Smart Digital Stethoscope - Phase II

Design Document
sdmay22-18

Client/Adviser: Dr. Khokhar

Team Members/Roles

- Austin Collins - WebRTC and Other Necessary Software Engineer
- Yilun Huang - Product Design Engineer
- Omar Alsaedi - User Interface Design Engineer
- Abdalla Saeed Alzaabi - Network and Communication Engineer
- Vignati Yalamanchili - Embedded System Engineer/Scrum Master
- Matthew Gasparaitis - Hardware Engineer
- Joyce Lai - Communications Engineer/Scrum Master

Team Email: sdmay22-18@iastate.edu
Team Website: http://sdmay22-18.sd.ece.iastate.edu

Executive Summary

Development Standards & Practices Used

Circuit, Hardware, and Software Practices

- Analog bandpass filter design
- Smart board usage
- Bluetooth data transmission
- Data live-streaming
- WebRTC server
- Networking protocols
- Data storage (database)

Engineering Standards

- IEEE 1073-10441-2013
- IEEE 1652-2004
- IEEE 1522-2004

Summary of Requirements

- Measure patient heart sounds
- Filter out noise and undesired frequencies
- Wirelessly send audio to computer
- Send the data to a server in real-time
- Server that can collect and store data
- Server that can query and send data
- Machine learning-driven detection of heart irregularities

Applicable Courses from Iowa State University Curriculum

- EE 201
- EE 224
- EE 230
- EE 321
- EE 422
- EE 423
- BIOL 212
- BIOL 256
- BME 220
- CprE 288
- SE 309
- CprE 489

New Skills/Knowledge acquired not taught in courses

Networking Protocols

- SDP: Session Description Protocol
- ICE: Interactive Connection Establishment
- DTLS: Data Transport Layer Security
- SRTP: Secure Real-time Protocol
- RTP: Real-Time Transport Protocol
- SCTP: Stream Control Transmission Protocol
- RTCP: Real-Time Transport Control Protocol

Languages, Software, Data Transmission

- Javascript
- CSS
- HTML
- WebRTC
- Web Sockets

Table of Contents

1. Team

1.1 Team Members

- Austin Collins - WebRTC and Other Necessary Software Engineer
- Yilun Huang - Product Design Engineer
- Omar Alsaedi - User Interface Design Engineer
- Abdalla Saeed Alzaabi - Network and Communication Engineer
- Vignati Yalamanchili - Embedded System Engineer/Scrum Master
- Matthew Gasparaitis - Hardware Engineer
- Joyce Lai - Communications Engineer/Scrum Master

1.2 Required Skill Sets for Our Project

- Basic analog and/or digital filter design & implementation – Filter out frequencies outside the range of heartbeat (and lung contraction) frequencies.
- CAD design & 3D printing – Condense existing stethoscope hardware setup in order to compactly contain circuit, microphone, and transmitter.
- ADC design & implementation – Convert analog/electrical signals associated with the patient's stethoscope reading into digital data.
- Embedded system design & implementation – ADC implementation, WebRTC implementation
- Wireless communication system design and/or knowledge – Bluetooth receiver/transmitter (RX/TX) design between smart stethoscope and computer/wireless headphones or knowledge to select and purchase an appropriate RX/TX.
- Bluetooth protocol
- WebRTC – Real-time communication/transmission of patient data over WiFi.
- Machine learning – Train the ML model to identify different heart irregularities (i.e. – arrhythmia, ischemia, murmur, tachycardia, etc.)

1.3 Skill Sets Covered By the Team

- Joyce, Yilun – Basic analog and/or digital filter design & implementation
- Yilun, Matt, Joyce – CAD design & 3D printing
- Abdalla, Omar, Austin, Matt – ADC design & implementation
- Abdalla, Omar, Austin, Vignati – Embedded system design & implementation
- Joyce, Yilun – Wireless communication system design and knowledge

1.4 Project Management Style Adopted by the Team

The project management style we have adopted is the agile project management style.

1.5 Initial Project Management Roles

- **Developers** – Joyce, Yilun, Matt, Austin, Omar, Abdalla, Vignati
- **Scrum Master(s)** – Vignati, Joyce
- **Product Owner** – Dr. Khokhar

2. Introduction

2.1 Problem Statement

Heart-related illnesses such as heart attacks and heart failure and lung-related illnesses such as lung failure often occur suddenly. These illnesses are typically chronic and occur in individuals who are genetically predisposed or have records of related illnesses.

In the event of a health emergency, patients may not have the ability to get to an emergency room due to financial factors, day-to-day obligations, predetermined travel plans, and other factors. Additionally, patients may not even be aware of the severity of their health condition.

Likewise, doctors may not be physically present (in the office) at the time of their patient's emergency due to similar factors. Also, the frequent relocation of healthcare workers is a common impediment to patients' consistent care.

Our project gives the ability for doctors to monitor patients with heart- and lung-related issues with minimized contact and increased efficiency. On the day-to-day, patients will be able to individually measure their heartbeats or lung contractions without needing to be in the presence of a healthcare professional. Patient data will automatically be converted into digital data that will be streamed in real-time via WiFi to a server that healthcare professionals can access at any time. Additionally, our project will enable automatic detection of heart and lung irregularities; any irregularities detected will result in notification to the healthcare professional.

Automatic patient data recording, transmission, streaming, and detection will increase the accessibility of healthcare for all patients, ease from the healthcare professional's side, and convenience for all.

2.2 Requirements & Constraints

This section outlines the requirements and constraints of the project into functional, resource, and qualitative aesthetics.

Functional

- Measure patient heart sounds
- Filter out noise and undesired frequencies
- Wirelessly send audio to computer
- Send the data to a server in real-time
- Server that can collect and store data
- Server that can query and send data
- Machine learning-driven detection of heart irregularities

Broadly, these can be grouped into the following categories: hardware circuit design, signal processing, data transmission, server architecture, network protocol, and machine learning.

Resource

- ADC – Conversion of analog/electrical signals associated with the patient's stethoscope reading into digital data that can be transmitted wirelessly and over WiFi.
- WebRTC server – Real-time communication/transmission of patient data over WiFi to the doctor via a secure server.
- Analog filter – Nullification of frequencies beyond the range of human heartbeat and lung contraction frequencies.
- Machine learning algorithm – Detection of heartbeat and lung contraction pattern abnormalities.

- Bluetooth chip- Wireless communication of patient data between digital(Tx) and computer (Rx)
- 3D printer – Manufacturing of a container for the compact digital stethoscope system and its associated hardware parts.
- PCB designer and Manufacturer – Design and manufacturing of PCB
- 500$ budget and previous project iteration supplies

Qualitative Aesthetics

- PCB – Compact chip containing all necessary circuitry as opposed to breadboard platform.
- Final product – Compact overall hardware system enclosed within a 3D printed container

2.3 Engineering Standards

- IEEE 1073-10441-2013 - Health Informatic -- Personal health device communication Part 10441: Device specialization -- Cardiovascular fitness and activity monitor
    - Chose this over the ECG standard because our project is developing a working stethoscope, and this was the closest guideline to monitoring cardiovascular health which fit the parameters.
- IEEE 1652-2004 - IEEE Standard for Rechargeable Batteries for Portable Computing
    - Our design will contain some type of Battery with(hopefully) rechargeable capabilities. We will have to safely apply this technology to power our circuit
- IEEE 1522-2004 - IEEE Standard for Testability and Diagnosability Characteristics and Metrics
    - We want to follow all testing requirements needed to possibly be considered for use in the medical field

2.4 Intended Users and Uses

Anyone who monitors, documents or interprets heart and lung sounds will benefit from this project. The product we aim to create is multifaceted: it documents the sounds in a database; provides an analysis tool with which the interpreter can be prompted with suggestions about its abnormalities, provides enhanced noise cancellation and sound enhancement, provides real time connectivity anywhere in the world and does so without the mess of wires. This has several obvious benefits to anyone who has vested interest in keeping themselves or others that way.

Healthcare workers (i.e. – doctors, nurses, etc.) and patients, especially those with chronic heart issues, will all benefit from our project. Doctors and nurses will be able to use the product to examine and monitor their patients' heartbeats and lung contractions without having to be in the clinic or hospital. Likewise, patients will be able to individually record their heartbeats and lung contractions without having to be in the clinic or hospital. Thus, this project will make healthcare more convenient for all and more accessible for individuals that live in areas with no medical care centers nearby.

3. Project Plan

3.1 Project Management/Tracking Procedures

We are adopting the waterfall and agile project management styles for our project. Since we have very specific requirements for our end project, we need to use waterfall to ensure that our end product has Bluetooth (BT) connectivity and a server that can stream live patient data between the patient(s) and doctor(s).

Since there are other functionalities that we would like to implement that are not of priority or required such as design compactness (PCB implementation and 3D printed case), cost-efficiency, we are also adopting agile project management style. We will first focus on creating and testing a basic prototype with the BT functionality and real-time streaming server. Then we will restart the design process to add other features and test them individually.

