Smart Digital Stethoscope

Design Document

Team Number: sdmay21-03

Client: Dr. Ashfaq Khokhar
Advisers: Dr. Ashfaq Khokhar
Meeting Scribe: Joseph Dobosenski
Meeting Facilitator: Megan Kasabian
Chief Hardware Engineer: Erik Becker
Chief Software Engineer: Andrew Holman
Test Engineer: Jordan Spidle
Report Manager: Maggie Heaslip

Team Email: sdmay21-03@iastate.edu
Team Website: http://sdmay21-03.sd.ece.iastate.edu

Revised: 10/4/2020 Ver. 1

Executive Summary

Development Standards & Practices Used

- We will follow an Agile methodology of development.
    - Two week sprints
    - Weekly standups
    - Backlog of tasks
- We will use Branch-Review-Merge workflow.
- We will implement Test Driven Development.
- We will keep robust documentation.
- We will manage our tasks via the GitLab issue board.
- Agile methodology of development.
- Test Driven Development.
- We will keep robust documentation.
- Manage our tasks via the GitLab issue board.
- Bluetooth
- Organized and thorough design schematics
- Utilize cheap, yet effective components that fulfill requirements

Summary of Requirements

- Digitize an analog signal
- Send data via bluetooth
- < 3 second delay from finishing the recording to having the file prepared
- Convert the data to a .wav file
- Desktop application can retrieve and store data
- Utilizes machine learning to identify certain anomalies
- The user interface must be intuitive to ensure ease of use by all users
- Cost of software licenses < $50
- Monthly expenses on data storage <$1/active user
- Collect pressure waves from the heart and lungs
- Convert pressure waves to usable electrical signal
- Filter out noise & amplify the result
- Run on battery power
- Minimum size
- Cost of hardware < $200

Applicable Courses from Iowa State University Curriculum

- Electrical Engineering 201
- Electrical Engineering 230
- Electrical Engineering 324
- Electrical Engineering 224
- Computer Engineering 288
- Software Engineering 319
- Computer Science 309
- Computer Science 474

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

1 Introduction

1.1 Acknowledgement

Firstly, thank you to Dr. Ashfaq Khokhar for providing technical assistance and support throughout the design process. Additionally, this project is, in part, made possible by Iowa State ECpE department funding.

1.2 Problem and Project Statement

Access to a doctor is often limited for billions of people around the world and this is only worsened during a global pandemic. These people would find it difficult to have a doctor personally diagnose their issue which may be easily treated if caught early on. Also, many times those who can see doctors have to talk to them virtually. However, these people may not have the means to easily send the high quality audio of their chest sounds that the doctor might need to diagnose the issue. This has created a need to introduce as much automation into the healthcare industry as possible.

The goal of this project is to design and manufacture a smart stethoscope that converts heart and lung sounds into a digital format while removing any perceived noises and amplifying signals of a certain range. Once in this digital format it will use machine learning to identify known irregularities in the sounds. The user will interface with the stethoscope via a web application on a device that is connected to the stethoscope by bluetooth. In this user interface users will also be able to easily send their audio recordings to doctors for further analysis.

1.3 Operational Environment

When designing a product, it is important to consider what environment a product will be used in throughout its lifetime. Most stethoscopes are typically used within the confines of a doctor's office. While this is accurate for this project, it could potentially be used in user's households as well. The goal is to make a stethoscope that does not need to be accompanied by a doctor. The smart stethoscope will be used indoors, in generally clean environments, similar to a typical stethoscope. However, the smart stethoscope will need to travel outside the doctor's office and into the homes of users, so handling magnitude will be increased, as well as the number of users. This must be considered when designing the stethoscope; it must be a durable product that can withstand heavy usage.

1.4 Requirements

For this project there are two categories of requirements, software and hardware.

For software requirements:

- Stethoscope can digitize an analog signal.
- It will be able to send data via bluetooth to a device of the user's choosing.
- There will be less than a 3 second delay from finishing the recording to having the file prepared.
- It must convert the data to a .wav file to store for later use.
- The desktop application can retrieve and store data using the .wav file.
- Project utilizes machine learning to identify certain anomalies within recorded sound.
- The user interface must be intuitive to ensure ease of use by all users.
- Cost of software licenses must be under $50 and monthly expenses on data storage should be no more than $1 per active user on average.

For hardware requirements:

- The stethoscope must be able to collect pressure waves from the heart and lungs
- Convert pressure waves to a usable electrical signal.
- It must filter out noise within the signal and then amplify the result.
- Stethoscope will run on battery power for up to 12 hours, replacing the battery when needed.
- Minimum possible size is needed.
- Cost of hardware not to exceed $200.

