



Home Health Monitoring System

2020-12-04

Ziyad Allehaibi, [zlehaibi](#)

Terry Edwards, [tedwa98](#)

Ben Whalin, [vaeca21](#)

Elijah Rose, [elirose](#)

Mentor: Gregory Myers (gmyers), [gmyers](#)

Instructor: Abdollah Mirbozorgi, [samir](#)

Abstract

The idea behind the senior design project through EE-498 and 499 is to demonstrate the ability to design, model, and eventually create a prototype of the chosen team project. Throughout this project, the team develops a further understanding of teamwork regarding communication and delegation. The idea behind this project is to develop a home health monitoring system controller that has the ability to connect multiple subsystems to approach different aspects of a client's needs. This was done by designing the systems hub around the goal of creating a modular system that would fit around a client's flexible needs. The design process consisted of creating a design table that would display all design decisions made throughout the process of creating the system. To ensure the best choices were made throughout the design process, each decision table was created to display and ultimately choose the best decision for each part of the design process. In each decision table, numerous weighted engineering characteristics were used to calculate the chosen option. These decision tables were compiled to create the design table used to create the prototype.

Table of Contents

| | |
|-------------------------------------|-----------|
| Introduction | 3 |
| Background | 4 |
| Existing Solutions | 4 |
| Project Description | 4 |
| Sample | 5 |
| Capabilities | 5 |
| Machine Learning | 5 |
| Dictionary | 5 |
| Design Process | 6 |
| Methods | 6 |
| Decision Tables | 6 |
| Design Tables | 7 |
| Input Table | 11 |
| Output Table | 11 |
| Timeline | 12 |
| Subsystem Analysis | 15 |
| System Design | 15 |
| Hardware Design | 15 |
| Sensor Subsystem Analysis | 16 |
| Hardware Simulation | 16 |
| Software Design | 16 |
| Software Simulation | 19 |
| Cost Analysis | 19 |
| Individual Tasks | 21 |
| Ziyad Allehaibi | 21 |
| Terry Edwards | 21 |
| Ben Whalin | 21 |
| Elijah Rose | 22 |
| Appendix | 23 |
| Appendix A: Poster | 23 |
| Appendix B: Final Slides | 23 |

Introduction

This report will outline and document the design methods and decisions made throughout the entire semester in the first semester of senior design. The project will demonstrate basic knowledge and understanding of both hardware and software systems as well as the capabilities of working in a virtual team environment. The goal of this first semester was to complete the entire design process in order to focus on the implementation next semester so that the group might arrive with a successful final senior design product at graduation. The communication abilities and knowledge of required tool sets were tested and displayed through the semester.

For the purpose of this project, the semester began with defining the problem: How can one monitor their partially dependent family member while the user is away from their home? Although there are many options that someone could choose to pursue such as a nursing home or a live-in nurse, these can be extremely expensive. Not only does the price of care create a ginormous burden on the family members, the additional factors of Covid-19 now come into play. Family members are now limited to either extremely limited and controlled visiting hours or in some cases visitors are no longer allowed to visit nursing homes to see their loved ones. In a time of desperation and near world-wide terror, being restricted from seeing family members can have a huge impact on the moral in elderly people - this can/could have a direct impact on the persons health.

Like mentioned in the previous paragraph, the competition that comes with monitoring loved ones are in-home nurses, nursing homes, and current in-home “smart” monitoring systems. So why not choose one of these options? To begin: in-home nurses can be extremely expensive, costing up to ninety dollars an hour, and in many cases have minimum hours each day. Over-night and whole day personal care can easily get out of hand when it comes to pricing and in many families, this is not a feasible option. Next, in the United States of America, the average nursing home or assisted living facility can cost upwards of \$4,000 a month. Over the course of five years, this will cost the family \$240,000.

So, what is the solution here? Develop a home health monitoring system hub that can connect multiple subsystems to approach different aspects of a client’s needs. The system would have the ability for a user to add on any type of subsystem/sensor that they desired as the system would be designed around the concept of modularity. This means that the system would be designed in such a way that the subsystems would be divided into their own independently created systems that will have the ability to be created, modified, replaced, or exchanged with other subsystems. The subsystems added would have the ability to measure everyday aspects of an individual’s health from information like their movement throughout the house to how many times they have used the restroom or even taken a shower. The limits to what aspects of the user’s day to day life that is monitored by the home health monitoring system is limited only to the user’s imagination – and of course knowledge of hardware.

To begin with the project, stakeholders as well as the lines that needed to be drawn regarding the scope of the project were discussed. This discussion was then taken and revised to create the problem statement that was used to brainstorm the initial top-level diagram. After multiple revisions, the final diagram can be seen in [Figure 1] and [Figure 2] below.

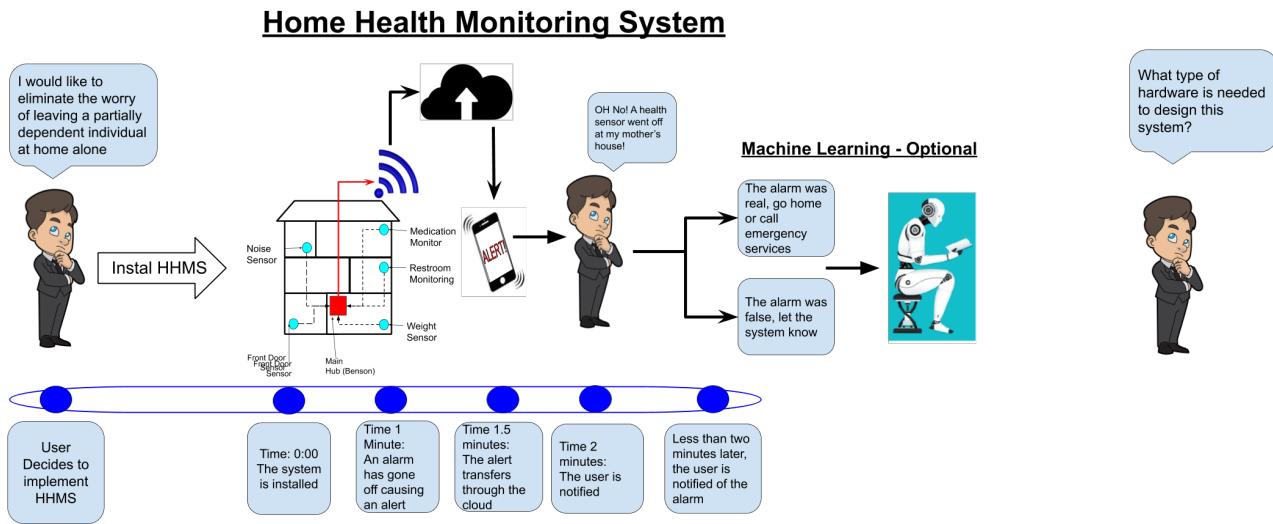


Figure 1: Top Level Diagram

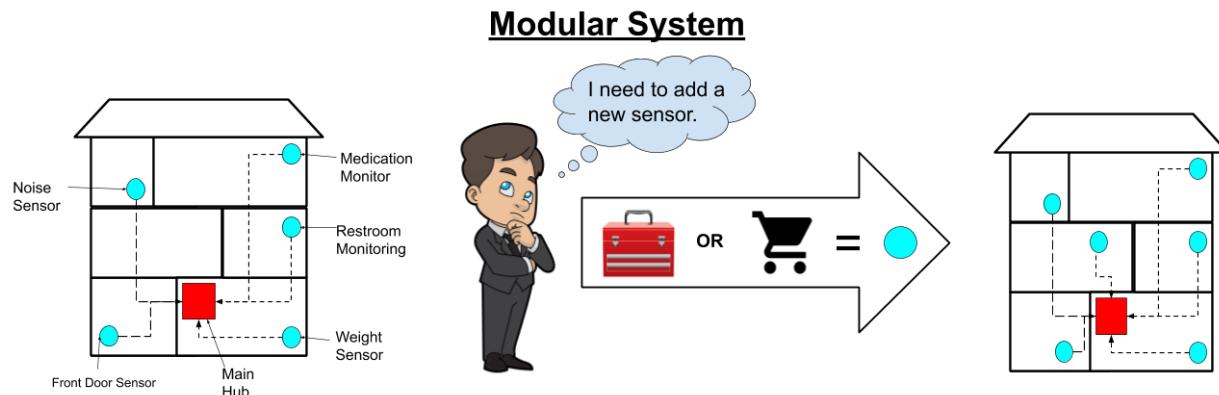


Figure 2: Top Level Diagram Part 2

Background

Existing Solutions

Project Description

In order to address the aforementioned problems, the team proposes an open-source home health monitoring system with an API to encourage extensibility to third-party devices. The system, initialized as **HHMS**, seeks to create and focus on a framework moreover than a single implementation, allowing relatively simple additions to the system without necessitating a recreation of the multi-step communication setups and data processing.

The project thusly focuses on establishing this framework and its API, allowing near any device to integrate with the system given it adheres to the protocols set by the system.

Sample

While the framework is the main focus, it is difficult to show that a framework exists without a working solution, and indeed multiple. If the system only demonstrated a single communicating device, it would hardly justify or prove the extensibility. Hence several sensor sub-systems will be set-up to work with the framework as an example of its capability and proof-of-concept.

Capabilities

- **Data Acquisition and Communication:** the user shall be able to read the data from their sensor subsystems, as well as communicate back to them in order to set their properties.
- **Data Processing:** data shall be processed according to set rules in order to indicate discrepancies with expected behavior.

Machine Learning

In order to further aid the efforts of the caretaker, behavior anomaly recognition should be implemented in order to detect irregular patterns or sharp steps in established patterns. The implementation will allow short-term recognition of emergencies (e.g. falls) as well as unhealthy longer-term patterns (such as a sudden drop in weight over a few days).

Dictionary

- Partially-Dependent Individual (PD): the stakeholder who will be monitored and cared for by the Caretaker
- Caretaker: a primary stakeholder, uses HHMS to monitor their PDs
- Data-Acquisition Hub (DA or DAH): the component that resides in the user's home to acquire and transmit data (as well as potentially more). Previously synonymous with "Hub" and at times "Microprocessor"
- Sensor Subsystems (SS): a setup that allows communication of sensor data to the DAH. Synonymous in the context of the DAH as a Slave, and at times synonymous with Microcontroller.
 - We additionally need a term that refers to the abstract collection of devices that can be enslaved to the DAH as mapped by software, undecided on this.
 - Do not confuse with sensors, which may comprise the entirety of the SS or be merely a component. The SS wraps the sensor. Likewise, the SS wraps the Microcontroller and/or Microprocessor, which is why it may be inappropriate to call it as such.

Design Process

After establishing the main [project goals](#), design began with the overarching view of the project and brain-storming solutions via conceptual block diagrams, as d

An early conceptual diagram for the outline of the software systems.

To this end, there is already a necessary separation in the software – data-streams of the connected devices and metadata used by the system to interpret the data-streams – as well as hardware – sensor component and system wrapping microcontrollers. Additionally, due to the emphasis on a DIY and simple extensible set-up, user communication through the system is inherent to the design.

After the broad design was established, the individual components and implementation details were established in decision tables using a calculated Figure of Merit [1] system. These decisions were then processed into the design table that outlines the final component-level decisions. Due to the inexperience of team Engies, this system is crucial in order to determine the proper materials for the project and develop the parts list without merely relying on the limited scope of products introduced in courses.

Methods

Decision Tables

The decision tables compile the research into the relevant specifications

| Micro Controllers | Input Types | | | | | | | | |
|--------------------------------|-------------|--------|---------|---------|---------|--------|--------|--------|-------------------|
| | Bluetooth | Wifi | USB 1.1 | USB 2.0 | USB 3.0 | USB C | RS-232 | HDMI | Type A - Standard |
| Weights: 0-10; Values: Boolean | 0 | 1 | 0 | 5 | 5 | 2 | 1 | | 4 |
| | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | |
| Raspberry Pi 4B 4GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | | 0 |
| Raspberry Pi 4B 8GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | | 0 |
| Arduino NANO 33 IOT | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 0 |
| Banana Pi M3 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Odroid XU4 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | | 1 |
| NanoPi NEO4 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | | 1 |
| UDOO BOLT V8 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | | 1 |
| UDOO X86 II ULTRA | 1 | 1 | 0 | 0 | 1 | 0 | 0 | | 1 |
| ASUS Tinker Board | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Onion Omega2+ | 0 | 1 | 0 | 1 | 0 | 0 | 0 | | 0 |
| Orange Pi 4B | 1 | 1 | 0 | 1 | 1 | 0 | 0 | | 1 |
| NanoPC-T3 Plus | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Le Potato - AML-S905X-CC | 0 | 0 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Orange Pi Zero Plus2 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | | 1 |
| Raspberry Pi Zero W | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 0 |
| Intel® NUC Board NUC7i3DNBE | 1 | 1 | 0 | 1 | 1 | 0 | 0 | | 1 |

Figure 2: Decision Table 1

| Microprocessors | Bluetooth | Wifi | USB 1.1 | USB 2.0 | USB 3.0 | USB C | HDMI | Type A - Standard | HDMI Type C - Mini |
|---------------------------------|-----------|------|---------|---------|---------|-------|------|-------------------|--------------------|
| Weight: 0-10 Values: Boolean | 10 | 7 | 1 | 5 | 5 | 2 | 1 | | 4 |
| Raspberry PI 4B 4GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | | 0 |
| Raspberry PI 4B 8GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | | 0 |
| Arduino NANO 33 IOT | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 0 |
| Banana Pi M3 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Odroid XU4 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | | 1 |
| NanoPi NEO4 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | | 1 |
| UDOO BOLT V8 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | | 1 |
| UDOO X86 II ULTRA | 1 | 1 | 0 | 0 | 1 | 0 | 0 | | 1 |
| ASUS Tinker Board | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Onion Omega2+ | 0 | 1 | 0 | 1 | 0 | 0 | 0 | | 0 |
| Orange Pi 4B | 1 | 1 | 0 | 1 | 1 | 0 | 0 | | 1 |
| NanoPC-T3 Plus | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Le Potato - AML-S905X-CC | 0 | 0 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Orange Pi Zero Plus2 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | | 1 |
| Raspberry Pi Zero | 1 | 1 | 0 | 1 | 0 | 0 | 0 | | 1 |
| Intel® NUC Board NUC7i3DNBE | 1 | 1 | 0 | 1 | 1 | 0 | 1 | | 0 |

Figure 3: Decision Table 2

Design Tables

To ease with the design process, a design table was created to document and display each design decision that was made throughout the semester. The table shown below in [Figure 4] shows each decision that was made from the weighted tables as well as the problem statement, approach, and necessary equipment in order to create the home health monitoring system. The design table was then used to create the parts list towards the end of the report.