We plan to use Git to share project-related code and also to set deadlines for the software development-related tasks. We also plan to use Google Drive where we will keep track of meeting minutes and to-dos for each week. Those will also be kept track of in our team's Discord server so they are more easily accessible.

3.2 Task Decomposition

The two major components of our project are Bluetooth connectivity and real-time streaming server. We have split our team into two groups to target these areas.

For the Bluetooth functionality, there are multiple subtasks:

I. Identify requirements for sending our real-time patient data (i.e. – data rate, etc.).
II. Identify a suitable microcontroller and BT chip/module combination given our requirements.
III. Identify a suitable ADC chip/module to work with our analog signals and BT transmitter.
IV. Connect and configure the microcontroller, ADC, BT chip/module.
    A. Test the configuration and initialization of ADC and microcontroller.
    B. Test the configuration and initialization of BT and microcontroller.
    C. Test the microcontroller with ADC and BT modules. (For initial stages, testing can be done without the physical stethoscope and bandpass filter and instead with a waveform generator as our analog signal.)

For the WebRTC real-time patient data streaming server, there are also multiple subtasks:

I. Identify the structure of WebRTC and server setup.
II. Establish a server with 1 channel.
    A. Test the server's channel for real-time sending and receiving of text messages.
III. Enable multiple channels within the server.
    A. Test the server's channels for simultaneous real-time sending and receiving of text messages.
IV. Enable automatic channel assignment for each user (patient or doctor).
V. Enable server GUI to only stream when both end-users have indicated they are ready.
VI. Adapt the server to allow real-time streaming of audio files.
    A. Test the server's channels for real-time streaming of audio files.
    B. Ensure that audio files are only streamed as long as they are being sent from one end (true live-streaming).
    C. Test server with real-time streaming of real-time microphone audio input.
VII. Enable two types of user accounts and corresponding login processes.

For the final product with BT connectivity and WebRTC server, there are the following subtasks:

I. Build bandpass filter (BPF) prototype on breadboard. (Second order filter should be sufficient.)
II. Connect battery, stethoscope, microphone, and BPF, to ADC, microcontroller, and BT modules.
    A. Test the transmission of stethoscope readings through the system.
    B. If needed, test after each component (i.e. – microphone, BPF, ADC, etc.) individually.
III. Configure server to take patient data readings after the computer platform receives BT signal.
    A. Test the whole system (all hardware and streaming of patient-data through WebRTC server).

IV. When BT connectivity and WebRTC are successfully integrated, replace BPF on breadboard with PCB design of BPF.
    A. Test the whole system accordingly.
    B. Likewise, repeat the process to replace or improve any component or aspect of the smart digital stethoscope system.

It should be noted that the BT connectivity and WebRTC development can be worked on in parallel.

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

For the Bluetooth functionality, we have proposed the following milestones and metrics:

I. Hardware systems (microcontroller, ADC, and BT modules) has sufficient data transfer and transmission rates for sending real-time stethoscope data.
II. Hardware system uses an appropriate amount of power (power equivalent to 3 AA batteries which the previous team used).
III. Hardware systems can capture sinusoidal signals at frequencies within the BPF range with gain of at least 0 dB (output/input magnitude = 1).
IV. Hardware system nullifies irrelevant frequencies with change in gain of approximately -3dB (output/input magnitude = $1/\sqrt{2}$).
V. Hardware systems have a good power efficiency.

For the WebRTC streaming server, we have proposed the following milestones and metrics:

I. Establish a server with 1 channel that can send and receive text messages with delay of less than 1μs.
II. Establish a server with multiple channels that can send and receive text messages with delay of less than 1μs.
III. Establish automatic channel assignment for each user (patient or doctor).
IV. Establish server GUI that only streams when both end-users have indicated they are ready.
V. Establish real-time streaming of audio files with delay of less than 1μs.
VI. Establish real-time streaming of real-time microphone audio input with delay less than 1μs.
VII. Establish a server with two types of user accounts and corresponding login processes.

3.4 Project Timeline/Schedule

The project timeline and schedule is given below in Fig. 1.

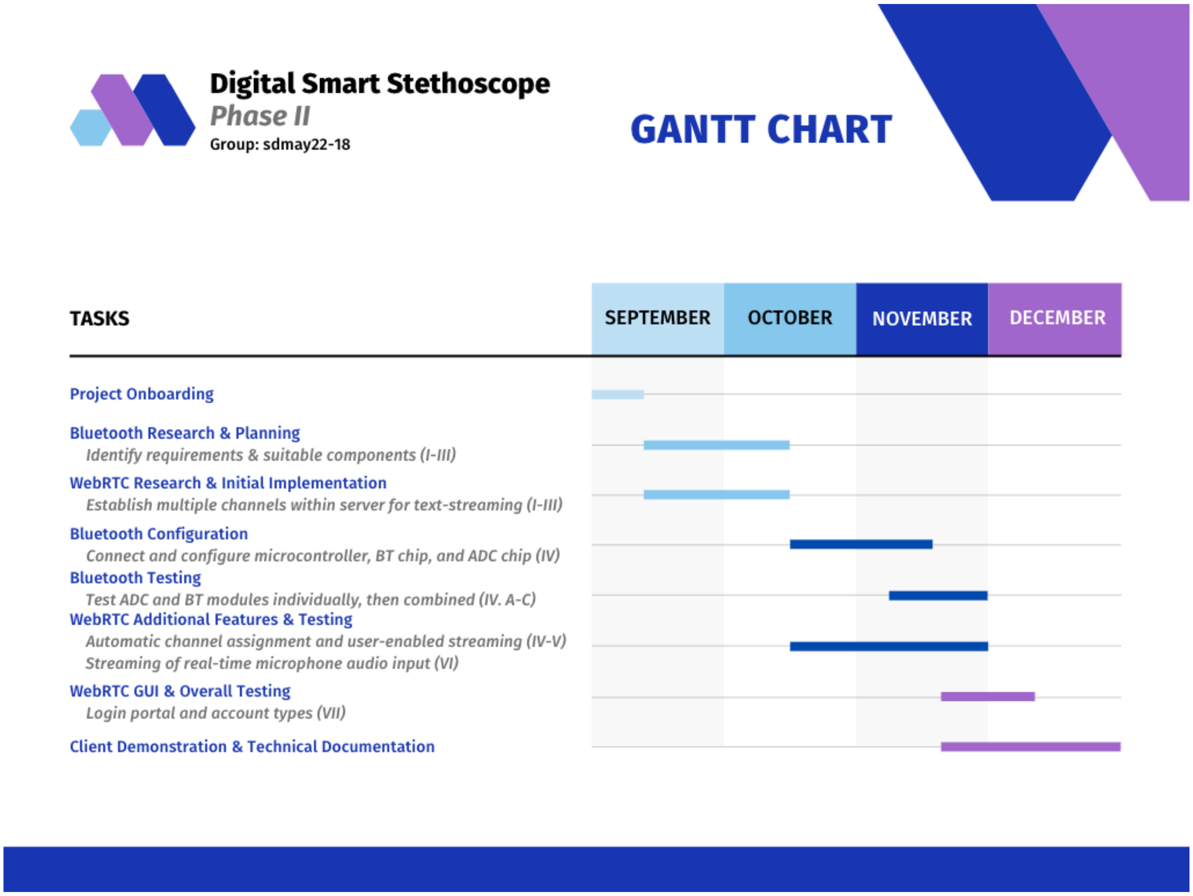

Figure 1. Gantt chart for the smart digital stethoscope phase II project

3.5 Risks And Risk Management/Mitigation

Scope Creep
We need to make sure we have clear boundaries in both our hardware and software designs so we do not do too much that is unnecessary or unwarranted.

Insufficient Testing
We need to make sure we are testing all of our work sufficiently prior to moving onto another part of the project.

Insufficient Resources
We need to ensure that we are purchasing the correct equipment so we do not squander our budget.

3.6 Personnel Effort Requirements

The efforts of the project team are distributed and summarized in the table below.

Task	Individual(s) Assigned	Time per Person (Est. Minimum)	Explanation
Bandpass Filter Design	Matthew Gasparaitis, Joyce Lai, Yilun Huang	20 min	Already done by previous group, just need to verify/adjust design as needed
Bluetooth Transmitter Research & Configuration with Microcontroller	Matthew Gasparaitis, Joyce Lai, Yilun Huang	4 hrs	Will need to find a both cost-efficient and energy efficient chip that will transfer the needed number of packets, then we will need to install said device to the basic circuit
ADC Integration	Matthew Gasparaitis, Joyce Lai, Yilun Huang	1 hr	Deciding on whether the ADC implementation will be on a breadboard or a software.
WebRTC Research & Initial Channel Testing	Austin, Abdalla, Omar	2 hrs	Will need to decide if we are using Spring-boot or linux libraries to send the audio through IP addresses
WebRTC Integration	Austin, Abdalla, Omar	1 hr	Will need to find a way to change the Server-Client connectivity from sending a String on chars to Audio files.
Testing	Joyce & Austin	5 hrs	Will send out audio files/test the real time streaming to verify the usefulness of our finished stethoscope. However, testing throughout the design process is also required.