1.5 Intended Users and Uses

For this project, there will be two users, the patient and the doctor. The patient will use this device to allow a doctor to listen to their heartbeat without needing to physically be in the office. Since the patient will be the one who has to do most of the setup, both the hardware and software need to be as simple as possible to be useful to the largest number of people. On the other hand, the doctor will likely primarily be using the web interface to interact with the patient and give them feedback on what they're hearing. However, the doctor will have multiple patients so they will need to be able to manage all their patients independently. Therefore, the interface for the doctor will be more intricate due to this increased volume of data to manage.

1.6 Assumptions and Limitations

Assumptions

- The user will have a device that can connect to bluetooth.
- The end product will be user friendly and easy to navigate.
- The product will resemble an average stethoscope.
- The product will be available for the common household.
- Batteries will be easy to interchange/ charge.

Limitations

- The end product will need to be roughly the size of a regular stethoscope.
- The end product will need to cost under ~$200.
- The user will need a computer to utilize the full functionality of the product.
- The product will need to be battery powered.

1.7 Expected End Product and Deliverables

Once our project is completed there will be a set of deliverables from the software and hardware side we intend to show to our client. For the software side one goal is the stethoscope is able to digitize an analog signal and then send that signal via bluetooth to a desktop application. In addition, the conversion will be a .wav file that can be used by the user on the desktop. Also, machine learning will be used to alert the user if the recording sounds problematic within a three second delay from the recording ending. Finally, the interface we create will be intuitive for the user and the software costs will remain under $50. The web app framework for our product will be completed by 9/13, the bluetooth transmitter will be finished by 10/2 and our machine learning algorithm will be refined by 1/24.

To continue, for the hardware side of the project our end project should collect pressure waves from the heart and lungs. These waves will then be converted into an electrical signal and a filter will eliminate unwanted frequencies. Furthermore, an amplifier will be used to strengthen the signal and our product will run on battery power. The battery consumption will be minimized and the overall size of the product will be small for easier usability. The standard length of a stethoscope is 27 inches so we will remain consistent with this size. Finally, we will also work to keep the cost of hardware aspects to no more than a few hundred dollars. We will work to have the pickup sensor completed by 10/2, the amplifier with the filter by 10/30, the buffer with the amplifier by 11/20 and overall user audio streaming by 3/21.

2 Project Plan

2.1 Task Decomposition

The various tasks that need to be completed can be seen in Figure 1 below. The pickup sensor will pick up the pulse signals of the user. Those sounds will then be sent through a preamplifier so it can be properly filtered. To ensure a clear signal, a low-pass filter will be added to remove any background noise. A buffer will ensure that the filter output does not affect our final amplifier. A final amplifier will increase the signal amplitude to match the ADC. The analog to digital converter (ADC) will take the analog signal and convert it into a digital signal that can be taken by the bluetooth transmitter and send it to the user device.

Once linked to the user's device the stethoscope can be interfaced with the web application. The web app will allow the user to record the audio from the stethoscope and play it back. Then the user will be able to use a machine learning algorithm to determine if the recording indicates a cardiovascular or respiratory illness. Also the user will be able to stream their audio to another user on the web application through a shared session connection. Finally if the user is logged into the website they will be able to save their recordings and listen back to previously saved recordings.