For each item in the design table a sub table was created to show the decision processes and to choose the best option based on the desired engineering characteristics and weights. Every support table lists each possible option that was found through extensive research and then the engineering characteristics that were chosen to help narrow down the most logical choice.

| Problem Statement | Approach | Solution | Features | Approach | Hardware Needed | Price (USD) | Link |
|---|---|-------------------|-------------------------------------|---|---|-------------|---|
| How can one monitor their partially dependant family member while the user is away from their home? | Develop a home health monitoring system hub that has the ability to connect multiple subsystems to approach different aspects of a client's needs | | | | | | |
| Hub (Benson) | | Hub Case Material | Data Processing Input Output | Use the hub as a transport to relay the data to an offsite software to process the data Raspberry Pi 4B 8GB WiFi Bluetooth USB 2.0 USB 3.0 USB C HDMI Type D - Micro Ethernet WiFi | Raspberry Pi 4B 8Gb | 75 | https://www.pishop.us/product/raspberry-pi-4-model-b-8gb?src=raspberrypi |
| | | | Store Bought Case | Raspberry Pi 4 Case, Red/White | Raspberry Pi 4 Case, Red/White | 5 | https://www.pishop.us/product/raspberry-pi-4-case-red-white/ |
| | | | Case Cooling Fan/Heatsink | Case Fan & Heatsink For Raspberry Pi 4 Case | Case Fan & Heatsink For Raspberry Pi 4 Case | 5 | https://www.pishop.us/product/case-fan-heatsink-for-raspberry-pi-4-case/ |
| | | | Temporary Data Storage (Hub) | 64GB Storage | Class 10 microSD card - 64GB | 14 | https://www.amazon.com/Sandisk-Extreme-microSD-UHS-I-Adapter/dp/B007FCMBLV/ref=sr_1_47?dchild=1&keywords=micro+s+d+card+64gb+class+10&qid=1603739378&r=4 |
| | | | Temporary Data Storage (SubSystems) | 32GB Storage | Class 10 microSD card - 32GB | 7 | https://www.amazon.com/Sandisk-Ultra-microSD-Memory-Adapter/dp/B007JWVGN7/ref=sr_1_32?dchild=1&keywords=micro+s+d+card+32gb+class+10&qid=1603739378&r=3 |
| | | | Power Supply | Wired | USB-C Power Supply | 8 | https://www.pishop.us/product/usb-c-power-supply-5-v-3-0-a-black-ul-listed/ |
| Subsystems | | Client Interface | Magnetic Contact Switch Sensor | Monitor movement in the house Monitor if a person has gone to the restroom Monitor exterior doors | Magnetic Alarm Contact Switch | \$27 | https://www.amazon.com/Gikfun-Magnetic-Contact-Sensor/ |
| | | | Button For User Input | Has the person taken their medication? Did the person take a shower or bath? Has the person had a meal? | Small Static Button | \$33 | https://www.newark.com/ak-devic-hvdcml#hvaladnt=hval |
| | | | Light Level Sensor | Ensure that lights have not been left one throughout the home Make sure the user is moving throughout the day (lights turning off and on) | Photoresistor Photo Light Sensitive Resistor Light Dependent Resistor | \$38 | robotica.com/shop/tmp36-temperature |
| | | | Temperature Sensor | Monitor the temperature throughout the house | TMP36 Temperature Sensor | \$29 | https://www.amazon.com/gp/pro |
| Client Interface | | Log Interface | dot Net Core | dot Net Core | dot Net Core | | |
| | | | Alert Types | Email | dot Net Core | | |

Figure 4: Design Table

Many of the decision tables revolved around the systems hub/controller. The first table that was created to help ease the decision on which microcontroller to implement, the output and input types of each microcontroller were added to the support tables seen below in [Figure 5] and [Figure 6]. The given outputs would be needed to connect to the systems data processing controller – meaning the outputs must be compatible with Wi-Fi.

| Output Types | | | | | | | | | | |
|--------------------------------|-----------|------|---------|---------|---------|-------|------------------------|--------------------|---------------------|---------|
| Microprocessors | Bluetooth | Wifi | USB 1.1 | USB 2.0 | USB 3.0 | USB C | HDMI Type A - Standard | HDMI Type C - Mini | HDMI Type D - Micro | RESULTS |
| Weight 0-10 Values: Boolean | 10 | 10 | 1 | 5 | 5 | 2 | 1 | 2 | 3 | |
| Raspberry Pi 4B 4GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 32 |
| Raspberry Pi 4B 8GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 32 |
| Arduino NANO 33 IOT | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 25 |
| Banana Pi M3 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 27 |
| Odroid XU4 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 12 |
| NanoPi NEO4 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 34 |
| UDOO BOLT V8 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 29 |
| UDOO X86 II ULTRA | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 27 |
| ASUSTinker Board | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 27 |
| Onion Omega2+ | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 15 |
| Orange Pi 4B | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 32 |
| NanoPC-T3 Plus | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 27 |
| Le Potato - AML-S905X-CC | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 7 |
| Orange Pi Zero Plus2 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 22 |
| Raspberry Pi Zero | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 27 |
| Intel® NUC Board NUC7i3DNBE | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 31 |

Figure 5: Micropocessor Output Types

| Input Types | | | | | | | | | | | | | | |
|-----------------------------|-----------|--------|---------|---------|---------|--------|--------|--------|-------------------|--------------------|---------------------|----------|---------------------------|--------|
| Micro Controllers | Bluetooth | WiFi | USB 1.1 | USB 2.0 | USB 3.0 | USB C | RS-232 | HDMI | Type A - Standard | HDMI Type C - Mini | HDMI Type D - Micro | Ethernet | Audio Jack - 4 Pole 3.5mm | |
| Weight: C10; Value: Boolean | 0 | 1 | 0 | 5 | 5 | 2 | 1 | 4 | 2 | 3 | 10 | 1 | | |
| Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: | Value: |
| Raspberry Pi 4B 4GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Raspberry Pi 4B 8GB | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Arduino NANO 33 IOT | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Banana Pi M3 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| Odroid XU4 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| NanoPi NEO4 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| UDOO BOLT V8 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| UDOO X86 II ULTRA | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| ASUS Tinker Board | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| Onion Omega2+ | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Orange Pi 4B | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| NanoPC-T3 Plus | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| Le Potato - AM-L-S905X-CC | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| Orange Pi Zero Plus2 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Raspberry Pi Zero W | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Intel® NUC Board NUC7i3DNBE | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |

Figure 6: Microprocessor Input Types

In both the input and output support tables for the microprocessor above, the Raspberry Pi 4B 8Gb was the obvious answer but before we could be sure that this was the decision that needed to be made for the final design, a support table regarding the speed and power of each microprocessor needed to be created. This table can be seen below in [Figure 7].

| Microprocessors | | | | | |
|-----------------------------|--|-----------------------------|---------------------|---------------------------|----------------------------------|
| Microprocessors | Processor Name | Processor Speed | Number of Cores | Onboard Ram | Integrated GPU |
| | | Weight: 10 Value in GHz: | Weight: 7 Value: | Weight: 7 Value in GB: | Weight: 3 Value: 1: yes 0: no |
| Raspberry Pi 4B 4GB | Cortex-A72 | 1.5 | 4 | 4 | 1 |
| Raspberry Pi 4B 8GB | Cortex-A72 | 1.5 | 4 | 8 | 1 |
| Arduino NANO 33 IOT | SAMD21 Cortex-M0+ | 0.048 | 1 | 0.000032 | 0 |
| Banana Pi M3 | Allwinner A83T ARM Cortex-A7 | 1.8 | 8 | 2 | |
| Odroid XU4 | Samsung Exynos422 Cortex-A15 and Cortex-A7 Octa core | 2/1.4 | 8 | 2 | 0 |
| NanoPI NEO4 | Rockchip RK3399 64-bit Dual Core Cortex-A72 + Quad Core Cortex-A53 | 2/1.5 | 2/4 | 1 | 1 |
| UDOO BOLT V8 | AMD Ryzen™ Embedded V1605B Quad Core | 2.0/3.6 | 4 | up to 32 | 1 |
| UDOO X86 II ULTRA | Intel Pentium N3710 | 2.56 | 4 | 8 | 1 |
| ASUS Tinker Board | Rockchip Quad-Core RK3288 | 1.8 | 4 | 2 | 1 |
| Onion Omega2+ | MT7688 SoC | 0.58 | 1 | 0.128 | 0 |
| Orange Pi 4B | Rockchip RK3399 | 2 | 6 | 4 | 1 |
| NanoPC-T3 Plus | Samsung S5P6818 | 1.4 | 8 | 2 | 0 |
| Le Potato - AML-S905X-CC | ARM Cortex-A53 | 1.512 | 4 | 2 | 1 |
| Orange Pi Zero Plus2 | Cortex-A53 | 1.2 | 4 | 0.512 | 1 |
| Raspberry Pi Zero W | BCM 2835 SOC | 1 | 1 | 0.512 | 0 |
| Intel® NUC Board NUC7i3DNBE | Intel® Core™ i3-7100U | 2.4 | 2 | up to 32 | 1 |

Figure 7: Microprocessor Support Table

The next decision that needed to be made regarding the hub/controller can be seen in [Figure 8]. The material the case that will hold the controller needed to be determined. After creating the table and beginning to fill out the engineering characteristics for each option, it was obvious due to the convenience and the extremely low price, that a store-bought case was the best option.

| Hub Case Material | | | | | | | | | |
|-------------------|--------|------------|---------------|------------------|------------|-------------|--------|--------|---|
| Material | Price | Durability | Availability | Production Speed | Lifespan | Replacement | Weight | Size | Link |
| | 10 | 5 | 10 | 5 | 3 | 7 | 6 | 8 | |
| | Value: | Value: 1-4 | Value: 0 or 1 | Value: 1-4 | Value: 1-4 | Value: | Value: | Value: | |
| 3d print Plastic | \$30 | 3 | 1 | 4 | 2 | | | | |
| Aluminum | \$60 | 3 | 1 | | 3 | | | | https://www.digikey.com |
| Steel | \$75 | 4 | 1 | | 4 | | | | |
| Plexi Glass | \$40 | 1 | 0 | 1 | 1 | | | | |
| Wood | \$15 | 4 | 0 | 1 | 4 | | | | |
| Store Bought | \$7 | 4 | 1 | 4 | 4 | | | | https://www.pishop.com |

Figure 8: Microprocessor Case Material Table

After choosing the previous decisions found in the support tables above, the amount of temporary data storage for the hub/controller needed to be determined. The reason for the storage is in the situation of a power outage inside the house. In this case the controller of the system would not be able to immediately relay data to the data processor and without a temporary storage, the data would be lost. Thus, the need for onboard storage. The most important part of this table came when determining the amount of storage needed. To do this, guesstimates were made in order to determine file size from each sensor type to then calculate the amount the system would need. The results can be seen below in [Figure 9].

| Temp Data Storage Types Hub | | | | | | |
|--------------------------------------|--------------|-------------|-----------|---------------|-----------|---|
| Storage Types | Capacity | Speed | Price | Physical Size | Life span | Link |
| | Weight: | Weight: | Weight: | Weight: | Weight: | |
| | 10 | 7 | 10 | 4 | 10 | |
| Yes = 1, No = 0 | Value: in GB | Value: mb/s | Value: \$ | Value: 0 - 4 | Value: | |
| Class 10 microSD card - 16GB | 16 | 100 | 10 | 0 | | https://www.ama |
| Class 10 microSD card - 32GB | 32 | 98 | 9 | 0 | | https://www.ama |
| Class 10 microSD card - 64GB | 64 | 160 | 15 | 0 | | https://www.ama |
| Class 10 microSD card - 128GB | 128 | 160 | 24 | 0 | | https://www.ama |
| Class 10 microSD card - 256GB | 256 | 100 | 48 | 0 | | https://www.ama |
| Class 10 microSD card - 512GB | 512 | 90 | 100 | 0 | | https://www.ama |
| 1TB Flashdrive/Thumbdrive | 1000 | 85 | 34 | 2 | | https://www.ama |

Figure 9: Data-Acquisition Temporary Storage Size Table

To continue in the design process, the support tables for the subsystems needed to be created. This consisted of a temporary data storage size and the microcontroller chosen for the design. These two tables can be seen below in [Figure 10] and [Figure 11].

