CpE 191 & EEE 193B

ASSIGNMENT #8: END OF PROJECT REPORT

May 3, 2021

R.F.C.T.S.

Team #9 – "*Team 9 Lives*"

Christopher Moncrief | Mitchell Allen | Yuruvith Saavedra | Andrew Sharp

Professor Neal Levine

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

Elevator Pitch: *A system of wearable radio frequency devices to help monitor and mitigate a pandemic through contact tracing.*

For the 2020 academic year (August 2020 to May 2021), four California State University Sacramento computer and electrical engineering students were tasked with developing a societal problem and solution based around a central concept; *An event that has an impact on twenty-million people over the course of at least six months.* The societal problem that we chose to focus on was a lack of preventative infrastructure when handling epidemics and pandemics outside of the annual strains of influenza. Through our research we discovered that most preventive measures provided by government health ordinances revolved around the most common diseases, leaving populations vulnerable susceptible to new strains of infectious diseases. To combat this, we found inspiration in the contact tracing applications provided by notable technology giants Apple and Google during the early stages of the COVID-19 pandemic. Our solution was to develop a system of radio frequency wearable devices to help mitigate the pandemic through contact tracing. This novel idea would allow medical personnel to operate this device in a minimal setting, allowing it to be used in several different medical situations and locations. We call it *R.F.C.T.S.* or the *Radio Frequency Contact Tracing System*.

While we did not need to make our project about the current COVID-19 pandemic, we wanted to help engineer a possible solution to help relieve the stress from contact tracers by offloading and automating some of their responsibilities. By bringing research and insight into the subject, we hope to persuade the public into seeing why this is an important topic that is often overlooked by public health organizations.

Each section of this report will give more insight into our project, with specific topics being broken down by section. With the exclusion of the introduction and conclusion, there will be sections involving our societal problem, the design of our prototype, the funding and marketability of our project, a breakdown of the work put into the project, a risk assessment of our prototype, our design philosophy, the testing that we put our prototype through, and the final prototype design. There will also be a collection of appendices at the end of the report that will be used for supplementary information and any additional material that could not fit into the main body of the report.

Using our societal problem as a starting point, we discuss and explain throughout the report the different processes that we underwent throughout the two-semester Senior Design course. Each section contains several subsections that help develop a stronger narrative and understanding of our design, why we went down certain paths, and hopefully create a better understanding for the general public as to why this is such an important issue to tackle.

Throughout this report, we (Moncrief, Allen, Saavedra, and Sharp) will be expanding on the different topics related throughout the project's implementation from its design idea all the way through the final iteration of the project prototype.

Abstract

As new viruses and diseases arise constantly, the lack of preventative infrastructure in relation to a spreading epidemic or pandemic is still common despite the modern technology available to society. Our goal is to create a design that will focus on contact tracing as a form of mitigation using a system of wearable radio frequency identification modules. This document outlines the technical work that was completed all throughout the Fall 2020 and Spring 2021 semesters. The report first introduces the societal problem along with a detailed design idea that proposes the solution. Additionally, information such as the cost and funding, work breakdown structure, research on the relevant market for our product, project milestones, risks involved all throughout the duration of the creation of the project, and testing results are all highlighted in this paper. Documentation since the very first day this idea was discussed, until the final day this product was presented is summarized and recorded in this report.

Keyword Index -

Contact tracing, Database, Pandemic, RFID, Wearable

I. INTRODUCTION

A. Societal Problem

In the Summer months of 2020, Team 9 was formed between two computer engineering students (Allen, Saavedra) and two electrical engineering students (Moncrief, Sharp), assembling a perfectly balanced team to take on the challenge of Senior Design. The problem placed before us (Allen, Moncrief, Saavedra, Sharp) was to research a societal problem that falls within the provided scenario; a crisis that will have an effect on a minimum of twenty-million people over a period of at least six months. Most of the Summer break was spent coming up with a societal problem and solution that was approved by Professor Neal Levine, Team 9's laboratory instructor and project mentor. While all members of the team contributed excellent project ideas, Team 9 decided to take on the societal problem of a lack of preventative infrastructure regarding epidemic and pandemic preparedness (Section II) and chose a modified version of contact tracing as our design idea solution (Section III).

In Section II of this report, we explain how there is a significant lack of preparedness when it comes to any epidemic or pandemic that is not related to influenza (seasonal flu). While this is mainly due to the yearly strains of the flu, there have been cases where epidemics have occurred due to a lack of preparation. The definition, transmission, and explanation of why major epidemiology organizations focus on influenza is discussed. Multiple forms of mitigation are also discussed before finalizing the report with an overview of contact tracing and why we chose a lack of preventative infrastructure as our societal problem.

B. Design Idea

Section III continues the discussion of the societal problem with a high-level concept of how our design idea of radio frequency-based contact tracing will work and be implemented. We included hardware and software concepts including interactive process that occur between the hardware. Software will be used to maintain medical databases and perform mass-text messaging protocols for a warning system that can be activated during any time of day. During this section of the report, we also discuss possible design ideas that would not work for our allotted time, skill set, and budget. We also introduce the feature set of the prototype for both the Fall and Spring semesters as well as the metrics that we will be using to demonstrate the completeness of our design to our professor and the general public with our end of project documentation.

Throughout this report, the prototype will begin to unfold as we introduce and present all the necessary steps that were taken to complete our Senior Design project.

C. Device Test Plan

As part of the design process, our laboratory prototype must endure different tests to allow it to develop into what will become our deployable prototype by the end of our Senior Design experience. These tests will access both the hardware and software aspects of our design. Team 9 will be developing and deploying a device test plan to ensure that we are able to advance with the project. The Device Test Plan will take place over several weeks and be document in Section IX and Appendix I of this report.

D. Funding

This brief portion of the report is in Section IV, where the project expense report table is located. The budget for the project was $1000.00 USD and the cost was divided evenly among the four team members.

E. Project Timeline

Much like the Work Breakdown Structure, this portion of the report is dedicated to the length of time each task took to complete. It will be broken down between both semesters and will include the time we spent outside of the Fall and Spring semester. This section of the report will be placed in Section V (Project Milestones) and will include descriptions of the notable progress that we make as we reach them during both the Fall and Spring semesters.

F. Work Breakdown Structure

The Work Breakdown Structure (WBS) will display Team 9's tentative schedule of what tasks should be completed, and by when they should be finished. The WBS resides in Section VI of this report and contains a long form breakdown of each project feature. Each member in the team was assigned a task and was expected to finish by the set deadline, to prevent the progress of the project from being delayed. A table containing a summary of the amount of work hours went into this project as well as the WBS in table format will be provided in Appendix G at the end of this report.

G. Risk Assessment

In Section VII of this report, we will discuss and evaluate the different risks that are associated with our project. These risks can come from system failures such as a malfunctioning piece or hardware or software that was implemented incorrectly. From outside environmental sources such as a global pandemic or a wildfire that shuts down access to school. Another form of outside risk involves an event that could effectively remove a team member from the project for a section of time or indefinitely such as illness, traumatic family emergency, or a similar life event.

H. Design Philosophy

In this section of the report, we discuss why we chose our societal problem, solution, and the design ideas that helped us make these decisions.

I. Testing Results Report

The testing process for our prototype lasted roughly two months from early February 2021 and coming to an end in April 2021. During this process each team member set out to collect viable and useful data on each of our self-assigned individual features. We were asked to reflect on the gained knowledge and testing data and present it in the second half of Section IX (Deployable Prototype Status) of this report. Any changes that we felt needed to be made to either the testing process or prototype itself was documented.

Testing results have been compiled into a table in Appendix I at the end of this document.

J. Market Review

To make best use of our prototype, it is essential we find the best areas, or settings, in which our product would be of great use. The market review allows for us to thoroughly research information that would help make our prototype a living idea. We want society to take great advantage of our product, but the only way for it to become a product of interest is to direct it towards people who would find it relevant to their daily activities. With the researched information, we were able to narrow down settings and groups of people our prototype would be highly useful and relevant to. The information gathered will also help us better explain why our product is better than already existing products that are similar to our prototype.

The market review, in simpler words, is to show why our prototype is a product of demand and why our targeted audience needs to buy it.

K. Conclusion

The conclusion of each individual section resides in Section XI of this report.

II. SOCIETAL PROBLEM

Team 9 chose the pandemic at large as our societal problem, with a focus on the lack of preventative infrastructure when handling widespread illness. Over the past century, the word has been affected by four influenza-based pandemics: H1N1 "The Spanish Flu" (1918-19), H2N2 "Asian Flu" (1957-58), H3N2 "Hong Kong Flu" (1968-69), and H1N1 "Swine Flu" (2009) [1]. Currently, the develop of the SARS-CoV-2 "COVID-19" (2019) pandemic has had a large effect on world with first sightings of cases during November 2019 in the Hubei province in China. While our project does not have to be directly related to the current pandemic, we decided that having firsthand experiences dealing with the current challenges created by the COVID-19 pandemic could help influence and innovate a viable solution to our societal problem.

A. Pandemic Definition

According to the "Dictionary of Epidemiology" a pandemic is defined as "an epidemic occurring over a very wide area, crossing international boundaries, and usually affecting a large number of people" [10]. While all four of the pandemics over the past hundred years have been from new (also known as novel) strains of influenza, the current pandemic is a novel strain of sudden acute respiratory syndrome (SARS) coronavirus. Listed below in Table 1 "Pandemic Total Deaths 1918 – 2009", we have listed the total deaths of the pandemics from the years 1918 through 2009.

Table I:
Total Pandemic Deaths 1918-2009 [1,14]

Pandemic	Death Rates
H1N1 (1918)	50 Million
H2N2	1.1 Million
H3N2	1 Million
H1N1 (2009)	150,000 – 575,000

B. Transmission

The transmission of infectious diseases normally happens in high traffic areas where people are in close proximities with each other. These high infection rate areas include airports, education institutions, major cities, and hospitals. Those who are the most at risk to contract these illnesses are infants, pregnant women, the elderly, immunocompromised individuals, transit workers, healthcare workers, and communities with underfunded healthcare systems [2, 8, 9]. There have even been accounts of young adults contracting illnesses thought to only affect the previously listed groups, providing evidence that infectious diseases do not discriminate and that all people, and even some animals, can become contaminated with an infectious disease or illness.

i. Pre and Asymptomatic Transmission

There have been numerous recent reports relating the amount of transmission between pre-symptomatic and asymptomatic carriers of COVID-19 and how the ability to observe and contain the disease before it comes widespread becomes marginally harder when a significant number of transmissions happen when infected individuals show no signs of the disease (asymptomatic) or they are about to (pre-symptomatic) but do not know it [40]. While this is considered unusual for a respiratory illness, it should be considered when implementing standard mitigation practices to help curb the infection rate.

C. Lack of Preventative Infrastructure

As seen in Table 1 above, the advances in medical technology since the reducing the death rate of each sequential pandemic. While this is an incredible scientific feat, there is still a need to increase the preparedness of the world for the *next* global pandemic [4, 5], which in most cases is referring to the current COVID-19 pandemic, which has a current (as of May 2, 2021) death rate of 3.19 million people worldwide [15].

There are some organizations that have created preventative measure guidelines that are readily available for public health officials and even the average citizen to read and practice. The World Health Organization (WHO) has successfully set up the Global Influenza Surveillance and Response System (GISRS) in 1952, which includes laboratories in several countries to allow epidemiologists around the world to work on vaccines. In the event such medical treatment is not readily available, the WHO also created non-pharmaceutical intervention (NPI) guidelines to help mitigate the spread of infectious diseases. These guidelines include social distancing, hand washing, cough etiquette, and wearing face masks as methods to help slow the expansion of a widespread illness. [3, 4]. While these laboratories do help provide the possibility of an advantage on the spread of an epidemic or pandemic, the cooperation of the general public will often be the deciding factor in how slow or fast an epidemic or pandemic can spread.

While these government funded public health organizations do have preventative measures available, they only focus on the possibility of influenza-based pandemics. For example, the last updated pandemic plan that was released by the U.S. Department of Health and Human Services (HHS) was in 2017, and it was also focused on an influenza-based pandemic.

i. Why Influenza?

The National Institute of Allergy and Infectious Disease (NIAID) states that due to the high volume of yearly cases and death that are related to influenza, it is a high priority to act and respond to on an annual basis [11]. There are even strands of influenza that can infect animals such as bats, dogs (canine), cats (feline), pigs (swine), and birds (avian). Some of these animal variations can be transmitted to humans [13]. Other types of influenza can form, but the annual occurrence creates a huge burden on society as large infection rates surface each year [13].

D. Mitigation

As stated in Section II.C above, there are specialized organizations that work towards solving medical crises before they happen or just as they are starting to occur. These organizations focus on influenza-based pandemics due to the high return rate of influenza virus strains. While one environmental expert suggested a system modeled after the National Hurricane Center (NHC) tropical storm alert system [5], which consists of risk-mapping, surveillance, and disease forecasting to better track and predict pandemic spread patterns. This collection of preventative mitigation is much larger in scope than we will be able to accomplish in two semesters due to the required server space to handle global amounts of medical data.

i. "Storm Watching"

Columbia University's Dr. Jeremy Shaman is an expert in the environmental factors that can influence the transmission and spread of infectious diseases. He has recommended an alert system that was modeled after the NHC's tropical storm alert system. He recommended a similar three-part system that could help with the preventative measures through risk mapping, pandemic forecasting, and surveillance.

Risk mapping involves using data from areas that have historically been severely impacted by an infectious disease. The surveillance system is used to watch current illness trends to help prevent the spread, but to also feed data into the other two systems. Forecasting is also used to predict the spread of a disease and the areas it will impact based off the historical data from the risk mapping system.

While this multi-system approach is a viable option for a project, it would be out of the scope of our two-semester course. This is due to the large number of servers that we would need to help process all of the medical data to properly implement this warning system.

ii. Vaccines

Vaccines are the optimum choice of mitigation of an epidemic or pandemic because they are the best way for a body to prepare and prevent an infectious disease from contaminating one person and infecting another. We see this kind of mitigation every year when seasonal flu shots are announced and advertised in the beginning of the colder months. Unfortunately, the time it takes to properly fund, research, test, and obtain approval from the Food and Drug Administration (FDA) and other governing bodies can take several years to complete. This is particularly difficult when novel strains of influenza appear and begin to spread faster than medical experts can keep up. Should a vaccine not be readily available non-pharmaceutical intervention methods are recommended by the WHO and CDC.

At the time of this report, there are two FDA approved vaccines for the COVID-19 pandemic; one provided by Pfizer and one from Moderna. Each company has claimed a high success rate with their respective vaccines. Both vaccines require two shots taken with a minimum of two weeks between them. Currently, there is a dosage shortage crisis, and the rollout of vaccines has been slow in the United States [39].

iii. NPIs

As listed in the previous subsection, NPIs are put into place when a vaccine is not readily available for use. There are different categories when implementing these interventions, and there are different requirements and standards for different settings. For example, the local grocery store will have more strict guidelines for maintaining a clean shopping environment and the amount of people allowed into the establishment [20]. The recommended mitigation techniques for personal NPIs are social distancing, mask wearing, cough etiquette, and handwashing.

Social distancing is a method of staying a recommended amount of distance away from another person or people in public. This is to help slow the potential spread of an infectious disease by lowering the chance of expelled droplets from spreading from one person to another. This means not attended gatherings of large people unless it is necessary like going shopping for essentials (food, medicine, cleaning supplies) or attending medical appointments. Crowds should be avoided at much as possible. Currently, the CDC recommends a six-foot gap between you and another person in order to slow the rate of infection through droplets that are expelled from the mouth when talking, sneezing, and coughing.

Cough etiquette works well with the previously NPI, social distancing, because it can directly affect the number of droplets that escape from one person to another. If a person feels the need to cough or sneeze, it is recommended that they cover their mouth either with their hands, elbow, or a tissue. Afterwards, they should throw away the tissue (if used) and wash their hands with warm water and soap or use hand sanitizer if those are not readily available.

Handwashing is an integral part of maintaining personal hygiene and should be practiced multiple times a day. If someone coughs or sneezes into their hands, it is important that they go wash them immediately before touching anything else that could be handled by another person. Using soap and warm water, a person should wash and scrub their hand for a minimum of twenty seconds. If a sink with soap and water are not readily available, hand sanitizer is the next best alternative.

The use of face masks or other face coverings will also help with lowering the transmission of droplets between people by creating a barrier around a person's mouth and nose [21]. It is important to maintain the use of masks when in crowded public places and should always be worn in public if a person might be contagious or infected.

Another key aspect to the success of NPIs is the willingness of a community to follow the recommended procedures implemented by their public health ordinance. Unfortunately, there are some who chose to see some of these methods as a political issue and not a public health issue and do so by ignoring epidemiologists and other medical professionals. It is important for community leaders to express the lifesaving potential of non-pharmaceutical interventions and encourage the use of the provided guidelines.

iii. Contact Tracing

Contact tracing is a method of mitigation that helps medical experts track, learn, predict, and slow the spread of a disease [6, 42], this method is especially effective during the early stages of an epidemic before it becomes a pandemic. Through a series of questions regarding symptoms, where an infected person has been, and who they had been in contact with, a collection of connections can be made to better track and record the spread of an illness from person to person either ending at the root of the illness or the most recent contamination.

For generations contact tracing was a physical paper process and could take a significant amount of time to discover the origin of an infectious disease. While it is unfair to judge a method by the limitations of its technology, the process and technology around it has made significant advances.

During an Ebola outbreak in Africa in 2014, a smart-phone application was tested in the northern regions of Sierra Leone as a solution to the geographic challenges of communities across the area [7]. While this method did have some issues regarding the hardware and software that was used (secondhand phones and complex application software) it did provide sufficient data to be considered a success.

Another instance of digital contact tracing comes from early 2020 when China and South Korea health ordinances had created applications that could perform contact tracing by continuously mapping where users were going and their proximity to those around them [41]. If too many infections would happen in a specific area, the application would enforce a period of lockdown. Despite the success in lowering cases, both applications did come under ethical question when handling user data.

In April 2020, Apple and Google announced contact tracing applications for their smartphones to allow citizens to know if they have been in contact with a recently infected person that utilized encryption for privacy and Bluetooth Low Energy (BLE) for collecting proximity data.

Our goal in senior design is to modify and engineer a contact tracing system using radio frequency identification (RFID) devices, encryption, and a medical database to create a system that is scalable, portable, and adaptable to different environments. This design should follow guidelines established for digital contact tracing as established by the CDC [36]. By meeting these base level guidelines, our project could be considered a viable option to aid in the mitigation efforts of the next epidemic or pandemic.

E. Conclusion

Refer to Section XI of this report.

III. DESIGN IDEA

Our design idea is centered around a collection of radio frequency (RF) based wearable sensors and a central receiving unit (CRU) for acquiring data in a medical setting. While we set out to design a near perfect system, we did run into trouble throughout the design process and did have to re-access several times, even as late as the integration process.

The main concept of our design is a series of wearable RF devices and a central receiving unit that will allow a hospital or other medical establishment to perform contact tracing inside its parameters on an expandible scale. This includes bringing the system to an educational institution, military triage, or any other kind of emergency medical setting. Designing the system to be focused on radio frequency wireless communication reduces the chance of errors occurring during the transfer of sensitive data as compared to a cellular or Wi-Fi centered system by faulty power, bad reception or connection, or geographical challenges as seen in [7]. This system also needs to be created at as little cost as possible to guarantee that almost any community can afford these devices in order to keep their citizens safe and healthy. Included in the affordability of this system is a privacy focused design to ensure that all of the sensitive medical data remains in the hands of the trained medical staff.

A. Hardware

As the technical backbone for the project, the hardware for this prototype needs to be solid in its execution. Our electrical engineering students Moncrief and Sharp will be focusing their talents into implementing the hardware designs to their maximum potential.

i. Wearable Devices

The main driving force behind our prototype is a system of wearable devices. Each of these wearables will contain a transceiver that will allow the device to send and receive data between the other wearables and the central receiving unit. These wearable devices will be equipped with a microcontroller (MCU), transceiver, antenna, power source (battery array, wireless charging, or other power source), memory, and storage.

The devices will be programmed to accept and receive data from other wearables should they come within the pre-programmed parameters. This proximity sensitivity level will be able to be changed based on the recommendations from a leading epidemiology organization like the CDC or WHO. Once the device has gained a significant amount of data, it will perform a data dump once it is in range of the central receiving unit.

Ideally, the battery of the device should last a significant amount of time. This is due to the unfortunately long shifts that hospital nurses partake in, with the average shift of a hospital staff nurse in the twelve hours or more range [16]. During the Fall 2020 semester, we were able to get the battery to last just under two hours and by the end of the Spring 2021 semester we had brought that time up to approximately seven and a half hours.

During our initial discussions regarding the size of the first version of the prototype, Professor Levine suggested that we aim for wallet sized devices for the end of the Fall 2020 semester. The May 2021 version is smaller than that, so that it could be worn on the waist band of a patient or medical staff to allow for maximum comfort and portability and minimizing awkward snagging or obstructions by being too obtuse or large. The consideration for implementing the International Electrotechnical Commission's (IEC) ingress protection code (IP) has not been fully explored at this time due to the inaccessibility of proper lab equipment and funding to create enough devices to implement the tests needed to establish proper ratings. These tests would analyze the ability of solid particles (such as dust, sand, or other abrasive physical substances) or moisture to enter the enclosures of the devices and rate it on a scale of 0 to 9. This scale is based on the worst to best possible protection where 0 indicates that there is not any kind of protection and a 9 means that the device has a high level of protection.

As stated previously, we are taking the privacy of the patients and medical staff seriously, so we will be implementing encryption into all of our devices to allow for the greatest amount of security that can be provided for these wearables.

ii. Central Receiver Unit (CRU)

The central receiver unit (CRU) will be programmed to always want to receive data from the wearable devices. It will be composed of an antenna, RF receiver, a microcontroller for processing incoming data, and a second processing unit (in this case a Raspberry Pi), and a large amount of storage for keeping track of all of the contact tracing data.

This is arguably the most important piece of hardware as it will be the main repository for storing all of the data that the wearables collect, sorting it into the correct database tables, and then executing the warning system should one of the patients or medical personnel becomes infected with the illness that is being studied for contact tracing.

The receiving unit will be powered through an AC outlet or some large battery or power source as needed. Its size is not as important as that of the wearables as it will be mostly stationary during its primary use. Should it need to be moved to a separate room, floor, building, or location, it will need to be compact and light enough to not be burdensome for the medical staff or information technology technician to install in its new location. We estimate that the device will be smaller than a shoebox. The devices IP rating is also being considered, but it also falls under the same challenges that the wearable devices do; a lack of funding and lab equipment to properly test this feature.

Just like the wearable devices, the central receiving unit will also feature encryption into its software design. The only individuals allowed to access the data stored inside the device are the trained medical personnel who will be allowed to add, delete, or update patient information throughout their shift. This will play an integral part of the warning system that we will be developing for this contact tracing system over the entirety of our Senior Design experience.

B. Software

Hardware components require software applications that allow them to work to their full capacity. Without the necessary software elements, the hardware we create will not be useful for this project since both are essential for the success of this project. The software elements we add onto this project will be implemented by Saavedra and Allen.

i. Databases

A well-structured and organized database will allow for a quicker and smoother process when it comes to collecting and storing data. Our devices will intake information for multiple people throughout various periods of time, as well as through multiple uses. We will be dealing with information from various individuals; thus, the use of a database will allow grouping individuals' information between those who are patients vs. medical staff. This organization will allow for proper release of information to specified members, i.e. patients having limited access whereas medical staff have full access to information, when the tags meet and release exposure notifications.

An ideal database, that we hoped to create for this project, can be referenced by looking at Figure 1. The table below demonstrates the initial approach that we had planned, but as we progressed further into the creation of our database, there were other ideas and designs that we took into consideration. In order to maintain a privacy first design, we chose to only user phone numbers be the identifying data in our database. As seen in the table, more data is collected on the patients, given that the medical staff; authorized personnel, will be able to monitor and quickly identify those that are infected. Medical staff will be able to update information on the patients' database if any changes in the patients' status occur. More information of who could handle and monitor sensitive information will be discussed under '*Encryption*.'