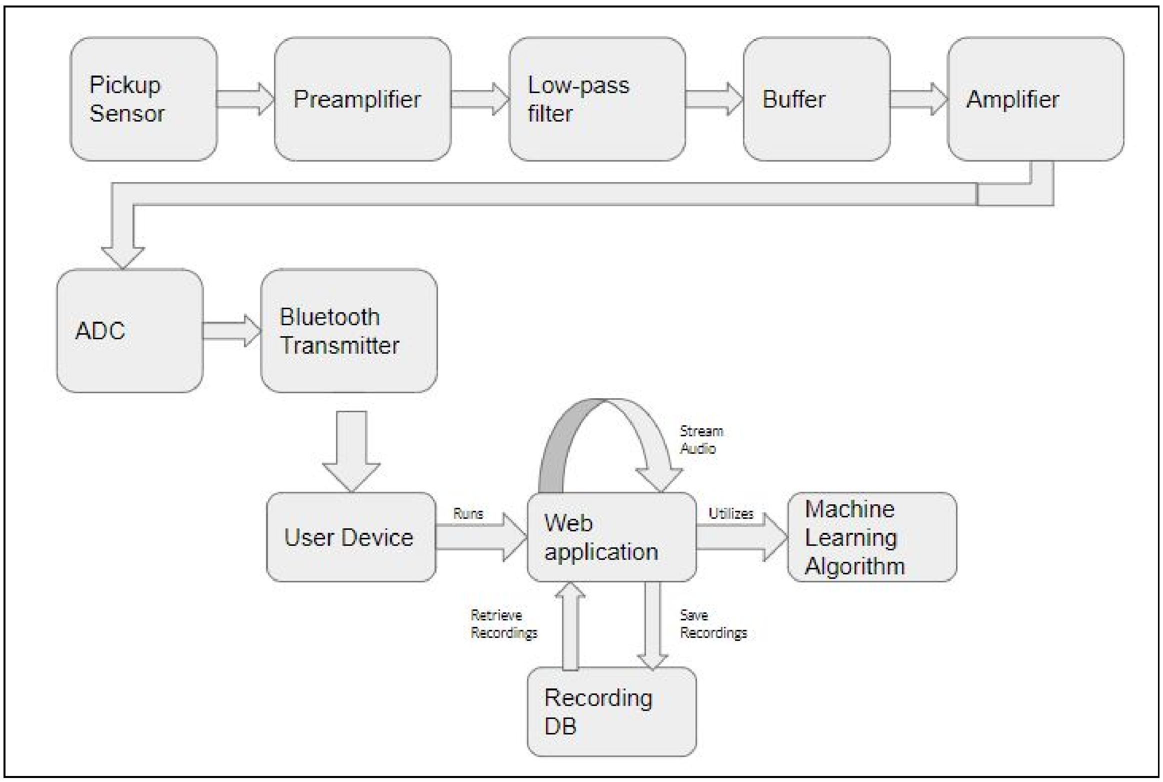

Figure 1: System Block Diagram

2.2 Risks And Risk Management/Mitigation

Our first task is creating a pickup sensor and the main risk is that the sensor is unable to detect the frequencies range of the patient's chest sounds. The probability of this occurring is 0.2, so not very high. Next, we will work on our preamplifier and the risk with this component is that it will not amplify the frequencies enough. The probability of this risk occurring is 0.3, which makes the component not very high risk. Then we will incorporate our low pass filter that has a risk of 0.5 because we could have calculated incorrect values for our filter to work with. Our risk mitigation plan would be to have multiple hardware members perform the calculations independently. After the filter, a buffer and amplifier will be used. Both of these components have a risk of 0.4 because it's possible they could not enhance the signal properly. Next, the analog to digital converter will be incorporated and has a risk of 0.2 because we will be purchasing it and can find a replacement if needed.

The next task is the bluetooth transmitter to send the digitized signals to the user device. The risk associated with this is that the transmitter won't be able to connect to the user device. This risk is relatively low at 0.05 since both the transmitter and user device will be products that we purchase off the shelf. After that is the web application. The risk here is that it won't run on a wide range of devices, this risk is rated at 0.6. With the wide range of devices and platforms that users can use it is very likely that for some it won't work. To mitigate this risk we are trying to narrow our focus to desktop applications. Also we are creating the app in Angular, this framework is very easy to style for mobile devices which should allow us to make it mobile friendly as a potential feature. Lastly is the machine learning algorithm, the risk is that the data set we are using to train the algorithm won't result in a trained model that can work off our own stethoscope. This is very possible at 0.5 so to mitigate it we have a few options. Firstly, we can try to find another data set that matches our own stethoscope data better. We could also adjust the stethoscope itself to match the data set. Lastly, we could do preprocessing on the data after it is transmitted to the program in order to make it match.

2.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

The smart stethoscope can be split into two major projects, the hardware of building the stethoscope, and the software of analyzing and presenting the signal from the stethoscope. These two projects each have their own milestones and metrics. The progress of significant tasks and milestones is recorded through interpersonal communication and within the team Gitlab task board.

Implementing the hardware of the stethoscope requires meeting specific criteria for each component of its circuit. Once a pulse signal is received from a patient, the hardware of the stethoscope will meet the following vital milestones and criteria:

- The wireless stethoscope should be able to operate on battery power for 12 hours.
- A preamplifier will increase the raw pulse signal with a gain of 2.
- A lowpass filter will filter out any frequencies not associated with human pulse signals; it will filter out any frequencies above 20Hz.
- A final amplifier will extend the filtered signal amplitude by a factor of 4.

- The final amplified signal should be digitized and transmitted via bluetooth for signal analysis.