The microcontroller chosen is the ESP32 board. This board was chosen based on speed and output connection types which consist of both the required Bluetooth and Wi-Fi. The other engineering characteristics that were used to determine the microcontroller decided for this design can be seen in the table below. Another important aspect of the chosen board was the number of input/output analog and digital pins that the board was compatible with.

| Microcontrollers | | | | | | | | | | |
|--------------------------------|--------------|---------------------|----------------------|----------------------|---------------|--------------|------------------|-----------------|------------------|----------------|
| Microcontrollers | Cost | Physical Size | Bluetooth | Wifi | Clock Speed | Flash Memory | Digital I/O Pins | Analog I/O Pins | Coding Languages | |
| | In USD | In mm | Weight: | Weight: | Weight: | Weight: | Weight: | Weight: | Weight: | Weight: |
| | | | 10 | 7 | 7 | 10 | 10 | 10 | 10 | 10 |
| | | | Value (0=no, 1=yes): | Value (0=no, 1=yes): | Value in MHz: | Value in KB: | Value: | Value: | Value: | Value: |
| Arduino UNO WRF Rev 2 | 35.84 | 53.4 x 68.6 | 1 | 1 | 16 | 48 | 14 | 6 | | Arduino |
| Arduino YUN Rev 2 | 46.90 | | 0 | 1 | 16 | 32 | 20 | 12 | | Arduino |
| Canaduino WEMO S D1 R32 | 7.66 | | 1 | 1 | 240 | 4 | 20 | 6 | | |
| Teensy 4.0 | 29.17 | 17.78 x 35.56 | 0 | 0 | 600 | 2048 | 40 | 14 | | |
| Digistump Oek | 10.95 | 23.4 x 30 | 0 | 1 | 80 | 4096 | 11 | 1 | | |
| SparkFun RedBoard Artemis Nano | 14.95 | 23.14 x 50.04 | 1 | 0 | 48 | 1024 | 17 | 8 | | |
| Raspberry Pi Zero | 5.00 | 30 x 65 | 0 | 0 | 1000 | 0 | 40 | 0 | | |
| ESP-12E Nodemcu | 4 for 16.99 | 25.6 x 48.55 | 0 | 1 | 80 | 4096 | 11 | 1 | | Arduino |
| HiLetgo ESP-WROOM-32 | 10.99 | 25.4 x 48.26 | 1 | 1 | 240 | 4096 | 32 | 0 | | Arduino |
| MSP-EXP430FR2355 | 15.59 | | 0 | 0 | 24 | | 44 | | 12 | |

Figure 10: Microcontroller Type Table

To complete the design process for the subsystem's boards, the Microprocessor support table ([Figure 11]) was created to determine the necessary storage types that would be required. Like mentioned above in the controller/hub storage, the data storage would only be needed in emergency situations where power was lost to the sensor. With the calculations made in the hub, the subsystem data size was also chosen.

| Temp Data Storage Types Subsystems | | | | | | | |
|--------------------------------------|-----------------|------------------|-----------------|-------------------|--------------------|--|--|
| Storage Types | Capacity | Speed | Price | Physical Size | Life span | Link | |
| Yes = 1, No = 0 | 10 Value: GB | 7 Value: mb/s | 10 Value: \$ | 4 Value: 0 - 4 | 10 Value: years | | |
| Class 10 microSD card - 16GB | 16 | 100 | 10 | 0 | 10 | https://www.a | |
| Class 10 microSD card - 32GB | 32 | 98 | 9 | 0 | 10 | https://www.a | |
| Class 10 microSD card - 64GB | 64 | 160 | 15 | 0 | 10 | https://www.a | |
| Class 10 microSD card - 128GB | 128 | 160 | 24 | 0 | 10 | https://www.a | |
| Class 10 microSD card - 256GB | 256 | 100 | 48 | 0 | 10 | https://www.a | |
| Class 10 microSD card - 512GB | 512 | 90 | 100 | 0 | 10 | https://www.a | |
| 1TB Flashdrive/Thumbdrive | 1000 | 85 | 34 | 2 | 10 | https://www.a | |

Figure 11: Sensor Subsystem Buffer Storage Table

Input Table

Output Table

| Output Table | | |
|--|---|------------------|
| Parameters | Expected Results | Measured Results |
| Data Rates | | TBD |
| Bluetooth | 3 MB/s | |
| Wi-Fi | 25 MB/s | |
| USB | 18 MB/s | |
| Cloud upload | 20 MB/s | |
| Battery Life | | |
| Subsystems - Door Sensor | 400 Hours | |
| Subsystems - Weight Scale | 400 Hours | |
| Subsystems - Light Sensor | 200 Hours | |
| Subsystems - Temperature Sensor | 100 Hours | |
| Hub | 0 Hours (Wired) | |
| Software | | |
| Expected file size of the string sent from the subsystems to the hub - Door Sensor | Format of String: Opened - Date - Time - Sensor ID 40 KB | |
| Expected file size sent from the subsystems to the hub - Button sensor, Sending a 1 when it's been pressed, storing for a week | 20 KB | |
| Expected file size sent from the subsystems to the hub - Temperature Sensor. Potential hourly log of room temp. | 100 KB | |
| Expected file size sent from the subsystems to the hub - Light Sensor. Potential hourly log to check if a light is on, sending 1 if on, 0 if off | 20 KB | |

Figure 12: Output Table

The Output Table, shown above in [Figure 12], for the project is a list of potential tests and their expected results, with the measured results table to be filled in in the next semester for EE499 after extensive testing on the project has been performed. With this output table, the expected tests are to be done involving the data transfer rates, such as transferring data from the system to the hub via Bluetooth or Wi-Fi. To allow for a battery backup, testing for the battery lives for some of the different subsystems will be performed to make

sure the battery back-up will be adequate for the project. And for the software aspect of the project, the file sizes of the data logs will be tested, expecting the files stored in the data acquisition hub, or the temporary storage of the subsystem itself in case of a power outage. The data filled in the expected column of the table was gathered by researching common data rate, battery lives, and based on the size of the strings generated for the activity logs, allowing for some overhead in the file size.

Timeline

To help with the organization of the project, a Gantt Chart for the Fall 2020 semester for EE498 was created. By giving each team member a color as well as a color for group activities, the work of the group can be visually organized and compiled in an easy-to-read manner, showing who's assigned to the given tasks with each task having a specific timeline in order to complete it. The figures below show the different tasks assigned over the course of the semester to each team member, or the group if need be.¹

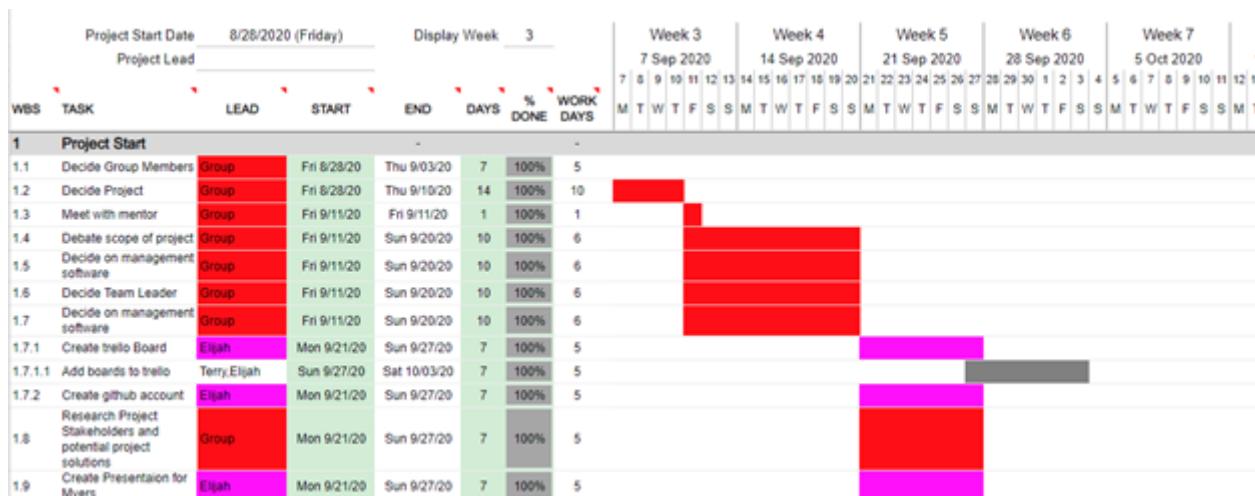
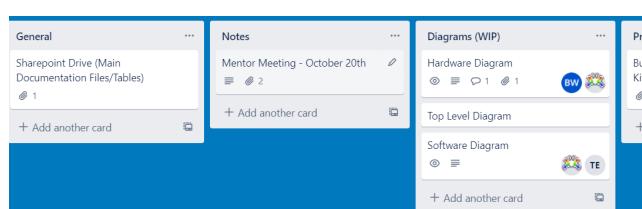


Figure 13: Gantt Chart: Initialization

¹ A Kanban-style project organization and task management structure was attempted as well via *Trello*.



However, due to the EE498 requirements, the project was switched to Gantt-style task management.

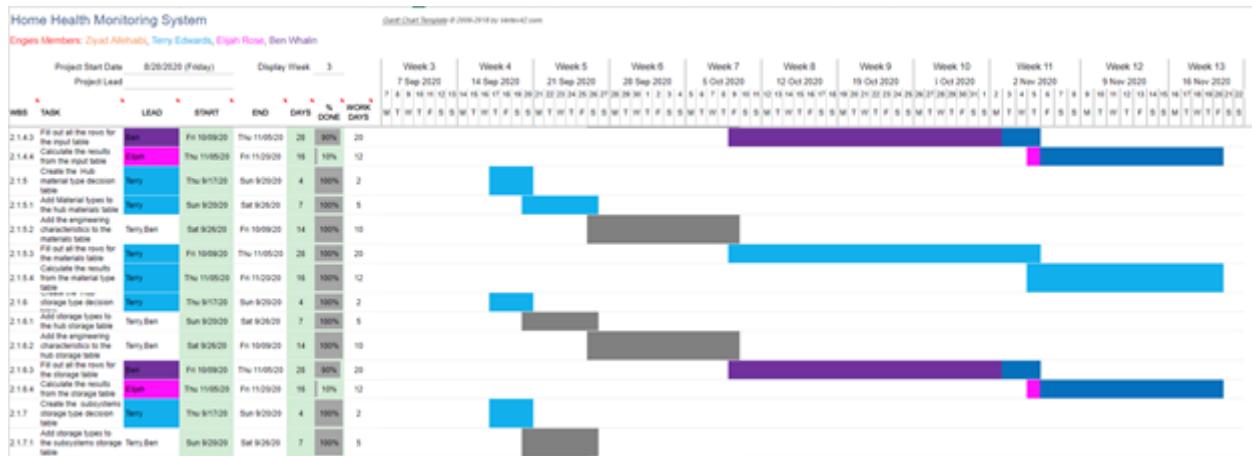
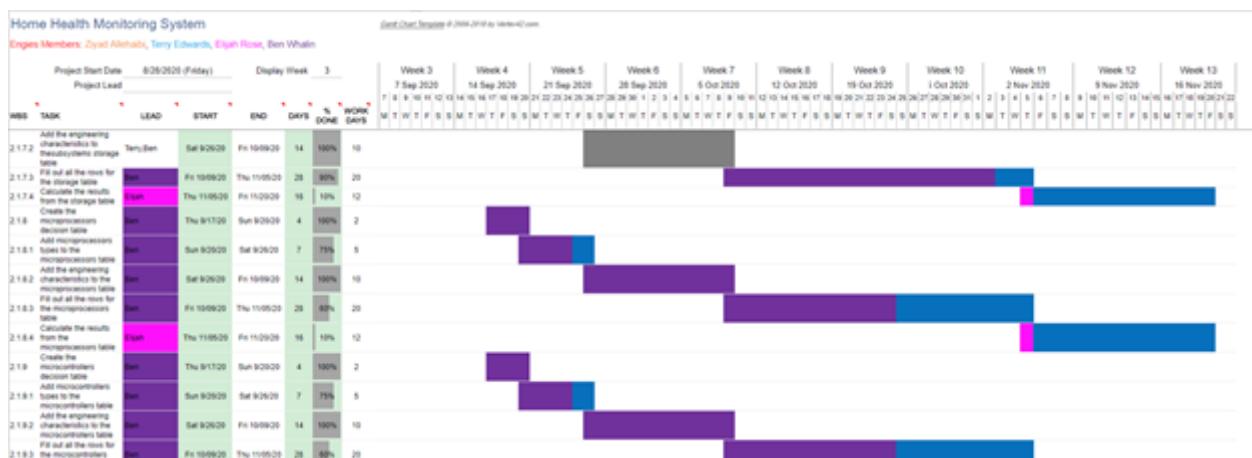

Figure 14: Gantt Chart: Design 1

Figure 15: Gantt Chart: Design 2

Figure 16: Gantt Chart: Design 3

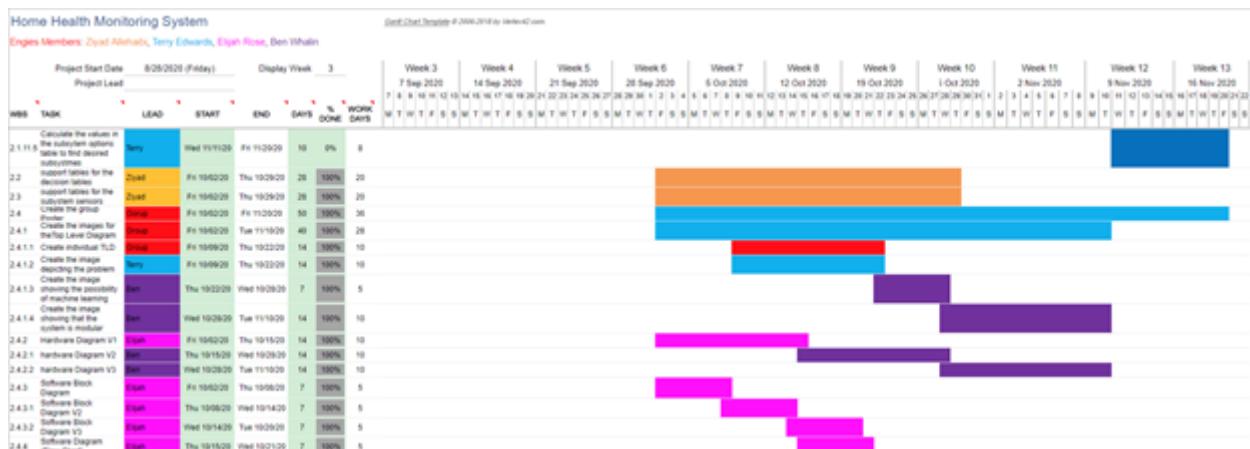

Figure 17: Gantt Chart: Design 4

Figure 18: Gantt Chart: Design 5

Figure 19: Gantt Chart: Design 6

**Figure 20:** Gantt Chart: Design 7

The project began in August 2020 towards the beginning of EE498 and the project will conclude in April of 2021 in EE499. Throughout the Fall 2020 semester, the project was selected, researched, and designed with each member of the group given certain tasks to complete, with group members helping each other out to try and complete the various tasks on time if need be. The implementation of the project's design will begin in the Spring 2021 semester beginning at the end of January.