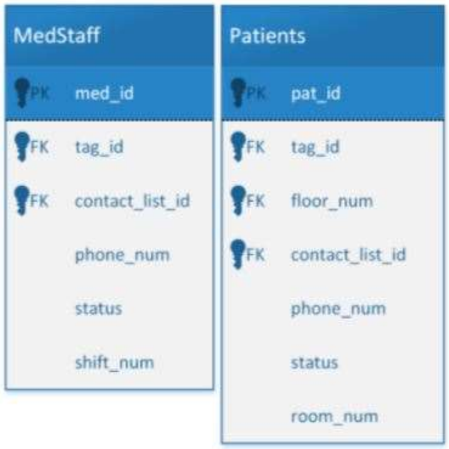

Figure 1:Med Staff vs Patient database [26]

The best way to create our database would be using SQL. Why an SQL database? SQL can retrieve, delete, update, and insert records in a database. It also allows for the creation of new tables, as well as setting permissions where needed, such as on the views, procedures, and tables. It is important to notice that, because we are handling very important and sensitive information, we will require encryption for our database. A SQL database can allow that feature to exist in a database [18]. More information about the encryption of data will be discussed under '*Encryption*.'

ii. Encryption

While discussing the database, it was mentioned that we would be storing private information from many individuals. Security is our key feature. Companies such as Apple and Google introduced the idea of contact tracing using Bluetooth for location pinging [19]. This, unfortunately, is not the best way of keeping information secure due to Bluetooth related vulnerabilities. Not securing an individual's information can cause exposure of personal data which is an invasion of the user's privacy. This makes users uncomfortable and less likely to use the intended device. Due to this, our project will encrypt the data we are intaking, to make sure our users can use our device without the need of worrying that their information will be leaked to other users. By encrypting our data, users will be notified about infected individuals with limited, but necessary information about that infected individual.

For example, let us refer to Figure 2 to analyze how the encryption will benefit users, in this case, patient-to-patient interaction. Ideally, our patients are all healthy. However, for the sake of this example, let us say that Patient C's status changes to 'infected'. If Patient C was near Patient A, this individual will be informed about their exposure to Patient C without giving out Patient C's identity. The only information Patient A will receive is the range of location, in this case the floor number, in which they became in contact with Patient C. Patient C's; the infected, personal information, as well as specific and exact location is kept confidential from other patients. On the other hand, in a patient-to-staff exposure, if Med Staff A were to receive this notification of exposure, they, unlike the patients, could access all information. With a specialized key of authorization, Med Staff A will be able to view Patient A's and Patient C's information to monitor and record any changes in their statuses.

pat_id	tag_id	gender	age	floor_num	contact_list_id	phone_num	status	room_num
Patient A	1	M	23	2	1	916-555-1234	healthy	22
Patient B	2	F	68	1	2	916-555-1212	healthy	17
Patient C	3	F	44	2	3	916-555-1001	healthy	24

Figure 2: Patient database and information. [27]

med_id	tag_id	gender	age	contact_list_id	phone_num	status	shift_num
Med Staff A	4	M	37	4	916-555-4321	healthy	1
Med Staff B	5	F	32	5	916-555-2435	healthy	2
Med Staff C	6	N	40	6	916-555-1324	healthy	3

Figure 3:Medical Staff database and information. [28]

All individuals using this device can benefit by learning where they encounter an infected, while having the confidence that their information will not be shared with unauthorized individuals. In order to ensure that we are protecting sensitive information, as already specified, a very small group of authorized people will have access to the database by using verification and their own specialized access key.

All throughout, it has been emphasized multiple times the importance of user privacy and data protection. This is what we want to achieve, given that it is a very vital component of our project. Our design is not just to make sure our device works for the public, but to demonstrate that they can confide in our developed idea without needing to feel anxious about a breach in security.

iii. Programming Languages & Features

The programming languages that we used for this project varied based on the hardware devices we bought. Collectively as a group, we are comfortable with C, and it is our preferred programming language for this project. However, we needed to learn how to use Python in order to utilize the radio frequency modules with the CRU.

Our project requires exposure alert notifications when a tag comes into proximity with another tag. We decided to implement a SMS notification that would automatically send a simple message stating the location range of where they could have been in contact with an infected individual. This message will alert the receiver without providing the other person's personal information and will urge to test themselves as soon as they can, in case they contract the virus. By having this notification feature, individuals will be notified to take preventative measures in order to decrease exposure of the virus to others.

Having this notification sent out through a text message, is much more convenient and much more effective. Notifications will be sent out quickly to devices, that we know people will always have on hand with them. Having this system provides flexibility and mobility to users of all ages that hold devices that can receive text notifications.

C. Processes

When we originally developed this project idea during the Summer, we created a series of flowcharts to better explain our design idea to Professor Levine during our final unofficial meeting in mid-August before the beginning of the Fall 2020 semester. These flowcharts covered the start of a shift, the end of a shift, wearable device interactions, and wearable-to-receiver unit interactions. These processes were designed with the concept that the contact tracing system activation, the devices will alternate between sending out signals and waiting for incoming ones from other devices. Once the wearable has received enough data, it will begin to send out longer signal until it receives a confirmation from the central receiving unit that communication has been made and data has been transferred. This process will repeat itself until the end of the designated shift.

iii. Wearable-to-Receiver Interaction

Much like the previous interaction, the wearable-to-receiver process is very similar. The receiving unit will always be in receiving mode, which will have it wait until a signal is sent from a wearable to the receiver. This phenomenon is created when a wearable has reached a point where it can no longer continue to record interactions with the other wearables and sends out a longer signal in the event that it is in a location that is farther away than anticipated.

Once the data is received, it will then sort and update the data to the main database. It will also send out a confirmation message to the wearable that it just received data from. It will promptly return to receiving mode until the next wearable is in need of dumping its data to the receiving unit. This process will repeat until the end of the shift.

iv. End of Shift/ Device Deactivation

Once the shift has been completed, all devices will be returned to their designated charging stations, which will most likely be the same area as the central receiving unit. Once the receiver has taken in the final parts of data for the shift, it will begin to sort the data and update it to the database. If a patient has become diagnosed with the illness that is being traced during the shift, any medical staff or patient that has been in contact with the infected will be flagged. Once the final amount of data has been collected, the following three events might happen.

The first being that the possibly contaminated people will receive a warning text from the warning system. This text will relay that they have recently been in contact with someone who has been diagnosed with the illness and will include instructions for social distancing and where they can get tested.

The second option is where there are no newly diagnosed patients, and the warning system does not activate. While this is the ideal scenario, it will most likely be the rarest event to occur. This is especially true when considering just how fast an illness can spread inside a hospital or other close proximity location with a high volume of ill people gathered.

Following this, the CRU will then create a .csv file of the current database, time stamp it and then encrypt ot.

v. Warning System Process

The warning system that will be in place can be activated through different points in the shift.

A SMS text message will be sent from the central receiving unit to anyone that had recently been in contact with the infected individual(s). This text will include sensitive information regarding the date of the contamination and instructions on how to receive testing. It should also ask the possibly contaminated to social distance for a recommended period of time.

D. Other Design Options

Version control and revisions are a major part of any engineering project. Throughout this process we have explored a few different options regarding what we wanted the prototype to be and how we wanted it to operate.

i. RFID Tags

In order for RFID Tags to be a nominal option for this project, we would need a significant amount of additional hardware including RFID tag read/write devices and specific active tags that usually came with a hard-shell case to protect them from extreme weather and from being crushed by a heavier object. Some tags could be secured to different types of surfaces, making it an ideal tracking device for overseas shipping and inventory management.

Most of the modules and tags that we found were prohibitively expensive and not within our budget (Section IV). The laboratory equipment we would need to alter these tags are not at our disposal with the closure of the labs on campus for the remainder of our senior design experience, we would need to find a way to design our own modules.

ii. Smartphone Application

The initial inspiration for this project idea was from hearing about the Apple and Google contact tracing applications in April 2020. The idea of using a practical software application that could easily be implemented into hardware that was already in use for a significantly large portion of the population was a case study in engineering practice.

Outside of this option primarily being a software engineering senior design project that would allow for very little electrical engineering work if at all, there are some other indicators that this was not the project idea we needed to go with. In order to successfully launch this type of project, we would need to design application software that is compatible with the two major mobile operating systems; Android (Google) and iOS (Apple). Both companies have provided source code for their notification applications through software development kits (SDK) [35].

Apple would have been particularly difficult to work with due to the monetary and hardware constraints surrounding their programming language, Swift, which is a version of Objective-C. In order to begin writing code for Apple devices, a programmer needs to own an Apple laptop, desktop computer, or tablet in addition to paying the licensing fees to become an Apple developer if they choose to publish it on the App Store. With the range of costs for their hardware starting in the low hundreds up to several thousand-dollar range, it was not a practical route from a monetary standing.

In order to gain access to Google's notification software SDK, we would need to prove that we were employed by a local health ordinance before they would release the source code to us so that we could develop our project. This would require a large amount of time and dishonesty without being employed by a local health ordinance.

E. Feature Set

Throughout the previous subsections of Section III, we have discussed and broken down the high-level concepts of our design idea. The feature sets that we will be using to define our prototype will be shown below in summaries of the separate feature sets for each semester.

i. Fall 2020

There will be four physical devices by the end of the semester: three wearable sensors and one central receiving unit. Each of these will be able to communicate with each other through radio frequency waves based off their proximity to one another. For the first semester prototype, the wearable devices will be significantly larger than they will be by the end of the project in May 2021 and may not be considered wearable by the end of the first semester in December 2020. Another physical requirement for the first version of the prototype is to have a minimum battery life of eight hours.

The software involved will contain scripting programs that will send and receive data from the central receiving unit to the database that we will have in place to help track infected patients. A SMS texting feature will be implemented into the software design to create the warning system that will be activated whenever a medical professional updates a patient's status from healthy to infected regardless of the state of the current shift.

ii. Spring 2021

The second semester will focus on refining our physical design of the wearable devices and providing more data security through encryption. We will have to reestablish our now smaller wearable devices, improve the battery life, and develop a more advanced database.

For the hardware, smaller wearable devices with improved battery life will be a key feature. Having smaller, lightweight, and compact devices will improve the chances of an individual (patient or medical staff) to wear the device. This will work especially well for the medical personnel if the wearables are comfortable and non-intrusive to their movements while they work. The added battery life will create less stress for an essential worker if they do not have to worry about the battery on their wearable device dying in the middle of their shift.

The advanced encryption and database software were also key features that were developed during the second semester. We have stated multiple times throughout this report that we are putting a huge focus on personal privacy, and we consider it incredibly important to uphold every ones' right to their medical privacy. Creating more complex database tables to obtain and store more data will help improve the contact tracing by allowing more storage space for longer contact lists of multiple patients over multiple floors or wings of medical establishments.

iii. Feature Set Summary

For the Fall 2020 semester, our prototype design will feature four devices that can communicate with each other, a simple database, and a warning system that can be implemented at any point before, during, or after a medical personnel's' shift in a hospital.

By the end of the Spring semester in May 2021, the overall design was revised, with the fully working features that we specified were to be improved. This system will give the ability for users to perform their daily activities with all stated features, conjoined, that will provide them flexibility, portability, and high-end security.

F. Measurable Metrics

The measurable metrics of this project are going to be used to show what features can be tested and measured in order to prove that our prototype works. We will provide video documentation of this process during the Senior Design Showcase at the end of the second semester in mid May 2021.

Through our video documentation, we will demonstrate the activation, wearable-to-wearable, wearable-to-receiver, and end of shift processes work based off pre-determined parameters such as distance and storage limitations. In addition, we will implement the warning system by having the central receiving unit text multiple members of the group with a pseudo-warning about needing to self-isolate and get tested. We will provide source code for all hardware and software processes as found in the appendix sections of this report.

G. Conclusion

Refer to section XI.

IV. FUNDING

During our initial Summer break meetings, we all agreed upon splitting the cost of the project evenly between the four of us. The average cost of a senior design project, as stated by the professors, was around $1000.00 USD. Based on this metric, each team member will be contributing up to $250.00 USD towards the project over the course of the two-semester. If one student had to go over the budget, the other students would help balance out the difference once the project was complete. This plan was also put in place to help mitigate any unnecessary in person meetings for trading off hardware by allowing a student to purchase hardware as they needed to complete their portion of the project.

During one of our earlier meetings with Professor Levine, he had stated that students can use any kind of premade software that they want in order to complete their project, but to try to not create highly modular systems when it came to hardware. Listed below in Table II is a list of all the hardware and software, quantity (Qty.), cost, and who purchased the unit(s) through out the project.

In the table below, the Raspberry Pi kit included the following items: Raspberry Pi 3B+, two aluminum heatsinks, a clear case, and the power supply (5V, 2.5A).

Table II:
Project Expenses [30]

Unit	Qty.	Cost	Purchaser
RFM69HCW Transceivers	4	$51.00	A. Sharp
Raspberry Pi 3B+ Kit*	1	$60.00	M. Allen
Batteries	4	$38.00	A. Sharp
SPI Flash	6	$19.00	A. Sharp
64GB micro-SD card	1	$14.00	M. Allen
Wearable PCBs	5	$20.00	A. Sharp
Parts for PCBs	3	$22.00	A. Sharp
MariaDB Database	1	$0.00	M. Allen
HeidiSQL	1	$0.00	M. Allen
Twilio Student Account	1	$0.00	Y. Saavedra

The final cost of our deployable prototype comes to around ***$224.00***, which is just under a quarter of our budget.

Initially there were restrictions set on what devices could be used for our Senior Design project, and that we were encouraged to not use prefabricated development boards such as the Raspberry Pi and Arduino microcontrollers. Due to the shipping crisis created by the pandemic during the first semester, we were granted permission from Professor Levine to use these pre-made development boards because there were more readily available that buying individual pieces that we wanted to use to create our own RFID tags.

In should be noted that we did not seek outside funding or sponsorships for the funding of our project prototypes.

V. PROJECT MILESTONES

Listed below are the project milestones that we reached throughout the three sections of our Senior Design experience: Summer 2020, Fall 2020, and Spring 2021.

A. Summer 2020

During the break between the Spring and Fall semesters, our team was formed in early June, assigned a team number (Team #9), and laboratory instructor (Prof. Levine). It was emphasized that teams should form as soon as possible and start working on each team member's individual societal problem based off of the prompt that was given to the incoming seniors during the orientation session in May 2020.

i. Team Formation

In early June 2020, our team formed over the Microsoft Teams forum provided by the professors and through email exchanges. After confirming our team with Professor Tatro, we started a group text as our main form of communication and began holding semi-regular meetings on Sundays, as that day worked best with everyone's individual schedules. Over these meetings we created some file sharing outlets through Google Drive and GitHub. After we started discussing project ideas, we began to contact Professor Levine to work out possible issues with our individual project ideas and to hold preliminary meetings to discuss our possible senior design project idea.

ii. Societal Problem Selection

By the end of our third and final pre-semester meeting with Professor Levine in August, we had decided to proceed with a project centered around using radio frequency devices as an alternative way of contact tracing. This project would be a substitute to the current smartphone applications and the traditional hardcopy method. Professor Levine approved of our idea and we started doing research on RFID technology and relevant medical research related to the spread of diseases and their evolution into epidemics and pandemics. Having a couple extra weeks of research helped us create a more solidified project idea and gave us some extra room between the beginning of the semester and the societal problem due date in mid-September.

B. Fall 2020

During the Fall 2020 semester we made significant progress on our laboratory prototype and our End of Project documentation. Through a series of writing and testing deadlines (see the Work Breakdown Structure in Section VI) we as a team defined and created our team societal problem and laboratory prototype based off of the required standards that were set by Professor Levine and Professor Tatro.

i. Societal Problem Presentation

On September 28, 2020, all senior design teams for the Fall 2020 – Spring 2021 year had to give a ten-minute presentation around their Team Societal Problem. This was significant because we got to explain why we chose our problem and give a small explanation of how we plan to engineer a solution to said problem. We were able to complete the presentation within the required time and were asked only a small number of questions regarding the topic and the style of encryption that we plan on using in our design.

ii. Feature Set & WBS

The feature set and Work Breakdown Structure of our project are integral to the project's progress, because they can define to the reader of this report and our senior design mentor, what exactly our project is supposed to do and who will work on and complete each feature of the project.

The feature set is a list of features that we have defined in the Design Idea (Section III of this report) and lists the high-level ideas and concepts of the features that will be available in both our laboratory prototype in December 2020 and our revised prototype in May 2021.

The WBS is significant because it is our detail-oriented plan on what needs to be completed and who is going to work on that portion of the project. We set aside a meeting out side of our regularly scheduled team meetings to really define this section of the report and after about four hours we felt that we had covered all of our feature set and everyone agreed on what parts they would be working on.

From this point on in our team instructor meetings, any progress that we make will be checked against our feature set and WBS with Professor Levine.

iii. Project Timeline

The project timeline is important because it takes the feature set and WBS and quantizes it. We will now be able to track and record the progress we make on our project each week as it happens. This creates a real time environment for us to work in and show our work.

With the addition of the PERT and Gantt charts, and our ability to update our progress in real time, our report will gain more depth as these additional documents add additional layers and perspectives on how and when we make significant progress on the project.

Much like the rest of the report, this is a living document that will have several revisions before the End of Project report is due in April 2021.

iv. Risk Assessment

This is where we will define how much risk there is involved with our laboratory prototype when it comes to features and outside forces that can alter our projects progress or function.

An example would be that a team member contracts COVID-19. This would mean that a team member would lose significant opportunities to work while they recover and would not allow them to physically meet to exchange hardware safely, unless significant precautions were put into place. COVID-19 has also been a risk in terms of the availability of parts for the project itself.

Another example would be the Lithium-Ion batteries that we are using for the wearable devices. If one of these gets punctures it creates a significant fire hazard as they are known to explode if ruptured.

More examples and a complete breakdown of the Risk Assessment report will be available on November 9, 2020 in Section VII of this report.

v. Technical Review

On November 16, 2020, all senior design teams participated in presenting the current status (as of the date of the presentation) of their laboratory prototype. Each team had fifteen minutes to present which allowed them to discuss the current working features of their project, as well as explain what still needed to be implemented between the technical review presentation and the Assignment 6 Prototype Review on December 7, 2020.

vi. Technical Design Evaluation

This was our final team instructor meeting with Professor Levine on Monday December 7, 2020. We presented to him our working laboratory prototype and went through each feature of our WBS to demonstrate its completeness. We had prepared additional documents and a video presentation that needed to be turned in to Professor Levine during our allotted team-instructor meeting time slot. After his approval we were able to move onto the final assignment for the first semester of our senior design experience, Assignment 7: Laboratory Prototype Presentation.

vii. Laboratory Prototype Presentation

Our final assignment for the Fall 2020 semester was our laboratory prototype presentation. This was to be a set block of time on Friday December 11, 2020 and was to act as our final exam for our first semester of senior design. During this presentation we would have been replaying a pre-recorded video that shows how our project works through various examples of our feature set in action. Following each video viewing we would have provided a set amount of time to answer any questions from our audience so that we could clarify anything in our presentation video that we might not have addressed or could possibly give insight into our revised prototype.

This presentation was to be virtual through Zoom as to keep the senior design teams and the general public safe. The presentation was the culmination of our work up until that point and provided evidence that we were able to complete our work to the best of our ability, demonstrated how we coordinated and executed tasks as a group, and provides some insight into how we solve problems as engineering students.

Unfortunately, due to technical issues and concern for the appearance of our project presentations, we were asked to only submit the required files by December 8th, 2020 in order to receive credit for this Assignment. We were disappointed at losing the opportunity to present our work but understood the reasoning behind the decision to cancel the event.

C. Spring 2021

The second half of our senior design project revolves around refining the progress that we have already made during the Fall 2020 semester. This includes adding and removing sections of our feature set, our WBS, and even some components of our laboratory prototype. While we did complete all of the required assignments, the last four events that had great significance for us were the prototypes integration, final prototype evaluation, end of project report, and the final presentation.

i. Prototype Integration

We were able to complete our project integration via direct interaction between software and hardware elements, showing that the functionality of the tags resulted in the Database being updated and the users being informed of contact tracing information. This was completed in person on April 25, 2021 the day before the prototype was due.

ii. Final Prototype Evaluation

On April 26, 2021 we had to present our final prototype design to Professor Levine during our allotted team-instructor meeting time. We had thirty minutes to demonstrate that our prototype design satisfied all of the required features in order to advance onto the final assignment.

After our presentation to him, he congratulated us on our achievement.

iii. End of Project Report

This report has a due date of May 3, 2021. We will be presenting this report and a summary video of our project to Professor Levine during our regularly scheduled laboratory period. It is there that we will receive the final confirmation that we have completed our senior design experience as long as all of our other documentation is complete. This has been an event roughly ten months in the making and we are all excited to move on from our Senior Design experience and move into our careers.

iii. Final Prototype Public Presentation

Our final assignment for our capstone senior design project was originally going to be a live presentation over Zoom, but due to technical reasons, this event was cancelled. It was originally going to take place in the morning on May 14, 2021 and much like our laboratory prototype presentation from the Fall 2020 semester, we will have a new version of our prerecorded video and visual aids that will be posted on a public access website. These will then be available to help the “attendee” The significance of completing this presentation is that it demonstrates our ability to adapt to and redefine our project as it progressed from June 2020 through May 2021. It displays how we operate as a team, as individuals, and how we engage problems from an engineering standpoint.

D. Conclusion

Refer to Section XI for the conclusion of the Project Timeline.

VI. WORK BREAKDOWN STRUCTURE

The main portion of the Work Breakdown Structure (WBS) is represented in two sections of this report: Section VI and Appendix G. In this section, we have the WBS represented in a text format, where each major task or feature of our prototype is broken down into multiple subsections with summaries of the subtasks that make up the larger feature of the project. We have completed this for both semesters starting with Fall 2020.

In Appendix G we have created the table representation of the WBS. In the table version we have also shown who is assigned to each task for both semesters.

It should be noted that due to time constraints and the limited availability of testing equipment and components, were not able to complete some of the additional features outside of the main feature set for this project.