With a filtered and amplified signal being transmitted to a web application of choice, the following software milestones and criteria will be met:

- The signal will be recorded and converted into a .wav file within 3 seconds of the recording finishing.
- The web application will save recordings to a user's account to allow playback.
- A machine learning algorithm will detect anomalies in the pulse signal and express the anomalies with 70% accuracy.
- The web application will allow for sharing of live audio with another user with less than a 2 second delay.

2.4 Project Timeline/Schedule

The Gnatt chart can be seen in figures 2 and 3 below. We have split the development between two teams, software and hardware. This division will allow for concurrent development between the two deliverables which are the physical hardware to capture the heartbeat and the software interface to allow transmission/sharing.

Development for the hardware will follow a linear direction since the various components will build off of each other. For example, the preamplifier will need the signal generated from the pickup sensor in order to function. Therefore, the timelines on the Gantt chart for the hardware components will generally not overlap.

Development for software can be somewhat more flexible. For example, the transmission of information via Bluetooth will not rely on the machine learning or web application being done. While they all still interact, we can work on these things separately and likely will need to. As a result, we can see on the chart that there is some overlapping.

Task	Start Date	End Date	Expected hours to complete
Build Initial Web App Framework	9/7	9/13	10 hrs
Design Document v1	9/7	10/5	15 hrs
Build Initial Machine Learning Algorithm	9/7	10/19	10 hrs
Refine Web App Framework	9/14	11/1	30 hrs
Refine Machine Learning Algorithm	10/19	11/30	30 hrs
Pickup Sensor	9/21	11/23	15 hrs
ADC	10/5	11/2	20 hrs
Bluetooth transmitter	10/5	11/2	20 hrs
Design Document v2	9/28	10/19	20 hrs
Preamplifier	9/28	11/9	20 hrs
Low-pass Filter	10/26	10/30	20 hrs
Saving and Retrieval functionality and	10/19	11/11	35 hrs
Design Document v3	10/26	12/7	25 hrs
Buffer	11/9	2/8	20 hrs
Amplifier	11/30	2/15	20 hrs
Condense circuitry into user friendly package	2/15	3/15	25 hrs
Streaming user audio	2/15	3/21	30 hrs
Intergrate Hardware and Software	1/25	2/22	30 hrs

Semester 1: 9/7, 9/14, 9/21, 9/28, 10/5, 10/12, 10/19, 10/26, 11/2, 11/9, 11/16, 11/23, 11/30

Figure 2: Semester 1 Gantt Chart

Task	Start Date	End Date	Expected hours to complete	Semester 2								
				Break	1/25	2/1	2/8	2/15	2/22	3/1	3/8	3/15
Build Initial Web App Framework	9/7	9/13	10 hrs									
Design Document v1	9/7	10/5	15 hrs									
Build Initial Machine Learning Algorithm	9/7	10/19	10 hrs									
Refine Web App Framework	9/14	11/1	30 hrs									
Refine Machine Learning Algorithm	10/19	11/30	30 hrs									
Pickup Sensor	9/21	11/23	15 hrs									
ADC	10/5	11/2	20 hrs									
Bluetooth transmitter	10/5	11/2	20 hrs									
Design Document v2	9/28	10/19	20 hrs									
Preamplifier	9/28	11/9	20 hrs									
Low-pass Filter	10/26	10/30	20 hrs									
Saving and Retrieval functionality and	10/19	11/11	35 hrs									
Design Document v3	10/26	12/7	25 hrs									
Buffer	11/9	2/8	20 hrs									
Amplifier	11/30	2/15	20 hrs									
Condense circuitry into user friendly package	2/15	3/15	25 hrs									
Streaming user audio	2/15	3/21	30 hrs									
Intergrate Hardware and Software	1/25	2/22	30 hrs									

Figure 3: Semester 2 Gantt Chart

2.5 Project Tracking Procedures

Our group has chosen to use GitLab to keep track of upcoming tasks, current tasks, and completed tasks. GitLab provides us a platform where we can assign different group members to specific tasks and is a great way to keep our team organized. Git will be used as a source control for the software development. All software developers will be able to branch off the master repository to implement their respective features without interfering with one another. In addition, we use Slack for fast communication between members because we can create different chat groups within our team that can still be visible to everyone. Finally, our team has taken advantage of Zoom and Webex to conduct weekly team meetings and weekly client meetings.

2.6 Personnel Effort Requirements

Include a detailed estimate in the form of a table accompanied by a textual reference and explanation. This estimate shall be done on a task-by-task basis and should be the projected effort in total number of person-hours required to perform the task.