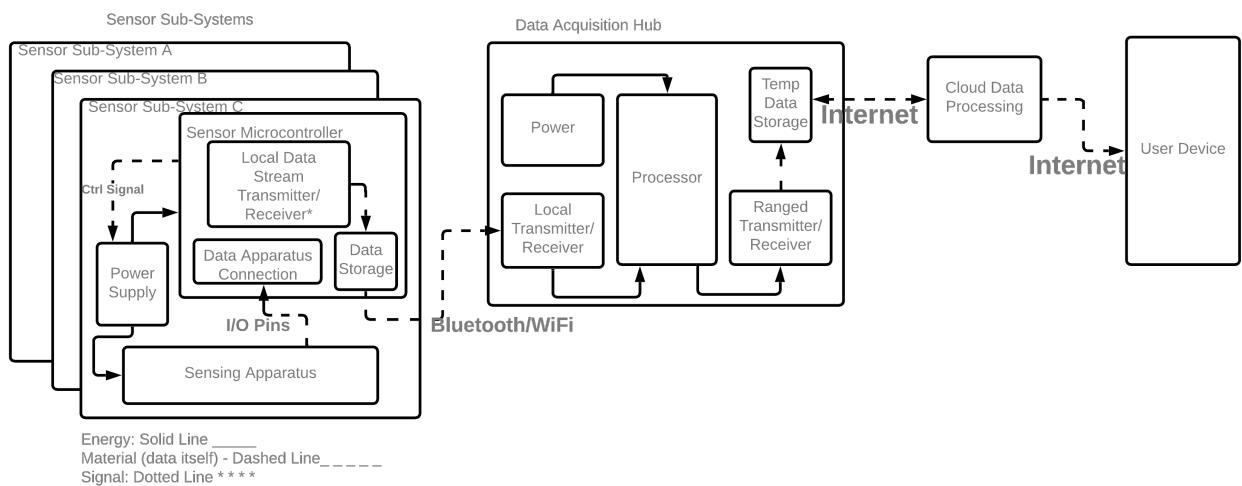
Subsystem Analysis

System Design

The following sections of show the respective hardware and software systems' designs, with the software simulations presented to give an idea of the look of the software aspect of the project. The hardware design will show the designs of the various hardware components of the project and the software design shows the expected flow of the project, that being the starting of the system and going through the ways the data is gathered and processed.

Hardware Design

The design of the hardware components hasn't been finalized for the project yet. While the specific components have all been decided upon, a concrete design for the entire system hasn't been decided upon. Though a final design will be decided upon and implemented in the Spring 2021 semester. While the design of the entire system is important, the software aspects of the system are more the focal point of the project since the project features a modular design, it's hardware design is more fluid and less concrete than other projects.



Hardware Block Diagram

Sensor Subsystem Analysis

Each of the sensor subsystems consists of the Microcontroller, the power supply, and the sensing apparatus. The sensing apparatus is connected to the microcontroller via the GPIO pins on the microcontroller, with the microcontroller collecting the data from the sensor and sending it via either Bluetooth or Wi-Fi to the Data Acquisition Hub to be logged in the storage for the Microprocessor and then transmitted and processed via the cloud to then be received by the user on their device.

Hardware Simulation

Software Design

The separation of several of the components necessitates at the minimum three separate software components: the sensor subsystems, the user's input/output of the system, and the intermediary communication and processing. Additionally, in order to better separate roles and allow the possibility of remote processing, the intermediary is further separated into a data-acquisition and processing component – theoretically, these could be separate program running simultaneously on the same device or two separate systems entirely, the software shall not be the limiting factor in this regard.

[Figure 21] outlines the four major components of the software.

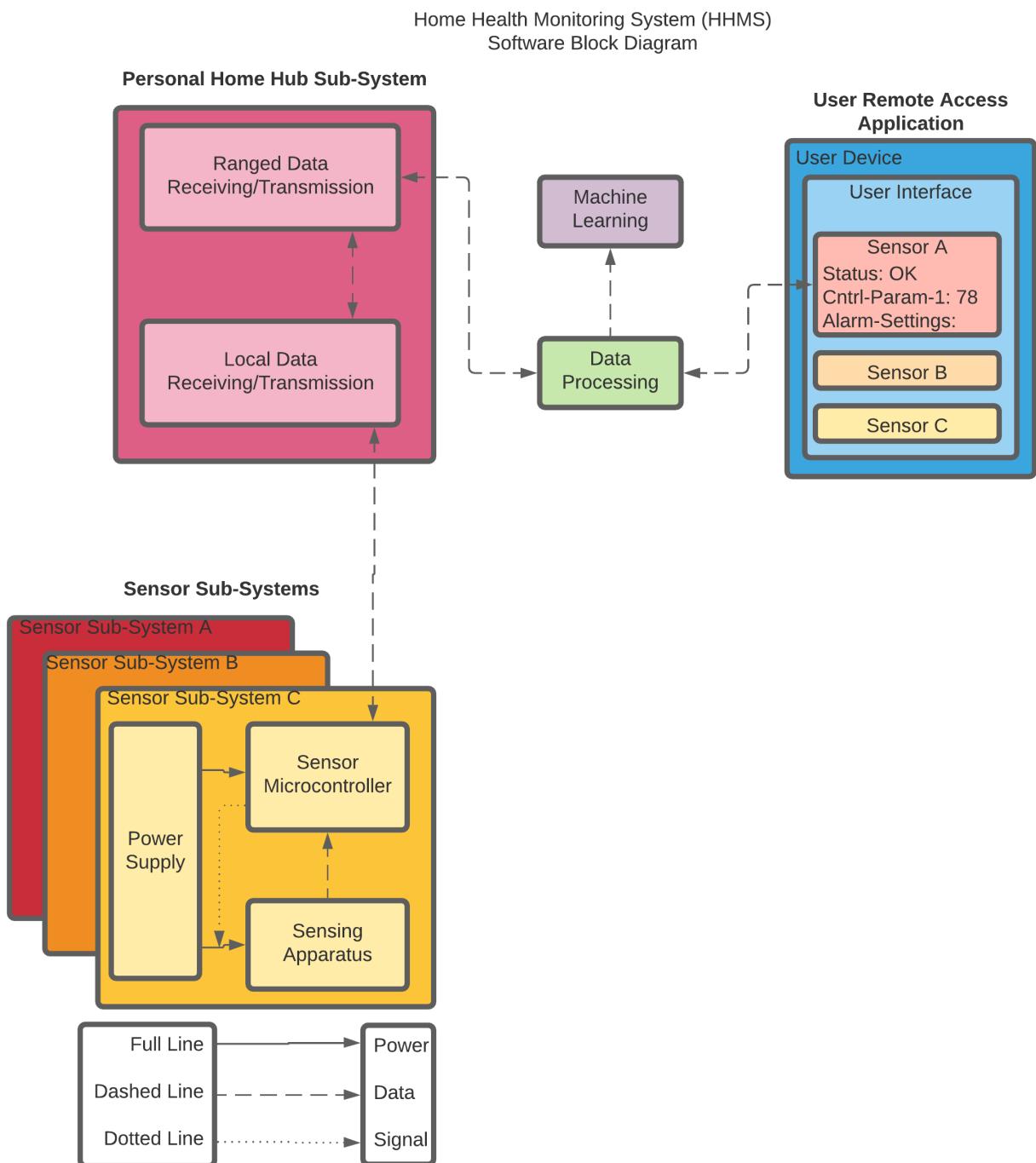


Figure 21: Software Block Diagram showing the major software components.

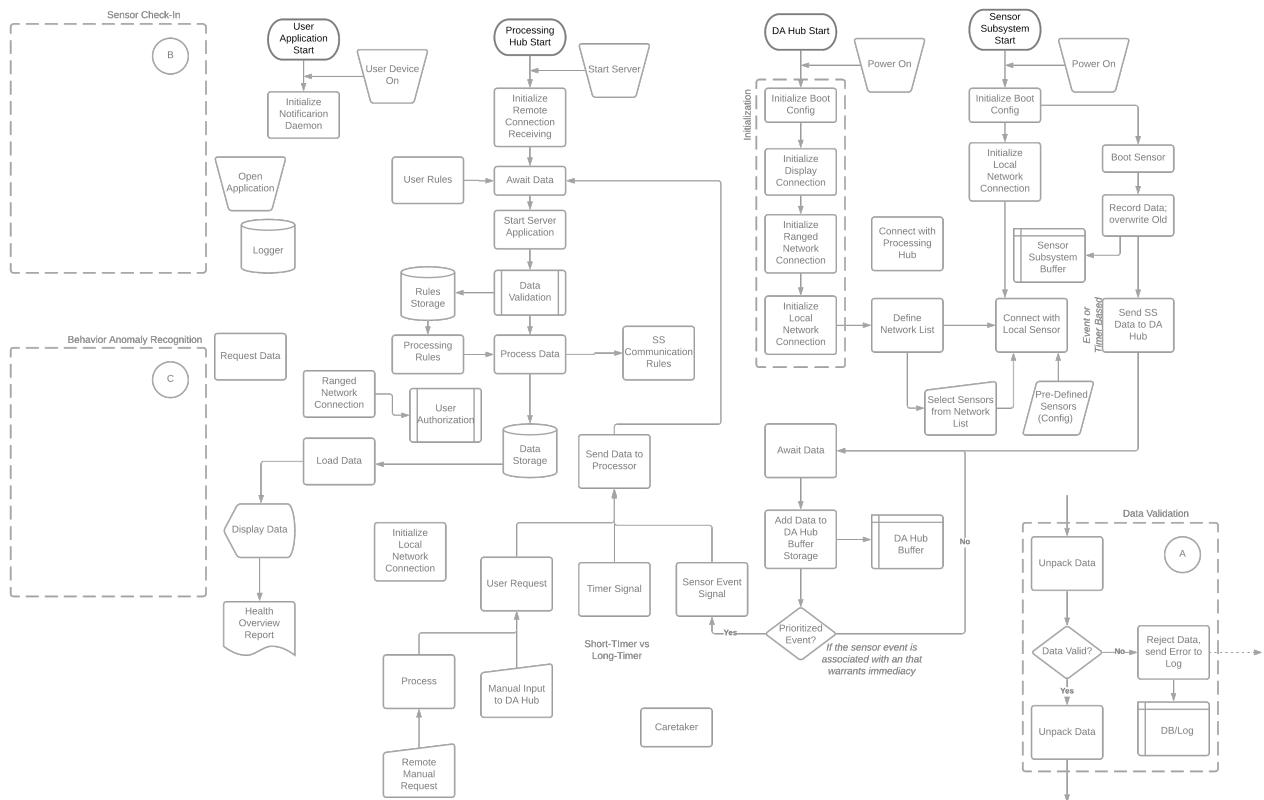


Figure 22: Software Flowchart Interaction showing the individual runtimes of each major software component.

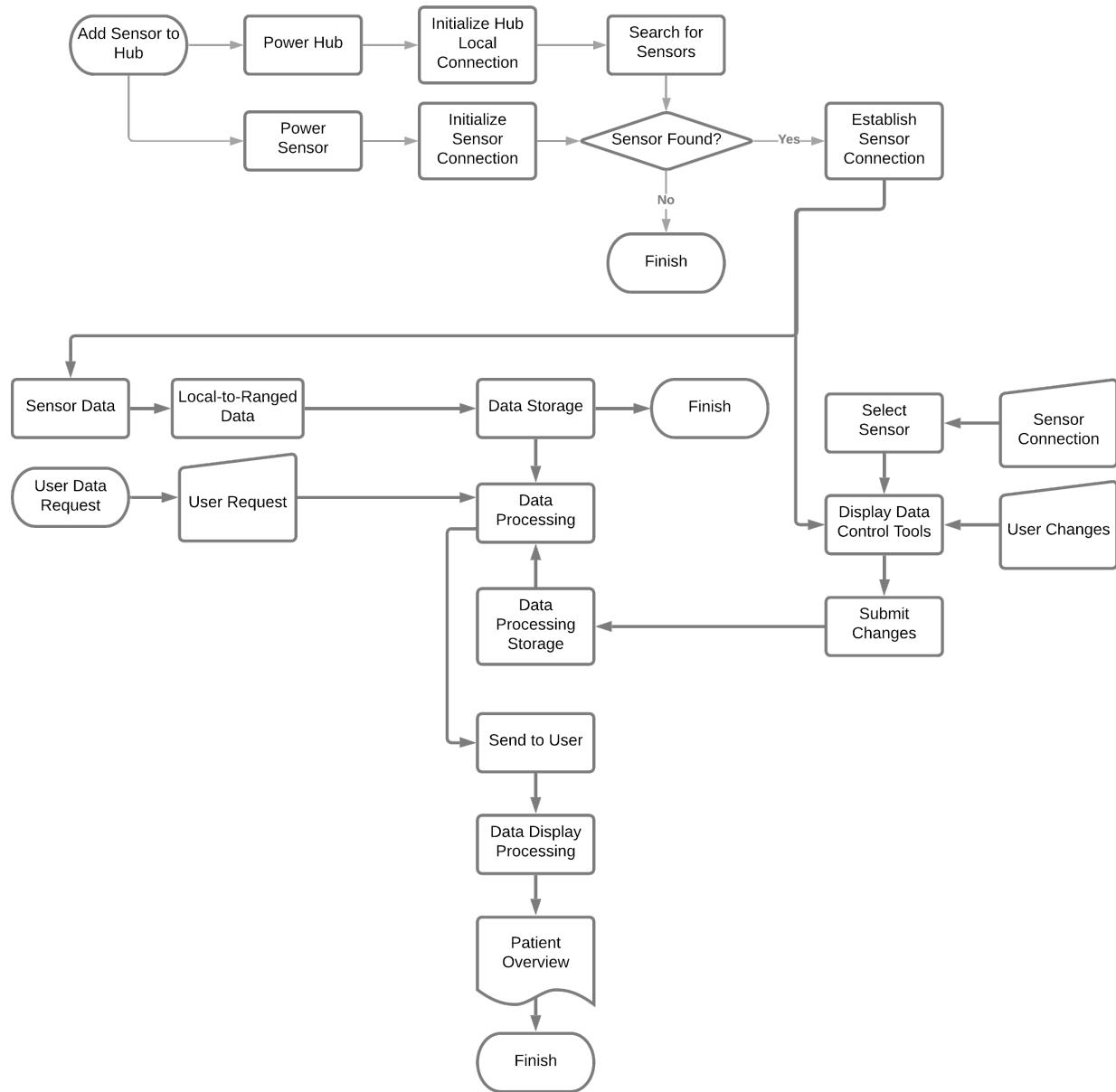


Figure 23: Software Functionality Flowchart showing some common use cases and how the users may interact and use the system,

Software Simulation

Cost Analysis

The budget that was given in senior design consisted of \$200 per group member. This amounted to \$800 for the entire project. Due to one of the goals of our design being that it would be affordable, the team desired to not use the entire budget for the baseline of our product. This meant that throughout the semester we

kept this idea in mind and chose hardware options based off reliability and dependability and not price. At the end of the design, the total cost of the hub/controller alone came out to be \$106.94 while the cost of the subsystem sensor package which includes six sensor systems, and a wearable watch came out to be \$253.24. Combined, this makes our entire product \$360.18, more than half of our budget.