Table 1. Personnel effort requirements for the smart digital stethoscope phase II

3.7 Other Resource Requirements

Other resource requirements are summarized below.

- Previous group members' contact information for potential technical guidance
- Previous group's hardware components
- Previous group's technical documentation (i.e. – schematic diagrams)
- Bandpass filter: Breadboard(s), resistors, capacitors, operational amplifier(s), PCB designer and manufacturer, connection wires
- Stethoscope (traditional, non-digital)
- Microphone (that converts pressure waves to electrical signals)
- Microcontroller (specific model TBD)
- ADC module/chip (specific model TBD)
- Bluetooth module/chip (specific model TBD)
- Computer for system testing (on which we will pull up the WebRTC server and use as BT receiver)

- Qualitative testing – Medical experts to evaluate accuracy of our digital audio
    - Dr. Asim Nisar, Cardiologist, Advocate Heart Institute, Elgin, IL
    - Austin (group project member), EMT

4. Design

4.1 Design Context

4.1.1 Broader Context

Smart digital stethoscopes are becoming a popular alternative to the traditional pressure-based stethoscope; however, the cost of such digital stethoscopes remains high at a few hundred dollars apiece. The goal of our digital smart stethoscope is to be affordable for a larger population in addition to being compact, portable, and achieving the same functionalities as existing smart stethoscopes.

The affordability, compactness, and accuracy of our device will be better suited for low-income communities and small, rural communities where access to healthcare is limited by distance. Additionally, our device will be an effective and cheaper alternative to current smart digital stethoscopes for all who prefer or require remote monitoring of chronic heart or lung conditions.

Area	**Description**	**Examples**
Public health, safety, and welfare	The solution is implemented in their communities, offers convenience, social distancing, remote health monitoring, affordability, and accessibility.	• Allows for remote health monitoring and potential prognosis or diagnosis of existing heart and lung conditions. • Remote monitoring has the potential to decrease the spread of viruses and other diseases such as COVID-19. • Saves doctors' and patients' time by eliminating or reducing travel time for physical medical appointments.
Global, cultural, and social	The solution reflects generally universal values of cost efficiency, sustainability (low-power), health-consciousness (remote, virtual, social distancing), and convenience.	Operation of the solution may: • Result in Health Insurance Portability and Accountability Act (HIPAA) violations if not done correctly. • Require changes in medical professionals' practices and routines.

Environmental	The solution offers energy efficiency; however, the manufacturing of the product itself contributes to materials manufacturing and consumerism.	• Increase manufacture of chips and microcontrollers • Increase production of non-recyclable materials • Decrease energy consumption and manufacturing in comparison to preexisting smart digital stethoscopes.
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups. The solution requires funding from the ISU ECpE Department, will be purchased by patients and medical institutions, and compete with existing smart digital stethoscopes.	• Product needs to remain affordable for target users. • Designing, prototyping, and testing the product needs to remain affordable for the Department of ECpE. • Product will affect the demand of competing smart digital stethoscopes once it enters the market.

Table 2. Smart Digital Stethoscope broader context and societal impacts summary

4.1.2 User Needs

The first user group is low-income communities. This product will help low-income communities reduce costs because their hospital visits will decrease and that will increase affordability chances to get better care.

The second user group is small, rural communities. The product will help the people in small, rural communities with increased accessibility to medical care, especially older citizens who live in the countryside or by themselves.

The third user group is patients with chronic heart or lung conditions. The product will help the patients get care remotely and continuously from their personal doctor.

The fourth user group is healthcare workers and medical professionals. The product will help the healthcare workers reduce the need for in-person appointments and decrease work stress.

Overall, the product will increase the quality of healthcare and give a chance for medical professionals to focus more on critical patients. This product will also protect the lives of healthcare workers and medical professionals amid the COVID-19 pandemic.

4.1.3 Prior Work/Solutions

Prior work has been done by past ISU students for the Smart Digital Stethoscope – Phase I project. They have successfully created a noise-canceling filter. They were also able to convert pressure waves into analog electrical signals which were then converted to digital signals (audio data) via a wired connection. Although the smart digital stethoscope achieved successful conversion of pressure waves into audio files, their final product did not include BT functionality or live data-streaming.

Smart digital stethoscope technology also already exists in the market. One of the products is the Eko CORE Digital Attachment which costs around two hundred and forty-nine dollars. The other product is the Thinklabs One Digital Stethoscope which costs around five hundred dollars. Thinklabs One Digital Stethoscope is similar to what we are trying to achieve in terms of physical design and functionality minus the LED screen for the graphical user interface. Although both products effectively achieve the desired functionality that we are also striving for, the main setback is in the high cost making it affordable only for a small population.

4.1.4 Technical Complexity

In this section we discuss the technical complexity of the project.

The design consists of multiple components that each utilize distinct scientific, mathematical, and engineering principles including:

- Analog filter design and implementation
- Analog-to-Digital Converter (ADC)
    - Quantization
- Microcontroller and submodules (BT, ADC) configuration & initialization
    - UART protocol
    - Bit and baud rate
    - Sampling rate
- Web server and network architecture
- Effective data storage
- Printed circuit board(PCB) design
- 3D Computer-aided design (CAD)

The problem scope contains multiple challenging requirements that match or exceed current solutions:

- Eko CORE Digital Attachment
- Thinklabs Digital One Stethoscope

4.2 Design Exploration

4.2.1 Design Decisions

Our key design decisions are listed in the table below where our final decisions are in the left column.

Chosen Design Decision	**Alternative Design Decision**
Microcontroller plus a separate Bluetooth chip	Using a microcontroller with an integrated Bluetooth (BT) module
No WiFi chip; Data is sent via BT to the computer where data is uploaded to the web server (computer is the client)	Using WiFi chip to send data directly from stethoscope (stethoscope is the client)

Basic noise-cancellation microphone (one microphone)	Equalizer for measured ambient noise-cancellation (two microphones)

Table 3. Main design decisions summary table

4.2.2 Ideation

In deciding the platform for the microcontroller and BT connectivity, we first consulted the project's phase I group about issues with their platform and possible solutions. In the following, we researched for possible setups that would achieve our desired functionality. Based on our research and ideas, we received feedback from our client and recommendations from our professors.

The required components for our product are listed below with potential options listed under each category. The bolded components are our final decision.

- Stethoscope head
- Microphone and pressure-wave transducer
  - **Electret Microphone Amplifier - MAX9814 with Auto Gain Control (original)**
- Microcontroller unit (MCU)
  - Launchpad (original)
  - Arduino
  - LPC55S6x High-Efficiency Arm Cortex-M33 Core MCU
  - Raspberry Pi
  - **Raspberry Pi Pico (RP2040)**
- Bluetooth module
  - Launchpad's integrated BT module (original)
  - Arduino's integrated BT module
  - NXH3670 BLE Chip for Audio Streaming
  - **HiLetgo HC-05 BT Wireless BT Chip**
- Bandpass filter
  - **Analog filter (original)**
  - Digital filter
- Battery
  - 2 AAA non-rechargeable batteries - 3v (original)
  - Lithium Ion Polymer Battery - 3.7v 1200mAh
  - **Lithium Ion Polymer Battery - 3.7v 500mAh**
- Battery charger
  - **Adafruit Micro-Lipo Charger for LiPo/LiIon Batt w/MicroUSB Jack - v1**

4.2.3 Decision-Making and Trade-Off

In the end, we decided to use the Raspberry Pi Pico and HiLetgo HC-05 BT Wireless BT Chip combination over all other microcontrollers with integrated BT or IoT platforms because of its low cost, compactness, and detailed documentation.

The stethoscope and microphone were inherited from the previous year's project group since the components worked effectively.

We ruled out the original platform, the Launchpad, since the integrated BT module's data rate could not be easily customized in order to send our desired frequency range. The Launchpad also exceeded our desired product size. Additionally, we ruled out the Arduino due to its size

We ruled out the NXH3670 because it had undesired data processing and noise cancelling functionalities, the LPC55S6x because it lacked BT functionality, and the Raspberry Pi due to its size and cost. Thus, we decided on the Raspberry Pi Pico.

We decided to use analog filtering over digital filtering since we wanted to eliminate noise before transmitting the data. Digital filtering has the potential to amplify noisy signals if they are not filtered out in the analog circuitry.

The battery was chosen based on the minimum power requirement of the Raspberry Pi Pico and was also the cheapest battery out of our choices. The charger is the standard charger component for our desired rechargeable battery.

4.3 Proposed Design

We have implemented our product using the following components:

- Microphone – Adafruit MAX9814
- Filter – Analog Bandpass Filter (BPF)
- MCU and ADC module – Raspberry Pi Pico
- BT module – HiLetgo HC-05 Wireless BT Chip
- Web Server Platform – WebRTC

The analog BPF has been physically realized and simulated in LTSpice. The microcontroller and BT chip have been purchased and their functionalities integrated.

