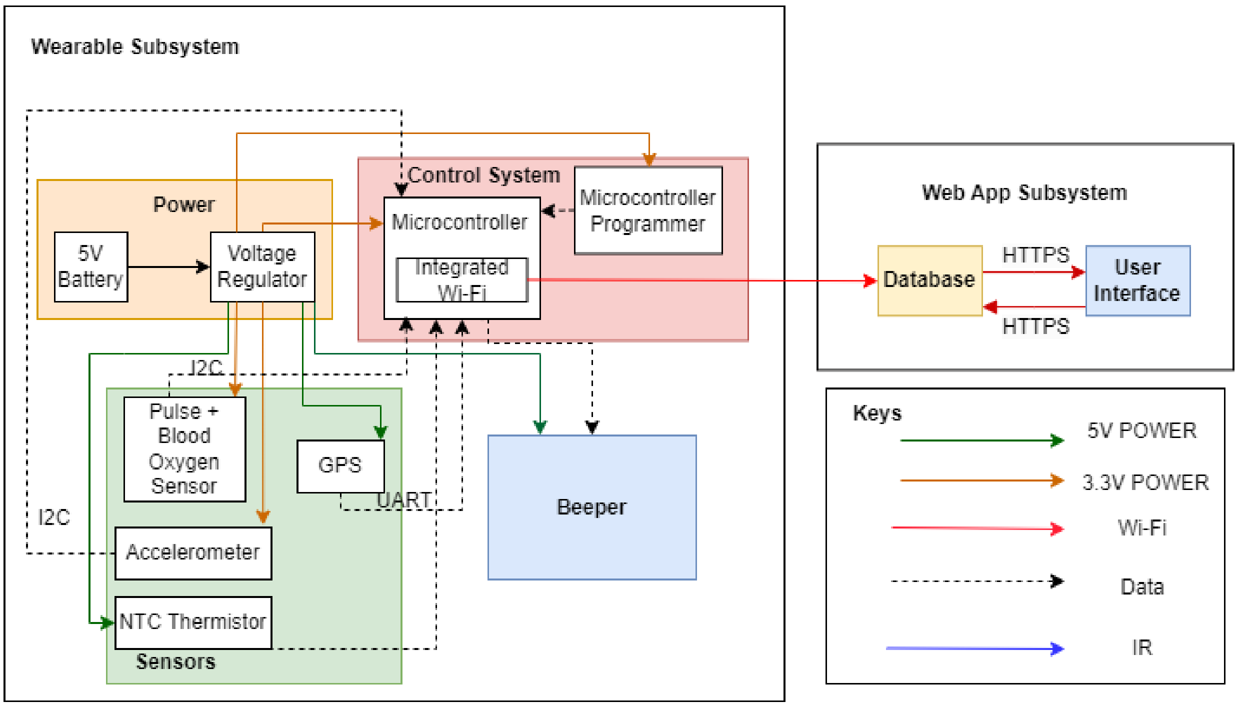

Figure 2. The Block Diagram of our project detailing the subsystems and corresponding connections

2.2 Subsystem Overview

<u>**Wearables System**</u>

Sensors Subsystem

The sensor subsystem collects various health parameters and indicators. These sensors will be placed in a box on a wearable belt with a system attached to the chest to measure pulse and blood oxygen levels. We are planning on implementing the following sensors:

- **Heart Rate and Blood Oxygen**: The pulse rate and oxygen concentration measurement (**MAX30102**) sensor [1]. This sensor will be placed near the person's chest in an insulated sticky pad to detect the person's heart rate. As richer-oxygen hemoglobin absorbs more infrared light, we observe the amount of infrared radiation absorbed to see the amount of oxygen in the blood.

Requirement: Provide an accurate (± 5 bpm w.r.t. to existing smartwatch measurement, ± 2% oxygen level) and continuous reading of the heart rate and blood oxygen levels with as close to real-time data (latency of 2 minutes ± 30 seconds)

- **Temperature**: The temperature sensor is placed near the chest in an insulated sticky pad and is used to constantly monitor the person's temperature to ensure it is within acceptable bounds. For the temperature sensor, we have two possible options-
    - A temperature sensor circuit can be created that uses the NTC thermistor (**GA3K3A1A**) and an analog-to-digital converter connected to the microcontroller. This circuit will allow us to calibrate the temperature sensor with the temperatures we want [2].
    - The second option is to get an already calibrated temperature sensor chip directly connected to our microcontroller.

  Requirement: Provide an accurate ± 2C of reading of body temperature. The user interface would display this temperature as close to real-time (2 minutes ± 30 seconds).

- **Accelerometer**: This sensor (**MPU-6050**) is used for fall detection with acceleration across three axes and a 3-axis gyroscope [3]. The accelerometer measures motion in three dimensions as well as orientation. When a person falls, there is a sudden increase in the magnitude of acceleration with a change of orientation and then a decrease in the magnitude of acceleration since the person is no longer moving [4]. This sensor will help us observe such drastic changes in motion and, therefore, be able to detect if a person has fallen. Additional Feature: If time permits, we are also thinking of detecting the amount of exercise using the accelerometer to keep track of the step count.