Due to one of our team members being overseas for the semester and the fact that we are supposed to be social distancing, we opted in spending the remainder of our budget on purchasing extra subsystems and hub/controllers in order for each group member to be able to experiment and aide the process of implementing our design next semester. After figuring out the amount of subsystems and hubs that we could order while staying under our budget, we came to a price of \$787.09. This price includes sixteen sensor subsystems and three sets of the hub/controller. The two sub parts tables and the total parts table can all be seen below.

| Part Name | Status | Supplier | Part Number | Description | Price | Quantity | Subtotal | URL |
|---|---------------|----------|--------------------|--------------------------------|---------|----------|----------|---|
| Amazon | | | | | | | | |
| SanDisk 64GB microSD Memory Card | Need to Order | Amazon | SDSQXA2-064G-GN6MA | Memory Card for Raspberry Pi | \$13.99 | 1 | \$13.99 | https://www.amazon.com/ |
| PiShopUS | | | | | | | | |
| Raspberry Pi 4B 8Gb | Need to Order | PiShopUS | 8GB-9006 | Raspberry Pi Development Board | \$75.00 | 1 | \$75.00 | https://www.pishop.us/pr |
| Raspberry Pi 4 Case, Red/White | Need to Order | PiShopUS | 644824914916 | Raspberry Pi Case | \$5.00 | 1 | \$5.00 | https://www.pishop.us/pr |
| Case Fan & Heatsink for Raspberry Pi 4 Case | Need to Order | PiShopUS | 1411 | Raspberry Pi Case Fan | \$5.00 | 1 | \$5.00 | https://www.pishop.us/pr |
| USB-C Power Supply, 5.1V 3.0A | Need to Order | PiShopUS | 1203 | Raspberry Pi Power Supply | \$7.95 | 1 | \$7.95 | https://www.pishop.us/pr |
| | | | | | | | | \$106.94 |

Parts List for Hub/Controller

| Part Name | Status | Supplier | Part Number | Description | Price | Quantity | Subtotal | URL |
|--|---------------|----------|-----------------------|-------------------------------------|---------|----------|----------|---|
| Amazon | | | | | | | | |
| 4PCs Breadboards Kit | Need to Order | Amazon | RQ-BK-002 | Two large breadboards and two small | \$9.99 | 2 | \$19.98 | https://www.amazon.com/ |
| Hilgeto 200pc/5x40pc Breadboard Jumper Wires | Need to Order | Amazon | 3-02-1141 | Assortment of Wires | \$6.99 | 1 | \$6.99 | https://www.amazon.com/ |
| 61 Values (Pack of 1095) 1% 0.25w Resistor Book kit | Need to Order | Amazon | EE902 | Assortment of Resistors | \$14.99 | 1 | \$14.99 | https://www.amazon.com/ |
| Virtuabotix SD Card Reader/Writer for Arduino and other Microcontroller | Need to Order | Amazon | SDREADWRITE_BO | SD Card Reader for ESP32 Board | \$4.99 | 6 | \$29.94 | https://www.amazon.com/ |
| SanDisk 32GB Ultra microSDHC | Need to Order | Amazon | SDSQUAR-032G-GN6MA | Micro SD Memory Card | \$6.99 | 6 | \$41.94 | https://www.amazon.com/ |
| MELIFE 2 Pack ESP32 ESP-32S Development Board | Need to Order | Amazon | B07QS76VVZ | with 3 Breadboards | \$20.22 | 1 | \$20.22 | https://www.amazon.com/ |
| MELIFE 2 Pack ESP32 ESP-32S Development Board | Need to Order | Amazon | B07QS76VVZ | Pack of 2 ESP32 Development Boards | \$14.99 | 2 | \$29.98 | https://www.amazon.com/ |
| Gikfun MC-38 Wired Door Sensor Magnetic Switch | Need to Order | Amazon | 3652995690 | Two pack of door sensor | \$7.29 | 1 | \$7.29 | https://www.amazon.com/ |
| Cylewet 70Pcs Momentary Tactile Push Button | Need to Order | Amazon | CYT1115 | 35 Momentary Push Buttons | \$8.99 | 1 | \$8.99 | https://www.amazon.com/ |
| Elechip 10Pcs 1.8B20 DS18B20 TO-92 3 Pins Wire Digital Thermometer Temp | Need to Order | Amazon | 1 | Pack of 10 Temperature sensors | \$9.99 | 1 | \$9.99 | https://www.amazon.com/ |
| eBoot 30 Pieces Photoreistor Photo Light Sensitive | Need to Order | Amazon | EBOOT-RESISTOR-05 | Pack of 30 Light Level Sensors | \$4.95 | 1 | \$4.95 | https://www.amazon.com/ |
| NodeMCU ESP8266 Programmable Development Board Built in 500mAh Battery with OLED Display, Wristband and 3D Printing Case | Need to Order | Amazon | ZWQW16U53EU12GO383O44 | Wearable Watch from ESP32 board | \$42.99 | 1 | \$42.99 | https://www.amazon.com/ |
| | | | | | | | | \$253.24 |

Parts List for Hub/Controller

| Part Name | Status | Supplier | Part Number | Description | Price | Quantity | Subtotal | URL |
|--|---------------|----------|-----------------------|-------------------------------------|---------|----------|----------|---|
| Amazon | | | | | | | | |
| SanDisk 64GB microSD Memory Card | Need to Order | Amazon | SDSQXA2-064G-GN6MA | Memory Card for Raspberry Pi | \$13.99 | 3 | \$41.97 | https://www.amazon.com/ |
| 4PCs Breadboards Kit | Need to Order | Amazon | RQ-BK-002 | Two large breadboards and two small | \$9.99 | 3 | \$29.97 | https://www.amazon.com/ |
| Hilgeto 200pc/5x40pc Breadboard Jumper Wires | Need to Order | Amazon | 3-02-1141 | Assortment of Wires | \$6.99 | 1 | \$6.99 | https://www.amazon.com/ |
| 61 Values (Pack of 1095) 1% 0.25w Resistor Book kit | Need to Order | Amazon | EE902 | Assortment of Resistors | \$14.99 | 2 | \$29.98 | https://www.amazon.com/ |
| Virtuabotix SD Card Reader/Writer for Arduino and other Microcontroller | Need to Order | Amazon | SDREADWRITE_BO | SD Card Reader for ESP32 Board | \$4.99 | 15 | \$74.85 | https://www.amazon.com/ |
| SanDisk 32GB Ultra microSDHC | Need to Order | Amazon | SDSQUAR-032G-GN6MA | Micro SD Memory Card | \$6.99 | 15 | \$104.85 | https://www.amazon.com/ |
| MELIFE 2 Pack ESP32 ESP-32S Development Board | Need to Order | Amazon | B07QS76VVZ | with 3 Breadboards | \$20.22 | 4 | \$80.88 | https://www.amazon.com/ |
| MELIFE 2 Pack ESP32 ESP-32S Development Board | Need to Order | Amazon | B07QS76VVZ | Pack of 2 ESP32 Development Boards | \$14.99 | 4 | \$59.96 | https://www.amazon.com/ |
| Gikfun MC-38 Wired Door Sensor Magnetic Switch | Need to Order | Amazon | 3652995690 | Two pack of door sensor | \$7.29 | 3 | \$21.87 | https://www.amazon.com/ |
| Cylewet 70Pcs Momentary Tactile Push Button | Need to Order | Amazon | CYT1115 | 35 Momentary Push Buttons | \$8.99 | 1 | \$8.99 | https://www.amazon.com/ |
| Elechip 10Pcs 1.8B20 DS18B20 TO-92 3 Pins Wire Digital Thermometer Temp | Need to Order | Amazon | 1 | Pack of 10 Temperature sensors | \$9.99 | 1 | \$9.99 | https://www.amazon.com/ |
| eBoot 30 Pieces Photoreistor Photo Light Sensitive | Need to Order | Amazon | EBOOT-RESISTOR-05 | Pack of 30 Light Level Sensors | \$4.95 | 1 | \$4.95 | https://www.amazon.com/ |
| NodeMCU ESP8266 Programmable Development Board Built in 500mAh Battery with OLED Display, Wristband and 3D Printing Case | Need to Order | Amazon | ZWQW16U53EU12GO383O44 | Wearable Watch from ESP32 board | \$42.99 | 1 | \$42.99 | https://www.amazon.com/ |
| PiShopUS | | | | | | | | |
| Raspberry Pi 4B 8Gb | Need to Order | PiShopUS | 8GB-9006 | Raspberry Pi Development Board | \$75.00 | 3 | \$225.00 | https://www.pishop.us/pr |
| Raspberry Pi 4 Case, Red/White | Need to Order | PiShopUS | 644824914916 | Raspberry Pi Case | \$5.00 | 3 | \$15.00 | https://www.pishop.us/pr |
| Case Fan & Heatsink for Raspberry Pi 4 Case | Need to Order | PiShopUS | 1411 | Raspberry Pi Case Fan | \$5.00 | 3 | \$15.00 | https://www.pishop.us/pr |
| USB-C Power Supply, 5.1V 3.0A | Need to Order | PiShopUS | 1203 | Raspberry Pi Power Supply | \$7.95 | 3 | \$23.85 | https://www.pishop.us/pr |
| | | | | | | | | \$797.09 |

Parts List for Total Package

Individual Tasks

Ziyad Allehaibi

Terry Edwards

Excluding the work put into this document, my tasks for this semester (After the project was chosen and researched) consisted of creating the initial rough draft sketch of the top-level design image used in the design poster. This sketch was later recreated in google drawing before updated and improved upon by other group members. After the initial problem statement was decided on, the possible subsystems needed to be brainstormed. To do this I created the subsystems table and added the different things that could be possibly done regarding monitoring a loved one. To fill out this table, I interviewed multiple nurses and a respiratory therapist to get their opinion on aspects of the subsystems that I could not think of. After compiling all the findings and my own personal ideas, a table of roughly sixty different subsystems was created. This table was then broken down and colorized based on the type of physical sensor that could be used in each subsystem. After researching sensor types, I broke the original subsystem table up into the ten sensor types that we have today and then compiled and categorized the different subsystems that were brainstormed and displayed them in a new table. This table was then used to create a weighted decision table based on the chosen engineering characteristics to determine the subsystems that would be implemented into the final design. Alongside the work with the subsystems table, I also aided creating and filling out the design table. This consisted of making many decision tables that required individual engineering characteristics and weights. After completing the decision tables I then compiled all the results into the design table used in our presentation. Next, I took the design table and the subsystem table and created the parts list. I researched parts options to make sure there were no cheaper options and then added them to the parts list. To determine the best combination of parts amongst the group, I split the parts list into multiple tables that showed individual prices for the hub itself and a set of subsystem sensors. Afterwards I combined the two to create the current parts list. I then created the output table used in the presentation. I filled out the initial contents of the table and another student later added on to these. I also recreated the Gantt chart after the initial one made by another student was corrupted and no longer functioned. This consisted of going through the entire semester and deciding on potential tasks that would be necessary to produce our product. Finally, I created the initial copy of the presentation that was used while the other students came behind me and added their individual slides

Ben Whalin

My tasks for the first semester one (EE498) included researching the potential Microcontrollers and Microprocessors that would be looked at and considered for use in the project. Throughout the semester, I helped work on the Design Table as well as the Decision Tables needed to help with the project. I created the Top-Level Diagram (TLD) used for the poster for the course as well as the Hardware Diagram that will be useful for helping create the project in EE499. I contributed to the PowerPoint Presentation by creating the slides used

for the TLD, Hardware Diagram and the Poster as well as helping look over everyone's slides to help with ideas and proof-reading.

Elijah Rose

Elijah focused primarily on the software design aspects of the project, along with the aid of Terry, as detailed in [software simulation](#). He additionally contributed various other pieces and work to the team, such as the hardware diagram V1, separating the Data-Acquisition and Processing hub, etc.

Appendix

Appendix A: Poster

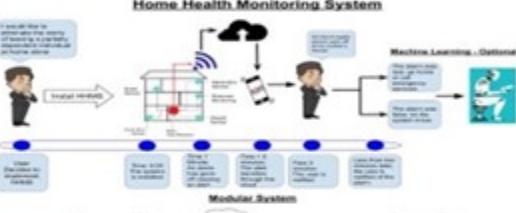
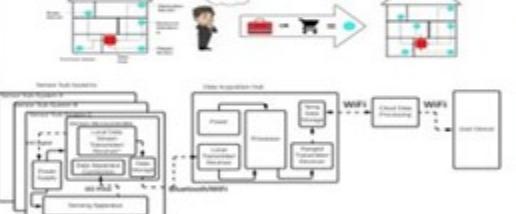
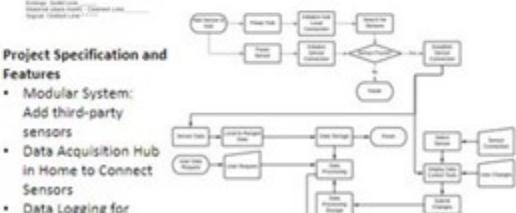


Home Health Monitoring System

Team Advisor: Gregory Myers (ECE)
Terry Edwards, Ben Whalin, Elijah Rose, Ziyad Allehaibi