A. Fall 2020

i. 1.1 Wearable Devices

Small devices (wallet sized or smaller) that are based upon the concept of tags in RFID system. They will include a central processing module for tag operations, a module for interfacing with the Receiver, one for interfacing with other tags, and finally a module that allows for true wireless charging from 2.4GHz transmitter devices.

a) Minimum X hour battery life

The wearable devices must maintain a minimum battery life through out a standard shift for any medical professional. We plan to use a combination of batteries and wireless charging to extend the total functioning time of these devices before they need to be charged.

i) Battery calculations

We are assuming an average shift of 8 hours, but knowing the medical field, the final battery life should be 12 hours. Some estimates were done for the batteries, and assuming an average current draw of 30 ma, a 190maH battery should last around 6 hours. With the right hardware, is should be possible to get 8 hours.

ii) Wireless charging

Based upon the original premise of Tag-to-Tag communications in which part of the power absorbed by the RF tag from the Receiver signals could then be used to allow the tag to undergo more complicated operations then simple backscattering, even if the tag is passive. The system would absorb power from the carrier waves created by the extremely common 2.4GHz class of devices. This would hopefully help augment the battery life of our tags.

b) Wearable to Wearable Communication

One of the key features of our whole project is the ability for these wearable devices to communicate with each other over a maximum distance six feet, as well as hastily transfer data between these devices. Ideally, we will be able to fine tune this feature to allow for maximum communication between these devices through implementing effective software into the main data processing unit on the wearables microcontroller board.

i) Transceiver function testing

To ensure the transceiver we plan to use would work, testing needed to be done. Range tests through open air and through obstructions were tested, and the RFM69HCW is suitable for our proposed design.

ii) Software Implementation

Implementing the software that is required to have the wearable devices communicate with each other through RF waves. This will most likely be Arduino based from the research that we've done on the radio transceivers that we have purchased and the software libraries that are available for these pieces of hardware.

c) Interference Limiting

For our design we will be using antennas rather than Bluetooth pinging. Due to this, we want to make sure we are limiting the amount of interference from unwanted, external signals from disrupting the communication between our tags and receiver unit.

d) Communication with the CRU

In order for the tags to communicate with each other, they need a middle source, in this case our receiver unit. Our receiver unit contains the necessary information and will allow the alerts to be sent wherever it is that they need to be reached. As long as our receiver unit works, the communication between our tags should work and communicate as well.

e) Proximity recording (X < 6')

As stated previously, a large part of the wearable devices features is the ability to record one device's distance from another if they enter the possible infection zone of six feet or less.

i) Tracking Wearable to Wearable Interactions

We will be attempting to record the proximity of each wearable device. As mentioned, one of our features is to notify users when they encounter an infected, but because we are not exactly certain of when this communication takes place, we want to record each time, how far these tags are from each other before they send the alert.

ii) Time Stamping

There are functions that can be called that can record the time of another function's execution. We plan on implementing these functions into our wearable devices to allow us to track the amount of time two users spend in close proximity to each other.

iii)Time Stamp Comparison/Processing

Comparing the results of the time stamp functions, we can calculate the amount of time two wearable devices spent in close proximity to each other.

ii. 1.2 Central Receiving Unit

a) Data organization

The organization of data is a key factor in making our prototype work. The CRU will be receiving data from the wearables intermittently throughout a shift and must be able to organize all of the packets of data that it receives into the right tables in the database.

i) Sorting Algorithms

The CRU will need to have sorting algorithms set in place to allow it to analyze and sort through packet information that is received from a wearable device. For the first semester, we will be focusing more on implementing a sorting function and focus on the execution speed during the second semester.

ii) Scripts to update Database

When the CRU receives data from a wearable device, it must update the required table with this new data. From our research, there are Python software libraries that can update, create, and delete databases and database tables.

b) *RPi/Arduino Implementation*

i) Software Implementation

We will setup the required libraries and the software to allow the Raspberry Pi to program and control an Arduino Microcontroller. This will allow us to have the Raspberry Pi act as a more sophisticated data processing device, while the Arduino microcontroller acts as a peripheral.

ii) Separate Arduino Serial Communication

Since it is likely that we will have separate microcontrollers interact in this project (possibly on the tag, if power draw allows), we will use SPI communications to facilitate an exchange of data, to better help with both tag to tag and Receiver operations.

i) Software Implementation

There are many available ways to program the MSP430 microcontroller. However, since we are using Arduino devices and peripherals, it makes sense to select a programming method more in line with those devices. To this end, the platform Energia was selected. Energia is an Arduino based platform, which allows us to use Arduino sketches on TI devices. The UI itself is almost identical to the Arduino UI.

ii) RPi Serial Communication

Since we will be using an Arduino compatible device-in the Receiver part of the CRU setup, we need to be able to "dump" the tag data it receives into a more sophisticated processing unit. In this case, the data dump will be sent to a Raspberry Pi, so serial communications (SPI, I2C, UART) are going to be established between the two devices to allow the data to be transferred. From the Raspberry Pi, we can send the processed data to an even more sophisticated device, such as a central server.

d) Warning System Activation

Our design incorporates a warning system that will alert the users of the device when they have encountered an infected. Listed below is the type of system we'll be using to warn the users.

i) SMS Text Message System

The implementation of a SMS text messaging system will allow for us to program a release of an automated message that will alert an individual if they encounter someone tested positive for the virus. This automated message will only send messages at certain periods of times, only when needed. The release of text messages will be controlled so that they are not bombarding the users of the devices with an influx of warning messages. This semester we are still working on a prototype, and therefore, the possibility of the consistency and amount of messages that are sent out to a user will probably not be how we want it to be in the first place.

ii) Python Script

Our receiver unit is compatible with Python, as well as most of the devices that are used for this project. By using this language, we will be able to create code for our databases and features that we stated will be implemented into the design. Additionally, since we are implementing a SMS text message warning system, many of the SMS APIs work best with Python.

e) Processes

There will be two main processes that the CRU performs, and that is the "Tag Setup" at the beginning of a medical staff shift and the "End of Shift" process.

i) "Tag Setup"

This will be the initial activation of a wearable device upon the beginning of a medical personnel's shift or be assigned to a new patient that has verified that they will contribute to the contact tracing process. The activation process will include clearing any non-essential data, making sure the device's battery is charged, and establishing a new tag ID if the device is being used for the first time.

ii) "End of Shift/Data Dump"

At the end of each medical professional's shift, they will turn in their wearable device that will dump any relevant data to the CRU. If needed, the CRU can activate the Warning System.

iii. 1.3 Database

We will be implementing a simple database to help with the tracking of medical staff and patients. We have chosen Listed below are short summaries of each table that we plan on implementing into our design.

a) Simple Database

The simple database will be comprised of six smaller tables that only have three to four attributes each. Most of the attributes can be found in multiple tables, but due to the rewritable nature of some of these attributes they will not be considered primary or foreign keys. This does create smaller challenges when creating connections between tables.

i) Tag Table

This will contain the device ID numbers (we will refer to them as tags for now), as well as a connecting phone number and status indication. The "tag_id" attribute will be the primary key for this table.

ii) Person_Type Table

This table will have attributes that help define the type of user that is wearing one of our devices. The table consists of attributes related to tag id, type of user ("med" for medical staff, "pat" for patient), phone number, and health status ("h" for health, "i" for infected).

iv) Infected Table

This table will contain the "tag_id" as a foreign key and will list all patients and medical staff that are currently under the infected status. There will be a trigger that will delete (drop) an infected individual from the

v) Phone Number Table

The will be a table of phone numbers that will once again contain the "tag_id" as a foreign key.

vi) Contact List Table

This table will contain a complex key that is created from a user's tag ID number and a pseudo list number for each user's contact list. An example could be "1001" for 1 (con_id) and 001 (tag_id).

b) Med Staff Access

Part of establishing a database is allowing an administrative user to have total access and control over the database. We will create a generic username and password for the first semester to test out the Warning System Activation and table updating. During the Spring 2021 semester we will create a more complex access system with encryption.

B. Spring 2021

i. 2.1 Wearable Devices

a) Extended battery life

i) Battery calculations

Similar to the previous semester, the boards we use are likely to be the biggest consumer of power. To maximize battery life, we will need to minimize the power consumption of both the microcontrollers and transceivers. Some battery life is regained by the Wireless charging, but the main focus will be on reduction in consumption.

ii) Wireless charging

An additional feature we would like to add onto this design, is the idea of having our devices have wireless charging. Our battery life will increase for the Spring 2021 semester and implementing a wireless charging feature would be an interesting concept for this design. There are similar designs that use this functionality in the industry, but due to the lack of resources and access to proper manufacturing facilities (as outlined in the Risk Management Section), the required effort would be outside the scope of the project, so further efforts were made into improving the battery life of the tags through other means.

b) Wearable to Wearable communication

The tags themselves would imitate an active tag-receiver system, so they would have the capacity to register proximity with each other without input from the receiver. To further expand upon this, the tags would temporarily store these interactions until prompted by the receiver to upload them.

i) Transceiver Antenna update/Optimization

While the range tests with a simple wire antenna were very promising, in spring we may want to improve our antennas. It may help with power consumption and could help if any range issues between a tag and the CRU come up.

ii) Software Update

The communication software will be updated to try and maximize the response time between two communicating wearable devices. This will be further defined in the Spring 2021 semester.

c) Design Update

i) Smaller Design

In the end the project does need to be wearable. The biggest size limitation is likely to be our transceivers, as the microcontrollers can be shrunken down to just about the chip. For the first semester, we are aiming for a wallet sized tag, but for the second semester we are hoping to fit into something that can be worn on the wrist.

ii) Make It Wearable

At this point in time, making it wearable is an unknown. We need to know the final dimensions after scaling the tags down to put into something that can be worn on the wrist. Some potential designs are a spring steel base, or repurposing an old fit-bit and putting the hardware for the tags into that.

d) Encryption Implementation

The wearable devices will contain information about the user. In other words, the device they hold will contain their name, health status, phone number, and tag ID. The user will be prone to having their information hacked if it does not have the necessary encryption. Implementing encryption to these devices will prevent the user's information from being sent to another user, as well as any other unauthorized personnel. If the devices were to be lost, the encryption will prevent others, i.e. non-owners of the device, from trying to obtain the information on it.

ii. 2.2 Central Receiver Unit

Updates to the CRU will include advances in the software (data organization, processes) , possible hardware updates, and the creation of some kind of housing unit to contain all of the physical components that make up the CRU.

a) Data organization

A significant portion of the Spring 2021 semester is reviewing and updating our prototype to better fit our more defined feature set. This includes updating the software portions of the CRU.

i) Update Data Structures

After reviewing the processes and how long they take to update the data in the database tables, we might need to update the sorting algorithms in order to speed up the flow of data in the CRU and database.

b) Encryption Implementation

The central receiver unit will be intaking information in order to relay back the necessary information to the receiving tag. As already stated, sensitive information is being handles, and thus requires data security. Implementing encryption will prevent unauthorized individuals from obtaining personal information, whether it be the person receiving the warning alert, or the person who triggered the warning.

c) RPi Updates

i) Software Updates

The software implementations from Fall 2020 will remain in this design. However, because of the new additional features, our software requires updates. We want to make sure our data is processing correctly and smoothly, especially as we increase the battery life. As we increase the battery life, we will need to modify certain components so that our hardware is able to withstand this new power source. As we continue to finish our project, we will add more details about the improvements in our software. More details about these improvements will be provided in Spring 2021.

d) Warning System Activation

Our design incorporates a warning system that will alert the users of the device when they have encountered an infected. Listed below is the type of system we'll be using to warn the users. It will include improvements and modifications from the same system used in Fall 2020. Additional details of these improvements will be provided in Spring 2021.

i) Update Scripts

Based on the system that was incorporated in Fall 2020, this semester we will use the same warning system, but with modifications. Our goal is to make sure everyone is receiving an alert when needed, but in a way where it is not overwhelming the users due to a bunch of warnings. In Fall 2020, it was stated that we wanted to have a controlled warning system that'll prevent an influx of messages to the user. Spring 2021 will be dedicated into updating and improving this system. Users will only be notified when they encounter an infected every certain, and specified, period of time. This will allow for an efficient message transmitting system.

e) Design Update

ii) Create Housing/Casing for unit

Similar to making the devices wearable, the final dimensions must be known for the casing of the CRU. It will either be plasma cut or laser cut acrylic walls, with a hole for the final antenna the CRU uses.

iii. 2.3 Database

a) Complex Database

A more complex database will be implemented through updating and possibly deleting tables that are no longer needed. More information will be available in the Spring 2021 semester. This could include using a different relational database system that utilizes a server or requires more security for a user to operate.

i) Update current tables

For this update we will be creating more complex tables for the database by including more specific attributes in each table. We will also consider removing some attributes or tables all together if they do not fit a specific need after the laboratory prototype review in the Spring 2021 semester.

b) Encryption Implementation

The database is the primary unit that holds each user's information. For this project, we are handling with 2 specific databases, the patients' and the medical staffs'. Each database contains personal and sensitive information, with the addition that the medical staff's database contains a bit more information about their schedule. It is very important both databases are encrypted. Encryption will prevent unauthorized individuals from trying to tamper into the system to obtain information they shouldn't. In this case, medical personnel will be allowed to view patients' information with an assigned encryption key. On the other hand, patients are not allowed to look at anyone else's information. Even when patients are alerted about their contact with an infected, the data encryption will prevent them from receiving all information that a medical staff can view.

C. Block Diagram

Shown below in Figure 4 is a block diagram representation of our project. It shows the main part of our project on top and breaks down the three major parts that make up our project: Proximity Detection, Data Organization, and Contacting Potentially Infected People.

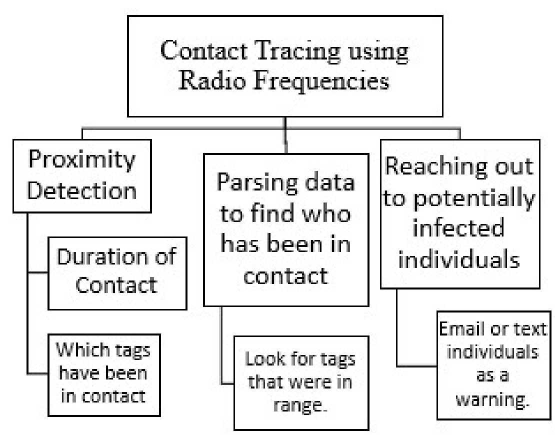

Figure 4: Block Diagram of Prototype Features. [29]

D. Project Timeline

Shown in the previous section of this report (Section V: Project Milestones) is a modified version of the Project Timeline, which is the quantized version of our WBS. Our timeline contains all of the same information as our feature set and WBS, but it is given exact start and stop dates for each feature and assignment over both the Fall 2020 and Spring 2021 semesters. Section V shows significant milestones in our projects progress that highlights events from our team's inception in June 2020 through the very last team presentation in May 2021.

E. Missed Features

As stated in the introduction subsection of this portion of the report, there were several ideas that were not fully implemented due to different constraints. These include the wireless charging feature, upgraded antenna for the radio transceivers, increasing the speed of program execution, and casing for the CRU.

While these would have greatly increased the technical and financial value of our project, we decided to drop these auxiliary features due to time, money, the feature had phased out of our development process, or due to a lack of proper testing equipment. Had the conditions for our Senior Design experience been different, there is a very real possibility that we would have had more time and resources to implement more of our design ideas.

F. Conclusion

Refer to Section XI for the conclusion of this section and Appendix G for the table representation of the WBS and the hours summary table.

VII. RISK ASSESSMENT

A. Introduction

Listed below are the three lists based off three different risk types. We have broken down what we feel are the systematic, specific, and broad technical risks that are involved with the process of engineering and deploying our laboratory prototype. Shown below in Figure 5 is our Risk Matrix that is based on the probability and impact of an event happening that could prevent progress or function of our laboratory prototype model.

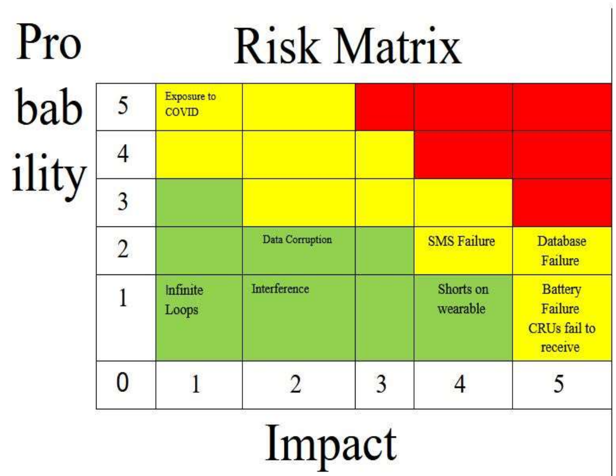

Figure 5: Risk Matrix [33]

B. Systematic Risks

Systematic risks include all risks that can stem from outside of the project and can have some form of influence on the direct outcome. These risks include environmental events (wildfires, floods, earthquakes), family crises, medical emergencies (COVID-19, long-term or sudden illness), and other individual life changing events.

i. Current Risks

The main systematic risk that poses a threat to the progress of our project's progress is the global pandemic, COVID-19. In March of 2020, CSUS and the rest of the California State Universities closed for the remainder of the Spring 2020 semester, once the state declared that a lockdown would be put in place. It was later decided that campuses would remain closed throughout the Fall 2020 and Spring 2021 semesters, with the only exception being that certain lab centric classes would be getting access to on campus facilities. Due to these closures, the entire senior design class will be without access to the laboratory classrooms and equipment that engineering students would normally have access to during their year-long capstone projects. This campus closure impedes our progress as we lose access to critical testing equipment, important analysis software, and in-person help from our professors.

It should be noted that there have been previous semesters in recent history that left senior design students without the labs due to campus closures (Camp Fire, 2018), the length at which this pandemic has prevailed and its effect on the community both locally and globally seems to be boundless. This is the first senior design class that will be completing the two-semester process through distanced learning methods at this establishment.

The pandemic has also slowed down the supply chain for every global industry. This includes businesses that would normally be able to ship components and parts to us from outside the United States in a timely manner for reasonable prices. For example, there was a specific part that we were looking to purchase for our receiver unit that is normally a lower cost item. We could only find it from a private seller in China and the mark up on shipping brought this item into the thousands of dollars price range due to shipping and regulatory fees. The wait time for this part was also outside of the time we had left to create our laboratory prototype.

ii. Team 9's Precautions

With campus being closed, our team sees a significant drop in the possibility of contracting COVID-19. The instructors explicitly stated that we are only to meet in person to exchange pieces of hardware while maintaining proper social distancing guidelines, such as remaining six feet apart, wearing masks, and only having brief interactions to perform the exchange [20].

To combat the possibility of losing additional project progress, we have established some transparency in our communication that comes with the understanding that there are some events in life that cannot be avoided and arrive at inopportune moments. It is why we have as much of an open conversation between the four of us so that if needed, one or more group members can take over for the absent team member.

This is another example of why having a balanced team of engineering disciplines is an advantage that Team 9 proudly holds. If one major student needs to step away from the project, there is another one that can take their place with little to no effort. It also helps that there are overlapping topics and courses between the computer engineering and electrical engineering major paths. So even if there is a major event or events that remove two students from the project, the other two can work together to continue making progress.

While we do sincerely hope that these events do not happen, we are trying our best to remain ready for any and all events by having open communication and filesharing platforms updated so that the impact on our progress is minimal.

C. Specific Technical Risks

Specific technical risks are errors that can form when the wrong pieces of hardware or software are implemented into the project that end up causing the system to partially or completely fail.

i. Failure to receive/send.

The CRU tells the wearable devices when to transmit their ids. If the CRU fails to send out the time, no receivers will transmit, and no range between devices can be calculated. If a wearable fails to send or receive, it can either have no impact, or stop one wearable from being traceable. If a wearable misses one cycle, but continues to transmit on time for the rest of the user's shift, then no issue is presented. If, however, the wearable stops receiving or transmitting, then it is functionally useless.

The best mitigation for ensuring tracing can continue if one device fails, is that the wearable devices do not completely depend on each other. If one out of five wearable fail, the other four will continue to work.

ii. SMS API Failures

Sending an alert at the correct time, to the correct individual, is fundamental for this project and helping mitigate the virus. The SMS system should send an alert warning to individuals who have encountered an infected person. Not having a correct warning system defeats the purpose of mitigating the virus. Even the smallest errors in the API can negatively affect the people who are receiving these alerts. If the script code had no control block, it is highly possible that the alert system would be stuck in an infinite loop. This infinite loop will, as the name suggests, send an infinite number of messages to the tag users when it should not be doing so. If this error would occur, it would discourage people from using the device.

Another potential risk in the SMS API, could be message alerts being sent to the wrong person or people, and/or messages being sent at the wrong occasion. Both these failures can cause panic to the individuals that are being alerted, even when they did not encounter an infected person. Alerts should only be sent to those who are confirmed to have met infected people. If messages were sent to the wrong people, and/or sent when there was no encounter, it could be that our API is incorrectly collecting data, or incorrectly searching through the database, hence the mistake in the alert's release.

Failing to send an alert message would continue the risk of healthy individuals contracting the virus if they're not notified of that potential exposure. Many peoples' health is put at risk when a warning alert is failed to be sent at the appropriate day and time. Not warning the person, could cause them to get sick, and/or get others around them sick without knowing that they were the ones who are carrying the virus. Delayed messages can also prevent the healthy individual from taking the necessary precautions, as soon as possible, to make sure they are not exposing others. By not correcting these issues, we are not helping mitigate the virus. Therefore, the main purpose of our project after we have taken all these factors into consideration and are approaching them in the best matter possible. Only then, will our project work as we intend for it.

iii. Data loss or interference

If noise interferes with data transmission, a wearable device may fail to transmit, or a wearable may receive the improper id, which could cause improper tracing. If the CRU transmits the wrong signal when it should be transmitting the time, then there is a chance none of the receivers send out their id when required.

The risk of interference is mitigated by short wires, as well as the frequency of the wearables being in a less used range, centered around 915MHz. The closest commonly used band in the spectrum to the one we are using is TV at 862MHz, which should not cause any interference.

iv. Database Failure and Security

The database contains and manages the sensitive contact tracing data that the wearable devices have collected, and medical staff have been updating throughout their shift. Having the information misplaced due to a faulty software process or malicious intent means that critical medical information is lost and efforts to trace an illness is reduced or halted until the database can be brought back to working order.

For this project, we are creating and maintaining a database that does not require the use of a server to record the medical information. This means that all of the data is saved to disk on the CRU. For this project we are using a Raspberry Pi as the main CPU to control the CRU and the rest of the system processes. This simultaneously solves and creates some problems in terms of the security of the database itself.

As stated earlier, the database is stored in the CRU's microSD card, which means that the device cannot be remotely compromised unless given the proper access through the internet. We are designing this system to not need an internet connection to work properly, meaning that the chances of being remotely compromised drops significantly.

Physically if the CRU were stolen, had a USB with malware scripts installed, or dismantled in anyway, the entire system could fail. The best way to mitigate this issue would either to have a secondary memory storage device present or to create a container for the CRU that makes it difficult and obvious when someone is tampering with it. Making it have bright colors and be obtuse is shape would prevent it from being easily stolen. We will be exploring container options for our wearables and CRU in the Spring 2021 semester.

C. Broader Technical Risks

Broader technical risks define our initial design ideas, measurable metrics, and feature set of our prototype and how the first idea is not always the best idea. In an earlier lecture from Fall 2020, the professors stated that senior design teams often reach for greater ideas than they can achieve in the roughly 40-week program. To combat losing excessive amounts of time and energy on professional level ideas, teams maintain weekly meetings with their lab instructor in order to clearly define their measurable metrics and feature set so that students can complete their projects in the already compacted timeline.

This type of risk can also reference general issues that can arise during the production or use of our laboratory prototype such as a microcontroller development board becoming unusable, bad wire connections, improper software implementation, and even random hardware and software failures.

i. Connection issues / Shorts

While a short on a wearable device at best does seemingly nothing, it at worst starts a fire. If the short kills the wearable, contact tracing between the other devices will not be affected, but whoever's wearable that is now no longer working can no longer be traced, and if they come into contact with someone who is infected, they will not be able to use our devices to tell. If the CRU has a short somewhere and is fried, then there cannot be any contact tracing before the device is fixed. Similarly, if a wire that is critical to send data shorts to ground, then no data will be sent, stopping all tracing.

Some ways to mitigate this particular risk are proper soldering, shrink tube, and other insulation techniques. If the wires are the correct length, it minimizes the chance that a wearable's internal connection is disturbed.

ii. Battery Failure

One of the biggest fears in any device that is on a person in any capacity is battery failure. If the battery fails, does it cause a fire or explode? If the battery does fail, it needs to fail in a boring, yet safe, way. For this project, if a battery fails, it will only stop that specific wearable from functioning. If it fails in a non-violent way, everyone can go about their day, and the other devices will continue to function as if nothing happened. If it fails in a violent way, which we will be avoiding, then it is likely that no one will want to wear the wearables anymore, and I cannot blame them for that.

To prevent battery failure, or at least prevent non-violent failure, one can implement a battery monitoring system. If the battery gets too hot, it the devices can be turned off. Of course, if the battery failure is caused by a short, nothing can be done, but a short between battery terminals can be mitigated by using the proper connectors, as well as proper insulation between wires in the circuit. If a battery is punctured, there is no way for it to fail safely, but the only way to make that as safe as possible, is to make the wearables easy to take off in an emergency.

iii. Software Malfunction

While there are many different pieces of software that will be implemented into our project, it does not reduce our chance of a software failure. This could simply manifest as a compiler error, improper function use, or even an infinite loop being implemented.

In order to prevent and mitigate these types of errors from happening, we are constantly testing our code before it is being implemented into the main portion of the main project. Other techniques include edge case testing, code review (over Zoom or GitHub), and debugging the code as errors occur during the software's creation.

iv. Random Failures

Just like unexpected life events happening at random, software and hardware can both experience random failures and faults based on different events happening at the same time. For example, there could be an issue with the CPU of one of the development boards that we are using for our wearable devices that could either be grounded or misconnected to the rest of the board, and could cause an issue when trying to execute different commands. Statistically, these types of errors do happen when the silicate is being prepared for these incredibly small processors and is almost guaranteed to happen to a percentage of processing units on the production line.

Issues with software can occur when using an open-source option over a commercial product that is not regularly updated or maintained by the community that originally created it. For instance, we were originally going to use an open-source integrated development environment (IDE) for one of the development boards that we were testing for our central receiver unit. We ended up running into issues when trying to import a header file that none of us could solve. This issue only seemed to happen in this IDE, and after troubleshooting the problem we abandoned it in order to prevent any more loss of time and energy on this portion of the project.

While random events are hard to prepare for, it seems like the only mitigation is to realize when it is time to stop working on a problem and switch to an alternative approach. This saves time overall and resumes progress on the project with very little effort. While it is good engineering practice to keep working at a solution, learning when to change directions can be just as good of a skill to have.

D. Conclusion

Refer to Section XI for the conclusion of this section.

VIII. DESIGN PHILOSOPHY

Our societal problem brings out to light that regardless of the time and location, as long as we lack the proper infrastructure to mitigate diseases and viruses, these will continue to spread quickly and broadly. Due to this, we have formulated a design idea that will provide this type of infrastructure. This will not only reduce the spread of diseases but also bring awareness to others about the importance and relevance of this issue. This section will discuss the approach that was taken to formulate this solution, as well as why we decided that contact tracing through a system of wearable radio frequency devices was ideal.

In order to formulate a suitable solution to our societal problem, we had to conduct research about the measures and solutions that were already in place to mitigate the ongoing pandemic. We had to analyze who our product would most benefit, how, and why it was appropriate to bring our design to life. Upon our research we were able to find an idea that also used contact tracing. However, after analyzing this idea, we found flaws in it; the biggest one, privacy. Having this in mind, we thought out of a way to make our product an item that would not only help mitigate the disease but also ensure privacy safety to our users. Rather than using Bluetooth, we decided that the use of radio antennas would grant better protection to the private information that is being transmitted from our central processing unit (CRU) to our wearable devices (tags). Essentially this became a very big and central focus for our design.