The WebRTC platform has already been established and is in the second stage of development. The server has a user login (identification) method and multiple instances or channels in which text messages and microphone input audio can be streamed. We have also explored a second software implementation using the doctor's IP address and C Libraries.

4.3.1 Design Visual and Description

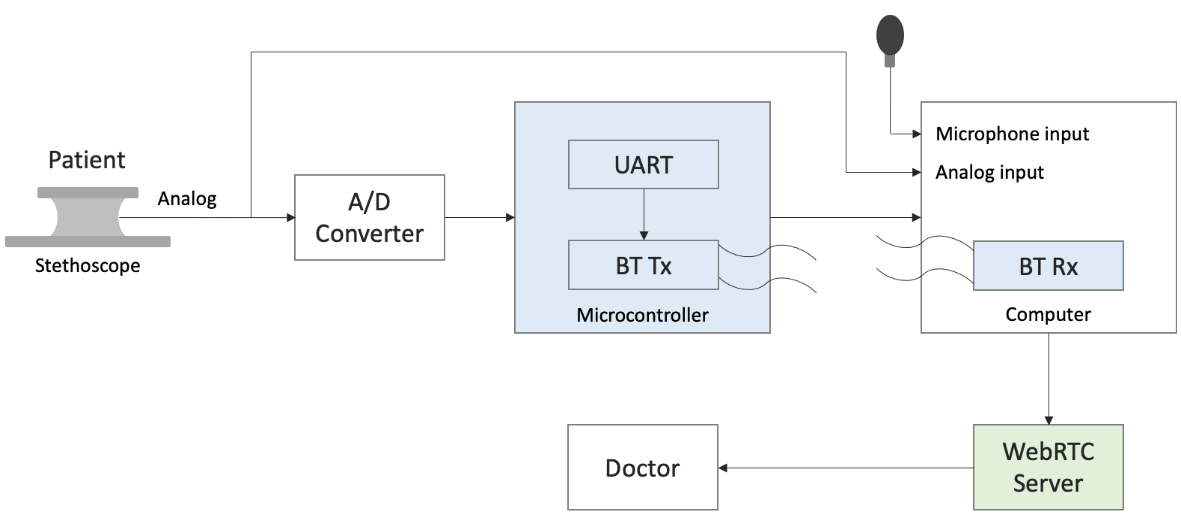

Figure 2. Smart digital stethoscope overall system schematic diagram.

Fig. 2 shows the smart digital stethoscope's overall system schematic diagram. It begins at the stethoscope where the patient will measure their heartbeat or lung contractions which will be immediately converted to analog electrical signals. The analog signals will pass through an ADC module where the signal will be quantized, sent via UART protocol to the BT transmitter, and received by the computer's built-in BT receiver. Through the computer's pre established Wi-Fi connection, the real-time audio data will be uploaded and streamed via the WebRTC server. The doctor will be able to listen to the audio stream via the same WebRTC server from any location as long as they are connected to Wi-Fi.

4.3.2 Functionality

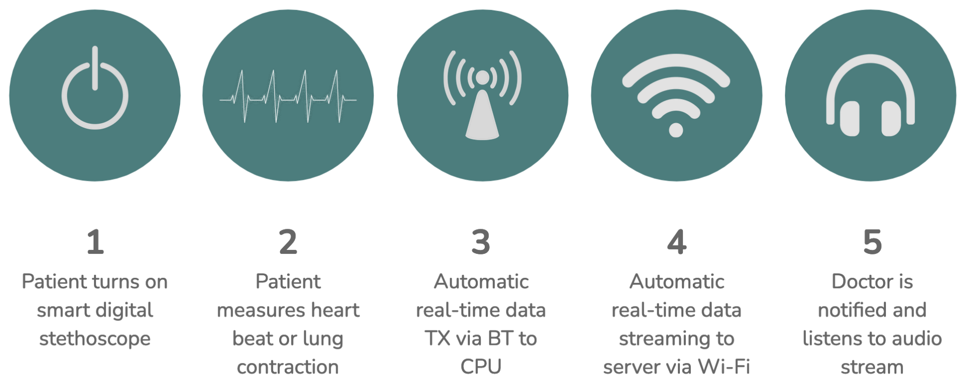

Figure 3. Smart digital stethoscope usage process

Fig. 3 shows the smart digital stethoscope's usage process. The patient will turn on the smart digital stethoscope, then measure their heartbeat or lung contraction. The associated electrical signals will be automatically converted to digital signals and transmitted in real-time via BT to the user's computer.

The computer will automatically stream the data through the server in real-time via WiFi, and the doctor will be notified on their end so they can tune in and listen to the patient's audio stream.

Our design satisfies all the technical functionalities (pressure to electrical to digital signal conversion, noise filtering, BT transmission, WiFi audio streaming through server); however, it does not yet satisfy industry standards in terms of final product presentation. The analog noise filter is implemented on a breadboard and has yet to be translated into a PCB. The overall system has yet to be encased by a 3D-printed case.

4.3.3 Areas of Concern and Development

Based on our current design, our primary concerns for delivering a successful smart digital stethoscope with all required functionalities and which meets the user and client needs are:

1. Making it compact and into one unit.
2. Keeping the "look" of a stethoscope.

To address these concerns, we plan on removing the tubular component of the stethoscope and attaching the microphone to the chest piece directly. The rest of the circuitry and components (analog BPF, Raspberry Pi Pico, HC-05 chip) will be wired and encapsulated compactly with a 3D printed case. that will be connected to the chest piece via a long durable wire so that the user can collect their data without holding the external circuitry and components and will only be required to hold the actual stethoscope chest piece.

4.4 Technology Considerations

The available microcontrollers have a very high ADC conversion/quantization rate; however, this is also a weakness in the context of our project. Since the frequencies where are seeking to capture are in the low range (< 5kHz), our system's required conversion and data transmission rate are also low. Additionally, the BT transmission will not be able to handle such high precision and high conversion rate from the ADC. Thus, a workaround will need to be established.

A possible solution is to manually program the ADC to a lower conversion rate using interrupts or timers. Another option is to use a data compression technique following the ADC before the BT transmission.

4.5 Design Analysis

So far, the server works at a very basic level. We have not gotten to test the proposed hardware design yet, as we have just decided on the hardware components and will need to place an order and wait for shipping.

If our proposed design fails, we have possible workarounds in the previous part, section 3.4.

4.6 Design Plan

We will develop the BT connection in parallel with the WebRTC server. Development and testing of the WebRTC server can be done independently from the BT module by instead using microphone input or preexisting audio files. The BT connection can be developed and tested independently from the other hardware (stethoscope and BPF) as well by instead using a waveform generator as the input voltage.

5. Testing

5.1 Unit Testing

The individual components of our system are the stethoscope chest piece and microphone, analog bandpass filter, Raspberry Pi Pico (microcontroller) which includes the ADC and is connected to the HC-05 Bluetooth chip (transmitter), and the computer (receiver) on the patient's side that we will be using as an interface for the WebRTC server as a web page, then the computer on the medical professional's side.

A unit test will be taken at the output of each individual component. The electrical signal of the patient's associated heartbeat pressure wave will be measured via oscilloscope and its frequency components observed. After the bandpass filter, we will measure the output via oscilloscope again and its frequency components to ensure that only the proper frequencies are passing through. We will then inspect the output of the ADC module to ensure that the electrical signals are properly quantized as per the Raspberry Pi Pico datasheet specifications and our code. The Bluetooth module will be evaluated based on how well the computer receives the BT signal and what the audio data looks like in terms of frequency components. Once those unit tests have been successfully completed, we can move onto interface testing.

All signal inspection tests for frequency components will be done using simple MATLAB programming or Simulink.

We will also have many different unit tests within the software itself. Instead of explaining each possible instance of unit testing, I will instead explain the types of unit testing we must complete. We use SQL unit testing to make sure the data is storing, querying and sending properly. We use WebRTC unit testing to make sure the stream of information is being sent and received in real time. We do Artificial Intelligence (AI) unit testing in order to ensure the AI results make sense when compared to the problem we are trying to solve. Finally, we do Web App Navigation Unit Testing to Ensure the app's UI switches pages and is responsive to User Commands.

5.2 Interface Testing

Our interfaces are as follows:

- Oscilloscope → Bandpass filter
- Bandpass filter → ADC module
- ADC module → Bluetooth module
- Bluetooth module → Web App
- Web App → SQL
- Web App → WebRTC
- Web App → AI
- SQL → Web App

For each interface, we test by ensuring that each of these are lining up in terms of transmission and receiving data. We will need to use different tools depending on each one we are doing. Some of these tests can be as simple as starting the software and seeing if data is transmitting, and others may be simulating the hardware (LTSpice).

We will test all the server-oriented interfaces with postman and MySQL workbench. Our tools will inevitably grow and develop throughout the process as we get a feel for what works and what does not.