THE UNIVERSITY OF
ALABAMA AT BIRMINGHAM

| | | |
|---|--|--|
| <p>Goals</p> <p>When someone oversees the care of their loved one for any number of reasons, be the person ill or elderly, the caretaker of the partially dependent person cannot always be home with their loved one at all points of a day. With this in mind, a system that can monitor the person in the home and send alerts when certain subsystems are triggered to the caretaker, with the system learning about different scenario's such as what patterns of behavior are acceptable and will no longer trigger alerts is a benefit to society at large. The system developed will be a modular system with a central hub unit that will communicate with the caretaker and learn based on the caretaker's responses to the alert. It will be a modular system so that many different subsystems can be easily added to the overall system.</p> <p>Methods and Technologies</p> <p>Modular Sensor Microcontrollers, Local-to-Ranged Communication Hub, API, Custom Sensor Integration Interfacing, Data Manipulation Interface/Commands, Possible Machine Learning and Behavioral Anomaly Recognition</p> <p>Research Issues</p> <p>With advances in medicine, developed countries have an increasing population of elderly and partially-dependent individuals. The choice between retirement homes and personal caregiving is increasingly relevant. Due to the price involved with retirement homes, a real-time monitoring solution can offer greater tools to caregivers for increased safety of the partially-dependent individual as well as convenience and peace of mind for the caregiver.</p> <p>References</p> <p>Microsoft (2020-10-10) https://dotnet.microsoft.com/apps/aspnet W3Schools (2020-15-11) https://www.w3schools.com/IEEEExplore (2020-09-10) https://ieeexplore.ieee.org.proxy3.lib.uab.edu/document/5335525 https://aws.amazon.com/api-gateway/?hp=title&so-exp=below</p> | <p>Project Top-Level Diagrams</p> <p>Home Health Monitoring System</p>  <p>Modular System</p>  <p>Sensor Hub Resources</p>  <p>User Application Unit</p>  <p>Cloud Data Processing Unit</p>  <p>Machine Learning Unit</p>  | <p>Software & Learning Materials</p> <ul style="list-style-type: none"> API Documentation Machine Learning (Pattern Break Recognition) .NetCore Windows IoT Core OS Arduino/Sketch/Launchpad Github <p>Required Hardware & Price</p> <ul style="list-style-type: none"> Raspberry Pi Microprocessor (\$75) Hiliteo ESP32 Microcontroller w/ Breadboards (\$55) Raspberry Pi Case w/ Heatsink and Fan (\$10) 64GB MicroSD Card (\$14) Sensors: Temperature, Light, Door (\$22) Momentary Pushbutton (\$9) Wires and Resistors (\$22) <p>Preparation and Plan</p> <ol style="list-style-type: none"> Create Data Acquisition Component; Integrate with Processing Hub Design Demo Sensor Microcontroller Send Data from Sensor to Data Acquisition Create Processing Software Send Data from Data Acquisition Hub to Processing Create Data Controller Interface; create Data View/Monitor Interface. Implement on local monitor. Demonstrate connection of Processor to Control Interfaces. Implement Remote Communication of Hub-to-Processor and Processor-to-Controller Implement Local Communication of Sensor Microcontroller-to-Hub Demonstrate feedback capability from processing to hub to sensor. <p>Instructor Information</p> <p>Dr. S. Abdallah Mirbezorgi jamir@uab.edu Phone: 205-934-8412 e-BioTech Lab Department of ECE, UAB BEC 2590</p> |
|---|--|--|

HHMS Poster

Appendix B: Final Slides



Home Health Monitoring System

2020-12-04

HHMS
OUTLINE

PROBLEM INTRODUCTION

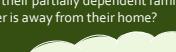
How can one monitor their partially dependent family member while the user is away from their home?

*13% of US
Population is 65+.

Average cost of assisted living is \$4,000/month.

Some are not comfortable sending family to retirement homes.

Covid-19: Separation of Family



- Sensors around home
- Alert User quickly in case of emergency
- Modularity
- Machine learning/Pattern Recognition optional

The diagram illustrates a hardware architecture for a home monitoring system. It starts with various sensors (Door Sensor, Motion Sensor, Light Sensor) located around the home. These sensors feed into a central hub labeled "Central Home Monitoring Hub". The hub then connects to a "Processor" unit, which includes a "Local Processor" and a "Cloud Processor". The "Processor" also has a "Transmitter" connected to it. The "Processor" and "Transmitter" are shown with dashed lines indicating they are part of the "Processor" block. Finally, the "Processor" connects to a "Cloud" icon, representing the final destination for the collected data.

The diagram illustrates the hardware components of a sensor system:

- Sensor Application:** Contains a Microcontroller, Sensors, and a Wireless Transmitter.
- Bluetooth Module:** A bridge between the Sensor Application and the Base Application Hub.
- Base Application Hub:** Contains a Processor, a Database, and a Wireless Transmitter.
- Data Line Representations:** Shows the connection between the Sensor Application and the Base Application Hub using a solid line, and the connection from the Base Application Hub to the Internet using a dashed line.
- Internet:** Represented as a cloud icon.
- Cloud Data Store:** Represented as a rectangular box.
- Next Device:** Represented as a rectangular box.

Annotations at the top right indicate "Hardware Diagram" and "Ben Whalin [7]."

- In order to ease with further development a design table was created to visualize the needed parts of the product.
- Parts of the design:
 - The controller/Hub
 - Subsystems
 - Client Interface

The slide has a blue background. At the top left, the word "BLE" is written in large white letters. To its right, the name "Terry Edwards" is displayed. The central part of the slide features a cartoon illustration of five people of diverse backgrounds (including a person in a hijab) sitting around a long table, looking at laptops. Above the people is a glowing yellow lightbulb, with dashed lines connecting it to various icons representing technology and connectivity: a smartphone, a laptop, a magnifying glass, a gear, and a location pin.

| HUB DESIGN TABLE | | | | | | | Terry Edwards |
|--|--|---|--|--|---|--|--|
| System Statement | Elements | Topics | Features | Approach | Motives | Role | Value |
| How can we make our heart palpitations dependent body movements so that it is easier for them to return? | Develop a heart palpitations system that can detect body movements and provide feedback to help users return to their normal heart rate. | Heart palpitations, body movement detection, feedback, user interface design. | Real-time monitoring, machine learning models for heart rate prediction, mobile application for user interaction, sensors for body movement, UI/UX design for user feedback. | Iterative design, user-centered design, prototyping, A/B testing, user research. | Improved health, user satisfaction, system reliability. | System designer, developer, researcher, user advocate. | Healthcare innovation, user-centered technology, accessible medical solutions. |

| SUBSYSTEM DESIGN TABLE | | | | | | Terry Edwards' 11 | |
|---|----------|----------|--|--|---|-----------------------|--|
| Problem Statement | Approach | Subtask | Features | Approach | Requirements Needed | Impact Index | Link |
| How can we monitor the health of an elderly family member while the user is away from their home? | Robotics | Robotics | Humanoid robot Mobile robot Telepresence robot | Humanoid Control Mobile Control Telepresence Control Robotics API Robotics Database Robotics Sensors Robotics Actuators Robotics Power Robotics Navigation Robotics Perception Robotics Decision Making Robotics Learning | Humanoid Action Mobile Action Telepresence Action Robotics API Robotics Database Robotics Sensors Robotics Actuators Robotics Power Robotics Navigation Robotics Perception Robotics Decision Making Robotics Learning | High Medium Low | https://www.researchgate.net |
| Robotics | Robotics | Robotics | Robotics | Robotics | Robotics | Medium | https://www.researchgate.net |



HHMS

What type of hardware is needed to design this system?

HHMS PRODUCT DECISION TABLES

- Used as support tables for the hardware chosen in the design table.
- Contains each design option from the different component on the design table.
- Shows the applicable engineering characteristic for all design options.
- Equations are used to calculate the results for each decision table.

Terry [Edwards]