Although there existed a design similar to what we thought of, the lack of privacy is what made us realize how it prevented people from wanting to use the device. Therefore, after much discussion, we decided that our device would need to stand-out. We had to create a feature that would guarantee the users' safety and information privacy. Only then would our product stand-out and would be an item that would promote usage. Our design is intended to help the community; help society, and the only way we'd be able to have it work as intended, and have it used by others, is to provide features that would be found appealing and where safety is guaranteed.

After finally establishing the set features for our product and agreeing on their priority, we had to make sure our product was affordable. Research about the amount and quality of parts was intensively carried out so that we could provide a quality product. We wanted a portable device that could be used by all types of communities. We were well aware that some communities lacked funds, to begin with, to even afford medicine or equipment for health improvement. As we conducted research and thoroughly planned out our design, we had all communities in the back of our mind. The idea of mitigating a disease through contract tracing should be established everywhere. Our design was thought out carefully to serve the community for better by providing an affordable wearable to all, that carries features that'd promote usage and safety.

IX. DEPLOYABLE PROTOTYPE STATUS

A. Current Status

Currently, the deployable prototype has been completely integrated to meet our feature set standards. The rest of this section contains our testing methods and data that we were able to fit into the format of the body of the report. We have documented the rest of our findings in Appendix I at the end of this report.

B. Device Test Plan

Listed below are some of the tests that will be taking place between February and the beginning of April 2021. Some versions of these tests have already been performed when working out issues with the laboratory prototype towards the end of the Fall 2020 semester.

i. Hardware Testing

a) Range tests

One set of tests that will be performed are related to range. The wearables and CRU must be able to communicate in a hospital setting. This means that the wearables need to get the data from the CRU, likely through walls, and the wearables need to know when they are in the specified range. Wearables will also have to be configured to *not* detect each other through walls, likely by reducing power output of the tags themselves. Overall, a way to test this is to simply have the wearables be varying distances apart, and with varying obstacles between them. The libraries used for the RF modules have functions to check the RSSI (for the RFM69, this value will range from –15 to –80 [47], Table III below shows ranges for ideal data transmission) of the last received message, and this can be used to ensure the devices function at the expected ranges and environmental conditions, particularly those found inside of University buildings and Hospitals. In order to systematically test this, we will ensure each tag can receive from the CRU at a given range, say 60 ft, for 10 or more transmissions. We will do the same with transmissions between wearables, albeit at a shorter range. In Table III below, we see the testing data from our RSSI range test.

Table III:
RFM69 RSSI Value Ranges [46, 47]

RSSI Range	Status
-15 to -25	Ideal
-25 to -35	
-35 to -45	Good
-45 to -55	
-55 to -70	Marginal Data Loss*
-70 and below	Presence of interference

**Data loss is proportional to the amount of data sent.*

b) Battery Life

While we can measure the current over a period, and use those measurements to estimate the battery life, as the voltage on the battery changes, it is likely that the current will change too. This means that the battery life will have to be tested while in circuit. The devices will have to be functioning, and preferably moving. Since it is difficult to move them simultaneously with the health guidelines in place, one device will be stationary while another is moved periodically. This will allow us to measure the current with differing distances between devices, as well as see how long a battery will last. Ideally this test would be performed at least 5 times, but given the time needed to do the test, 2 tests per wearable should suffice. Once again, these tests will need to be performed with all wearables, and their battery life will need to meet a given minimum, which is generally a fraction of a "shift". This minimum will be determined by placing two tags within 6 feet of each other and have them function (by logging each other's presence in the preset 5 second intervals) until battery life becomes low enough to preclude continued operations. Table IV shows ranges of battery life in this type of test that constitute a "pass" or a "fail." A "fail" means that it is very likely (that is, too inconsistent to consider in the final product) the battery on the tags will not last the duration of the shift, which we set at 8 hours.

Table IV:
Battery Life Stress Test Ranges [46]

Battery Life (Minutes)	Status	Pass/Fail
0-45	Poor	Fail
45-60	Below Average	Fail
60-65	Average	Fail
65-70	Above Average	Pass
70+	Ideal	Pass

c) Data Transfer

These tests include both wearable to SPI flash chip, as well as flash chip to the Raspberry Pi in order to be uploaded into the database. One potential test for the wearable to chip transfer, is to transfer data like what will be transferred when the device is in use. We will likely be transmitting 10 bytes of data every 5 seconds to the SPI flash chips. In order to test the transmissions, we will upload random data, say 20 transmissions of 50 bytes, with each wearable to its SPI chip. Then the data will need to be read back, to ensure that the correct data was transferred. The same write/read test will need to be done when uploading data to the Raspberry Pi. Another way is to investigate the CS line, which will allow us to determine where the data is being sent, be it tag storage or Receiver storage. This will allow us to see if there are any issues with writing to the chips before we put the whole thing together.

Our project design utilizes several pieces of hardware, three of which will have the potential to be constantly moving around. It is important that we perform the correct hardware tests so that we can make sure that all of the hardware is working correctly and efficiently.

ii. Software Testing

Due to the large amount of software that goes into our prototype design, we will be testing and debugging our code as we write it to ensure that it is working properly. This process will also aid in optimizing our code's efficiency. Listed below are some of the heaviest software-centric portions of our project that we will be testing over the next few months before the final prototype is due at the end of April 2021.

a) SMS Text Message Alert System Testing

The SMS alert system will only work with the help of the tags' communication. The alerts are sent based on the distance of the 3 tags (hardware) that we are using, as well as the health statuses of the individuals carrying those tags. As long as any of the tags come into a certain range of distance from one another, and at least one individual is considered infected, a text message will be sent. Text messages can be sent simultaneously if one infected individual encounters more than one healthy individual. Testing results are not limited to just sending alert messages to local numbers. We are also implementing the idea of having alert messages sent to individuals with phone numbers with different area codes.

Testing results should demonstrate a timestamp on the screen of when the message was sent. The timestamp should match the same timestamp of that on the users' phones if they are expected to receive an alert message. Messages being sent vary based on the location of proximity as well as the users' individual statuses. If all 3 tags were to be within a close range and 1 individual were to be infected, the other 2 tag users would be notified of the encounter. This would also work in the case of 2 of 3 tag users were people who were considered sick and were all within the proximity range; the heathy individual will be notified of their encounter with 2 people. No testing outputs would be provided in case of the tags never meeting. No testing outputs would also be provided if it is certain all individuals are healthy despite their proximity. The only testing result that could be provided during a "no message sent" event, would be a timestamp on our screen of the program running with no message sent to the user to verify our alert system working.

Currently, there are holder values that substitute as tags. The SMS alert system should work after integrating the range test code into this portion of the project. After integration, the outputs of the range tests, and the data in the databases, determine which tag owners get notified of exposure.

Examples of this code can be found in Appendix C – Software. Testing data and results can be found in Appendix I – Device Testing.

b) Data Encryption Testing

The data that our databases hold contain sensitive information. It is very important that we create a system where this information stays private, and in case there was a sudden glitch or an attempt of breach, the information is not viewable to the public. Encrypting the data will allow only authorized individuals to access the information when they need it.

The code will output encrypted, or "locked", information if there was a sudden unwanted distribution of alerts through our alert system containing private information of our users. As mentioned, if there ever was a glitch to occur, or someone wanted to tamper with the information in our database and send it to themselves through our already established SMS alert system, the encryption that is being established will make sure that information that is being sent is just a bunch of random numbers or letters, rather than real data.

Testing outputs for the data encryption testing should show admin users receiving a warning message of an attempt of security breach when any of the mentioned scenarios above happens. Just as in the SMS alert system testing portion, a timestamp would be shown on our screen of when these messages are sent. The timestamps should match those on the admins' phones of when they receive this urgent message. It'd be safe to only notify the admins so they could address the problem quickly without needing to worry the tag users. Tag users should only continue to receive notifications of exposure to sick individuals. On the other hand, if the person trying to breach into the database tries to send themselves information from this database, they would only be receiving encrypted data. The information found in the database would not be viewable to them.

Examples of this code can be found in Appendix C – Software. Testing data and results can be found in Appendix I – Device Testing.

c) Database Testing

When building any database there are several factors that need to be put in place in order to make sure that the data you are collecting is secured and can be manipulated as needed by the allowed users. While we did not have enough time to explore advanced security options through penetration testing, the database will experience user access testing via the user authentication methods provided by MariaDB and HeidiSQL (the GUI that we plan on utilizing for our new MariaDB database) as well as running tests to make sure that the tables in the database are properly updating.

i) User Authentication Testing

Due to the sensitive nature of the data being created for this project we need to create a system that only certain people can access. In order to do this, we will need to create different users through the HeidiSQL database management service and give each of these users different roles such as administrator or medical staff. After creating these different user accounts, we will need to successfully fail to login to each of these user accounts. This could easily be performed any number of times by entering random passwords and receiving a failed login message.

When testing to make sure that each user only has access to their specific tools, we can login to a specific user's account and attempt to perform functions that are not normally part of their permitted status. For example, we could have a medical professional attempt to perform a task that only an administrator could perform, like creating a new user. This test can be performed multiple times for each user to make sure that there are no accidental approvals of power that are not allowed in this setting.

ii) *Update Tables*

We will also need to test to make sure that the transferred data is being updated correctly when interacting with the wearables and CRU. Before we start the integration process, we will need to make sure that the code works in a more controlled environment. We performed similar testing during our first semester when proving that our code worked conceptually during the Prototype Progress Review (Assignment #6, Fall 2020).

We can achieve this through entering random data into a user input function on the terminal after calling a specific SQL function (INSERT, DELETE, ALTER, etc.). Updating the table can come in different forms, and due to the finality of our overall database design, for our purposes it is mainly adding, updating, and deleting information from different database tables. These updates can be performed from the command-line or through the HeidiSQL software. These updates can also be automated using triggers.

Triggers can update a specific database table in relation to another action happening on the *same* table. For example, last semester we created a trigger that would remove a patient or medical staff user from the "Infected List" table once their health status was switched from infected to healthy. Our new database we will be recycling that trigger, so we must re-test the trigger by continuously adding sick patients and then changing their status between sick and healthy.

We can check to make sure that the database is properly updating by viewing the database through HeidiSQL and the command line on the CRU. Once we felt confident that our code worked effectively and individually of the wearables, we performed and completed the integration process to handle our live data collection.

Examples of this code can be found in Appendix C (Software) while testing data can be found in Appendix I (Device Testing).

d) Processes Testing

Part of the "processes" code is to be able to update and modify the database through commands. We will be creating functions of code that can pull, update, and push data back into the database. Other processes will include pulling data from the wearable devices when they connect with the CRU, implementing the warning system, and performing the "End of Shift" procedures. Each of these will be tested for accuracy and evaluated after each consecutive attempt for ways to make the source code more efficient.

i) End of Shift Activation

At the end of a "shift" (to be determined by our battery life tests) the wearables will need to perform a data dump and send their collected contact tracing information to the CRU. Then the CRU will sort through the data and push updates to the database. This can be tested through the same methods as seen in the subsection above (Sec. IX.B.ii.c.ii. "Update Tables"), however these tests will need to be modified to better suit our design.

After securing positive results from the "Update Tables" testing, we can then add on different processes to be tested. For the "End of Shift" process, we will need to have the CRU accept data from the wearables. In order to test this before integration we can send large amounts of data through and check the number of new infections and create a list of users that need to be contacted. This is the first step of the warning system activation process.

ii) Warning System Activation

Part of creating an effective warning system with our SMS code is to create a system that can easily activate it when needed. This will be an optional activation that will only happen if there are any recent infections during the most recent shift. In order to properly test this process, we will need to push infected user data to the CRU so that it can properly sort through and create a list of people that the warning system needs to text (via our SMS texting process). We can test the success rate of this by adding or omitting a different number of infected patients so that their contacts can be warned and given instructions as soon as possible. A successful test would include confirmation that a contact of an infected individual received a text message stating that they need to self-isolate until they can receive medical attention. This can be implemented by adding in our own phone numbers or those that have agreed to participate in our testing and getting screenshots of received text messages.

Examples of this code can be found in Appendix C – Software. Testing data and results can be found in Appendix I – Device Testing.

iii. Testing Considerations

During the process of creating our design we have considered looking into and developing parts of our prototype that would normally be included in a modern product. Due to the limitations set on our senior design experience by the pandemic, we lost out on opportunities to explore. Other determining factors include financial and scheduling conflicts. Ideally, we would have wanted to explore the Ingress Protection rating (how water and particle proof our devices could be) and security penetration testing.

Having concrete data on how well our device could work against sweat and other bodily fluids while attached to a patient or medical professional could have been integral in altering other aspects of the design. Considering that the main users would either be medical staff constantly moving around or a potentially sick patient, we would have wanted to ensure that the device could still function if it were to be covered in a wet substance or encountered a particle heavy environment such as dust or sand [17].

Being able to perform penetration testing could have given us information on how we could better secure the sensitive medical data from being exposed to the general public either through malicious intent or accidental leaking. This could have been through more conventional penetration testing by utilizing an ethical hacking environment such as Kali Linux, or through data gathering execute-able files hidden inside an email or a removable hardware device such as a USB drive or SD card.

When we first pitched our design idea to Professor Levine in June 2020, we stated that we wanted to prioritize user privacy to better fall in line with HIPAA security standards [45], stating that we wanted to develop a "Privacy First" methodology to gain user trust and make our prototype more marketable and ethically sound.

iv. Integration Testing

The final part of our testing process was the integration of our accumulated hardware and software elements. We had planed on starting the integration process by the end of March 2021 at the very latest, and were able to complete it by April 25, 2021 which was the day before the prototype was due.

This portion of testing was more difficult as we will not be able to properly debug certain portions of the tests in together in person. Extra precautions need to be made and instructions should be written out to the fullest to ensure that minimal issues arise during this important part of our project.

Hopefully, if we collectively feel like we can given our schedule, we may be able to go back and add in additional features that we originally had to pass over due to time constraints.

This continued session of testing will allow us to acquire the information that we need to continue smoothing out our design and present the best possible version of our deployable prototype at the end of April 2021.

C. Testing Results

i. Hardware Testing

a) Range tests

The range tests were done at varying distances, and the wearables were able to see the CRU at all ranges, the RSSI received is consistent enough to trace contact, which was expected.

Table V:
Transmission Range Test Data [##]

	Transmissions Sent	Transmissions Received	Average RSSI	Pass/Fail
6 ft Clear Los	308	308	-26	Pass
6ft Through Wall	104	104	-43	Pass
50 ft through many walls	191	191	-66	Pass

b) Storage tests

Data was stored on the tags and then read back. This was done in circuit as well as in the breadboard. When a tag was connected, the reading and writing tag successfully received the message, and then recorded the RSSI and tag id.

c) Battery Life Tests

The tests involved were to evaluate the *minimum* amount of time the batteries connected to the tags would run. This was done by programming a sample tag to transmit constantly every three seconds until the battery was unable to power the device. Several different battery types were used:

Table VI:
Battery Testing Data [##]

Battery Type	Battery Life (Minutes)	Pass/Fail
Alkaline (Mananese Dioxide)	98	F
Zinc Choloride	95	F
Lithium	112	P

ii. Software Testing

a) SMS Text Message Alert System Testing

The code available for this feature works as it should. Various tests were conducted simulating 2-tag communication and 3-tag communication. These various tests were not limited to sending alert notifications to phone numbers within the Sacramento area. Tests using phone numbers from out of city and out of state were conducted with text messages being sent successfully to each number. It was made sure that the phone numbers used came from varying carriers, as well as the model of the phone itself being different.

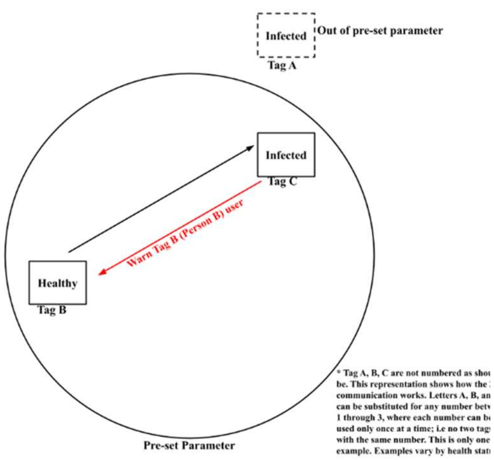

Figure 6: 2-tag Communication Simulation Graph [51]

2-tag communication

	Contact	No contact	Pass/Fail
Person 1: healthy Person 2: healthy	No warning sent.	No warning sent.	Pass | Pass
Person 1: infected Person 2: infected	No warning sent.	No warning sent.	Pass | Pass
Person 1: healthy Person 2: infected	**Person 1 is notified of exposure.**	No warning sent.	Pass | Pass
Person 1: infected Person 2: healthy	**Person 2 is notified of exposure.**	No warning sent.	Pass | Pass
Person 2: healthy Person 3: healthy	No warning is sent.	No warning sent.	Pass | Pass
Person 2: infected Person 3: infected	No warning is sent.	No warning sent.	Pass | Pass
Person 2: healthy Person 3: infected	**Person 2 is notified of exposure.**	No warning sent.	Pass | Pass
Person 2: infected Person 3: healthy	**Person 3 is notified of exposure.**	No warning sent.	Pass | Pass
Person 1: healthy Person 3: healthy	No warning is sent.	No warning sent.	Pass | Pass
Person 1: infected Person 3: infected	No warning is sent.	No warning sent.	Pass | Pass
Person 1: healthy Person 3: infected	**Person 1 is notified of exposure.**	No warning sent.	Pass | Pass
Person 1: infected Person 3: healthy	**Person 3 is notified of exposure.**	No warning sent.	Pass | Pass

Figure 7: 2-tag Communication Results [52]

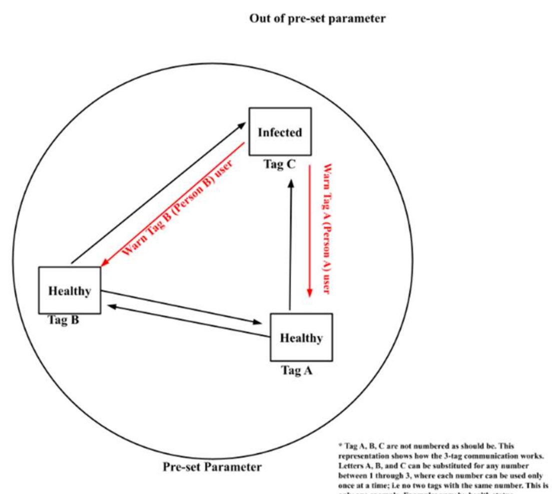

Figure 8: 3-tag Communication Simulation Graph [53]

3 tag communication

	Contact	No Contact	Pass/Fail
Person 1: healthy Person 2: healthy Person 3: healthy	No warning sent.	No warning sent.	Pass | Pass
Person 1: healthy Person 2: healthy Person 3: infected	**Person 1 and Person 2 are notified of exposure.**	No warning sent.	Pass | Pass
Person 1: healthy Person 2: infected Person 3: healthy	**Person 1 and Person 3 are notified of exposure.**	No warning sent.	Pass | Pass
Person 1: infected Person 2: healthy Person 3: healthy	**Person 2 and Person 3 are notified of exposure.**	No warning sent.	Pass | Pass
Person 1: healthy Person 2: infected Person 3: infected	**Person 1 is notified of exposure.**	No warning sent.	Pass | Pass
Person 1: infected Person 2: infected Person 3: healthy	**Person 3 is notified of exposure.**	No warning sent.	Pass | Pass
Person 1: infected Person 2: healthy Person 3: infected	**Person 2 is notified of exposure.**	No warning sent.	Pass | Pass
Person 1: infected Person 2: infected Person 3: infected	No warning sent.	No warning sent.	Pass | Pass

Figure 9: 3-tag Communication Results [54]

For each graph, please refer to their corresponding images; Figure 6 with Figure 7, and Figure 8 with Figure 9. Please note that all passing results were true with the usage of varying phone numbers. In other words, the passing results were true for variations between tests using all local numbers, all out-of-state numbers, all out-of-city numbers, and varying combinations of a mixture of two or three of these numbers; i.e. two local and one out-of-state, one out-of-state and one out-of-city, and so on and forth.

b) Data Encryption Testing

The information in the database is encrypted pre- and post-access. Regardless of whether the user has authorized permission or not, our program already encrypts the information found in our database. An authorized user will insert his or her entry key and the database is triggered to decrypt that encrypted data. On the other hand, when an unauthorized user accesses the database, they are not able to see any decrypted data, i.e., they gain access to the encrypted data, in the worst-case scenario, which is a bunch of jargon. Simultaneously, the admin in charge of the database receives an SMS alert system of this unauthorized access attempt. All test results have passed. These results include data encryption and decryption even after authorized personnel modify the database. New information added into the database will always be encrypted.

Table IVII:
Data Encryption and Decryption Results [55]

	Authorized Individual	Unauthorized Individual	Pass/Fail (Authorized | Unauthorized)
Encryption	Data must encrypt.	Data must encrypt.	Pass | Pass
Decryption	Data must decrypt for usage of authorized person.	Data remains encrypted. SMS warning message dispatched to admin in charge of database.	Pass | Pass

The warning message that is dispatched during an unauthorized attempt incorporates the SMS notification system.

c) Database Testing

The database testing was conducted on both a test CRU and the HeidiSQL GUI. Verifiable tests were achieved by applying changes to the database on either the CRU or GUI and then checking that the changes happened on the other. Triggers were also implemented in order to create a sense of automation for the final prototype. User authentication was tested for security measures while table updates were tested to make sure that data was properly being updated to the database at the CRU and GUI levels.

i) User Authentication

As stated earlier in this document, one of the main reasons why we switched our database management from SQLite to MariaDB is that SQLite did not support user authentication. This posed a problem as user authentication is one of the main security features of our project. In order to properly test this function, we attempted to login through the CRU Python files, the CLI, and the GUI. Unsuccessful tests threw an error message that could potentially lead to a system exit command being executed or the program asking us to try again. A more in depth testing result can be seen in the table in Appendix I at the end of the document.

ii) Update Tables

Performing table updates in MariaDB requires a standard SQL query, but the inclusion of triggers and using custom created Python functions we were able to automate some aspects of updating the tables.

There are a total of five triggers in this database, all of which are designed to create a sense of automation for our potential client users. These triggers are activated after the tags table has been updated in a specific way.

Auto_pbk: Automatically creates an entry in the phone book table whenever a new entry has been added to the tag table.

Add_infct: Automatically adds entries into the infected_list table whenever a status changes from healthy ("h") to infected ("i") in the tags table.

Updt_infct: Whenever a status changes from infected back to healthy in the tags table, that same entry in the infected_list table will change from infected to cleared ("c"). This works alongside the *auto_infct_del* function in the *cru_processes.py* file which is used to remove these entries at the end of the program.

Updt_map: Adds a new entry to the tag_map table whenever a new entry is added to the tags table.

Updt_room: Whenever an entry's room number is changed in the tags table, this change is then reflected on the tag_map table.

d) CRU Processes Testing

The CRU contains four Python files to operate, *cru.py, cru_processes.py*, *cru_sec.*py, and *cru_decrypt.py* where *cru_processes.py* acts as a module for the main file that contains the functions needed for the CRU to operate properly. Each of these functions operates in conjunction with each other depending on the area of the overall data collection process they reside in. The following subsections will contain breakdowns of each function and where they are placed in the CRU processes.

i) Functions

There are several different functions that are called during the CRU's operation. Each of which has a relation to the database itself through data retrieval or manipulation. For this subsection we will briefly breakdown what these functions do.

Main: This is the main function that allows the CRU to properly function from start to finish. It is comprised of several different while loops that help with user decision making and data retrieval.

Logo: This function prints the acronym for the project in asterisks. This is more for presentation purposes than functional purposes.

Sort_convert: This takes lines of input data and breaks it apart into strings with a length of 10. These are then divided, sorted, and converted into their proper data type in order to push this information to the interactions table in the database. This function has some custom error codes and performs a function call of *push_to_inter.* If the data was successfully pushed to the interactions table, it will return a success message.

Push_to_inter: This function takes the recently sorted and converted wearable data and pushes it to the interactions table in the database.

Pull_from_infect: This function is used to pull all of the infected tag IDs from the infected_list table of the database. It returns a list of these infected tag IDs.

Warning_system_protocol: This is the first portion of the "End of Shift" protocol function that is called at the very end of the *main* function. It will compile a list of people to text based off of the contact_list table combined with the infected tag ID list. If there is a list of phone numbers, the SMS API will text those who have been in contact with a recently infected individual the appropriate steps to take regarding their health and the health of the general public.