5.3 Integration Testing

The stethoscope will be getting a sound reading from wherever we hold the device, presumably the heart and lung area(s) in our case. The stethoscope will then communicate with the microphone, which is secured at the end of the stethoscope tube. This can be tested by getting an oscilloscope reading detecting frequencies when connected to the breadboard.

The bandpass filter will filter out high and low frequencies in order to get readings, this will also be tested by using an oscilloscope until it can be hooked up to the website. Once a website/digital interface can be used, we will no longer have to test using the oscilloscope, other than to verify the functionality of that specific part.

The Bluetooth and website will display the information read by the device. The last group had hooked up a USB to the breadboard directly; however, we will use bluetooth to communicate with the device in our phase. A website can be tested by comparing the results on screen vs. results from the bandpass oscillation that should be correct. We will then compare a hardwired result from the website vs. the bluetooth connection to determine which component is failing.

The WebRTC server will provide real-time audio streaming between the patient and medical professional. This will be tested by simply being able to hear the audio in real time between two group members trying to communicate with each other. We must also test that the audio is able to pick up the correct frequencies of sound to the point that a medical professional would consider the device usable in their workplace. We will verify this by having actual medical professionals: Cardiologist Dr. Asjm Nisar, Dr. Vincent Gasparaitis, and our group member Austin who is a trained EMT.

The secondary server to store recorded audio from patients can be tested by being able to retrieve the audio file from said server, which was previously used in real-time communication.

5.4 System Testing

Successful unit, interface, and integration tests are all imperative for the success of the overall system. Assuming the unit, interface, and integration tests outlined in previous sections have been successfully completed, we can then test the functionality of the overall system.

The test will require a healthy subject to act as a patient and measure their heart beat. The audio data sent through the WebRTC server and stored in the database will be compared to validated, sample heartbeat audio data of a healthy patient available on online databases such as PhysioNet.

We will use a coherence test to determine coherence between the two signals' frequency components. The coherence value will be our test vector. As of now, our threshold for a sufficient output signal is a coherence with the validated healthy patient's data of greater or equal to 0.9, where coherence near zero implies little correlation, and coherence near one implies significant correlation. The coherence will help us to quantify how much our digital heart beat signal resembles a true heart beat signal, as the main functionality of the overall smart digital stethoscope is to simply imitate a traditional stethoscope in a digital format.

To accomplish this system-level test, we will utilize MATLAB signal analysis tools and functions, specifically the mscohere function and other data visualization tools documented here.

5.5 Regression Testing

We will only run test cases that are relevant to the set of changes that can ensure the new additions do not break the old functionality. We will begin with critical systems and go to functional systems and then to non-functional systems. Also, we can set up a regression test tracking mechanism. This enables us to keep track of program modifications. If a software requires performance testing, we can conduct it. This test verifies the output's quality by evaluating the system and monitoring the system's response to feedback. As a whole, this procedure verifies that the system functions as expected by the customer. Additionally, it assesses the outcome in terms of user experience.

5.6 Acceptance Testing

The testing of the product is based on meeting the client's requirements. Assuming the product meets the technical requirements by successfully passing the unit, interface, integration, and system tests (functional requirements), we can then test for user-friendliness (non-functional requirements).

Before the test, we will have a team member(s) explain how to use the system with basic instructions. We will then have our client test the system from the patient's side and from the medical professional's side. While our client is testing from the medical professional's side, one of our team members can act as the patient and record their data. We will take note of any questions or confusions the client has for each qualitative test.

When the client is satisfied with the product and has minimal confusion or concerns about using the product, we will consider the acceptance testing criteria as being met.

5.7 Security Testing (if applicable)

The security for this project is not a challenging part since the details that will be shared in the Client - Server connection will be a sequence of audio files.

Since the majority of the project will be implemented using hardware (on breadboard, etc), if the Server-Client connection was secured, the only possible way to access the information will be through establishing a physical connection with the prototype.

For testing, since the Client-Server will be using TCP connection, there will be a detection of packets not being acknowledged by the Server, meaning that there is either a connection error or other user receiving the data.

5.8 Results

The results verify that the individual units, interface, and overall system work as expected and work as specified by the client and other user groups. By having each component work correctly, we reduce the spread of errors, ensure that the data transmission and storage are successful, and optimize audio data accuracy. Furthermore, successful testing ensures that the user interface allows for simultaneous real-time streaming of several patient-doctor connections. Altogether, the results indicate whether or not the functional and non-functional requirements have been met.

6. Implementation

The preliminary implementation plan for the next semester consists of several broad tasks which we list below.

1. Full software architecture implementation
2. Signal processing and audio packet generation code development
3. Hardware and software integration and testing
4. Integrate machine learning pattern recognition
5. Put together final product
6. Product testing and validation

The details can be found in the design section.

7. Professionalism

7.1 Areas of Responsibility

Area of Responsibility: Hardware accountability

Definition: To take accountability for one's decisions and actions that may have led to a failed design in any aspect(environmental impact, technical failure, etc.).

IEEE v. NSPE Canon: The main difference between NSPE and the others is that NSPE is typically more strict when it comes to citing sources. It is a direct violation of the NSPE code of ethics to not give credit, where credit is due. The others do not necessarily see this as an ethics violation.

IEEE's Values: It satisfies all of these responsibilities shown in the table because it upholds them. Specifically, it is honoring the idea of transparency of communication and allows for improved sustainability in doing so. There is a lot of overlap but this is the big idea.

How does this apply to the other 7?
While the table within the paper talks about delivering high quality products with responsible design choices, this entry created above would have the engineers behind any project be responsible for any unforeseen failures as well after the design is completed.

7.2 Project Specific Professional Responsibility Areas

Area of Responsibility	Does it apply to our project's professional context? Why yes or why not?	How well is our team performing (high, medium, low, n/a) in this area?
Work Competence	Yes – In regards to planning since we are still in the planning stage and half in the implementation stage. Planning the hardware and software architecture of our smart digital stethoscope requires significant technical detail. If the planning is done without proper, professional, high quality planning, it will make progress difficult later. Poor planning will likely translate into a poor final product if we don't otherwise re-plan in the future.	Medium – We have hardware planning mostly finished and ideas for software, however they have not been fully flushed out and discussed as a group. We are completing the planning steps in a slow but steady-ish manner, but could definitely be more timely. We did do a good job of splitting group members up into technical areas (Bluetooth, WebRTC server, PCB design, product design) based on their skills, interests, and expertise.
Financial Responsibility	Yes – Since technology with teh same concept as ours already exists, one of our main focuses is to create one that is significantly cost-effective (i.e. – cheap).	High – Although the total cost of the project has not been calculated because some of it depends on actual implementation and manufacturing (e.g. – PCB design), we have chosen very cheap hardware components for our main system's functionality. The expected total cost is expected to be fairly cheap, around $20 in total to make.
Communication Honesty	Yes – Our main stakeholder as of now is our client, and it is extremely important to be transparent with him so he can give us effective and proper guidance based on his desired outcome of the project. There aren't too many concerns for our end users, but of course communication and transparency about how the product will work and how secure the patient data will be is necessary.	High – We meet regularly with our stakeholder/client and are transparent about our progress, or lack of progress, any teamwork or internal communication issues we may be having within our group. We have also discussed privacy and security, but in our stage of the project it isn't an initial concern/priority.

Health, Safety, Well-Being	No – There are few concerns regarding harm to the patient. The main concern is health data privacy, but the product does not have many safety or health concerns since it is simply a digital implementation of a traditional stethoscope which is widely accepted as unharmful and safe.	n/a
Property Ownership	Yes – but in a very vague manner. Our technology will be measuring and handling patient health data which requires us as the product developers to respect specifically the information of clients (patient data).	Medium – We are implementing a "handshake" protocol to authenticate the patient and doctor. Higher levels of security are not being explored in our initial stages.
Sustainability	Yes – Since this product would ideally be distributed and used on a large scale, the product would need to be manufactured sustainably to prevent high-level environmental harm.	Medium – Environmental sustainability has not been explicitly factored into our technical decisions, however we are using cheap, readily available, and widely used hardware parts as opposed to rare natural materials that are expensive and difficult to obtain.
Social Responsibility	Yes – Our product's goal is to benefit society (both patients and medical professionals) by offering remote, and thus, convenient yet secure health monitoring and potential prognosis and diagnosis of diseases/illnesses.	High – There is not much action we need to take to uphold this area of responsibility. The project description and end-product goal is inherently meeting this social responsibility.

Table 3. Project specific professional responsibility areas summary table

7.3 Most Applicable Professional Responsibility Area

The group demonstrated a moderate level of proficiency in the work competence area, particularly in distribution of tasks by splitting into groups for the Bluetooth connectivity, WebRTC server, PCB design, and product design.

The goal of a WebRTC server is to stream audio in real-time via wifi/server. WebRTC already has two versions (user identification/login + multiple channels + sending microphone input audio). Bluetooth connectivity aims to send heartbeat audio from the microcontroller + Bluetooth chip (transmitter) to the computer (which already has a Bluetooth receiver). When it comes to PCB design, we have referenced the past group's circuit in order to deliver a product that already seems to work as anticipated.