| SYSTEM DECISION TABLE | | | | | | Terry Edwards |
|-----------------------|----------------|--------|-------|---------------|--|-------------------------|
| Microcontrollers | | | | | | |
| System | Processor | Memory | Time | Time Required | Cost | Notes |
| 1 | Intel 8051 | 2Kb | 100ns | 100ns | \$10 | Fast, Low Cost |
| 2 | Motorola 6809 | 16Kb | 100ns | 100ns | \$100 | Medium Speed, High Cost |
| 3 | Motorola 68000 | 64Kb | 10ns | 10ns | \$1000 | Very Fast, High Cost |
| 4 | Motorola 68020 | 128Kb | 10ns | 10ns | \$2000 | Very Fast, High Cost |
| 5 | Motorola 68030 | 256Kb | 10ns | 10ns | \$4000 | Very Fast, High Cost |
| 6 | Motorola 68040 | 512Kb | 10ns | 10ns | \$8000 | Very Fast, High Cost |
| 7 | Motorola 68060 | 1MB | 10ns | 10ns | \$15000 | Very Fast, High Cost |
| 8 | Motorola 68882 | 16Kb | 100ns | 100ns | \$100 | Medium Speed, Low Cost |
| 9 | Motorola 68886 | 32Kb | 100ns | 100ns | \$200 | Medium Speed, Low Cost |
| 10 | Motorola 68888 | 64Kb | 100ns | 100ns | \$400 | Medium Speed, Low Cost |
| 11 | Motorola 68889 | 128Kb | 100ns | 100ns | \$800 | Medium Speed, Low Cost |
| 12 | Motorola 68890 | 256Kb | 100ns | 100ns | \$1600 | Medium Speed, Low Cost |
| 13 | Motorola 68891 | 512Kb | 100ns | 100ns | \$3200 | Medium Speed, Low Cost |
| 14 | Motorola 68892 | 1MB | 100ns | 100ns | \$6400 | Medium Speed, Low Cost |
| 15 | Motorola 68893 | 2MB | 100ns | 100ns | \$12800 | Medium Speed, Low Cost |
| 16 | Motorola 68894 | 4MB | 100ns | 100ns | \$25600 | Medium Speed, Low Cost |
| 17 | Motorola 68895 | 8MB | 100ns | 100ns | \$51200 | Medium Speed, Low Cost |
| 18 | Motorola 68896 | 16MB | 100ns | 100ns | \$102400 | Medium Speed, Low Cost |
| 19 | Motorola 68897 | 32MB | 100ns | 100ns | \$204800 | Medium Speed, Low Cost |
| 20 | Motorola 68898 | 64MB | 100ns | 100ns | \$409600 | Medium Speed, Low Cost |
| 21 | Motorola 68899 | 128MB | 100ns | 100ns | \$819200 | Medium Speed, Low Cost |
| 22 | Motorola 6889A | 256MB | 100ns | 100ns | \$1638400 | Medium Speed, Low Cost |
| 23 | Motorola 6889B | 512MB | 100ns | 100ns | \$3276800 | Medium Speed, Low Cost |
| 24 | Motorola 6889C | 1GB | 100ns | 100ns | \$6553600 | Medium Speed, Low Cost |
| 25 | Motorola 6889D | 2GB | 100ns | 100ns | \$13107200 | Medium Speed, Low Cost |
| 26 | Motorola 6889E | 4GB | 100ns | 100ns | \$26214400 | Medium Speed, Low Cost |
| 27 | Motorola 6889F | 8GB | 100ns | 100ns | \$52428800 | Medium Speed, Low Cost |
| 28 | Motorola 6889G | 16GB | 100ns | 100ns | \$104857600 | Medium Speed, Low Cost |
| 29 | Motorola 6889H | 32GB | 100ns | 100ns | \$209715200 | Medium Speed, Low Cost |
| 30 | Motorola 6889I | 64GB | 100ns | 100ns | \$419430400 | Medium Speed, Low Cost |
| 31 | Motorola 6889J | 128GB | 100ns | 100ns | \$838860800 | Medium Speed, Low Cost |
| 32 | Motorola 6889K | 256GB | 100ns | 100ns | \$1677721600 | Medium Speed, Low Cost |
| 33 | Motorola 6889L | 512GB | 100ns | 100ns | \$3355443200 | Medium Speed, Low Cost |
| 34 | Motorola 6889M | 1TB | 100ns | 100ns | \$6710886400 | Medium Speed, Low Cost |
| 35 | Motorola 6889N | 2TB | 100ns | 100ns | \$13421772800 | Medium Speed, Low Cost |
| 36 | Motorola 6889O | 4TB | 100ns | 100ns | \$26843545600 | Medium Speed, Low Cost |
| 37 | Motorola 6889P | 8TB | 100ns | 100ns | \$53687091200 | Medium Speed, Low Cost |
| 38 | Motorola 6889Q | 16TB | 100ns | 100ns | \$107374182400 | Medium Speed, Low Cost |
| 39 | Motorola 6889R | 32TB | 100ns | 100ns | \$214748364800 | Medium Speed, Low Cost |
| 40 | Motorola 6889S | 64TB | 100ns | 100ns | \$429496729600 | Medium Speed, Low Cost |
| 41 | Motorola 6889T | 128TB | 100ns | 100ns | \$858993459200 | Medium Speed, Low Cost |
| 42 | Motorola 6889U | 256TB | 100ns | 100ns | \$1717986918400 | Medium Speed, Low Cost |
| 43 | Motorola 6889V | 512TB | 100ns | 100ns | \$3435973836800 | Medium Speed, Low Cost |
| 44 | Motorola 6889W | 1PB | 100ns | 100ns | \$6871947673600 | Medium Speed, Low Cost |
| 45 | Motorola 6889X | 2PB | 100ns | 100ns | \$13743895347200 | Medium Speed, Low Cost |
| 46 | Motorola 6889Y | 4PB | 100ns | 100ns | \$27487790694400 | Medium Speed, Low Cost |
| 47 | Motorola 6889Z | 8PB | 100ns | 100ns | \$54975581388800 | Medium Speed, Low Cost |
| 48 | Motorola 6889A | 16PB | 100ns | 100ns | \$109951162777600 | Medium Speed, Low Cost |
| 49 | Motorola 6889B | 32PB | 100ns | 100ns | \$219902325555200 | Medium Speed, Low Cost |
| 50 | Motorola 6889C | 64PB | 100ns | 100ns | \$439804651110400 | Medium Speed, Low Cost |
| 51 | Motorola 6889D | 128PB | 100ns | 100ns | \$879609302220800 | Medium Speed, Low Cost |
| 52 | Motorola 6889E | 256PB | 100ns | 100ns | \$1759218604441600 | Medium Speed, Low Cost |
| 53 | Motorola 6889F | 512PB | 100ns | 100ns | \$3518437208883200 | Medium Speed, Low Cost |
| 54 | Motorola 6889G | 1EB | 100ns | 100ns | \$7036874417766400 | Medium Speed, Low Cost |
| 55 | Motorola 6889H | 2EB | 100ns | 100ns | \$14073748835532800 | Medium Speed, Low Cost |
| 56 | Motorola 6889I | 4EB | 100ns | 100ns | \$28147497671065600 | Medium Speed, Low Cost |
| 57 | Motorola 6889J | 8EB | 100ns | 100ns | \$56294995342131200 | Medium Speed, Low Cost |
| 58 | Motorola 6889K | 16EB | 100ns | 100ns | \$112589990684262400 | Medium Speed, Low Cost |
| 59 | Motorola 6889L | 32EB | 100ns | 100ns | \$225179981368524800 | Medium Speed, Low Cost |
| 60 | Motorola 6889M | 64EB | 100ns | 100ns | \$450359962737049600 | Medium Speed, Low Cost |
| 61 | Motorola 6889N | 128EB | 100ns | 100ns | \$900719925474099200 | Medium Speed, Low Cost |
| 62 | Motorola 6889O | 256EB | 100ns | 100ns | \$1801439850948198400 | Medium Speed, Low Cost |
| 63 | Motorola 6889P | 512EB | 100ns | 100ns | \$3602879701896396800 | Medium Speed, Low Cost |
| 64 | Motorola 6889Q | 1PB | 100ns | 100ns | \$7205759403792793600 | Medium Speed, Low Cost |
| 65 | Motorola 6889R | 2PB | 100ns | 100ns | \$14411518807585587200 | Medium Speed, Low Cost |
| 66 | Motorola 6889S | 4PB | 100ns | 100ns | \$28823037615171174400 | Medium Speed, Low Cost |
| 67 | Motorola 6889T | 8PB | 100ns | 100ns | \$57646075230342348800 | Medium Speed, Low Cost |
| 68 | Motorola 6889U | 16PB | 100ns | 100ns | \$11529215046068469600 | Medium Speed, Low Cost |
| 69 | Motorola 6889V | 32PB | 100ns | 100ns | \$23058430092136939200 | Medium Speed, Low Cost |
| 70 | Motorola 6889W | 64PB | 100ns | 100ns | \$46116860184273878400 | Medium Speed, Low Cost |
| 71 | Motorola 6889X | 128PB | 100ns | 100ns | \$92233720368547756800 | Medium Speed, Low Cost |
| 72 | Motorola 6889Y | 256PB | 100ns | 100ns | \$184467440737095513600 | Medium Speed, Low Cost |
| 73 | Motorola 6889Z | 512PB | 100ns | 100ns | \$368934881474191027200 | Medium Speed, Low Cost |
| 74 | Motorola 6889A | 1EB | 100ns | 100ns | \$737869762948382054400 | Medium Speed, Low Cost |
| 75 | Motorola 6889B | 2EB | 100ns | 100ns | \$1475739525896764108800 | Medium Speed, Low Cost |
| 76 | Motorola 6889C | 4EB | 100ns | 100ns | \$2951479051793528217600 | Medium Speed, Low Cost |
| 77 | Motorola 6889D | 8EB | 100ns | 100ns | \$5902958103587056435200 | Medium Speed, Low Cost |
| 78 | Motorola 6889E | 16EB | 100ns | 100ns | \$11805916207174112870400 | Medium Speed, Low Cost |
| 79 | Motorola 6889F | 32EB | 100ns | 100ns | \$23611832414348225740800 | Medium Speed, Low Cost |
| 80 | Motorola 6889G | 64EB | 100ns | 100ns | \$47223664828696451481600 | Medium Speed, Low Cost |
| 81 | Motorola 6889H | 128EB | 100ns | 100ns | \$94447329657392902963200 | Medium Speed, Low Cost |
| 82 | Motorola 6889I | 256EB | 100ns | 100ns | \$188894659314785805926400 | Medium Speed, Low Cost |
| 83 | Motorola 6889J | 512EB | 100ns | 100ns | \$377789318629571611852800 | Medium Speed, Low Cost |
| 84 | Motorola 6889K | 1PB | 100ns | 100ns | \$755578637259143223705600 | Medium Speed, Low Cost |
| 85 | Motorola 6889L | 2PB | 100ns | 100ns | \$1511157274518286447411200 | Medium Speed, Low Cost |
| 86 | Motorola 6889M | 4PB | 100ns | 100ns | \$3022314549036572894822400 | Medium Speed, Low Cost |
| 87 | Motorola 6889N | 8PB | 100ns | 100ns | \$6044629098073145789644800 | Medium Speed, Low Cost |
| 88 | Motorola 6889O | 16PB | 100ns | 100ns | \$12089258196146291579289600 | Medium Speed, Low Cost |
| 89 | Motorola 6889P | 32PB | 100ns | 100ns | \$24178516392292583158579200 | Medium Speed, Low Cost |
| 90 | Motorola 6889Q | 64PB | 100ns | 100ns | \$48357032784585166317158400 | Medium Speed, Low Cost |
| 91 | Motorola 6889R | 128PB | 100ns | 100ns | \$96714065569170332634316800 | Medium Speed, Low Cost |
| 92 | Motorola 6889S | 256PB | 100ns | 100ns | \$193428131138340665268633600 | Medium Speed, Low Cost |
| 93 | Motorola 6889T | 512PB | 100ns | 100ns | \$386856262276681330537267200 | Medium Speed, Low Cost |
| 94 | Motorola 6889U | 1EB | 100ns | 100ns | \$773712524553362661074534400 | Medium Speed, Low Cost |
| 95 | Motorola 6889V | 2EB | 100ns | 100ns | \$1547425049106725322149068800 | Medium Speed, Low Cost |
| 96 | Motorola 6889W | 4EB | 100ns | 100ns | \$3094850098213450644298136000 | Medium Speed, Low Cost |
| 97 | Motorola 6889X | 8EB | 100ns | 100ns | \$6189700196426901288596272000 | Medium Speed, Low Cost |
| 98 | Motorola 6889Y | 16EB | 100ns | 100ns | \$12379400392853802577192544000 | Medium Speed, Low Cost |
| 99 | Motorola 6889Z | 32EB | 100ns | 100ns | \$24758800785707605154385088000 | Medium Speed, Low Cost |
| 100 | Motorola 6889A | 64EB | 100ns | 100ns | \$49517601571415210308770176000 | Medium Speed, Low Cost |
| 101 | Motorola 6889B | 128EB | 100ns | 100ns | \$99035203142830420617540352000 | Medium Speed, Low Cost |
| 102 | Motorola 6889C | 256EB | 100ns | 100ns | \$198070406285660841235080704000 | Medium Speed, Low Cost |
| 103 | Motorola 6889D | 512EB | 100ns | 100ns | \$396140812571321682470161408000 | Medium Speed, Low Cost |
| 104 | Motorola 6889E | 1PB | 100ns | 100ns | \$792281625142643364940322816000 | Medium Speed, Low Cost |
| 105 | Motorola 6889F | 2PB | 100ns | 100ns | \$1584563252885286729880645632000 | Medium Speed, Low Cost |
| 106 | Motorola 6889G | 4PB | 100ns | 100ns | \$3169126505770573459761291264000 | Medium Speed, Low Cost |
| 107 | Motorola 6889H | 8PB | 100ns | 100ns | \$6338253011541146919522582528000 | Medium Speed, Low Cost |
| 108 | Motorola 6889I | 16PB | 100ns | 100ns | \$12676506023082293839045165056000 | Medium Speed, Low Cost |
| 109 | Motorola 6889J | 32PB | 100ns | 100ns | \$25353012046164587678090330112000 | Medium Speed, Low Cost |
| 110 | Motorola 6889K | 64PB | 100ns | 100ns | \$50706024092329175356180660224000 | Medium Speed, Low Cost |
| 111 | Motorola 6889L | 128PB | 100ns | 100ns | \$101412048184658350712361320448000 | Medium Speed, Low Cost |
| 112 | Motorola 6889M | 256PB | 100ns | 100ns | \$202824096369316701424722660896000 | Medium Speed, Low Cost |
| 113 | Motorola 6889N | 512PB | 100ns | 100ns | \$405648192738633402849445331792000 | Medium Speed, Low Cost |
| 114 | Motorola 6889O | 1PB | 100ns | 100ns | \$811296385477266805698890663584000 | Medium Speed, Low Cost |
| 115 | Motorola 6889P | 2PB | 100ns | 100ns | \$1622592770954533611397781327168000 | Medium Speed, Low Cost |
| 116 | Motorola 6889Q | 4PB | 100ns | 100ns | \$3245185541909067222795562654336000 | Medium Speed, Low Cost |
| 117 | Motorola 6889R | 8PB | 100ns | 100ns | \$6490371083818134445591125308672000 | Medium Speed, Low Cost |
| 118 | Motorola 6889S | 16PB | 100ns | 100ns | \$1298074216763626889118250661744000 | Medium Speed, Low Cost |
| 119 | Motorola 6889T | 32PB | 100ns | 100ns | \$2596148433527253778236501323488000 | Medium Speed, Low Cost |
| 120 | Motorola 6889U | 64PB | 100ns | 100ns | \$5192296867054507556473002646976000 | Medium Speed, Low Cost |
| 121 | Motorola 6889V | 128PB | 100ns | 100ns | \$10384593734109015112946005293952000 | Medium Speed, Low Cost |
| 122 | Motorola 6889W | 256PB | 100ns | 100ns | \$20769187468218030225892010587904000 | Medium Speed, Low Cost |
| 123 | Motorola 6889X | 512PB | 100ns | 100ns | \$41538374936436060451784021175808000 | Medium Speed, Low Cost |
| 124 | Motorola 6889Y | 1PB | 100ns | 100ns | \$83076749872872120903568042351616000 | Medium Speed, Low Cost |
| 125 | Motorola 6889Z | 2PB | 100ns | 100ns | \$166153499745744241807136084703232000 | Medium Speed, Low Cost |
| 126 | Motorola 6889A | 4PB | 100ns | 100ns | \$332306999491488483614272169406464000 | Medium Speed, Low Cost |
| 127 | Motorola 6889B | 8PB | 100ns | 100ns | \$664613998982976967228544338812928000 | Medium Speed, Low Cost |
| 128 | Motorola 6889C | 16PB | 100ns | 100ns | \$1329227997965953934457088677625856000 | Medium Speed, Low Cost |
| 129 | Motorola 6889D | 32PB | 100ns | 100ns | \$2658455995931907868914177355251712000 | Medium Speed, Low Cost |
| 130 | Motorola 6889E | 64PB | 100ns | 100ns | \$5316911991863815737828354710503424000 | Medium Speed, Low Cost |
| 131 | Motorola 6889F | 128PB | 100ns | 100ns | \$1063382398372763147565670942100848000 | Medium Speed, Low Cost |
| 132 | Motorola 6889G | 256PB | 100ns | 100ns | \$2126764796745526295131341884201696000 | Medium Speed, Low Cost |
| 133 | Motorola 6889H | 512PB | 100ns | 100ns | \$4253529593491152590262683768403392000 | Medium Speed, Low Cost |
| 134 | Motorola 6889I | 1PB | 100ns | 100ns | \$8507059186982305180525367536806784000 | Medium Speed, Low Cost |
| 135 | Motorola 6889J | 2PB | 100ns | 100ns | \$17014118373964610361050745073613568000 | Medium Speed, Low Cost |
| 136 | Motorola 6889K | 4PB | 100ns | 100ns | \$3402823674792922072210149014672712000 | Medium Speed, Low Cost |
| 137 | Motorola 6889L | 8PB | 100ns | 100ns | \$6805647349585844144420298029345424000 | Medium Speed, Low Cost |
| 138 | Motorola 6889M | 16PB | 100ns | 100ns | \$1361129469917168828884058058688848000 | Medium Speed, Low Cost |
| 139 | Motorola 6889N | 32PB | 100ns | 100ns | \$2722258939834337657768116117377696000 | |

| SENSORS SUB TABLE | | | | | Ziyad Alaliabi (2) |
|-----------------------------------|--|--|----------|-----------------|---|
| Sensors sub-table | | | | | |
| Sensor | Description | Use | Accuracy | Range | Notes/Details |
| Finger movement/pressure | Measures the pressure applied to a finger or the movement of a finger. | Measuring force or tracking hand movements. | ±1% | 0-100 N | High accuracy, suitable for medical applications. |
| Fall detection & Monitoring | Detects falls and monitors physical activity levels. | Assisting elderly individuals and monitoring patient activity. | ±5% | 0-100% activity | Reliable for fall detection. |
| Pulse estimator | Estimates heart rate based on blood flow measurements. | Monitoring heart health and fitness tracking. | ±10% | 60-180 BPM | Accurate for general monitoring. |
| Weight scale | Measures body weight. | Health monitoring and fitness tracking. | ±1% | 0-150 kg | High accuracy for weight measurement. |
| Blood pressure (sphygmomanometer) | Measures blood pressure using a sphygmomanometer. | Health monitoring and medical diagnosis. | ±3% | 0-200 mmHg | Accurate for blood pressure measurement. |