Auto_infct_del: Due to MariaDB's inability to create triggers that will delete rows from a table, we had to create our own that will cycle through the infected list table and remove any wearable user who has the status "c" for cleared. This will reduce the workload for the program administrator and reduce redundant text messages from being sent out.

ii) Modes

In order to organize the CRU processes better, we implemented two different modes: Automatic Data Entry and Manual Data Entry modes. We were able to get the project to work full y by the prototype evaluation's deadline due to our integration sessions during the later weeks of our project's design schedule. With proper radio frequency communication between the test CRU and wearable devices we are able to produce testing data.

Automatic Data Entry mode involves allowing the CRU to continuously receive radio data from the different wearables over a period of time. While the CRU is collecting data, it will periodically send that data to be sorted and sent to the database. When this happens a message will appear on screen confirming a successful data transfer. This process can continue on for several minutes or longer based on what the counter variable is set to. Once the time is up, the user will receive an on-screen prompt asking if they would like to continue receiving data or start the end of shift protocol.

Manual mode is the mode that allows the user to manually enter in wearable data almost directly to the database. Much like Automatic Mode, it calls the *sort*_convert function, and after making this call it will ask the user if they want to continue manually entering data until the user selects "No". At this point the device will continue to the "End of Shift" protocol.

D. Conclusion

Refer to Section XI for the conclusion of this section.

X. MARKETABILITY FORECAST

A. *Target Audience/Market*

Based upon the nature of the product and the Societal Problem (Contact Tracing) we selected, the market for the product itself will largely be focused on the healthcare industry or government funded healthcare organizations, with emphasis on the enclosed spaces within Hospitals. For example, this product could be used effectively by the American Red Cross, UNICEF, Doctors Without Borders, etc. The product itself involves two separate marketable products: an RFID-related proximity detection system, and the data given by the physical system that is tailored towards use in Contact Tracing. The end goal of the system is to give Medical experts and statisticians access to a *secure* body of real-time proximity data, instead of requiring Contact Tracing agents be deployed to the site to collect that data.

B. SWOT Analysis

SWOT Analysis is a marketing outline strategy that allows for organizations to internally and externally analyze their product. Internal inspection refers to the strengths (S) and weaknesses (W) of the product, while external inspection refers to the opportunities (O) and threats (T). The following subsections will expand upon our SWOT analysis for our project.

i. Strengths and Weaknesses

An even balanced team of engineers gives a great advantage to the development of a product. What one member may lack in terms of skills, another may best work with those same set of skills. By having members from two different majors, we can help and complement each other's work with the different skill sets we each possess. This balanced team allows for a broader perspective on how to correct issues on our product if any arise while working in its completion. We have those who have the ability to help with software issues and are better at working and fixing in that area, while the other members have a better ability and understanding in fixing the hardware portion. We are able to share our ideas and provide information from different perspectives so that our product can work to its best potential.

Our project requires the insight of people who work in medical settings. These individuals would help us improve our product, if needed, by voicing out the strengths and weakness of our project. Fortunately, a couple of us have a few connections with people who can give us intel. The intel we can gather through our connections will allow for us to make any future modifications to our project that could lead it into the creation of a product our intended audience would want to invest on.

Having an even balanced team, as well as relevant connections, are some strengths that we have. However, besides strengths, we also possess weaknesses that prevent us from creating an 'ideal' product. For example, we lack major funds. Without the necessary funds, we are unable to purchase the items we had originally planned due to high shipping costs, or the costs of the products itself. We had to find alternative parts that would still allow for the functionality of our product, while also making sure we stuck with a set budget. The lack of market experience is another weakness that we have. We had to do our own research about how we would make our product an item that would appeal to our intended audience. If we aren't meticulous with our research, we won't have the slightest clue as towards how to make our project a product of need.

Despite our team working hard to achieve the completion of this project, we each have other academic courses that we need to complete. We are all putting as much time as we can into this project to create a product of quality, but because we aren't giving it the amount of time that we would like, we lack quick and constant progress. We probably could have tested the working portions much more than we have already, but due to the current situation, we have limited ourselves from having any physical contact with one another. We are having to test portions of our product remotely and have had to contact each other via virtual meetings to help each other with issues that arise. The pandemic has kept us limited on resources and away from advanced testing equipment. Besides the lack of funds, as already mentioned, the pandemic has also prevented us from buying products due to the excessive amount of wait time of product arrival.

ii. Opportunities and Threats

Opportunities consist of external influences that create a better case for the design and use of our product. Our project design is highly modular and due to the high availability of the parts that we are using in our prototype (See Appendix B - Hardware for more information and a parts list), it is easy for us to replace or repair parts of the wearable devices. This reduces the amount of downtime required to get the system back up and running again as well as increase the number of organizations adopting these devices due to the ease of use.

Another opportunity is the overall low cost of these modular parts. This will also take away from the amount of downtime that a normal product would create if there is any form of internal failure.

Due to our design of a closed system for medical information sharing, our project has a lower chance of having personal information leaked to the general public. This creates a unique advantage as our wearable devices will not be allowed to leave the medical setting that they are deployed to. The tags can only transmit data correctly if they are within an environment that also contains other tags and a CRU. This creates a closed system of information that has a lower risk of a security breach from outside sources if one device is removed as the only information that is being exchanged between the two tags are the tag identification numbers, time of interaction(s), and the RSSI values for each interaction. No personal identification information such as name, address, or contact information will be exchanged between tags.

Our final opportunity is the rapid adoption of new technology by the FDA to help combat the current COVID-19 pandemic [48]. Due to the increased threat of infection, the FDA has started the emergency rapid approval of different medical devices and mitigation techniques to help prevent the growth of infection rates inside the United States. Other health organizations around the world are also adopting these emergency measures as well.

Threats are characterized as any negative outside influence that could potentially take business away from our product, including competitors in the medical technology industry. Our biggest competitor is the exposure notification application that was developed and released jointly by Apple and Google in April 2020 [49]. The incredible amount of funding and resources at their individual disposal is enough to remove any minor competition out of their path, but to have these two tech giants combine forces creates an enormous obstacle for smaller projects and companies to overcome. The instant availability of their application on all iOS and Android based smartphones creates a built-in market for these companies to utilize.

The other main threat to our project is the general public's willingness to cooperate with the product guidelines. As reported in Section II, a community's adoption of emergency medical regulations does depend largely on a government's readiness and ability to enforce emergency public health ordinances. There are also instances where some lower- and middle-income communities do not have the financial needs or proper administration to accommodate major medical advances. Politically motivated groups may also persuade government health officials to adopt other mitigation methods over the others, regardless of how effective they may be in order to take advantage of a global health crisis for financial gain [50].

C. Cost

The cost of a prototype tag at this point is around $15. Final values will be closer to $10, as if production was increased unit prices would be much lower. On the software side of things, most of the software used is open source. MariaDB and Twilio do have subscription options, but for the prototype the subscriptions were not needed. When the product is deployed, there will be a need for someone to manage the devices, likely an IT professional.

D. Market Growth

The market growth at the time of writing is huge. Due to the Covid-19 pandemic, the need for contact tracing is huge, and the FDA is approving more medical devices. When the pandemic is over, contact tracing will not go away, but its growth will diminish. Our devices can be used for nearly any form of infectious disease contact tracing, so while the growth will slow, the need will never go away.

E. Conclusion

The conclusion for this section can be found in Section XI.

XI. CONCLUSION

At the time of this document being completed, the project prototype has been fully integrated and tested to meet our feature set standards. As discussed in Section II, our societal problem is based around a lack of preventative infrastructure that is needed to properly track a spreading disease. Over the course of the past two semesters, we had been documenting our progress towards our modified contact tracing system of radio frequency based wearable sensors and a receiver that will continually add sensitive medical data into a database to properly track an infectious disease in a close proximity setting.

As discussed in the Design Idea section of this report (Section III), our overall design is centered around a system of radio frequency based wearable sensors and a central receiver. This design will allow for medical staff to perform contact tracing in just about any setting provided they have enough data storage and wearable devices to cover both staff and patients. The software parameters can be adjusted by trained medical personnel or a CRU technician to allow for better tracking of a specific illness. This software also includes encryption and a SMS text messaging system to allow us to put privacy first for both patients and the hospital staff that are saving lives.

Specific hardware and software information is discussed in Appendix B through Appendix D.

A. Societal Problem

Team 9 had selected to focus on the lack of preventative measures leading up to global health crises as our societal problem. While large amounts of preventive measures do exist through the CDC, WHO, and NIAID websites, these all focus on influenza-based pandemics, with very little mention of other forms of illnesses. We see this as a serious problem as there are still diseases out there that could have the potential to become epidemics or pandemics. Focusing on one type of illness should not be viewed as a successful mitigation plan.

At the time of this report's release the COVID-19 pandemic has had several devastating peaks in cases that have left a large amount of smaller underfunded communities with overloaded contact tracers. These public servants are so overwhelmed to the point where some local health ordinances are relying on the general public to do their own contact tracing [38]. This creates problems regarding medical data being skewed and corrupted due to contact tracing being performed by the general public and not professionals. This reiterates our need to create a contact tracing system that can help alleviate the burden of smaller communities.

During the creation of this project there was a major shift in political control of the United States government that brought in a new administration. Through this new administration, the President of the United States established executive orders that help further scientific needs to create new forms of preventative infrastructure [37].

B. Design Idea

We have defined the features and provided a general, high-level outline of how our design idea will function and how it was implemented. This project design idea contains a series of wearable radio frequency devices that will communicate with each other and a central receiving unit in order to maintain a database for contact tracing in a limited area. The goal is to create a system that is cost effective, adaptable to different environments, privacy focused, scalable, and portable.

This design implements multiple communication processes such as: wearable-to-wearable interactions, wearable-to-receiver interactions, and receiver to database. There are two main modes of data collection and there are end of shift and warning system activation protocols set into place. Each of these are achieved through our hardware and software designs that were successfully implemented, executed, and refined over the better part of a year.

After reviewing our design idea between the first and second semester, we feel that we have established a strong prototype that could very well succeed outside of our learning environment.

For more specific information on the hardware and software that we chose to use for our project, Appendix B (Hardware) and C (Software) are available to view. These appendices will include pin-outs, datasheets, source code, and flowcharts to help us explain exactly how our prototype devices will work.

C. Device Test Plan

Team 9 has thoroughly discussed and planned what aspects of the project will be tested. We have divided our project into phases, or stages, where each portion is critically examined to make sure each of these are working properly before integrating them. All testing data can be found in Appendix I while explanations behind our testing can be found in Section IX of this report.

D. Funding

As stated earlier in the report, this was a student funded project that did not have any outside sponsorship involved. We as a group decided early on to evenly divide up the cost of the project at the end of the Spring 2021 semester.

We were able to keep our project below a quarter of the average Senior Design budget of $1000.00 USD.

E. Project Timeline

Just like the Work Breakdown Structure, the Project Timeline presents the project progress. However, the timeline gives more details about said progress by providing the expected timeframe of completion for each task that makes up the whole design. As a team, we set completion dates for ourselves to motivate us to work harder and complete the project in a timely manner without the necessity of worrying that we're falling behind.

F. Work Breakdown Structure

With this having said, Team 9 has built this work breakdown structure to make sure the project progresses is completed by the expected time. Each member focused on their tasks, while making sure that every task is completed within the planned timeframe. Although tentative, Team 9 tried their best to make sure their work matched with the Work Breakdown Structure and Timeline as much as possible.

This structure was also used for Team 9 to prepare themselves ahead of time, with the future tasks that are to be completed. If any issues were to arise that could cause a discrepancy in our planned schedule, the Work Breakdown Structure would allow Team 9 to physically see where changes should be made to make sure the completion of the project is finished on time.

G. Risk Assessment

After reviewing our laboratory prototype, we have developed a list of risks that could impact the production and progress of our project. This list is based off the probability of an event happening and the intensity of the impact it could have on our project. Some of these risks include environmental factors or events that are considered impossible to prepare for (COVID-19, wildfires, family emergency, etc.), other risks include hardware or software failures that could either be a minor annoyance (infinite loop in a software execution) or cause serious damage (batter puncture) or bodily harm if not properly addressed.

H. Design Philosophy

Through our research we were able to design a product that we could stand behind due to our devotion to maintaining user privacy and creating a cost effective system of devices that any one can use or afford. We are engineering students who set out to help those in need and we feel that we have succeeded by creating an almost universal medical monitoring device.

I. Deployable Prototype Status

With our project protype completed, we are well on our way to completing our Senior Design experience.

J. Testing Results Report

After two months of rigorous testing, we have each tested our individual feature sets and are now swiftly moving on towards the end of our prototype's design cycle. The next steps involve fully integrating all of our parts together and performing the final tests to make sure that our individual features still work according to our initial testing results.

K. Market Review

After careful consideration of our project's marketability, we have concluded that our product has a high change of becoming a successful device based on the current conditions with the onset of mitigation approval from global health organizations. We have come to this conclusion based on our internal analysis through the SWOT (Strengths, Weaknesses, Opportunities, Threats) method and from what market research we were able to compare and apply to our unique project.

With large technology corporations Apple and Google being our main competitors, we were able to design a closed contact tracing system that significantly decreases the probability of a security breach from misplaced or stolen wearable devices or malicious intent via hacking. While Apple and Google have created the exposure notification application for their fleet of smartphones, we have developed a collection of devices that can ease consumer doubts about confidential information being exposed to the wrong community.

While we ultimately want our project to no longer have any use globally due to the eradication of infectious diseases, we are confident that our project could be adopted for emergency health organizations for helping gain the needed contact tracing data.

L. Conclusion

As we enter the end of the documentation, all that is left to review is the References, Glossary, and the Appendices. For the past eleven months we have put a lot of time and effort into creating this document and the prototype to go along with it. As we conclude our Senior Design project and step into our careers after graduating, we plan to bring all that we have learned from this experience with us.

To review, we are Team 9, also known as "Team 9 Lives", from the College of Engineering & Computer Science at California State University Sacramento (CSUS). We are a perfectly balanced team comprised of two CpE and two EEE students. Our project was centered around the idea of creating a novel way of digital contact tracing to help mitigate the ongoing efforts of professional contact tracers. Our radio frequency contact tracing system (R.F.C.T.S.) has been rigorously tested to make sure that it matches our feature set (Section VI and IX) and we have maintained this living document to make sure that all progress was recorded.

This project was created during a global pandemic and primarily over the Zoom video conference service with minimal physical interaction in order to maintain proper social distancing guidelines as requested by the course professors. We were able to physically meet up over two days at the end of April 2021 in order to complete the integration process of the project.

We would like to thank our laboratory instructor and mentor Professor Levine, our lecture instructors Professor Tatro and Dr. Cottle, and our families for the support over the entire Senior Design project life cycle.

REFERENCES

[1] Center for Disease Control and Prevention. "Past Pandemics." cdc.gov. https://www.cdc.gov/flu/pandemic-resources/basics/past-pandemics.html. (accessed Sep. 10, 2020).

[2] Ada WC Yan et al. "The Distribution of the Time Taken for an Epidemic to Spread Between Two Communities". In: *Mathematical Biosciences* 303 (2018), pp. 139-147.

[3] Patrick Saunders-Hastings et al. "Effectiveness of Personal Protective Measures in Reducing Pandemic Influenza Transmission: A Systematic Review and Meta-Analysis". In: *Epidemics* 20 (2017), pp 1-20.

[4] Barbara Jester, Timothy Uyeki, and Daniel Jernigan. "Readiness for Responding to a Severe Pandemic 100 Years After 1918". In: *American Journal of Epidemiology* 187.12 (2018), pp. 2596-2602.

[5] Jeffery Shaman. "Pandemic Preparedness and Forecast". In: *Nature Microbiology* 3.3 (2018), pp. 265-267.

[6] Narges Montazeri Shahtori et al. "Quantifying the Impact of Early-Stage Contact Tracing on Controlling Ebola Diffusion". In: *Mathematical Biosciences & Engineering* 15.5 (2018), p. 1165.

[7] Lisa O Danquah et al. "Use of a Mobile Application for Ebola Contact Tracing and Monitoring in Northern Sierra Leone: A Proof-of-Concept Study". In: *BMC Infectious Diseases* 19.1 (2019), p. 810.

[8] Vicki Stover Hertzberg et al. "Behaviors, Movements, and Transmission of Droplet-Mediated Respiratory Diseases During Transcontinental Airline Flights". In: *Proceedings of the National Academy of Sciences* 115.14 (2018), pp. 3623-3627.

[9] Lorenzo Isella et al. "Close Encounters in a Pediatric Ward: Measuring Face-to-Face Proximity and Mixing Patterns with Wearable Sensors". In: *PloS One* 6.2 (2011), e17144.

[10] "pandemic" *A Dictionary of Epidemiology*, edited by Miquel Porta, Oxford University Press, Incorporated, 2014. ProQuest Ebook Central, https://ebookcentral.proquest.com/lib/csus/detail.action?docID=1679277.

[11] National Institute of Allergy and Infectious Disease. "Influenza." niaid.nih.gov. https://www.niaid.nih.gov/diseases-conditions/influenza . (accessed Sep. 26, 2020).

[12] "epidemic" *A Dictionary of Epidemiology*, edited by Miquel Porta, Oxford University Press, Incorporated, 2014. ProQuest Ebook Central, https://ebookcentral.proquest.com/lib/csus/detail.action?docID=1679277.

[13] Center for Disease Control and Prevention. "Influenza in Animals." cdc.gov. https://www.cdc.gov/flu/other/index.html (accessed Sep. 26, 2020).

[14] Center for Disease Control and Prevention. "First Global Estimate of 2009 H1N1 Pandemic Mortality Released by CDC-Led Collaboration." cdc.gov. https://www.cdc.gov/flu/spotlights/pandemic-global-estimates.htm (accessed Sep. 27, 2020).

[15] Google. "COVID-19 Map." google.com.

https://news.google.com/covid19/map?hl=en-US&gl=US&ceid=US%3Aen.(accessed May 2, 2021).

[16] Stimpfel, Amy Witkoski, Douglas M. Sloane, and Linda H. Aiken. "The Longer the Shifts for Hospital Nurses, the Higher the Levels of Burnout and Patient Dissatisfaction." *Health affairs* 31.11 (2012): 2501-2509.

[17] Steven Schriber. "Ingress Protection: Where and Why It Matters". mouser.com. https://www.mouser.com/applications/ingress-protection-matters/. (accessed Oct. 3, 2020)

[18] Suciu, Dan. "SQL on an Encrypted Database." *Communications of the ACM*, vol. 55, no. 9, 2012, pp. 102–102., doi:10.1145/2330667.2330690.

[19] Apple & Google. "Exposure Notification - Bluetooth Specification". Apple.com. https://covid19-static.cdn-apple.com/applications/covid19/current/static/contact-tracing/pdf/ExposureNotification-BluetoothSpecificationv1.2.pdf?1. (accessed Oct. 3, 2020)

[20] Center for Disease Control and Prevention. "Nonpharmaceutical Interventions (NPIs)". Cdc.gov. https://www.cdc.gov/nonpharmaceutical-interventions/index.html. (accessed Oct. 4, 2020)

[21] Center for Disease Control and Prevention. "Respiratory Hygiene/Cough Etiquette in Healthcare Settings". Cdc.gov. https://www.cdc.gov/flu/professionals/infectioncontrol/resphygiene.html. (accessed Oct. 4, 2020)

[22] Y. Saavedra. "Resume". Oct. 5, 2020.

[23] C. Moncrief. "Resume". Oct. 5, 2020.

[24] A. Sharp. "Resume". Oct. 5, 2020.

[25] M. Allen. "Resume". Updated May. 3, 2021.

[26] M. Allen. "MedStaff vs Patient Database Outline". Oct. 5, 2020.

[27] Y. Saavedra, M. Allen. "Patient Information Database Outline". Oct. 5, 2020.

[28] Y. Saavedra, M. Allen. "Medical Staff Information Database Outline". Oct. 5, 2020.

[29] A. Sharp. "Block Diagram of Prototype Features". Oct. 12, 2020.

[30] C. Moncrief, A. Sharp, Y. Saavedra, M. Allen. "Project Expenses Table". Oct. 26, 2020.

[31] Y. Saavedra, A. Sharp, C. Moncrief, M. Allen. "WBS Table". Oct. 23, 2020.

[32] A. Sharp. "Project Timeline Graphic." Nov. 2, 2020.

[33] A. Sharp, C. Moncrief, M. Allen, Y. Saavedra. "Risk Matrix". Nov. 6, 2020.

[35] Apple & Google. "Exposure Notification – Frequently Asked Questions". Apple.com. Exposure Notifications - FAQ v1.2 FINALbetter (cdn-apple.com). Sep. 2020. (accessed Jan. 31, 2021)

[36] Center for Disease Control and Prevention. "Guidelines for the Implementation and Use of Digital Tools to Augment Traditional Contact Tracing". cdc.gov/coronavirus. https://www.cdc.gov/coronavirus/2019-ncov/downloads/php/guidelines-digital-tools-contact-tracing.pdf. Dec. 15, 2020. (accessed Jan. 31, 2021)

[37] Federal Register. "2021 Joe Biden Executive Orders". federalregister.gov. https://www.federalregister.gov/presidential-documents/executive-orders/joe-biden/2021. (accessed Jan. 31, 2021)

[38] B. Dahlberg. "Do-It-Yourself Contact Tracing Is A 'Last Resort' In Communities Besieged By COVID-19". *NPR*, (2020). Available:

https://www.npr.org/sections/health-shots/2020/12/27/949246298/do-it-yourself-contact-tracing-is-a-last-resort-in-communities-besieged-by-covid. (Accessed Jan. 31, 2021).

[39] R. Treisman. “The Vaccine Rollout Will Take Time. Here’s What The U.S. Can Do Now to Save Lives”. *NRP, (2020)*. Available: https://www.npr.org/sections/health-shots/2021/01/23/959786884/the-vaccine-rollout-will-take-time-heres-what-the-u-s-can-do-now-to-save-lives . (Accessed Jan. 31, 2021).

[40] Moghadas, Seyed M., et al. "The Implications of Silent Transmission for the Control of COVID-19 Outbreaks." *Proceedings of the National Academy of Sciences* 117.30 (2020): 17513-17515.

[41] Ferretti, Luca, et al. "Quantifying Dynamics of SARS-CoV-2 Transmission Suggests that Epidemic Control and Avoidance is Feasible Through Instantaneous Digital Contact Tracing." *medRxiv* (2020)

[42] Kelbert, Mark, Igor Sazonov, and Michael B. Gravenor. "Critical Reaction Time During a Disease Outbreak." *Ecological Complexity* 8.4 (2011): 326-335.

[43] M. Allen, Y. Saavedra, A. Sharp, C. Moncrief. “Project Timeline Table”. Feb. 1, 2021.

[44] M.Allen, Y. Saavedra, A. Sharp, C. Moncrief. “Device Test Plan Table”. Feb. 1, 2021.

[45] United States Department of Health and Human Services. “The Security Rule”. hhs.gov. https://www.hhs.gov/hipaa/for-professionals/security/index.html. (Accessed Feb. 8, 2021).

[46] *RFM69HCW ISM Transceiver Module v1.1*, HOPERF Electronic, XiLi Town, Nanshan District, Shenzhen, Guangdong, China, 2006. https://cdn-shop.adafruit.com/product-files/3076/RFM69HCW-V1.1.pdf (Accessed Feb. 8, 2021)

[47] Adafruit, “Arduino Wiring”, April 15, 2016. https://learn.adafruit.com/adafruit-rfm69hcw-and-rfm96-rfm95-rfm98-lora-packet-padio-breakouts/arduino-wiring (Accessed Feb. 8, 2021)

[48] United States Food and Drug Administration. “FDA Combating COVID-19 WITH MEDICAL DEVICES”. fda.gov. https://www.feda.gov/media/136702/download (Accessed Feb. 28, 2021)

[49] Apple and Google. “Exposure Notification: Using technology to help public health authorities fight COVID-19”. google.com. https://www.google.com/covid19/exposurenotifications/ . (Accessed Feb. 28, 2021).

[50] Diaconu, Karin, et al. "Methods for medical device and equipment procurement and prioritization within low-and middle-income countries: findings of a systematic literature review." *Globalization and health* 13.1 (2017): 1-16.

[51] Y. Saavedra. “2-tag Communication Simulation Graph”. Apr. 4, 2021.

[52] Y. Saavedra. “2-tag Communication Results”. Apr. 4, 2021.

[53] Y. Saavedra. “3-tag Communication Simulation Graph”. Apr. 4, 2021.

[54] Y. Saavedra. “3-tag Communication Results”. Apr. 4, 2021.

[55] Y. Saavedra. “Data Encryption and Decryption Results”. Apr. 4, 2021.

[56] M. Allen. “Database Relation Graph”. April 18, 2021.