By hooking up and testing the breadboard design of the last group, the filters seem to be obtaining the correct frequency cutoffs for the heart and lungs. Significant research has been conducted in search of integrated Bluetooth processors and thorough planning has been done in this area.

8. Closing Material

8.1 Discussion

At this point, the main results of our project are the solidification of the hardware components, successful baseline communication between patient and medical professional via handshake protocol, Bluetooth communication, and the final product design.

They are meeting the requirements at a very basic level but are not quite up to the desired functionalities. However, this is the stage of the project that we expected to be in at this time.

8.2 Conclusion

At this point in our project, we have implemented the core of the WebRTC server at a very low level as it is able to transmit messages in real-time and live-stream microphone and camera audio. Plans for the future are to implement the remaining servers and data storage methodology (i.e. stun server, database, etc.)

The hardware circuit for pressure sound measurement to analog electrical signal and bandpass filter have also been built. Additionally, the PCB design of the analog filter has been completed. The sensor or reading of the analog signal by the MCU's ADC module has been completed as well. Future plans are to integrate the hardware with the MCU and BT module.

In regards to the data transmission, the Bluetooth transmission is near completion and the future goals consist of developing the signal processing mechanism and audio packet generation once it has been received by the patient's computer, ultimately allowing for real-time streaming by WebRTC.

Implementing the machine learning algorithm for patient data pattern recognition will be integrated and tested in the future as well. We will start with the algorithm developed by the previous group and add additional features.

All of the desired components have been finalized and the final product CAD design has been finalized reflecting the components we will be using.

8.3 References

Andrew-Holman, “Andrew-Holman/smart-digital-stethoscope-mirror,” GitHub. [Online]. Available: https://github.com/andrew-holman/smart-digital-stethoscope-mirror. [Accessed: 06-Dec-2021].

False, “Introduction to digital stethoscopes and: Maxim integrated,” Introduction to Digital Stethoscopes and | Maxim Integrated. [Online]. Available: https://www.maximintegrated.com/en/design/technical-documents/tutorials/4/4694.html. [Accessed: 06-Dec-2021].

“Getting started with Raspberry Pi Pico,” Projects.raspberrypi.org. [Online]. Available: https://projects.raspberrypi.org/en/projects/getting-started-with-the-pico/. [Accessed: 06-Dec-2021].

“How do I power a raspberry pi pico? – the Pi Hut.” [Online]. Available: https://support.thepihut.com/hc/en-us/articles/360017209818-How-do-I-power-a-Raspberry-Pi-Pico-. [Accessed: 06-Dec-2021].

L. Pounder, “How to set up and program Raspberry Pi Pico,” Tom's Hardware, 20-Apr-2021. [Online]. Available: https://www.tomshardware.com/how-to/raspberry-pi-pico-setup. [Accessed: 06-Dec-2021].

“MicroPython libraries¶,” MicroPython libraries - MicroPython 1.17 documentation, 06-Dec-2021. [Online]. Available: https://docs.micropython.org/en/latest/library/index.html. [Accessed: 06-Dec-2021].

P. Bhatt, “Connect Arduino with computer using HC05 Bluetooth module,” AVR Geeks, 23-May-2020. [Online]. Available: https://avrgeeks.com/connect-arduino-with-computer-using-hc05/. [Accessed: 06-Dec-2021].

8.4 Appendices

A. Product Design (CAD)

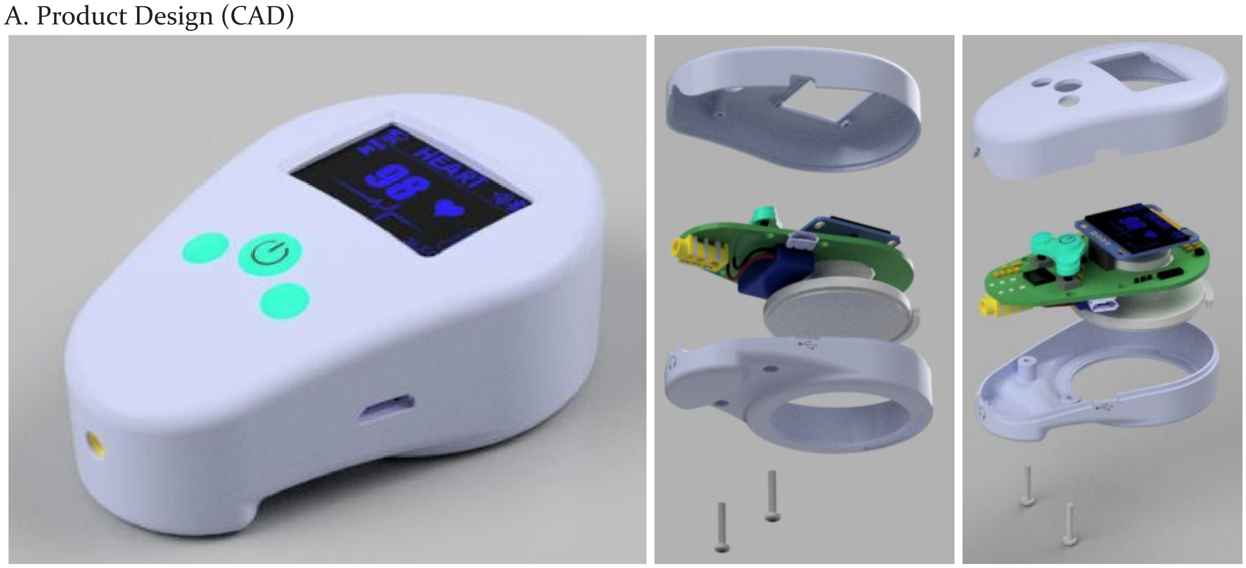

Figure 4. Ideal final product design

Figure 5. Current product design

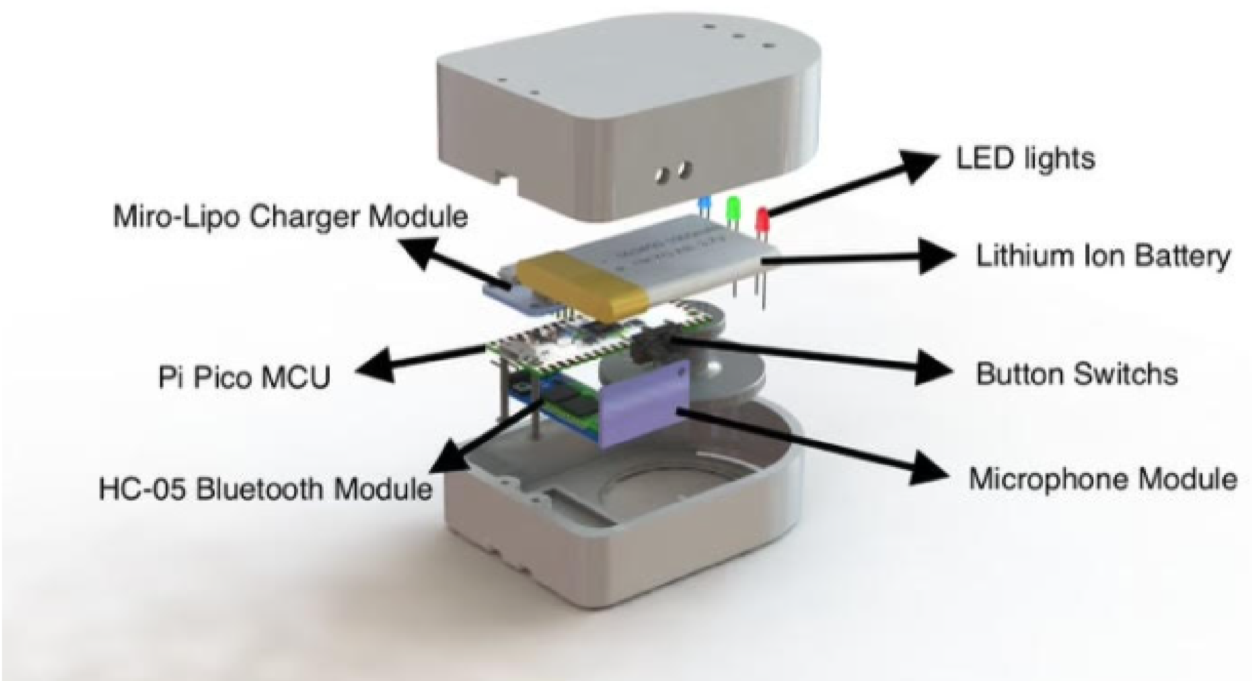

Figure 6. Current product design with component labels

B. MCU Notes & Planning

- frequency ranges we are measuring
  - human heart beat: 20-200Hz
  - human lung contraction: 25-1500Hz
- sampling frequency

  $f_s > 2f_{max}$ where f_{max} is the highest frequency in our analog signal (heart beat or lung contraction)

  for long-term purposes, it is better to use the 1500 Hz as our signal's highest frequency so our work can be extended to lung contraction-related illnesses.