| Output Table | | Bari Whalin 2 |
|--------------|------------------|------------------|
| | Expected Results | Measured Results |
| | | 100 |
| | 1.0000 | |
| | 0.9999 | |
| | 0.9998 | |
| | 0.9997 | |
| | 0.9996 | |
| | 0.9995 | |
| | 0.9994 | |
| | 0.9993 | |
| | 0.9992 | |
| | 0.9991 | |
| | 0.9990 | |
| | 0.9989 | |
| | 0.9988 | |
| | 0.9987 | |
| | 0.9986 | |
| | 0.9985 | |
| | 0.9984 | |
| | 0.9983 | |
| | 0.9982 | |
| | 0.9981 | |
| | 0.9980 | |
| | 0.9979 | |
| | 0.9978 | |
| | 0.9977 | |
| | 0.9976 | |
| | 0.9975 | |
| | 0.9974 | |
| | 0.9973 | |
| | 0.9972 | |
| | 0.9971 | |
| | 0.9970 | |
| | 0.9969 | |
| | 0.9968 | |
| | 0.9967 | |
| | 0.9966 | |
| | 0.9965 | |
| | 0.9964 | |
| | 0.9963 | |
| | 0.9962 | |
| | 0.9961 | |
| | 0.9960 | |
| | 0.9959 | |
| | 0.9958 | |
| | 0.9957 | |
| | 0.9956 | |
| | 0.9955 | |
| | 0.9954 | |
| | 0.9953 | |
| | 0.9952 | |
| | 0.9951 | |
| | 0.9950 | |
| | 0.9949 | |
| | 0.9948 | |
| | 0.9947 | |
| | 0.9946 | |
| | 0.9945 | |
| | 0.9944 | |
| | 0.9943 | |
| | 0.9942 | |
| | 0.9941 | |
| | 0.9940 | |
| | 0.9939 | |
| | 0.9938 | |
| | 0.9937 | |
| | 0.9936 | |
| | 0.9935 | |
| | 0.9934 | |
| | 0.9933 | |
| | 0.9932 | |
| | 0.9931 | |
| | 0.9930 | |
| | 0.9929 | |
| | 0.9928 | |
| | 0.9927 | |
| | 0.9926 | |
| | 0.9925 | |
| | 0.9924 | |
| | 0.9923 | |
| | 0.9922 | |
| | 0.9921 | |
| | 0.9920 | |
| | 0.9919 | |
| | 0.9918 | |
| | 0.9917 | |
| | 0.9916 | |
| | 0.9915 | |
| | 0.9914 | |
| | 0.9913 | |
| | 0.9912 | |
| | 0.9911 | |
| | 0.9910 | |
| | 0.9909 | |
| | 0.9908 | |
| | 0.9907 | |
| | 0.9906 | |
| | 0.9905 | |
| | 0.9904 | |
| | 0.9903 | |
| | 0.9902 | |
| | 0.9901 | |
| | 0.9900 | |
| | 0.9899 | |
| | 0.9898 | |
| | 0.9897 | |
| | 0.9896 | |
| | 0.9895 | |
| | 0.9894 | |
| | 0.9893 | |
| | 0.9892 | |
| | 0.9891 | |
| | 0.9890 | |
| | 0.9889 | |
| | 0.9888 | |
| | 0.9887 | |
| | 0.9886 | |
| | 0.9885 | |
| | 0.9884 | |
| | 0.9883 | |
| | 0.9882 | |
| | 0.9881 | |
| | 0.9880 | |
| | 0.9879 | |
| | 0.9878 | |
| | 0.9877 | |
| | 0.9876 | |
| | 0.9875 | |
| | 0.9874 | |
| | 0.9873 | |
| | 0.9872 | |
| | 0.9871 | |
| | 0.9870 | |
| | 0.9869 | |
| | 0.9868 | |
| | 0.9867 | |
| | 0.9866 | |
| | 0.9865 | |
| | 0.9864 | |
| | 0.9863 | |
| | 0.9862 | |
| | 0.9861 | |
| | 0.9860 | |
| | 0.9859 | |
| | 0.9858 | |
| | 0.9857 | |
| | 0.9856 | |
| | 0.9855 | |
| | 0.9854 | |
| | 0.9853 | |
| | 0.9852 | |
| | 0.9851 | |
| | 0.9850 | |
| | 0.9849 | |
| | 0.9848 | |
| | 0.9847 | |
| | 0.9846 | |
| | 0.9845 | |
| | 0.9844 | |
| | 0.9843 | |
| | 0.9842 | |
| | 0.9841 | |
| | 0.9840 | |
| | 0.9839 | |
| | 0.9838 | |
| | 0.9837 | |
| | 0.9836 | |
| | 0.9835 | |
| | 0.9834 | |
| | 0.9833 | |
| | 0.9832 | |
| | 0.9831 | |
| | 0.9830 | |
| | 0.9829 | |
| | 0.9828 | |
| | 0.9827 | |
| | 0.9826 | |
| | 0.9825 | |
| | 0.9824 | |
| | 0.9823 | |
| | 0.9822 | |
| | 0.9821 | |
| | 0.9820 | |
| | 0.9819 | |
| | 0.9818 | |
| | 0.9817 | |
| | 0.9816 | |
| | 0.9815 | |
| | 0.9814 | |
| | 0.9813 | |
| | 0.9812 | |
| | 0.9811 | |
| | 0.9810 | |
| | 0.9809 | |
| | 0.9808 | |
| | 0.9807 | |
| | 0.9806 | |
| | 0.9805 | |
| | 0.9804 | |
| | 0.9803 | |
| | 0.9802 | |
| | 0.9801 | |
| | 0.9800 | |
| | 0.9799 | |
| | 0.9798 | |
| | 0.9797 | |
| | 0.9796 | |
| | 0.9795 | |
| | 0.9794 | |
| | 0.9793 | |
| | 0.9792 | |
| | 0.9791 | |
| | 0.9790 | |
| | 0.9789 | |
| | 0.9788 | |
| | 0.9787 | |
| | 0.9786 | |
| | 0.9785 | |
| | 0.9784 | |
| | 0.9783 | |
| | 0.9782 | |
| | 0.9781 | |
| | 0.9780 | |
| | 0.9779 | |
| | 0.9778 | |
| | 0.9777 | |
| | 0.9776 | |
| | 0.9775 | |
| | 0.9774 | |
| | 0.9773 | |
| | 0.9772 | |
| | 0.9771 | |
| | 0.9770 | |
| | 0.9769 | |
| | 0.9768 | |
| | 0.9767 | |
| | 0.9766 | |
| | 0.9765 | |
| | 0.9764 | |
| | 0.9763 | |
| | 0.9762 | |
| | 0.9761 | |
| | 0.9760 | |
| | 0.9759 | |
| | 0.9758 | |
| | 0.9757 | |
| | 0.9756 | |
| | 0.9755 | |
| | 0.9754 | |
| | 0.9753 | |
| | 0.9752 | |
| | 0.9751 | |
| | 0.9750 | |
| | 0.9749 | |
| | 0.9748 | |
| | 0.9747 | |
| | 0.9746 | |
| | 0.9745 | |
| | 0.9744 | |
| | 0.9743 | |
| | 0.9742 | |
| | 0.9741 | |
| | 0.9740 | |
| | 0.9739 | |
| | 0.9738 | |
| | 0.9737 | |
| | 0.9736 | |
| | 0.9735 | |
| | 0.9734 | |
| | 0.9733 | |
| | 0.9732 | |
| | 0.9731 | |
| | 0.9730 | |
| | 0.9729 | |
| | 0.9728 | |
| | 0.9727 | |
| | 0.9726 | |
| | 0.9725 | |
| | 0.9724 | |
| | 0.9723 | |
| | 0.9722 | |
| | 0.9721 | |
| | 0.9720 | |
| | 0.9719 | |
| | 0.9718 | |
| | 0.9717 | |
| | 0.9716 | |
| | 0.9715 | |
| | 0.9714 | |
| | 0.9713 | |
| | 0.9712 | |
| | 0.9711 | |
| | 0.9710 | |
| | 0.9709 | |
| | 0.9708 | |
| | 0.9707 | |
| | 0.9706 | |
| | 0.9705 | |
| | 0.9704 | |
| | 0.9703 | |
| | 0.9702 | |
| | 0.9701 | |
| | 0.9700 | |
| | 0.9699 | |
| | 0.9698 | |
| | 0.9697 | |
| | 0.9696 | |
| | 0.9695 | |
| | 0.9694 | |
| | 0.9693 | |
| | 0.9692 | |
| | 0.9691 | |
| | 0.9690 | |
| | 0.9689 | |
| | 0.9688 | |
| | 0.9687 | |
| | 0.9686 | |
| | 0.9685 | |
| | 0.9684 | |
| | 0.9683 | |
| | 0.9682 | |
| | 0.9681 | |
| | 0.9680 | |
| | 0.9679 | |
| | 0.9678 | |
| | 0.9677 | |
| | 0.9676 | |
| | 0.9675 | |
| | 0.9674 | |
| | 0.9673 | |
| | 0.9672 | |
| | 0.9671 | |
| | 0.9670 | |
| | 0.9669 | |
| | 0.9668 | |
| | 0.9667 | |
| | 0.9666 | |
| | 0.9665 | |
| | 0.9664 | |
| | 0.9663 | |
| | 0.9662 | |
| | 0.9661 | |
| | 0.9660 | |
| | 0.9659 | |
| | 0.9658 | |
| | 0.9657 | |
| | 0.9656 | |
| | 0.9655 | |
| | 0.9654 | |
| | 0.9653 | |
| | 0.9652 | |
| | 0.9651 | |
| | 0.9650 | |
| | 0.9649 | |
| | 0.9648 | |
| | 0.9647 | |
| | 0.9646 | |
| | 0.9645 | |
| | 0.9644 | |
| | 0.9643 | |
| | 0.9642 | |
| | 0.9641 | |
| | 0.9640 | |
| | 0.9639 | |
| | 0.9638 | |
| | 0.9637 | |
| | 0.9636 | |
| | 0.9635 | |
| | 0.9634 | |
| | 0.9633 | |
| | 0.9632 | |
| | 0.9631 | |
| | 0.9630 | |
| | 0.9629 | |
| | 0.9628 | |
| | 0.9627 | |
| | 0.9626 | |
| | 0.9625 | |
| | 0.9624 | |
| | 0.9623 | |
| | 0.9622 | |
| | 0.9621 | |
| | 0.9620 | |
| | 0.9619 | |
| | 0.9618 | |
| | 0.9617 | |
| | 0.9616 | |
| | 0.9615 | |
| | 0.9614 | |
| | 0.9613 | |
| | 0.9612 | |
| | 0.9611 | |
| | 0.9610 | |
| | 0.9609 | |
| | 0.9608 | |
| | 0.9607 | |
| | 0.9606 | |
| | 0.9605 | |
| | 0.9604 | |
| | 0.9603 | |
| | 0.9602 | |
| | 0.9601 | |
| | 0.9600 | |
| | 0.9599 | |
| | 0.9598 | |
| | 0.9597 | |
| | 0.9596 | |
| | 0.9595 | |
| | 0.9594 | |
| | 0.9593 | |
| | 0.9592 | |
| | 0.9591 | |
| | 0.9590 | |
| | 0.9589 | |
| | 0.9588 | |
| | 0.9587 | |
| | 0.9586 | |
| | 0.9585 | |
| | 0.9584 | |
| | 0.9583 | |
| | 0.9582 | |
| | 0.9581 | |
| | 0.9580 | |
| | 0.9579 | |
| | 0.9578 | |
| | 0.9577 | |
| | 0.9576 | |
| | 0.9575 | |
| | 0.9574 | |
| | 0.9573 | |
| | 0.9572 | |
| | 0.9571 | |
| | 0.9570 | |
| | 0.9569 | |
| | 0.9568 | |
| | 0.9567 | |
| | 0.9566 | |
| | 0.9565 | |
| | 0.9564 | |
| | 0.9563 | |
| | 0.9562 | |
| | 0.9561 | |
| | 0.9560 | |
| | 0.9559 | |
| | 0.9558 | |
| | 0.9557 | |
| | 0.9556 | |
| | 0.9555 | |
| | 0.9554 | |
| | 0.9553 | |
| | 0.9552 | |
| | 0.9551 | |
| | 0.9550 | |
| | 0.9549 | |
| | 0.9548 | |
| | 0.9547 | |
| | 0.9546 | |
| | 0.9545 | |
| | 0.9544 | |
| | 0.9543 | |
| | 0.9542 | |
| | 0.9541 | |
| | 0.9540 | |
| | 0.9539 | |
| | 0.9538 | |
| | 0.9537 | |
| | 0.9536 | |
| | 0.9535 | |
| | 0.9534 | |
| | 0.9533 | |
| | 0.9532 | |
| | 0.9531 | |
| | 0.9530 | |
| | 0.9529 | |
| | 0.9528 | |
| | 0.9527 | |
| | 0.9526 | |
| | 0.9525 | |
| | 0.9524 | |
| | 0.9523 | |
| | 0.9522 | |
| | 0.9521 | |
| | 0.9520 | |
| | 0.9519 | |
| | 0.9518 | |
| | 0.9517 | |
| | 0.9516 | |
| | 0.9515 | |
| | 0.9514 | |
| | 0.9513 | |
| | 0.9512 | |
| | 0.9511 | |
| | 0.9510 | |
| | 0.9509 | |
| | 0.9508 | |
| | 0.9507 | |
| | 0.9506 | |
| | 0.9505 | |
| | 0.9504 | |
| | 0.9503 | |
| | 0.9502 | |
| | 0.9501 | |
| | 0.9500 | |
| | 0.9499 | |
| | 0.9498 | |
| | 0.9497 | |
| | 0.9496 | |
| | 0.9495 | |
| | 0.9494 | |
| | 0.9493 | |
| | 0.9492 | |
| | 0.9491 | |
| | 0.9490 | |
| | 0.9489 | |
| | 0.9488 | |
| | 0.9487 | |
| | 0.9486 | |
| | 0.9485 | |
| | 0.9484 | |
| | 0.9483 | |
| | 0.9482 | |
| | 0.9481 | |
| | 0.9480 | |
| | 0.9479 | |
| | 0.9478 | |
| </td | | |

Home Health Monitoring System



Bern Whalin |

The Home Health Monitoring System (HHMS) is a web-based application designed to facilitate communication between healthcare providers and patients. It includes features for managing patient records, tracking vital signs, and providing secure messaging. The system is built using Java and MySQL, and it is currently being used by several healthcare facilities.

System Components:

- Frontend:** A web-based interface for users to log in and access their information.
- Backend:** A Java-based application that manages patient data and sends notifications.
- Database:** MySQL database to store patient records and vital sign data.
- Cloud Integration:** Integration with various cloud services for data storage and processing.

Key Features:

- Secure Patient Data:** Stores sensitive patient information such as medical history, medications, and vital signs.
- Remote Monitoring:** Monitors patient vital signs (e.g., blood pressure, heart rate) and sends alerts to healthcare providers if values fall outside normal ranges.
- Secure Messaging:** Provides a secure way for healthcare providers to communicate with patients.
- Reporting:** Generates reports on patient activity and trends.

System Architecture:

```

graph TD
    Frontend[Frontend] --> Backend[Backend]
    Backend --> Database[Database]
    Backend --> Cloud[Cloud]
    Database --> VitalSigns[Vital Signs]
    VitalSigns --> Alerts[Alerts]
    Alerts --> Providers[Providers]
    Alerts --> Patients[Patients]
    Providers --> Messages[Messages]
    Patients --> Messages
    
```

Future Enhancements:

- Integration with more cloud services.
- Mobile application for healthcare providers.
- AI-powered predictive analysis for early disease detection.

Conclusion:

The Home Health Monitoring System has shown promise in improving patient care and communication. As technology continues to evolve, we expect to see even more advanced features and integrations in the future.

REFERENCES

- Carrie A. Werner (2019). *The Older Population: 2010*. <https://www.census.gov/policy/censos2010/info/censusbr.pdf>
- Average Cost of Senior Living Facilities. <https://www.seniorliving.org/assisted-living/costs/>
- <https://medekpm.com/how-does-pm-work>
- Powerpoint Concept Establishment: <https://synapsiscreative.com/concepting-animation-for-process-powerpoint/>



1. Figure of merit (n.d.) n.d. Available from: <https://www.sciencedirect.com/topics/chemistry/figure-of-merit>.