[57] M. Allen. “Wearable Tag Hardware Diagram”. February 2, 2021.

[58] M. Allen. “CRU Hardware Diagram”. February 2, 2021.

[59] A. Sharp. “Wearable Schematic”. May 2, 2021.

[60] A. Sharp. “3D Model of a Wearable”. May 2, 2021.

[61] A. Sharp. “Wearable Code”. 2021

[62] M. Allen. “data_pull source code”. May 3, 2021.

[63] M. Allen. “HeidiSQL Platform”. April 2021.

[64] M. Allen. “CRU High-Level Flowchart.” April 2021.

[65] C. Moncrief, M. Allen, Y. Saavedra, A. Sharp. “W2W, W2C, CRU2DB”. April 25, 2021

[66] M. Allen. “Database Triggers Source Code”. May 3, 2021.

[67] A. Sharp. “Wearable Device with Casing”. Apr. 25, 2021

[68] A. Sharp. “Wearable Device Casing Clip”. Apr. 25, 2021

[69] M. Allen. “Table Transfer Partial Source Code”. May 3, 2021.

[70] C. Moncrief, M. Allen, Y. Saavedra, A. Sharp. “Hours Summary Table:. May 3, 2021

GLOSSARY

A

API: *Application Programming Interface*

B

BLE: *Bluetooth Low Energy*

C

CDC: *Center for Disease Control and Prevention*.

Contact Tracing: *A method of tracking all known contacts of a person or persons who have been infected with a highly contagious disease.*

CLI: *Command-Line Interface*

CpE: *Computer Engineering*

CRU: *Central Receiver Unit*

CSUS: *California State University Sacramento*

E

EEE: *Electrical & Electronics Engineering*

Epidemic: *The occurrence in a community or region of cases of an illness, specific health-related behavior, or other health-related events clearly in excess of normal expectancy* [12].

F

FDA: *United States Food and Drug Administration*

G

GISRS: *Global Influenza Surveillance and Response System*

GUI: *Graphical User Interface*

H

HHS: *United States Department of Health and Human Services*

HIPAA: *Health Insurance Portability and Accountability Act.*

I

IDE: *Integrated Development Environment*

IEC: *International Electrotechnical Commission.*

Influenza: *Also known as the flu. A contagious respiratory infection caused by several flu viruses that infect the nose, throat, and lungs. Appears annually as seasonal influenza* [11].

IP: *Ingress Protection Code. The rating system dedicated to determining how secure* *a device is from moisture or particle infiltration. Established by the IEC.*

<u>M</u>

MCU: *Microcontroller Unit*

mRNA: *Messenger RNA*

<u>N</u>

NIAID: *National Institute of Allergy and Infectious Diseases*

NHC: *National Hurricane Center*

NPI: *Non-Pharmaceutical Intervention*.

<u>P</u>

Pandemic: *An epidemic occurring over a very wide area, crossing international boundaries, and usually affecting a large number of people* [10].

<u>R</u>

RF: *Radio Frequency*

RFID: *Radio Frequency Identification.*

<u>S</u>

SARS: *Sudden Acute Respiratory Syndrome.*

SDK: *Software Development Kit*

SMS: *Short Message Service.*

<u>W</u>

WBS: *Work Breakdown Structure*

WHO: *World Health Organizat*

APPENDIX A: USER MANUAL

This page is left blank.

Overview

R.F.C.T.S. is a contact tracing system currently consisting of three wearable devices (tags) and a central processing unit (CRU). Each wearable device contains a transceiver that sends and receives data between the other wearables and the CRU. All devices are equipped with security for the sake of user privacy, and an alert system to notify healthy tag users of exposure to an infected user.

Features

- Rechargeable battery
- Flexible SMS alert system: Ability to notify users regardless of location (within USA only!), cellphone carrier, cellphone type, area codes, etc.
- Portability
- Encrypted database: Ensures users' information privacy

Instructions

After receiving the devices, make sure to charge the wearables. The batteries should be fully charged after an hour of charging and typically last between seven and seven and a half hours. Be sure to place each tag on a charger when leaving your department.

In order to being the data collection process, make sure that the CRU is powered on and the program has been started. To start the program, open a terminal window and type in the following command:

```
Python3 cru.py
```

The program will display the logo and ask for your login credentials. If the wrong credentials are entered, the Administrator will receive a warning text about an unauthorized or failed login. Depending on your administrator's protocol, you may need to wait to login again.

Once the correct credentials are entered, you will then be prompted to choose a mode of data collection: Automatic mode or Manual mode.

Automatic Mode:

With the tags turned on, data will be periodically sent from the tags to each other before sending data to each other before sending their collected data to the CRU. If you wish to change the interval between CRU to database communication type in the following command:

```
nano cru_processes.py
```

From here, navigate to the *data_pull(conn,curs)* function and change the value of the *cnt_down* variable. When performing tests, we found that the default value creates a cycle that lasts roughly five minutes.

```
## DATA_PULL(RFM69, CONN, CURS)
 # pulls data from wearable devices
def data_pull(conn, curs):
 ser = serial.Serial('/dev/ttyACM0', 57600, timeout = 0.5)
 ser.flush()
 cnt_down = 10

 while cnt_down != 0:
  read_serial = ser.readline().decode('utf-8').rstrip()
  s = read_serial

  print(s) # used for demo, do not print in final version

  if len(s) > 6:
    s = read_serial
    sort_convert(s, conn, curs)
    cnt_down = cnt_down - 1
```

Figure A1: data_pull source code [62]

Successful data transfers will display on the terminal. Once the cycle is over you will be prompted to either continue accepting wearable data or move onto the End of Shift protocols.

Manual Mode:

Manual mode consists of a CRU technician manually inserting data for the CRU to process and transfer directly to the *interactions* table in the database. When entering this data, be sure to entering string values of at least 10, or else the *sort_convert* function will not process the data properly. This function does have some error control included and will display an error if one is made.

Much like Automatic Mode, Manual Mode will ask you after each entry if you want to continue entering data. If not, you will be taken to the End of Shift Protocol.

End of Shift Protocol:

During this phase, the CRU prepares the list of possible text messaging candidates to warn them about their possible contamination. It is important to maintain the correct phone numbers in the system by updating the *contact_list* table in the database. Tags registerd as the *user_tag* are actual tag users, while the *con_tag* column contains contacts of the user tag. Be sure to update this as much as you can either through the command line or via HeidiSQL. Non-user contacts get the *con_tag* of 0.

The End of Shift protocol also encrypts all database information at the end of the shift. This can be accessed via the file folder system. Look for a folder titled ‘rfcts-csv’.

At the end of each shift, the system starts over and will ask you for your user login again.

Decrypting the CSV File:

If data needs to be reviewed, running the *cru_decrypt.py* program will decrypt the .csv file. To activate this program, enter the following command in the terminal:

```
Python3 cru_decrypt.py
```

You will then be prompted to enter your R.F.C.T.S. login information. If your login information is incorrect, a Admin will be notified. So be sure to use the correct login credentials.

After a successful login, you will see a "DECRYPTION SUCCESSFUL" message populate on the screen. This means that the .csv file was successfully decrypted. You can double check this by clicking on the file inside the 'rfcts-csv' folder.

HeidiSQL:

Much like every other login for this system, CRUs can be accessed through the HeidiSQL software by using your R.F.C.T.S. login credentials. If you enter the wrong information, you will be asked to try again.

Successful entry will allow the user to view and manipulate the database information as they see fit, but only if they have the right credentials.

It is important when manipulating data in HeidiSQL that you be careful as to what you're editing. The tags table is the main table for the database and can create events that cause one of the five triggers to start manipulating the other tables.

Also, the Person_Type and Contact_List tables do not update on their own, so it is up to the administrator or a registered medical staff personnel to enter in the required data for those tables.

Shown below in Figure A2 is a screen shot of the side menu for HeidiSQL.

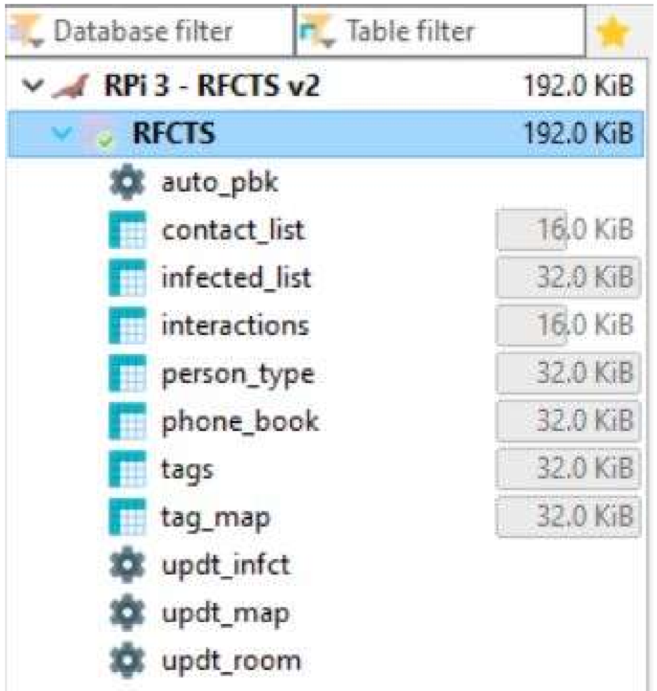

Figure A2: HeidiSQL Platform [63]

APPENDIX B: HARDWARE

Shown below are hardware diagrams and schematics for the wearables and CRU devices.

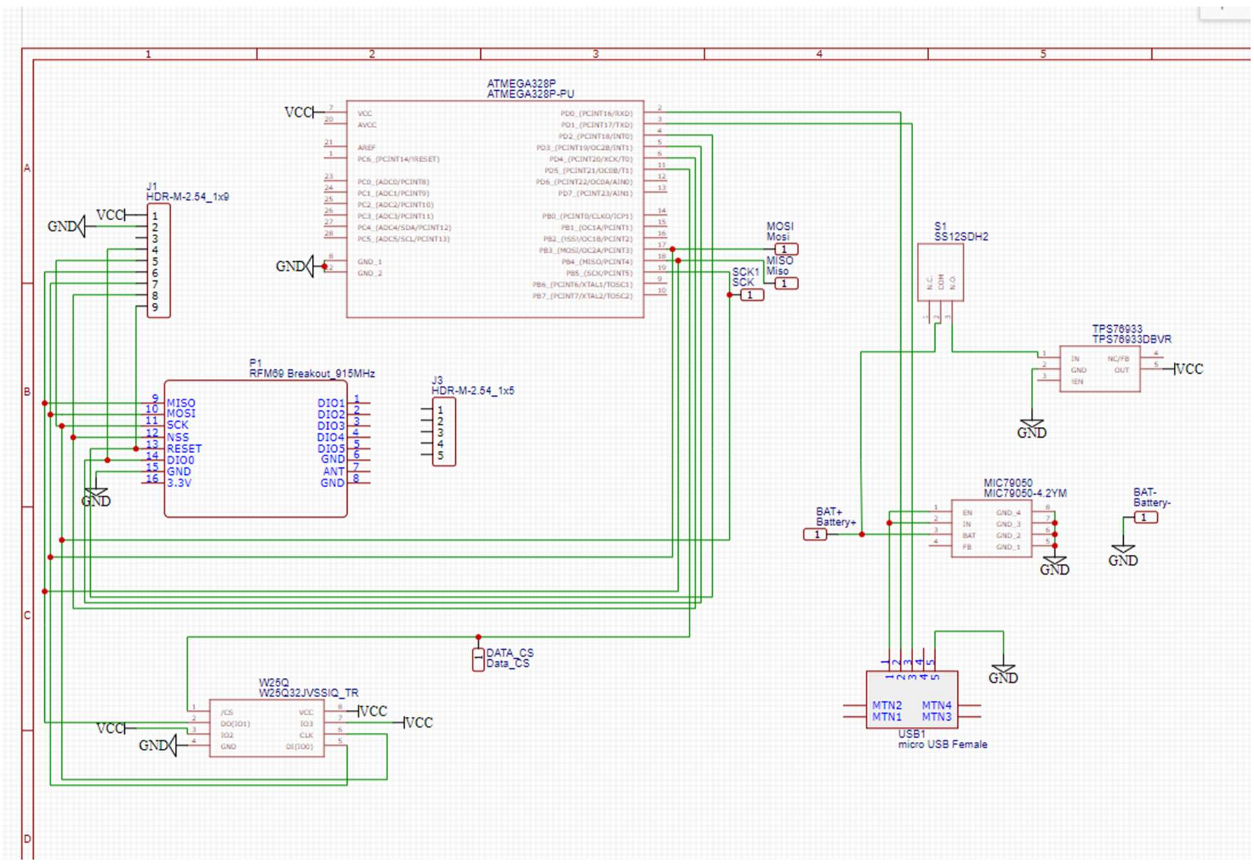

Figure B1. Wearable Schematic [59]

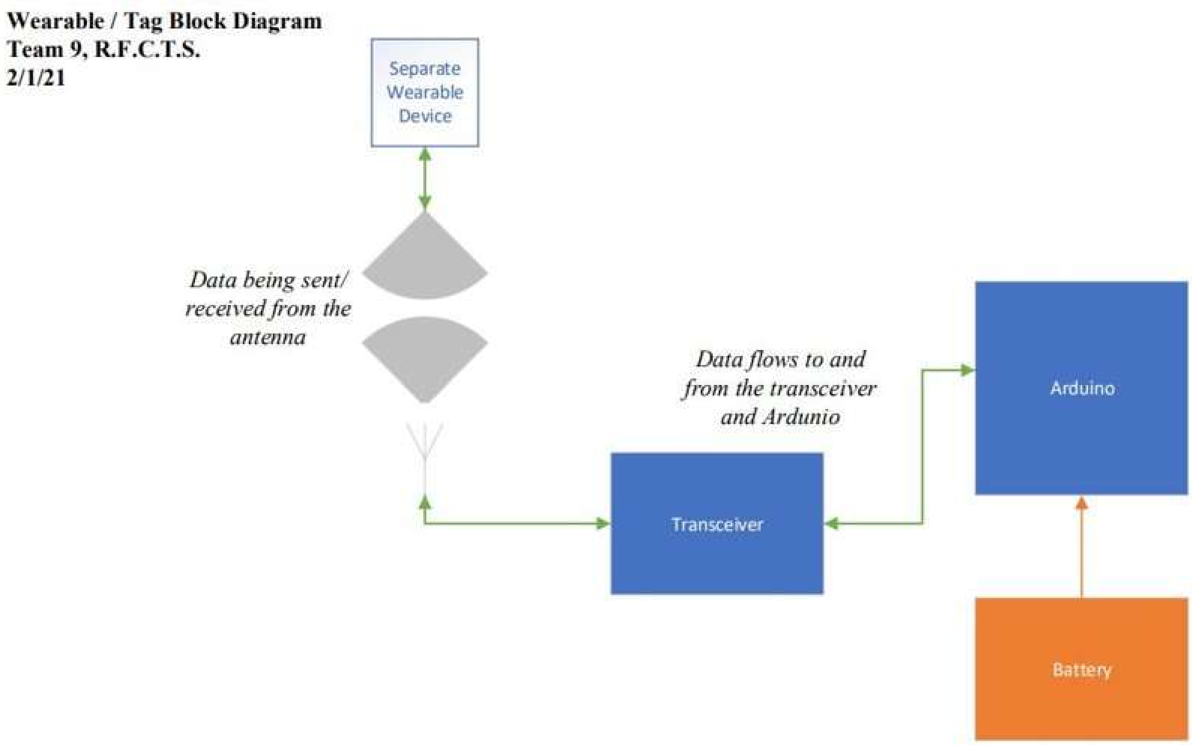

Figure B2. Wearable Tag Hardware Diagram [57]

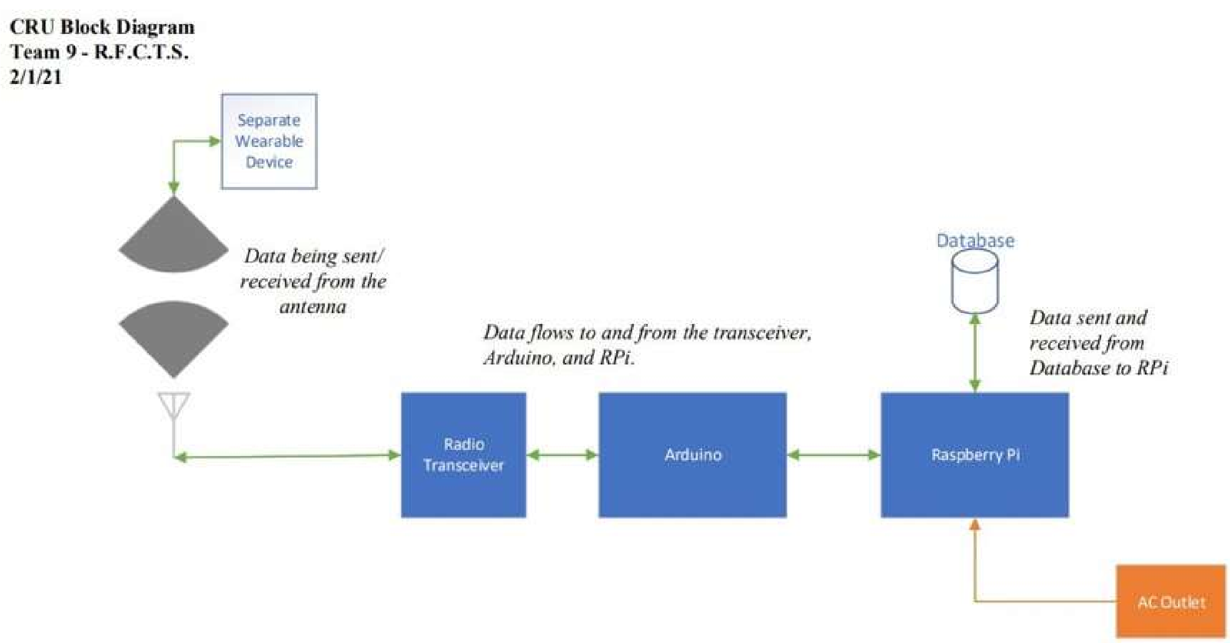

Figure B3. CRU Hardware Diagram [58]

APPENDIX C: SOFTWARE

The software involved in the production of this project can be broken down into several different categories: Wearables, SMS Text Messaging, Data Encryption, CRU Processes, and Database. This appendix will display diagrams, flowcharts, and screen shots of our code and software implementation.

Wearables

The code for the wearables is written in C++. Wearables transmit and receive on a cycle, so they receive a message from other wearables, then record the rssi of the last message. In figure C1, it is time for Tag 1 to save the strength of the message from Tag 3.

```
// Store Radio 3s RSSI and Tag id To send when it is time
if (rf69.waitAvailableTimeout(200)) {
  int Q = 0;
  for (; index < indexFinal + 6; index++) {
    DataToSend[index] = Time[Q];
    Q++;
  }
  indexFinal = index;
  if (rf69.recv(buf, &len)) {
    Serial.print("Radio 3 Rssi: ");  Serial.println(rf69.lastRssi());
    LastRssi = rf69.lastRssi();
    LastRssi = abs(LastRssi);

    LastRssiDigit1 = (LastRssi / 10) % 10;
    LastRssiDigit2 = (LastRssi % 10);


    itoa(LastRssiDigit1, DataToSend + index, 10);
    index++;
    itoa(LastRssiDigit2, DataToSend + index, 10);
    index++;
    itoa(3, DataToSend + index, 10);
    index++;
    itoa(1, DataToSend + index, 10);
    index++;
    indexFinal = index;
  }
}
```

Figure C1: Wearable Code [##]

SMS Text Messaging

The SMS notification system incorporates Python and Twilio, a communication API service. Twilio provided a temporary phone number that was used for the notification system. All recipients in the database would receive a notification that would alert them of exposure to an infected by the provided number. The SMS alert system will only work if a certain criterion is met. In this case, for a notification to be sent to a tag user, the tag user must be within a pre-set parameter *and* there must be someone categorized as "sick" within that same parameter. The flowchart is *Figure C2* demonstrates how this text message system works. Please also reference to the diagrams (*Figure C3* and *Figure C4*).

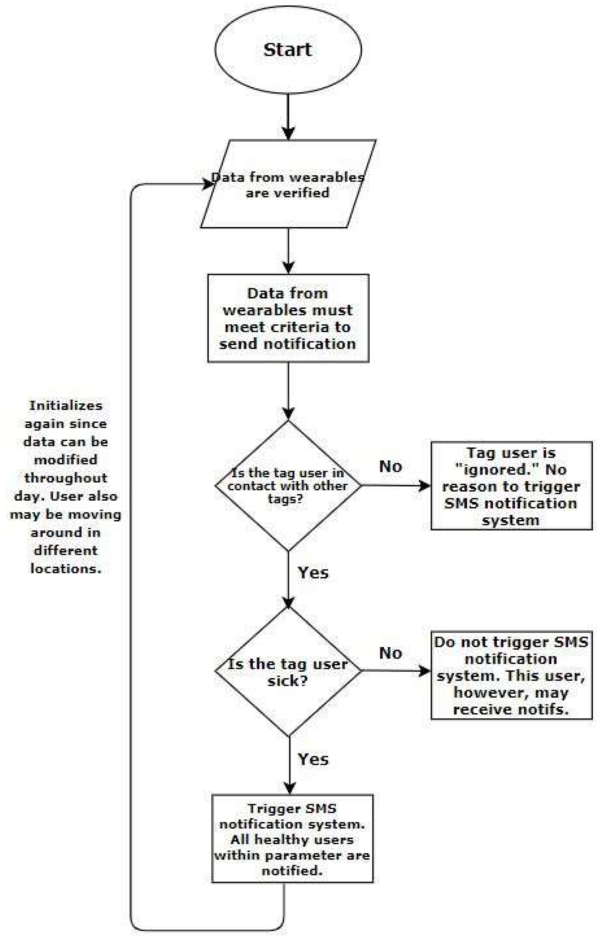

Figure C2: SMS Notification System Flowchart [##]

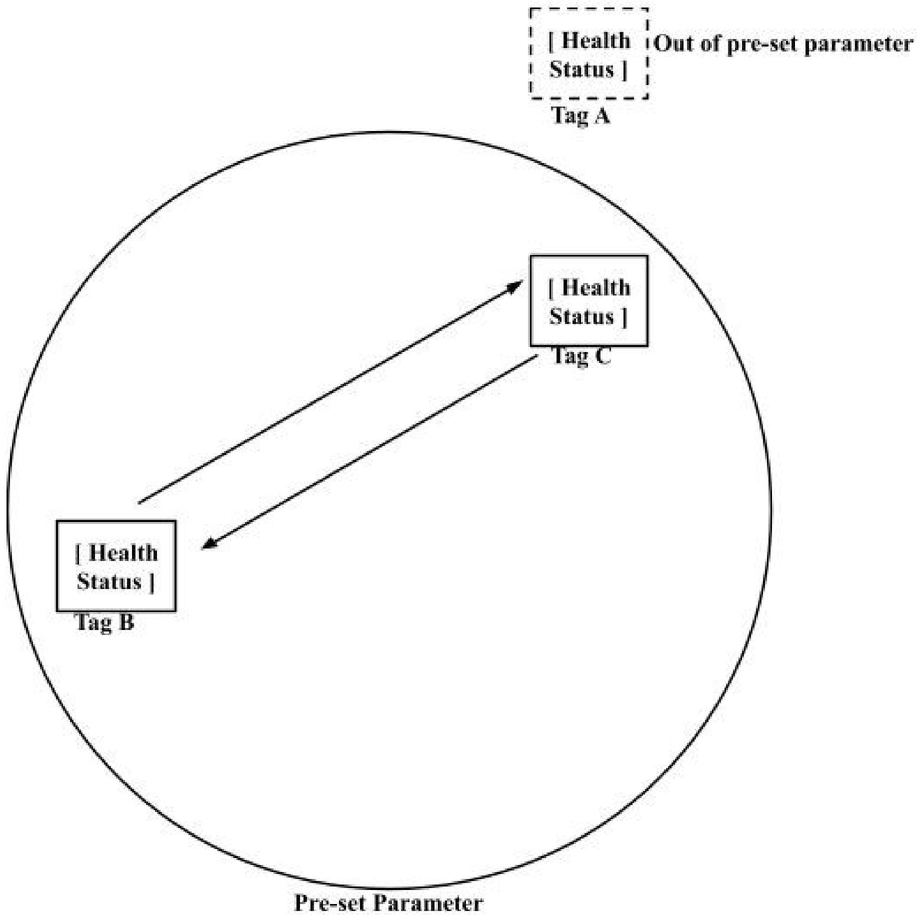

Figure C3: 2-tag Communication Diagram [51]

In *Figure C3*, Tags A, B, and C are not numbered as they should originally be. This representation shows how the 2-tag communication works. Letters A, B, and C can be substituted for any number between 1 through 3, where each number can be used only once at a time; i.e. no two tags with the same number. This is only one example. Examples vary by health status. The arrows indicate the communication between the two tags inside the parameter. If *only one* of the tags within the parameter was worn by a sick individual, the other tag would be notified of their exposure.