  $\therefore f_s > 2 * 1500Hz \rightarrow f_s > 3kHz$
- raspberry pi pico ADC sampling frequency
  - i.e. – every 2µs, the raspberry pi pico will take a reading and convert it to a 12-bit number (quantization) corresponding to the read voltage
  - the ADC sampling frequency is determined by the raspberry pi pico's system clock speed and design of the ADC itself
    - system clock frequency: $f_{clk} = 48MHz \rightarrow T_{clk} = \frac{1}{f_{clk}} = 20.83ns$
    - ADC takes 96 clock cycles per conversion/sample quantization: $f_{s,ADC} = \frac{f_{clk}}{96} = 96 * T_{clk} = 500kHz \rightarrow T_{s,ADC} = 2\mu s$
    - **this is over 166x the sampling frequency we need which may be overwhelming for the BLE module – but *I think* we can always decrease the ADC output rate using interrupts or timers.**
- raspberry pi pico ADC quantization resolution
  - 12-bit ADC – can be decreased (and possibly increased with use of additional registers)
- raspberry pi pico voltage input
  - ***GPIO 26-28*** *are able to be used either as digital GPIO or as ADC inputs (software selectable)* – from raspberry pi pico datasheet
  - **setback – maximum voltage input for raspberry pi pico GPIO pin is 3.3V (but we can always decrease the input signal gain using a voltage divider.)**

Figure 7. MCU data rate planning and pin information

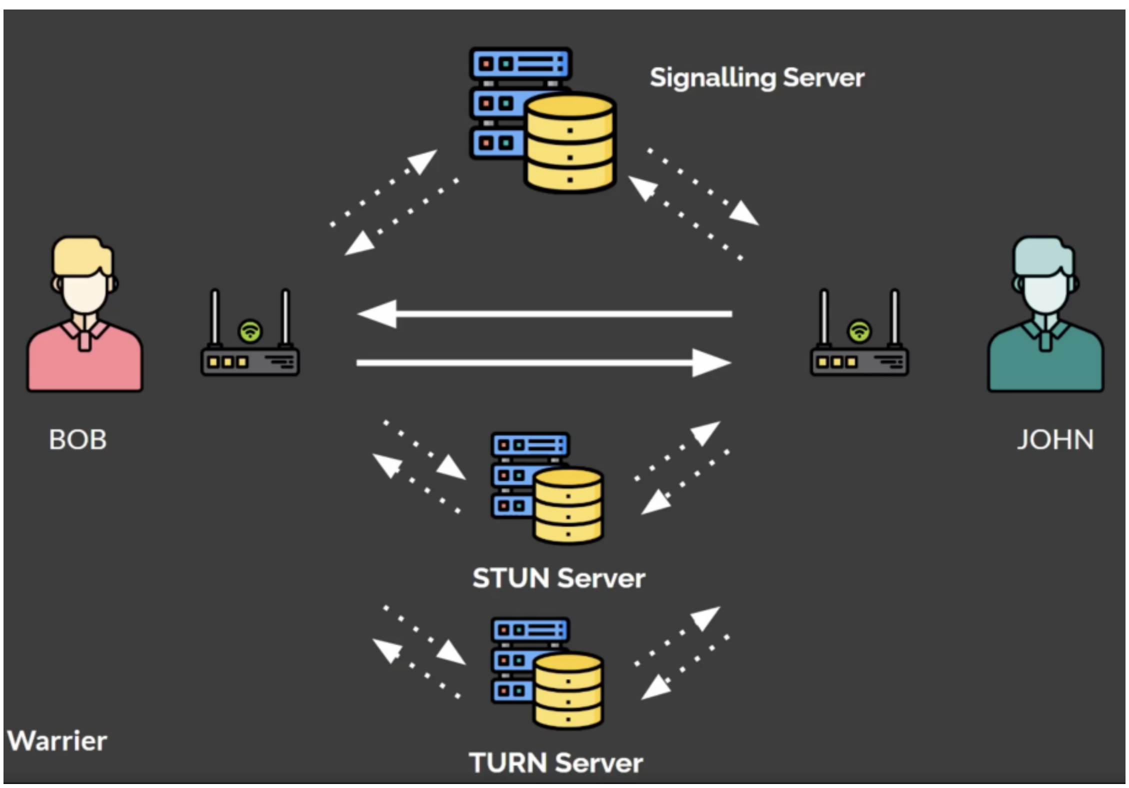

Figure 8. Our software architecture schematic inspiration

C. Software Architecture

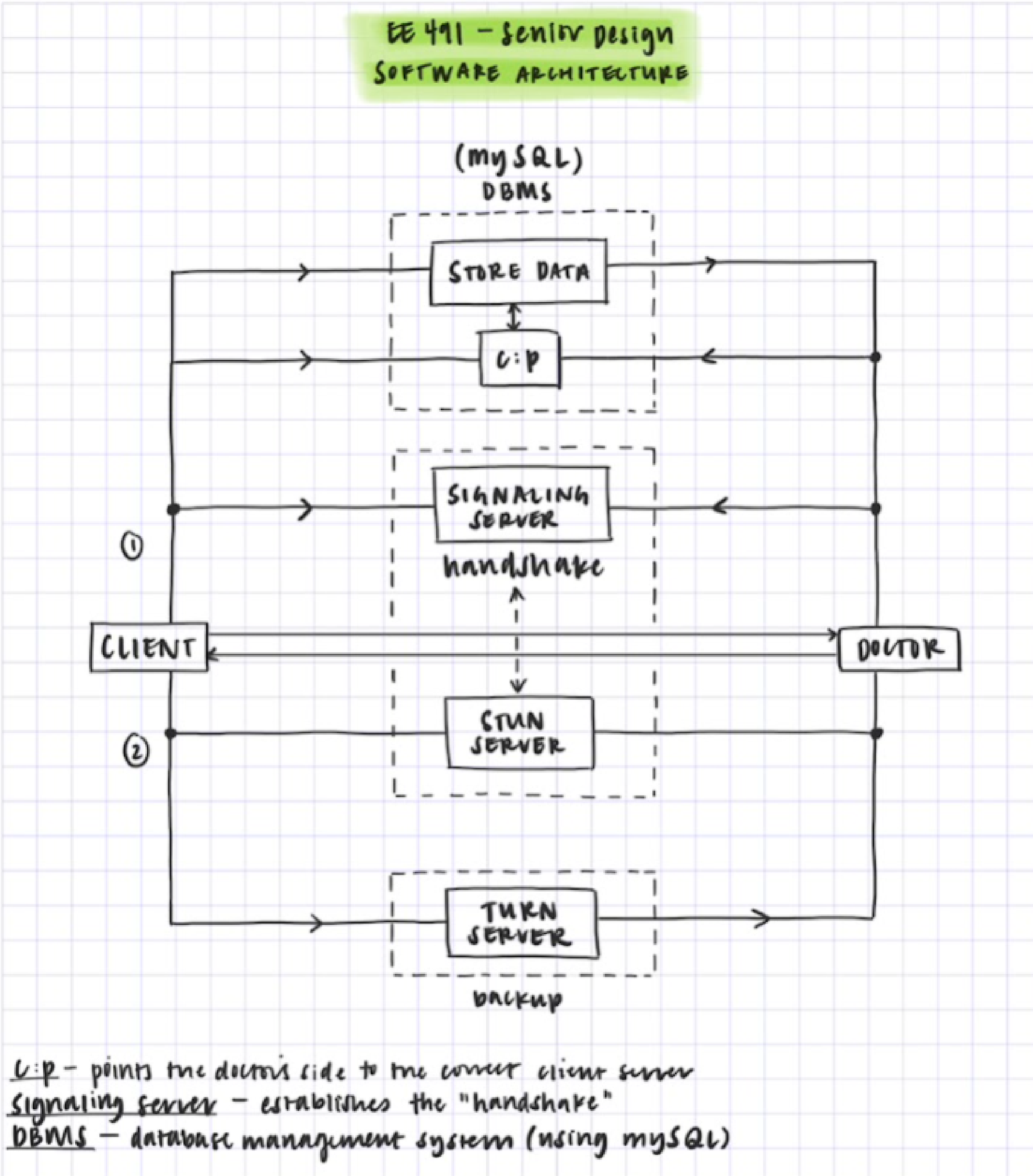

Figure 9. Our software architecture schematic sketch

D. Hardware Components

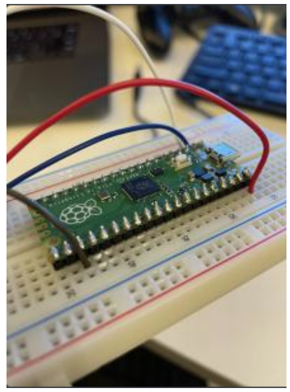

Figure 10. Raspberry Pi Pico connected to HC-05 BT chip on a breadboard for testing