  Requirement: Accurately detect if a person has fallen and send an alert on the beeper within 2 minutes ± 30 seconds. We would also send a notification on the user interface for assistance.

- **GPS**: We will use the **NEO-6MV2** GPS sensor, which will provide us with the latitude and longitudinal coordinates to pinpoint the individual's exact location.

  Requirement: Provide an accurate reading of where the elderly person is within ±50m (horizontal position)

Control Subsystem

The control subsystem processes the collected health data. With the integrated Wi-Fi, we would send POST requests to our backend API to store this health information in our database.

- **Microcontroller (ESP32)** - The microcontroller sends a signal to the beeper to sound an alarm when there is an irregular heartbeat or temperature, or the person has fallen. This microcontroller already has Wi-Fi integrated, so we do not need an additional Wi-Fi module [5].

Requirement: Process and send the health data to be stored to the backend API within 1 minute ± 30 seconds

Power Subsystem

The power subsystem powers our entire circuit. This subsystem consists of the following components:

- 5V rechargeable battery, which can also be easily replaced.
- The voltage regulator (**LM1117-3.3V**) ensures that we supply a steady, constant voltage supply through all the components depending on the voltage requirements of the component [6] [7].

Requirement: To provide the correct voltages to each subsystem from the regulator, which would be 3.3V ± 0.5V or 5V ± 0.5V.

Beeper Subsystem

The beeper subsystem serves to sound an alarm to call for assistance to the person. This component will play the sound when the microcontroller sends a signal in case of irregularity with the heartbeat, body temperature, and a person falling. The component used in this subsystem is a beeper (**CMI-1295IC-0585T**).

Requirement: Be able to make a sound when heart rate/ temperature is outside the normal range with a latency of 1 minute ± 30 seconds.

Web Application System

Database Subsystem

The database subsystem stores the processed health information we collected from the wearable sensors. We intend to use MongoDB as it is more scalable and allows more flexibility to the structure of the stored data compared to MySQL.

Requirements: Storage of heart rate information of the last week ± one day, temperature data for a week ± one day, and blood oxygen levels for a week ± one day.

User Interface Subsystem

This web interface allows the staff to detect everyone's vitals, and where the staff receives a notification when a person's vitals are outside of the normal range. We will be using React and Javascript to create this front-end web platform. The frontend web page would display close to real-time heart rate and body temperature of multiple elderly.

Requirements: Display the heart rate and body temperature within 2 minutes and 30 seconds after measurements.

2.3 Tolerance Analysis

When using a voltage regulator to provide each subsystem with the required amount of constant voltage, it is possible that the regulator can overheat based on the power used at this regulator [8]. Therefore, we will calculate and analyze the amount of power used at this regulator based on the different components it supplies voltage to and at what voltage it does. Based on how high it heats up, we can check whether this is below the maximum temperature range for the device, and this method can prevent overheating.