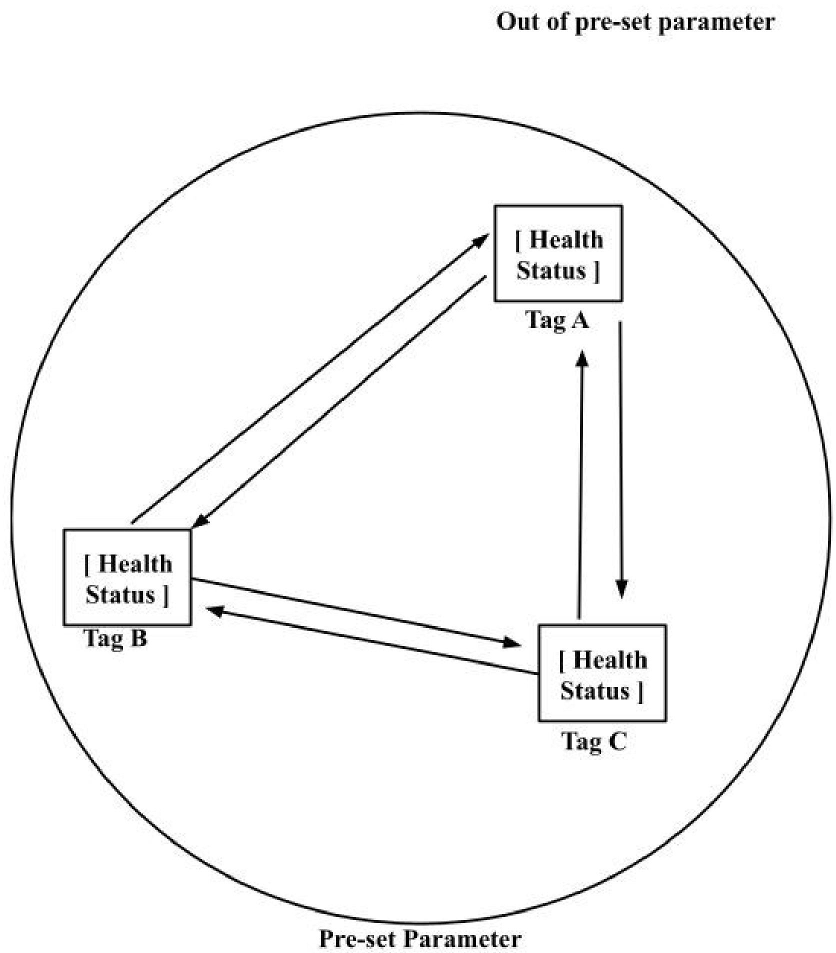

Figure C4: 3-tag Communication Diagram [53]

In *Figure C4*, Tags A, B, and C are not numbered as they should originally be. This representation shows how the 3-tag communication works. Letters A, B, and C can be substituted for any number between 1 through 3, where each number can be used only once at a time; i.e. no two tags with the same number. This is only one example. Examples vary by health status. The arrows indicate the communication between the three tags inside the parameter. If *one or two (maximum)* of the tags within the parameter are worn by sick individuals, the other tag(s) would be notified of exposure.

```python
# control block for Tag 1 and Tag 2
# uncensored number is twilio's number, i.e. sender's number
if contact_one == True:
   if infected_one == False and infected_two == False or infected_one == True and infected_two == True:
      print()  #do nothing
   elif infected_one == True:
      message = client.messages.create(
         to = "+1##########",
         from_ = "+16156714691",
         body = "came into contact with infected individual [person 1 is infected, person 2 is notified]")
   elif infected_two == True:
      message = client.messages.create(
         to = "+1##########",
         from_ = "+16156714691",
         body = "came into contact with infected individual [person 2 is infected, person 1 is notified]")
   else:
      print() #do nothing
```

Figure C5: SMS Notification System Partial Code [##]

The above code is only a sample of a control block for communication between two tags. As previously mentioned, the code will go through and verify the criteria and dispatch a message if necessary.

Data Encryption

The data from the database will always be encrypted throughout the whole day of the tag usage. Access to the database is limited to a small number of medical personnel. The only way to decrypt the encrypted data is to have a valid username and password. If an incorrect username and/or password was input into the login portal, the administrator in charge of the database is immediately notified of this unauthorized attempt. The flowchart shown in *Figure C6*, demonstrates how the data encryption, decryption, and warning dispatch works,

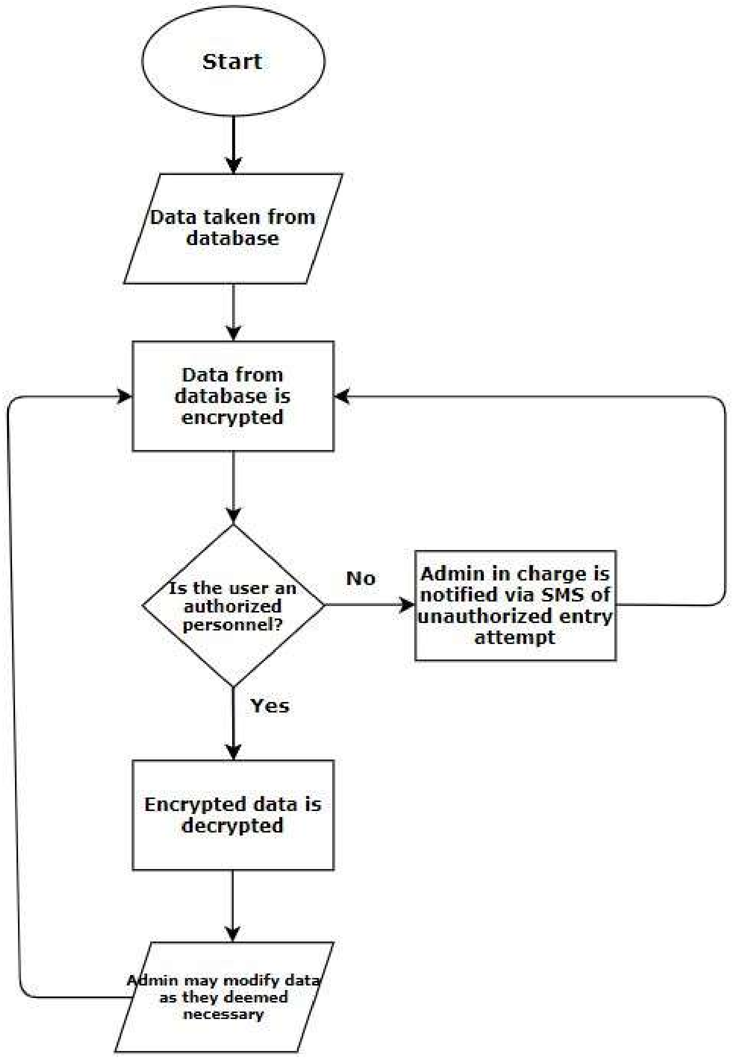

Figure C6: Data Encryption Flowchart [##]

```
#encrypting file
with open('sample.csv', 'rb') as original_file:
   original = original_file.read()
   encrypted = auth.encrypt(original)

with open ('enc_sample.csv', 'wb') as encrypted_file:
   encrypted_file.write(encrypted)

#verifying access; file will remain encrypted unless the user trying to access the files
#is actually within the list of authorized personnel
if access == False:
   #warning message should be sent to admin
   message = client.messages.create(
         to = "+1##########",
         from_ = "+16156714691",
         body = "WARNING!! AN UNAUTHORIZED ACCESS ATTEMPT WAS MADE! PLEASE REVIEW THE DETAILS ASAP!")
elif access == True:
   #decrypt files
   with open('enc_sample.csv', 'rb') as encrypted_file:
      encrypted = encrypted_file.read()
      decrypted = auth.decrypt(encrypted)

   with open('dec_sample.csv', 'wb') as decrypted_file:
      decrypted_file.write(decrypted)
else:
   #do nothing
   #it is expected that the file will remain encrypted throughout the whole time to maintain privacy safety
   print();
```

Figure C7: Data Encryption and Decryption Partial Code [##]

Figure C7 shows instances when the code is decrypted and encrypted. Twilio is also incorporated in this code to notify the admin in charge.

CRU Processes

The CRU contains four main Python files that are used to control the CRU's processes. These are *cru.py*, *cru_proceses.py*, *cru_sec.py*, and *cru_decrypt.py*. These four work in conjunction with each other to power the CRU, show below in Figure C8, we see the high-level flowchart for the CRU.

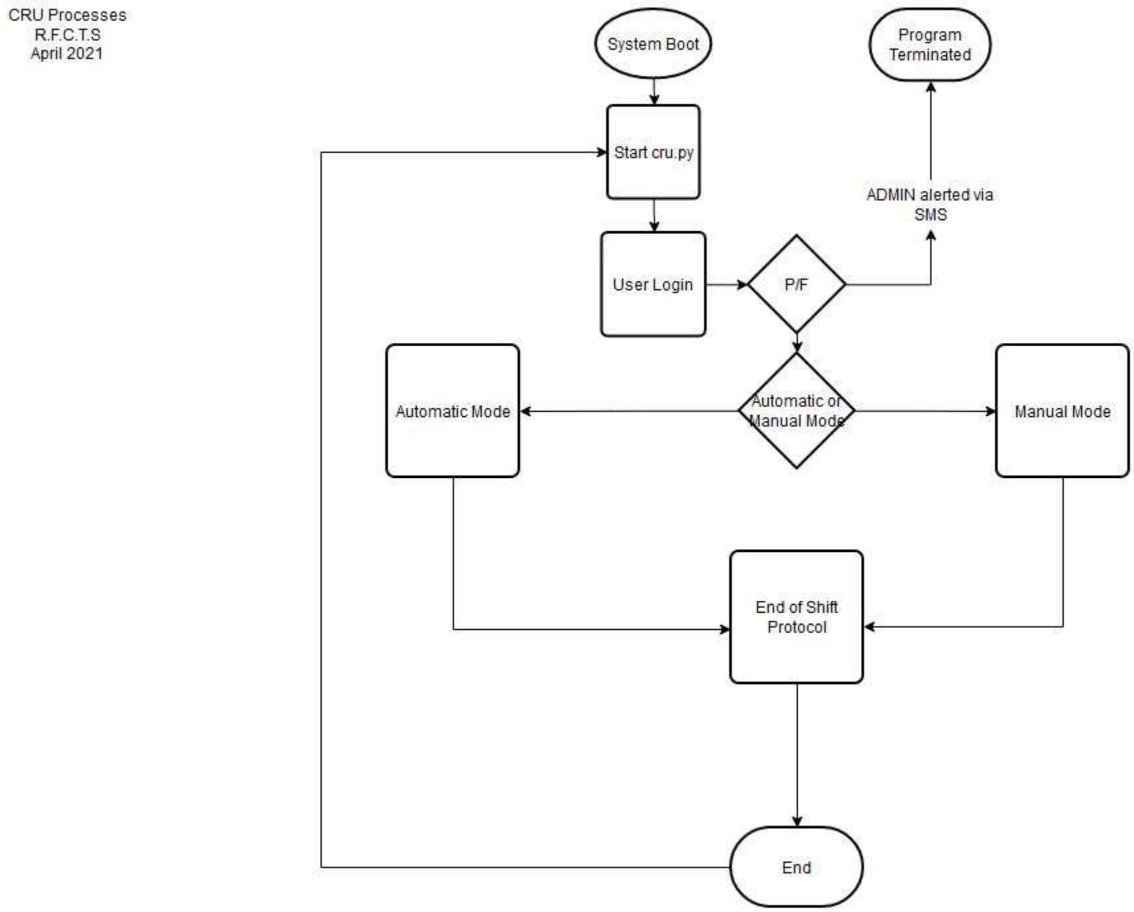

Figure C8: CRU High-Level Flowchart [64]

Figure C8 above displays the high-level explanation for the CRU's processes. Shown below in Figure C9 is the execution of the CRU in automatic mode accepting data from the tags and then routinely sending data to the database, which is being updated in the right window. This is the prime example of all three of our communication protocols acting in unison. Tag-to-Tag, Tag-to-CRU, and CRU-to-DB.

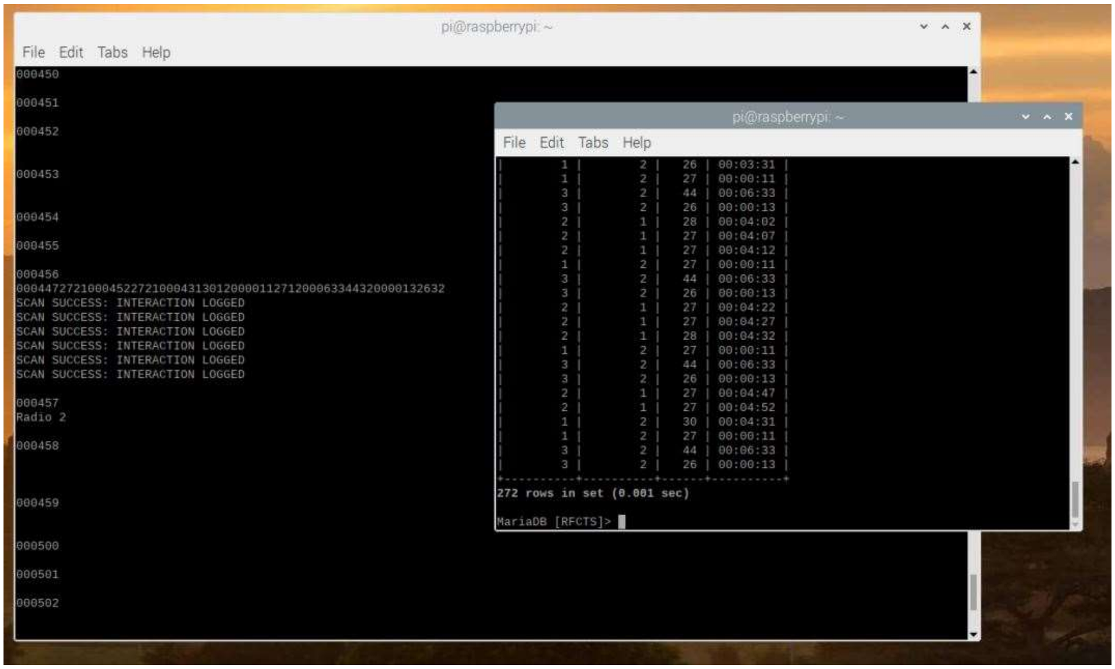

Figure C9: W2W, W2C, CRU2DB [65]

In addition, at the end of each shift the CRU performs a function that writes the database's contents to a .csv file. Some of the source code can be seen in Figure C10 below:

```
### TABLE_TRANSFER(CONN,CURS)

def table_transfer(conn, curs):
  now = datetime.datetime.now()

# Write Tags Table
  print("Writing TAGS Table")
  curs.execute("SELECT * FROM tags")
  r = curs.fetchall()
  h = [col[0] for col in curs.description] # copies table headers
  dw = open*'/home/pi/rfcts-csv/test1.csv', 'w')
  dw.write(now.strftime('%Y-%m-%d %H:%M:%S \n'))
  dw.write('TAGS TABLE \n')
  dat = csv.writer(dw, delimiter = '|', lineterminator = '\n')
  dat.writerow(h)
  dat.writerows(r)
  dw.close()

 # Write Tag_Map table
  print("Writing TAG_MAP Table")
  curs.execute("SELECT * FROM tag_map")
  r = curs.fetchall()
  h = [col[0] for col in curs.description] # copies table headers
  dw = open*'/home/pi/rfcts-csv/test1.csv', 'a')
  dw.write(now.strftime('%Y-%m-%d %H:%M:%S \n'))
  dw.write('TAG MAP TABLE \n')
  dat = csv.writer(dw, delimiter = '|', lineterminator = '\n')
  dat.writerow(h)
  dat.writerows(r)
```

Figure C10: Table Transfer Partial Source Code [69]

Database

The database was written using MariaDB, which is a fork from the MySQL relational database system. Due to this, a lot of the commands and syntax used when programming databases in MariaDB are the same as MySQL. As stated earlier in the report, MariaDB was the first database system that we tried during the development of the prototype from the first semester and was ultimately the database system that we returned to during the second semester. It was chosen due to its user authentication processes being built into the setup and it is an open-source database system which helps greatly reduce the cost of the project overall.

For the GUI, we went with another open-source product, HeidiSQL. This database management software can be used to rapidly adjust the database from a remote location, adding to the practicality and expands the number of CRUs that an administrator can monitor at a time.

The database contains seven tables and five triggers, shown below in Figure C[#]. Attributes in bold text are the Primary Key ([PK]) while attributes in italics are Foreign Keys ([FK]). Numbers inside of parentheses relate to size of the attribute, for example the *p_num* attribute is the data type varchar and has a size of 10. This means that a full phone number can be added to the database and possibly to the text list during the CRU's "End of Shift" processes.

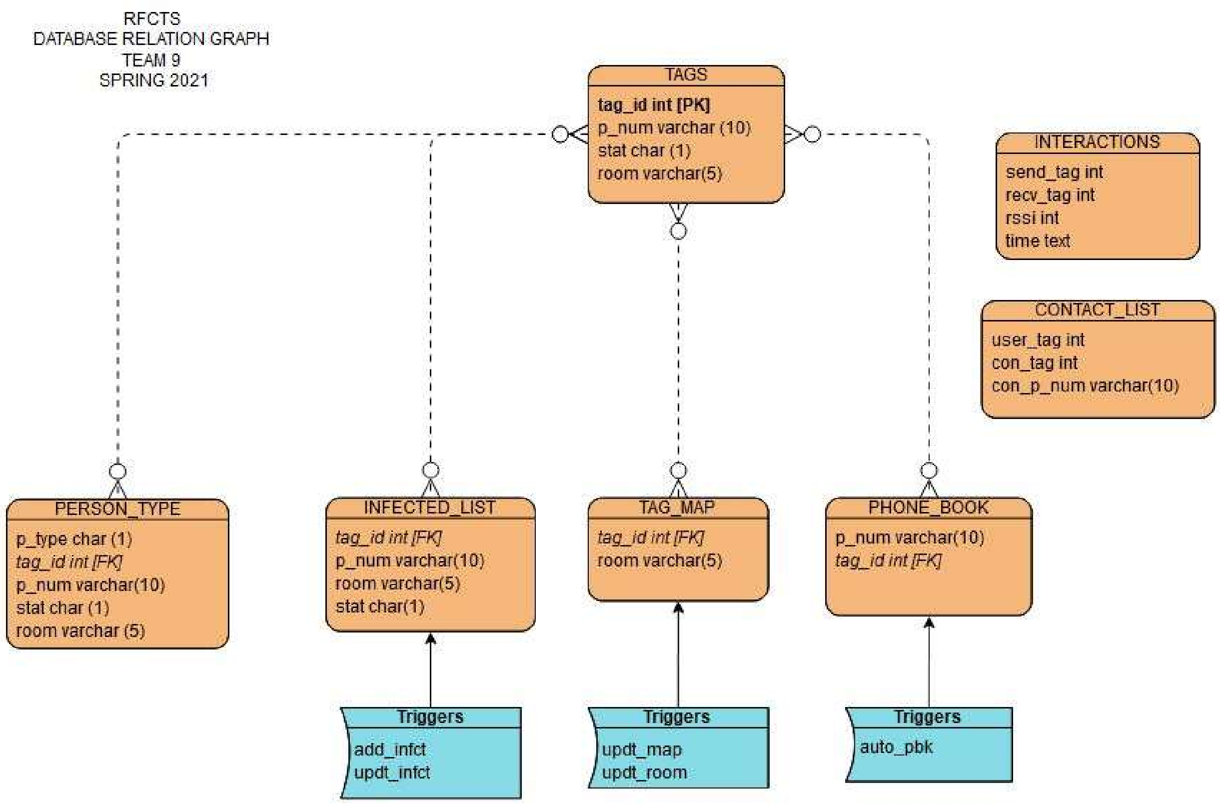

Figure C11: Database Relation Graph [56]

As shown the figure above, most of the database tables contain attributes from the Tags table. It can also be seen that the Interactions and Contact_List tables contain attributes that are not common with the other database tables, but their use in this system is just as important. With the Interactions table being used to log tag to tag interactions and the Contact_List table provided a table of interactions between tag users and non-tag users such as people that have come into contact with these people outside of the hospital setting. These non-users will have a contact tag ID (*con_tag*) of 0.

Shown below in Figure C11 is some of the source code for the database triggers that aid with the automation of some of the table updates that happen. Importantly, all triggers are activated whenever the tags table is manipulated in some way.

```
--updates infected_list table when tags tag_id stat is updated from 'h' to 'i'. 'c' stands for "cleared".
-- will write a python script to delete all 'C' status users during end of shift protocol
DELIMITER #
CREATE TRIGGER updt_infct
AFTER UPDATE ON tags
FOR EACH ROW
BEGIN
        IF NEW.stat = 'h' THEN
                UPDATE infected_list SET stat = 'c' WHERE tag_id = OLD.tag_id;
        END IF;
END#
DELIITER ;
--Automatically adds tags and room to the tag_map table anytime a new entry is added to the tags table
DELIMITER #
CREATE TRIGGER updt_map
AFTER INSERT ON tags
FOR EACH ROW
BEGIN
        INSERT INTO tag_map VALUES (NEW.tag_id, NEW.room);
END#
DELIMITER ;

--Automatically changes room # on tag_map based on tags table update
DELIMITER #
CREATE TRIGGER updt_room
AFTER UPDATE ON tags
FOR EACH ROW
BEGIN
        UPDATE tag_map SET room = NEW.room WHERE tag_id = OLD.tag_id;
END#
DELIMITER ;
```

Figure C12: Database Triggers Source Code [66]

Additional Software Information

Outside of the software that we used to enable our project to perform its intended functions, we used other software to help with other aspects of the project such as version control and documentation. Listed below is the different software that we used throughout our project including the programming languages that we used to create our code.

Diagrams & Flowcharts: Microsoft Visio, Visual Paradigm Online
Documentation: Microsoft Office Suite
File Sharing: Microsoft OneDrive
Programming Languages: Arduino (C++), Python, SQL
Version Control: GitHub

APPENDIX D: MECHANICAL ASPECTS

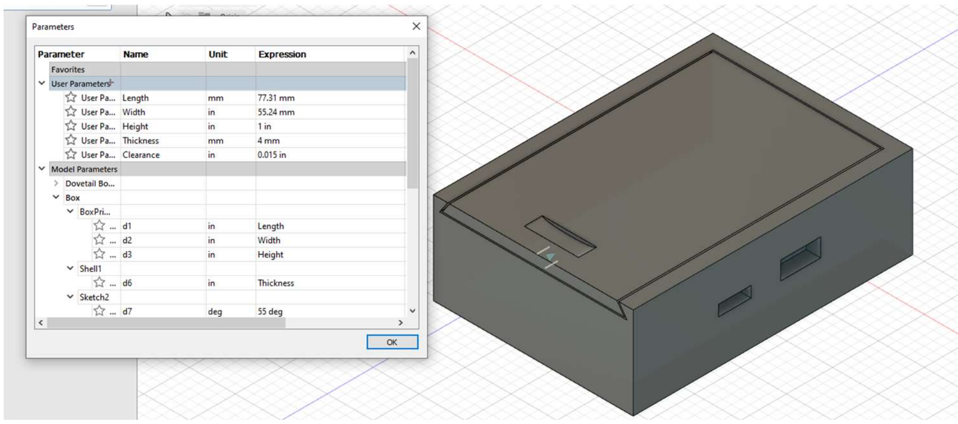

Figure D1: 3D Model Of a Wearable [60]

Figure D2: Wearable Device with Casing [67]

Figure D3: Wearable Device Casing Clip [68]

APPENDIX E: VENDOR CONTACTS

Professor Neal F. Levine
Part Time Professor, California State University Sacramento – Sacramento, CA
College of Engineering & Computer Science, EEE Department
email: neal.levine@csus.edu

Professor Levine was our project mentor and laboratory section instructor. Each week throughout the two semesters we met with him to discuss our project's progress. He offered us guidance and support throughout the entire Senior Design experience, and he was able to help use secure locations for us to meet in person to finish the project in time.

APPENDIX F: RESUMES

This page is left blank intentionally to allow for the maximum space needed for resumes.

Yuruvith T. Saavedra [22]

OBJECTIVE - Acquire new skills, as well as gain new industry experiences, while working in a team, as I complete my bachelor's degree in Computer Engineering.