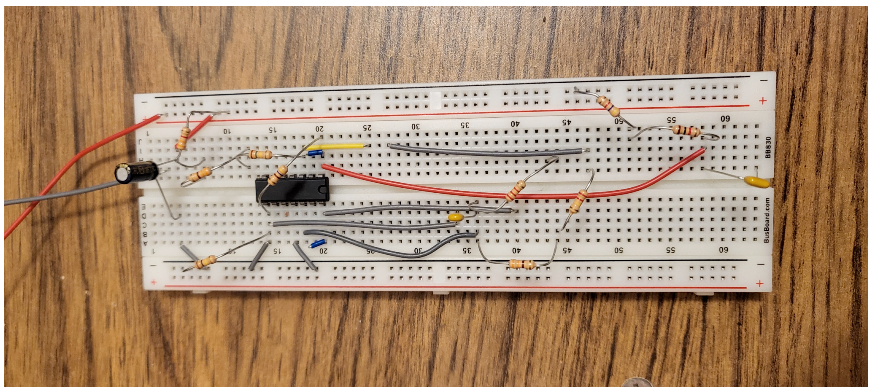

Figure 11. Analog bandpass filter hardware implementation

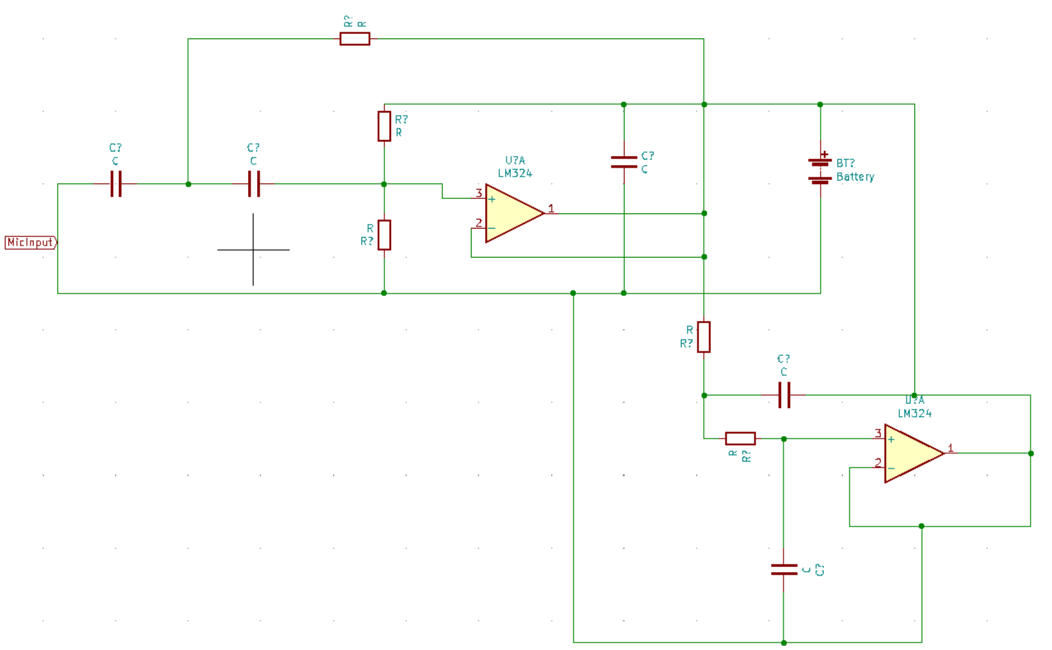

Figure 12. PCB design of the bandpass filter

8.4.1 Team contract

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:
    a. M 4:30 - 5 (F2F)
    b. F 4:00 - 5:00 (F2F)
2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):
    a. GroupMe: Quick updates/reminders.
    b. Discord: Sharing files, keeping track of links, some reminders, online meetings (when needed), screen-sharing.
3. Decision-making policy (e.g., consensus, majority vote):
    a. Majority vote.
4. Procedures for record-keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
    a. Joyce Lai will save meeting minutes within a folder in our Google Drive.
    b. Create a new document for each week (including all 491-related meetings).

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:
    a. Clear communication about absences prior to meeting times.
    b. Be on time.
2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
    a. Complete assignments prior to the due date with time for teammates to review and make revisions when necessary.
3. Expected level of communication with other team members:
    a. Regular monitoring of Groupme chat and Discord.
4. Expected level of commitment to team decisions and tasks:
    a. Completing the assigned tasks on time.
    b. Be as thorough as possible.
    c. Review agreed-upon resources prior to the next meeting.

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
    a. Team Organization (Vignati, Joyce)
    b. Client Interaction (Everyone)
        i. Email/Scheduling (Vignati)
        ii. F2F Update (Austin)
    c. Testing (Everyone)
        i. Autocorrelation (Joyce)
        ii. Hardware (Yilun, Matthew, Joyce, Abdalla)
        iii. Software (Austin, Omar)
    d. Individual component design:
        i. Hardware
            1. Compact Circuit Design (Matthew)
            2. 3D Printing/CAD (Yilun)
            3. Circuit Improvement (Joyce, Yilun, Matthew)
        ii. Digital Conversion Improvement (Vignati, Abdalla)
        iii. Bluetooth Connectivity
            1. Bluetooth RX/TX Design (Joyce, Yilun, Abdalla)
        iv. Create WebRTC server

            1. Setting Up Server (Omar, Austin)
        v. Signal Filtering (Digital)
            1. Same BW Noise Cancelling (Austin, Joyce)
        vi. Classification of Heartbeats
            1. Machine Learning Implementation (Austin, Omar, Joyce)
        vii. Design Validation
            1. Autocorrelation (Joyce)
            2. Doctor connections (Matthew, Austin, Vignati)
2. Strategies for supporting and guiding the work of all team members:
    a. Messaging people and finding them in person.
    b. Keeping meetings in person with near to full attendance.
3. Strategies for recognizing the contributions of all team members:
    a. Updating team on work that has been done (informal update).

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
    a. Joyce
        i. Expertise: Communications, signal processing, math, MATLAB, Simulink, C-programming
        ii. Soft Skills: Organizing, prioritizing, delegating tasks, communication
        iii. Perspective: Methodical; have worked on various biomedical technology in research settings; biomedical engineering minor
    b. Austin
        i. Skills: Algorithms and AI
        ii. Expertise: Software
        iii. Perspective: EMT experience
    c. Abdalla
        i. Skills: Verilog, C-programming, Java-programming.
        ii. Expertise: Embedded systems
    iii. Perspective: Logical
    d. Vignati
        i. Skills: C-programming, Java-programming, Python, Verilog, Android studio
        ii. Expertise: Embedded systems
        iii. Perspective: n/a
    e. Matthew
        i. Skills: MATLAB/SIMULINK, C-programming, CAD, Power-systems
        ii. Expertise: Electrical engineering with a focus on hardware, biomedical engineering classes
        iii. Perspective: Try not to fail =]
    f. Yilun:
        i. Skills: CAD, communications
        ii. Expertise: Electrical engineering
        iii. Perspective: n/a
    g. Omar:
        i. Skills: Java-programing, C-programing, Python, Android
        ii. Expertise: Software
        iii. Perspective: Logical
2. Strategies for encouraging and support contributions and ideas from all team members:
    a. People who are comfortable speaking should ask questions to other members.
    b. People who are comfortable speaking should encourage others to give their ideas.
    c. Be open to and reflect upon the solutions proposed by the group members to encourage sharing our thoughts and openness.

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)
    a. Take a vote.
    b. Bring up concerns at team meetings.

Goal-Setting, Planning, and Execution

1. Team goals for this semester:
    a. Design the smart stethoscope system.
        i. Enable Bluetooth connectivity.
    b. Start building a prototype with the required functionalities.
    c. Meet all the deliverables assigned to us in a timely manner while maintaining high-quality work.
    d. Study the prototype delivered in stethoscope phase I.

2. Strategies for planning and assigning individual and teamwork:
    a. Set up a timeline (e.g. Gantt chart) at the beginning of the project with main benchmarks.
    b. Discuss all tasks and divide.
    c. Utilize a Discord channel to keep track of group and individual tasks.
    d. Utilize a Discord channel for meeting minutes and weekly progress updates.
3. Strategies for keeping on task:
    a. Establish goals for the coming week in a group meeting (after the client meeting).
    b. Be prepared to give good progress updates to our client/professor.

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?
    a. Step 1. Speak to the person(s) involved about it.
    b. Step 2. Speak to the group about it.
    c. Step 3. Speak to TA about it.
    d. Step 4. Speak to the course professor about it.
2. What will your team do if the infractions continue?
    a. Allow the professor or TA to deal with it how they see fit.

****

a) I participated in formulating the standards, roles, and procedures as stated in this contract.
b) I understand that I am obligated to abide by these terms and conditions.
c) I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.

NAME	DATE
Abdalla Alzaabi	9-17-2021
Omar Alsaedi	9-17-2021
Matthew Gasparaitis	9-17-2021
Austin Collins	9-17-2021
Yilun Huang	9-17-2021
Vignati Yalamanchili	9-17-2021
Joyce Lai	9-17-2021