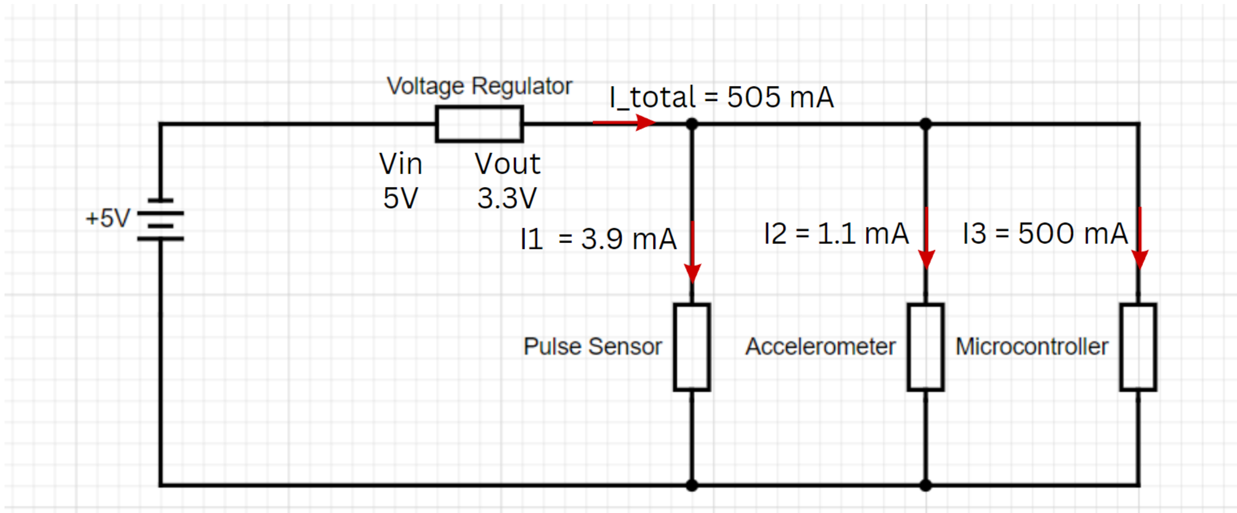

Figure 3: Abstract circuit for tolerance analysis calculation

The thermal resistance of the voltage regulator **(LM1117-3pin)**, $\theta_{Ja} = 23.8$ degree C/W [6]
For the sensors which are connected to a 5V battery and the corresponding voltage regulator using 3.3V within the **Wearable Device**: Pulse Sensor, Accelerometer and Microcontroller draw the current as following:
Pulse Sensor: At 3.3V, for resistance of $3k\Omega$, current supplied, $I_1 = 1.1mA$ [1]
Accelerometer Sensor: At 3.3V, for resistance of $10k\Omega$, current supplied, $I_2 = 3.9mA$ [3]
Microcontroller: At 3.3V, current supplied, $I_3 = 500mA$ [5]

Adding up all the current, we get total current from the voltage regulator at 3.3V, $I_{total} = I_1 + I_2 + I_3$
= 3.9 + 1.1 + 500 mA = 505 mA
Voltage drop, $V = V_{in} - V_{out}$ = 5 - 3.3 V = 1.7V
Power consumed, P = $V * I_{total} = 1.7 * 505 * 10^{-3} = 0.8585W$
Temperature change, $\Delta T = P * \theta_{Ja} = 23.8 * 0.8585 = 20.4323$ degree Celsius.

If we add the temperature difference to the room temperature, max temperature achieved = 25 degree C + 20.4323 degree C = 45.4323 degree Celsius.

Since this is well within the maximum temperature of 125 degree C of the voltage regulator, the circuit is safe for operation within the wearable device and the voltage regulator will not blow up.

Figure 4: Mathematical verification of tolerance analysis for Wearable subsystem

3. Ethics and Safety

1) Patients' healthcare information should not be disclosed and should be protected under secure databases. We would not collect any data or healthcare information of the patient without consent (under the Privacy Rule and Security Rule of HIPAA). This issue is also closely related to the code IEEE 7.8.I.1, wherein the patient's health and privacy should be the foremost concern, and our design will ensure in no way are these harmed or violated [9]. We would avoid this ethical breach by ensuring that the elderly are aware of the health activities collected before putting the device on. We would also ensure that our database is secured before releasing this innovation to the public to prevent any protected health information leaks.

2) Related to IEEE code 7.8.II.9, we would ensure that our product does not physically harm any individual [9]. We will avoid this ethical breach by ensuring that covering and material in contact with the patient are not toxic. Additionally, we will ensure that radiation levels are not harmful, that all parts are encased in a protective covering, and that no live wires or power connections could be in contact with the person.

3) According to the ACM Code of Ethics 2.9, we should design and build robust and usable systems [10]. To ensure we can follow this ethical code, we would robustly test our software to ensure that our web application works and can handle displaying close to real-time health information.

References

[1] Maxim Integrated, "High-Sensitivity Pulse Oximeter and Heart-Rate Sensor for Wearable Health", MAX30102 datasheet, Oct. 2018.

[2] L. Ada, "Thermistor," *Adafruit Learning System*. [Online]. Available: https://learn.adafruit.com/thermistor/using-a-thermistor. [Accessed: 07-Feb-2023].

[3] InvenseSense, "MPU-6000 and MPU-6050 Product Specification Revision 3.4", PS-MPU-6000A-00 datasheet, Aug. 2013.

[4] D. Patel, "How to build a fall detector with Arduino: Arduino," *Maker Pro*, 07-Feb-2023. [Online]. Available: https://maker.pro/arduino/tutorial/how-to-build-a-fall-detector-with-arduino. [Accessed: 07-Feb-2023].

[5] Expressif Systems, "ESP-32 Series En", ESP-32 datasheet, 2023.

[6] Texas Instruments, "LM1117 800-mA Low-Dropout Linear Regulator", LM1117 datasheet, Jan. 2016.

[7] M. Staff, "What is a voltage regulator?," *Digi*, 01-Jul-2020. [Online]. Available: https://www.digikey.com/en/maker/blogs/2020/what-is-a-voltage-regulator. [Accessed: 07-Feb-2023].

[8] A. Hughes, "What is thermal resistance modeling?: Simscale Blog," *SimScale*, 21-Apr-2022. [Online]. Available: https://www.simscale.com/blog/what-is-thermal-resistance/. [Accessed: 07-Feb-2023].

[9] "IEEE Code of Ethics", IEEE. [Online]. Available:https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed: 06-Feb-2023].

[10] "ACM Code of Ethics", ACM. [Online]. Available: https://www.acm.org/code-of-ethics [Accessed: 06-Feb-2023].