EDUCATION

California State University – Sacramento — Sacramento, CA

B.S. Computer Engineering (May 2021)

SKILLS & RELEVANT COURSEWORK

- Experience with Java, C, C++, Python, and x86
- Experience with hardware languages: VHDL and Verilog
- Flexible with different settings
- Know the basic features and functions for Word, Excel, PowerPoint

Data Structure + Algorithm Analysis
Computer Architecture
Advanced Logic Design
Advanced Computer Organization
Computer Interfacing
Computer Hardware Design
Program Concept + Method I/II

EXPERIENCE

Senior Design Project
RF-Based Contact Tracing, Team #09 — June 2020 – May 2021

Wearable RFID tags aid with contact tracing within medical settings. Users of the devices will be alerted if, and when, they encounter someone who has tested positive with the virus. Database for this device is highly encrypted to ensure user privacy, and data is only accessible to authorized personnel. The software portion of this project will be done with some of my contribution. I will be able to apply my coding skills.

Advanced Computer Organization Project
Pipeline System Simulation — May 2020

The objective of the project was to design and simulate the datapath and control unit for a pipeline system. I was able to apply my skills with Verilog and arithmetic logic into this project. The project consisted understanding and the creation of truth tables, datapaths, simulations, basic and complex logical functions, and reading and writing to memory. This was a solo project, and therefore, I had to use all the knowledge gathered from the course, as well as past courses, to implement into this project.

Christopher A. Moncrief [23]

OBJECTIVE - Obtain a degree in Electrical Engineering, with a concentration in Communications technology

---

EDUCATION

Communications Engineering, Sac State, Sacramento, CA. GPA: 3.2. Jan 2017- Current

Associate of Science, Natural Science, Sierra College, Rocklin, CA. GPA: 3.30 Dean's List. Jan 2013- Dec 2015

Ground Communications Organizational Repair, Communications-Electronic School, 29 Palms, CA. Top 10 percent of Class. Aug 2008- Sep 2008

Basic Electronics, Communications-Electronic School, 29 Palms, CA. Top 8 Percent of class. Jul 2008 – Aug 2008

SKILLS & RELEVANT COURSEWORK

Skills

Able to learn new tasks quickly and adapt to new situations.

Proficient in using Excel to modify data into a digestible form.

Experienced in writing reports on Microsoft Word.

Works well in both a team setting and alone.

Relevant Courses:

Differential Equations and Linear Algebra

Engineering Drawing and CAD

Applied Electromagnetics

Modern Communication Systems

Introduction to Circuit Theory

Principles of Physics: Electricity/Magnetism

Applied Wave Propagation

EXPERIENCE

Marine Corps Communications Maintenance Supervisor Oct 2012- Jan 2013

- Performed Quality Control on equipment, ensuring 100 percent of gear is returned without defects.
- Enforced an efficient and effective team setting, which reduced equipment turn-around by 25%.
- Utilized programs such as Excel, Word, and Power Point, generating accurate reports quickly.

Marine Corps Communications Technician Feb 2009 – Oct 2012

- Successfully identified faults for 300 pieces of communications equipment, ensuring the equipment was received.
- While working alone, maintained over 200 individual pieces of communications, properly troubleshooting all degraded equipment and ensuring all repair parts arrived in a timely manner.
- Provided field survey assistance to remote locations, insured that radio equipment was installed correctly and functioned as intended.

<u>Senior Design Project</u>

RF-Based Contact Tracing, Team #09 June 2020 – May 2021

Provided expertise into information security as well as providing a feasible model to the project itself, despite the research about tag to tag communications being relatively unexplored.

<u>Second Most Recent Course Project</u>

Blind Spot and Proximity Sensor Spring Semester 2020

Provide a suite of sensors to a car to help with detecting other cars in the "blind spot." The suite required communications between a Propeller Activity Kit and a Raspberry Pi, which I facilitated by developing a simple bus system that incorporated 4 lines, using both the Python and C programming, and a listener and transmitter program. Furthermore, I was able to have the Raspberry Pi transmit a website over the wireless network it was connected to that constantly showed updates on the status of the "alarm" states on the sensors attached to the system.

Andrew M. Sharp [24]

OBJECTIVE - Obtain a job involving either machine learning or lasers.

EDUCATION

California State University – Sacramento Sacramento, CA

B.S. Electrical & Electronics Engineering (May 2021)

SKILLS & RELEVANT COURSEWORK

Skills

MATLAB

Soldering

Welding

Use of RTL-SDRS including integration with MathWorks' Simulink

Autodesk Inventor

Relevant Courses

Applied Electromagnetics

Introduction to Circuit Analysis

Principles of Physics: Electricity/Magnetism

Differential Equations

Advanced Math for Science and Engineering

EXPERIENCE

Senior Design Project

RF-Based Contact Tracing, Team #09 June 2020 – May 2021

Finished this part when the project is finished

Towitlow.com

Design team member for a scalable startup manufacturing company
2016-Current

Identified needs and designed custom parts for a prototype trailer dolly utilizing Torchmate CAD/CAM

Assisted with the build of a Torchmate 4x4 Growth Series plasma cutter, learned its use alongside the team, use it for cutting custom parts

Helped assemble the now-patented prototype including grinding, cutting and welding

Sky Partners
Assistant to the Maintenance Team for a national private air carrier December 2019

Assisted mechanics with troubleshooting recurring problems based on my knowledge of circuit diagrams. Specific examples include a thermocouple circuit plus a door control circuit of a Gulfstream G200

Offered IT support to keep new computer systems running, printer setup

Mitchell D. Allen [25]

OBJECTIVE – Seeking an internship or entry level position in the computer engineering field.

EDUCATION

California State University – Sacramento — Sacramento, CA

B.S. Computer Engineering, Minor in Music (May 2021)

Sierra College — Rocklin, CA

A.A. Music (Dec. 2013)

SKILLS & RELEVANT COURSEWORK

Hardware: FPGA, Microcontrollers, Microprocessor development boards
Programming Languages: C, C++, Java, Python, Assembly, Verilog, VHDL, SQL
Additional Skills: Project Management, LaTeX, Lab Equipment, Agile development, Version Control, PSpice

Adv. Computer Organization
Adv. Logic Design
CMOS & VLSI
Comp. Hardware Design
Comp. Networks & Internets
Comp. Software Engineering
Data Structures & Algorithm Analysis
Operating System Pragmatics
Probability & Random Signals
Signals & Systems

EXPERIENCE

Senior Design Project
RF-Based Contact Tracing, Team #09 — June 2020 – May 2021

Engineered and documented a solution to the Fall 2020 – Spring 2021 Senior Design prompt of a societal problem. Performed Team Leader duties for a limited term as well as contributed to the overall hardware and software design of our project.

Computer Software Engineering Course Project — Summer 2020
Electric Utility Web Application

Created an electric utility RESTful web application over a three-week period using the Postman API, Java programming language, and the Spring Tool Suite IDE as an exercise in the Agile development process.

RC Willey, Rocklin CA || *Small Electronics Specialist*, Oct. 2015 – Apr. 2021
Harbor Freight Tools, **Rocklin CA** || *Various Positions*, Aug. 2010 – Oct. 2010

APPENDIX G: WBS AND HOURS SUMMARY TABLE

Table G-1:
Work Breakdown Structure for Both Semesters [31]

R.F.C.T.S. – FALL 2020			
Level 1	**Level 2**	**Level 3**	**Team Member**
1.1 Wearable Devices			
	1.1.1: minimum X hour battery life		
		1.1.1.1 Battery calculations	Andrew Sharp
		1.1.1.2 Wireless charging	Chris Moncrief
	1.1.2: Wearable to Wearable communication		
		1.1.2.1 Transceiver function testing	Andrew Sharp
		1.1.2.2 Software Implementation	Andrew Sharp
	1.1.3: Interference Limiting		
	1.1.4: Wearable to CRU Communication		Mitchell Allen
	1.1.5: Proximity recording (X < 6’)		
		1.1.5.1 Tracking Wearable to Wearable Interactions	Everyone
		1.1.5.2 Time Stamping	Yuri Saavedra, Mitchell Allen
		1.1.5.3 Time Stamp Comparison/Processing	
1.2 Central Receiver Unit			
	1.2.1 Data organization		Mitchell Allen
		1.2.1.1: Sorting Algorithms	
		1.2.1.2: Scripts to update data in Database	
	1.2.2 RPi / Arduino Implementation		Andrew Sharp, Christopher Moncrief
		1.2.2.1 Software Implementation	
		1.2.2.2 Arduino Serial Communication	

		1.2.2.3 RPi Communication	
	1.2.3 Warning System Activation		Yuri Saavedra
		1.2.3.1 SMS Text message system	
		1.2.3.2 Python Script	
	1.2.4 Processes		Everyone
		Setup	
		End of Shift Data Dump	
1.3 Database			
	1.3.1 Simple Database		Mitchell Allen
		1.3.1.1 Tag Table	
		1.3.1.2 Person_Type Table	
		1.3.1.3 Infected Table	
		1.3.1.4 Phone Number Table	
		1.3.1.5 Contact List Table	
		1.3.1.6 Interactions Table	
	1.3.2 Med Staff Access		
Team Activity Reports			
	Weeks 4 - 13		Everyone
Outgoing Team Leader Reports			
	1		Mitchell Allen
	2		Yuri Saavedra
	3		Andrew Sharp
	4		Chris Moncrief
Team Member / Teamwork Evaluations			
	Week 7		Everyone
	Week 14		
Assignment 1A: Individual Societal Problem			
	Research		Everyone
	Writing		
Assignment 1B: Team Societal Problem			
	Research		
	Writing		Everyone
	Presentation		

Assignment 2: Design Idea Contract			
	Research		Everyone
	Writing		
Assignment 3: WBS			
	Writing		Everyone
Assignment 4: Timeline			
	Writing		Everyone
Assignment 5: Risk Assessment			
	Writing		Everyone
Technical Review			
	Presentation		Everyone
Assignment 6: Prototype Evaluation			
	Video		Everyone
		Filming	
		Editing	
	Hour Distribution		
	Tasks Remaining List for Spring 2021		
	Poster		
	Statistics Report		
	Copy of Punch List		
Assignment 7: Semester 1 Showcase			
	Turn in required files		Everyone
R.F.C.T.S. – SPRING 2021			
2.1 Wearable Devices			
	2.1.1: Improved Battery Life		
		2.1.1.1 Battery calculations	Andrew Sharp
		2.1.1.2 Wireless charging	Chris Moncrief
	2.1.2: Wearable to Wearable communication		
		2.1.2.1: Transceiver Antenna update	Andrew Sharp, Chris Moncrief
		2.1.2.2 Software Update	Mitchell Allen
	2.1.3: Design Update		
		2.1.3.1: Smaller Design	Andrew Sharp, Chris Moncrief
		2.1.3.2: Make it wearable	Andrew Sharp

	2.1.4 Encryption Implementation		Yuri Saavedra
2.2 Central Receiver Unit			
	2.2.1 Data organization		Mitchell Allen
		2.2.1.1 Update Data Structures	
	2.2.2 Encryption Implementation		Yuri Saavedra
	2.2.3 RPi/Arduino Updates		Chris Moncrief
		2.2.3.1 Software Updates	
	2.2.4 Warning System Activation		Yuri Saavedra
		2.2.4.1 Update Scripts	
	2.2.5: Design Update		Andrew Sharp
		2.2.5.1: Create Housing/Casing for unit	
2.3 Database			
	2.3.1 Complex Database		
		2.3.1.1 Update current tables with more precise attributes	Mitchell Allen
	2.3.2 Encryption Implementation		Yuri Saavedra
Team Activity Reports			
	Weeks 1 - 13		Everyone
Outgoing Team Leader Reports			
	1		Mitchell Allen
	2		Yuri Saavedra
	3		Andrew Sharp
	4		Chris Moncrief
Team Member / Teamwork Evaluations			
	Week 7		Everyone
	Week 14		
Assignment 1A: Problem Statement Revision			
	Research		Everyone
	Writing		
Assignment 1B: Design Idea Review & Change Orders			

	Research		
	Writing		Everyone
Assignment 1C: Spring Timeline Update			
	Research		Everyone
	Writing		
Assignment 2: Device Test Plan			
	Writing		Everyone
	Presentation		
Assignment 3: Market Review			
	Writing		Everyone
Assignment 4: Feature Report & Presentation			
	Writing		Everyone
	Presentation		
Midterm Progress Review			
	Presentation		Everyone
Assignment 5: Testing Results Report			
	Writing		Everyone
Assignment 6: Ethics Quiz			
	Complete the Quiz		Everyone
Assignment 7: Deployable Prototype Review			
	Presentation		Everyone
	Post-Project Audit Report		Everyone
Assignment 8: End of Project Documentation			
	Writing		Everyone
	3 – 5 minute summary video		Everyone
Assignment 9: Deployable Prototype Presentation			
	Turn in required files		Everyone

Listed below is Table G-2, which contains the summary of hours that each team member put into our project over both semesters. The rows have been summarized down to the essence of the assignment in order to lower the confusion of all of the subsubsections related to certain assignment tasks. For example, under the CRU portion for the Fall 2020 semester, that includes the SMS and hardware portions of work, not just the communication and database portions.

Table G-2:
Hours Summary [70]

Assignment	***Moncrief***	***Allen***	***Saavedra***	***Sharp***	***Team Total***
			FALL 2020 SEMESTER		
Assign. 1A	12	10	7	6	**35**
Assign. 1B	6	11	4	5	**26**
Assign. 2	6	15	8		**29**
Assign. 3	8	15	11	8	**42**
Assign. 4	1	6	1	5	**13**
Assign. 5	6	10	3	2	**21**
Tech. Review	4	9	6	4	**23**
Assign. 6	3	12	5	7	**27**
Team Activity Reports	10	11	10	10	**41**
Evaluations	5	3	1	1	**10**
Team Leader Reports	1	1	1		**3**
Meetings	28	28	28	28	**112**
Hardware Research	22	20	10	9	**61**
Software Research	5	9	15		**29**
Wearables	18	7	7	43	**75**
CRU	8	39	25	5	**77**
			SPRING 2021 SEMESTER		
Assign. 1	6	10	2	4	**22**
Assign. 2	1	9	2	2	**14**
Assign. 3	12	13	7	3	**25**
Assign. 4	6	12	8	2	**28**
Assign. 5	16	8	2	1	**27**
Assign. 6	3	3	3	3	**12**
Assign. 7	1	2	2	3	**8**
Assign. 8		24	6		**30**
Team Activity Reports		8	11	10	**29**
Evaluations	2	6	4	3	**15**
Team Leader Reports	1	1	1	2	**5**
Meetings	23	23	23	23	**92**

Wearables	42			80	**122**
CRU	6	45	39	10	**100**
Integration	14	21	9	11	
TOTALS		389	**261**	**290**	

APPENDIX H: TIMELINE TABLE

Table H-1:
Project Timeline Table [43]

Task	Start Date	End Date	Team Member
Assignment 1A	June 2020	Sep. 14, 2020	Everyone
T.A.R. #1	June 2020	Sep. 21, 2020	Everyone
Assignment 1B	June 2020	Sep. 28, 2020	Everyone
T.A.R. #2	Sep. 21, 2020		
O.T.L.R. #1	Aug. 31, 2020	Sep. 28, 2020	Allen
Assignment 2	Sep. 28, 2020	Oct. 5, 2020	Everyone
T.A.R. #3			
T.M.T.E. #1	June 2020	Oct. 12, 2020	Everyone
T.A.R. #4	Oct. 5, 2020		
T.A.R. #5	Oct. 12, 2020	Oct. 19, 2020	Everyone
Assignment 3	Oct. 5, 2020	Oct. 26, 2020	Everyone
Safety Form			
T.A.R. #6	Oct. 19, 2020		
O.T.L.R #2	Sep. 28, 2020	Oct. 26, 2020	Saavedra
Assignment 4	Oct. 26, 2020	Nov. 2, 2020	Everyone
T.A.R. #7			
T.A.R. #8	Nov. 2, 2020	Nov. 9, 2020	Everyone
Assignment 5			
T.A.R. #9	Nov. 9, 2020	Nov. 16, 2020	Everyone
Technical Review Presentation	Aug. 31, 2020		
O.T.L.R #3	Aug. 31, 2020	Nov. 16, 2020	Sharp
T.A.R. #10	Oct. 26, 2020	Nov. 23, 2020	Everyone
T.M.T.E. #2	Nov. 16, 2020	Nov. 23, 2020	Everyone
Assignment 6	Oct. 12, 2020	Dec. 7, 2020	Everyone
SMS Program	Aug. 31, 2020	Dec. 7, 2020	Saavedra
Simple Database	Aug. 31, 2020	Dec. 7, 2020	Allen
Wearables	Aug. 31, 2020	Dec. 7, 2020	Sharp
CRU Processes	Aug. 31, 2020	Dec. 7, 2020	Allen, Sharp
Battery Life	Aug. 31, 2020	Dec. 7, 2020	Sharp, Moncrief
Assignment 7	Aug. 31, 2020	Dec. 8 2020	Everyone
T.A.R. #1	Aug. 31, 2020	Jan 25, 2021	Everyone
O.T.L.R #4	Nov. 23, 2020	Jan. 25, 2021	Everyone
Assignment 1	Nov. 23, 2020	Feb. 1, 2021	Moncrief
Assignment 1C	Jan. 25, 2021	Feb. 1, 2021	Everyone
T.A.R. #2			
Assignment #2			
T.A.R. #3	Feb. 1, 2021	Feb. 8, 2021	Everyone
T.A.R. #4	Feb. 8, 2021	Feb. 15, 2021	

Progress Review Presentation	Feb. 8, 2021	Feb. 15,2021	Everyone
O.T.L.R. #1	Jan. 25, 2021		
T.A.R. #5	Jan. 25, 2021	Feb. 22, 2021	Allen
Assignment #3	Feb. 15, 2021	Mar. 1, 2021	Everyone
T.A.R. #6	Feb. 22, 2021	Mar. 1, 2021	Everyone
Assignment #4	Feb. 22, 2021		
T.A.R. #7	Mar. 1, 2021	Mar. 8, 2021	Everyone
T.M.T.E. #1			
T.A.R. #8	Mar. 8, 2021	Mar. 15, 2021	Everyone
O.T.L.R. #2	Feb. 22, 2021	Mar. 29, 2021	Saavedra
Assignment #5	Feb. 1, 2021	Mar. 29, 2021	Everyone
T.A.R. #10	Mar. 29, 2021	Apr. 5, 2021	Everyone
T.A.R. #11	Apr. 5, 2021	Apr. 12, 2021	Everyone
O.T.L.R. #3	Mar. 29, 2021	Apr. 19, 2021	Sharp
T.M.T.E. #2	Mar. 22, 2021	Apr. 15, 2021	Everyone
T.A.R. #13	Apr. 12, 2021	Apr. 19, 2021	Everyone
Assignment #6	Apr. 12, 2021	Apr. 19, 2021	Everyone
Assignment #7	Apr. 12, 2021	Apr. 26, 2021	Everyone
Wearables	Aug. 31, 2020	Apr. 26, 2021	Sharp, Moncrief
CRU	Aug. 31, 2020	Apr. 26, 2021	Sharp, Moncrief, Allen
Database	Aug. 31, 2020	Apr. 26, 2021	Allen
Encryption	Jan. 25, 2021	Apr. 26, 2021	Saavedra
SMS Texting Program			
Assignment #8	Aug. 31, 2020	May 3, 2021	Everyone
Assignment #9	June, 2020	May 10, 2021	Everyone
O.T.L.R. #4	Apr. 12, 2021	May 10, 2021	Moncrief

APPENDIX I: DEVICE TESTING

For the database and CRU process related tests, a key has been created for each test ID:

CP = CRU Process
DB = Database
TR = Trigger

Table I-1:
Device Test Plan [44]

Test ID	Description	Expected Results	Viewed Results	Pass/Fail
CP-01	Sort_convert() function takes input data, separates it, and converts it to the appropriate data type so that it can be sent to the RFCTS database.	Entering character strings of data with lengths that are multiples of 10 will correctly be separated and converted into the appropriate data type. Use a print function to display correct data changes.	After performing this function I was able to view accurate separation and conversion of manually entered data.	**P**
CP-02	Push_to_inter() function takes recently sort_convert() data and pushes it to the interactions table in the RFCTS database.	Checking the interactions table through the GUI or CLI shows the recent converted data. Entries are based on multiples of ten.	Checking the interactions table through the GUI or CLI shows the recent converted data.	**P**
CP-03	Pull_from_infect() function creates a list of infected individuals by retreiving data from the infected_list table.	The function will print out the infected tag IDs from the infected_list table.	The function printed out the infected tag IDs from the infected_list table.	**P**
CP-04	Warning_sys_protocol() function will create a list of phone numbers to text that have been in contact with infected individuals and then text them.	The function will create and execute a for loop to SMS text people who could be infected based off of their recent contact with an infected individual.	The SMS messaging system texted a team members phone number in succession to demonstrate that the SMS system works based off infected tag ID data.	**P**
CP-05	Auto_infct_del() function automatically deletes entries from the infected_list table that have a status of cleared (“c”).	After performing a final check, the function will automatically delete any table entries with the “c” status.	The function sucessfully deleted multiple “c” status entries from the infected_list table.	**P**
CP-06	Data_pull() is used during Automatic mode to pull data from the wearable devices and push them to the CRU and DB.	Leaving Automatic mode running will produce accurate data uploads over a period of time.	Accurate data uploads happened at precise intervals over a short period of time. Data was doublechecked via HeidiSQL and the RPi terminal.	**P**
CP-07	Table_transfer() pushes data from the database into a .csv file at the end of a shift.	Data is pulled from the DB at the end of shift and time stamped into a .csv file	Successful transfer of table to .csv file.	**P**
CP-08	Encrypt_csv() encrypts the newly created .csv file.	New .csv file will be encrypted right after being created.	Data is encrypted and cannot be read unless given the right information.	
CP-09	Decrypt_rfcts() decrypts the CSV file that is output at the end of a shift.	With a successful loging attempt the .csv file will be decrypted.	After logging in correctly the .csv file was decrypted and available to read.	**P**
DB-01	Access the CRU (RPi) database remotely through the HeidiSQL GUI on the laptop.	When entering the correct login credentials, I will be able to access and modify the RFCTS database that is normally accessed through the	After entering the correct login credentials on HeidiSQL on my laptop I was able to access the RFCTS database.	**P**

		CRU. Entering the wrong credentials will result in an error.		
DB-02	Access the database through the *cru_processes.py* program.	Performing the provided connector function through the *mariadb* python library allows the CRU to take input data and send it to the database.	*cru_processes.py* can successfully connect to the database to perform and does not throw any connection error codes.	P
TR-01	"auto_pbk" trigger for database. Automatically adds "tag_id" and "p_num" values into the "phone_book" table whenever a new tag_id is initialized in the "tags" table.	We added six tags to the tags table and performed a SELECT * FROM command both the tags and tag_map tables. Both tables displayed accurate information. I double checked these results on HeidiSQL by refreshing the database and both tables showed the correct entries.	We added six tags to the tags table and performed a SELECT * FROM command both the tags and phone_book tables. Both tables displayed accurate information. We then double checked these results on HeidiSQL by refreshing the database and both tables showed the correct entries.	P
TR-02	"add_infct" trigger will automatically add any infected tag ID entry from the tags table to the infected_list table.	Whenever the tags table is updated by setting an entry's status to infected ("i") this change is reflected by adding the same information to the infected_list table.	After updating six tags table entries to infected, the infected_list table reflected these changes with six infected_list entries.	P
TR-03	"updt_infct" trigger will change the status of an infected tag ID in the infected_list table to "c" whenever the same tag ID changes back to the healthy status in the tags table.	Cycling back through the six infected_list entries, we changed their status in the tags table from infected ("i") to healthy ("h") which then changed their status in the infected_list table to cleared ("c").	After updating the tags table entries from infected to healthy, the infected_list table entries also changed from infected to cleared.	P
TR-04	"updt_map" trigger for database. Automatically adds the tag_id and "room" values into the tag_map table whenever a new tag_id is initialized in the tags table.	When we perform an INSERT command (MariaDB) or add a new entry (HeidiSQL) to the tags table, the values for "tag_id" and "room" will also be added to the tag_map table.	We added six tags to the tags table and performed a SELECT * FROM command both the tags and tag_map tables. Both tables displayed accurate information. We then double checked these results on HeidiSQL by refreshing the database and both tables showed the correct entries.	P
TR-05	"updt_room" trigger automatically updates the room of an entry in the tag_map table when that same entry changes in the tags table.	Changing the room number of several entries in the tags table will reflect that change in the tags_map table.	The room number of several entries changed in the tags table, this change also occurred in the tags_map table.	P