Task	Person-Hours Required
Pressure sensor detects pulse signals of the user.	15 Hours
Create a preamplifier and filter to remove noise from the signal.	20 Hours + 20 Hours
Create a buffer and amplifier to ensure signal integrity and increase signal amplitudes.	20 Hours + 20 Hours
Convert the analog electric signal to a digital signal.	20 Hours
Integrate bluetooth capabilities between the stethoscope and user device.	20 Hours
Create a database to store and retrieve signals within the application.	10 Hours
Define a machine learning algorithm to identify cardiovascular and respiratory illnesses.	10 Hours + 30 Hours
Securely share confidential data to health professionals through a network connection.	30 Hours
User's can replay previously recorded signals and save other recordings.	35 Hours

2.7 Other Resource Requirements

Additional resources we require include:

- Circuit components
- Circuit board
- A small microphone
- Batteries
- Casing for the batteries
- Amplifier
- Firebase for hosting and storage

2.8 Financial Requirements

The individual components are listed below and expected to cost a total of $135:

- Software licenses will be under $50 and the cost of hosting will be less than $1 per active user on average.
- Analog to Digital Converter will cost around $40.
- A bluetooth transmitter will be around $30.
- Circuit components vary between $0.2 and $5
- Microphone $10
- Battery Casing $3

There will also be a cost associated with development. For this we are expecting no more than an additional $200. This will cover purchasing test components, software licenses, and hosting fees that are not relevant to the final product. We will seek funds for this project from Iowa State but can purchase what we need to if we can't get the funding.

3 Design

3.1 Previous Work And Literature

Include relevant background/literature review for the project

– If similar products exist in the market, describe what has already been done

– If you are following previous work, cite that and discuss the **advantages/shortcomings**

– Note that while you are not expected to "compete" with other existing products / research groups, you should be able to differentiate your project from what is available

Detail any similar products or research done on this topic previously. Please cite your sources and include them in your references. All figures must be captioned and referenced in your text.

3.2 Design Thinking

Detail any design thinking driven design "define" aspects that shape your design. Enumerate some of the other design choices that came up in your design thinking "ideate" phase.

3.3 Proposed Design

Include any/all possible methods of approach to solving the problem:

- Discuss what you have done so far – what have you tried/implemented/tested?
- Some discussion of how this design satisfies the **functional and non-functional requirements** of the project.
- If any **standards** are relevant to your project (e.g. IEEE standards, NIST standards) discuss the applicability of those standards here
- This design description should be in **sufficient detail** that another team of engineers can look through it and implement it.

3.4 Technology Considerations

Highlight the strengths, weakness, and trade-offs made in technology available.

Discuss possible solutions and design alternatives

3.5 Design Analysis

- Did your proposed design from 3.3 work? Why or why not?
- What are your observations, thoughts, and ideas to modify or iterate over the design?

3.6 Development Process

Discuss what development process you are following with a rationale for it – Waterfall, TDD, Agile. Note that this is not necessarily only for software projects. Development processes are applicable for all design projects.

3.7 Design Plan

Describe a design plan with respect to use-cases within the context of requirements, modules in your design (dependency/concurrency of modules through a module diagram, interfaces, architectural overview), module constraints tied to requirements.

4 Testing

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, or software.

1. Define the needed types of tests (unit testing for modules, integrity testing for interfaces, user-study or acceptance testing for functional and non-functional requirements).
2. Define/identify the individual items/units and interfaces to be tested.
3. Define, design, and develop the actual test cases.
4. Determine the anticipated test results for each test case
5. Perform the actual tests.
6. Evaluate the actual test results.
7. Make the necessary changes to the product being tested
8. Perform any necessary retesting
9. Document the entire testing process and its results

Include Functional and Non-Functional Testing, Modeling and Simulations, challenges you have determined.

4.1 Unit Testing

– Discuss any hardware/software units being tested in isolation

4.2 Interface Testing

– Discuss how the composition of two or more units (interfaces) are to be tested. Enumerate all the relevant interfaces in your design.

4.3 Acceptance Testing

How will you demonstrate that the design requirements, both functional and non-functional are being met? How would you involve your client in the acceptance testing?

4.4 Results

– List and explain any and all results obtained so far during the testing phase

- Include failures and successes
- Explain what you learned and how you are planning to change the design iteratively as you progress with your project
- If you are including figures, please include captions and cite it in the text

5 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3.

6 Closing Material

6.1 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

6.2 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

6.3 Appendices

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.