SONOMA STATE UNIVERSITY

DEPARTMENT OF ENGINEERING SCIENCE

# Sense-It

*Juan Soto* *Anthony Hargrove*

ADVISER: Dr. Don Estreich

INDUSTRY ADVISER: Rob Gee

PROJECT WEB PAGE:
`http://hargrove7.wix.com/sense-it`
MAY 12, 2016

SUBMITTED TO THE DEPARTMENT OF ENGINEERING SCIENCE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF:

BACHELOR OF SCIENCE IN ELECTRICAL ENGINEERING

# Abstract

The driving force behind our idea is to provide an alert system for the visually impaired. Our product could be paired with other assistive devices to enhance the independent experience of someone with limited vision. Canes and dogs can keep people off of walls and other objects but knowing when one has arrived at their final destination can be a challenge. Many wall placards have braille writing on them to notify persons when they have reached a bathroom or other important location but our device would be able to tell them without them having to remove gloves, empty hands or touch potentially filthy walls.

We hope to design a wearable device that will allow its wearer to be notified about features of objects that they are passing by. Often times new office buildings can be intimidating. Without someone to guide us around, we may never even realize that we have arrived at our destination. RFID Notification would let the wearer know when they have come in close proximity to their destination. We are developing our device so that we can essentially allow for people with a visual impairment to navigate new areas, much like GPS, but inside buildings. Of course our device will be usable for the general public as well.

Radio Frequency Identification (RFID) is commonly found in libraries and retail stores. They often involve a system of antennas, some powered (active) and some not (passive), to determine what objects have passed by. In libraries, books carry unique passive antennas and as they pass through larger powered antennas, the libraries can tell what book has passed through its gates. We hope to use this technology to program a users own active antenna to recognize when it reaches preprogrammed locations, using bluetooth. Why bluetooth? In today's technology we have something called Bluetooth 4.0 and Bluetooth Low Energy (BLE). With Bluetooth 4.0 it allows for successfully toggling between mulpliple devices, instead of the standard one device. Essentially allowing for multiple communication lines without having to stay synched to one device. We are using BLE in a unique way which is why our device can be marketed if pursued later in the future. We are using BLE with iBeacon technology that will interface with an IOS application that we will develope. An iBeacon is a configured bluetooth module working with BLE configurations that will interface with any IOS device. These beacons are only used to advertise data, that means they are creating a signal advertising their position. This is a very high level explanation of how our system will work. We would like to temporarily install our antennas/beacons in discrete locations within the Salazar Electrical Engineering Department. There we would conduct our experiment for the next two semesters.

In order to achieve our goal, we will need to purchase bluetooth modules that support bluetooth 4.0 and are iAP(ipod Application Protocol) certified. We will need to develop an IOS App that will allow for a systemic demonstration of all devices integrating togethor to demonstrate concept. At the heart of directing all communication traffic we plan to use a $4.09 Microchip PIC32MX270F256B because of its small form and low voltage as well as its UART interface. We

have the tools and facilities in the engineering department to create our own printed circuit board (PCB) for the users device as well as our beacons, so we will be building it in-house.

Our device will be able to notify its user when they come in close proximity of any predefined location that holds our Sense-It Device and downloaded app.

## System Overview

Development of our IOS application is a critical part of our system. It allows for the actual detection of our beacons and therefore is the pivot point for our real world demonstration of our whole concept. The following is a description of its importance to our system and a technical overview of how we went about creating it.

For simplicity I will refer to the IOS application as an app(application). Our app was coded up by brainstorming at the highest level what we are trying to achieve. Proof of concept, user friendly, and a library of unique id's. Our proof of concept is that our team will be able to create iBeacons with BLE configurations and a wearable Sense-It device. Once the system is integrated the app will utilize Siri to voice location audibly once the signal of a beacon is within range of the phone. Our Sense-It device will be able to direct communication traffic from the app and iBeacon through UART(Universal Asynchronous Recieve/Transmit) utilizing the microcontrollers from both ends of the iBeacon and Sense-It.The app will be able to send messages to each iBeacon so that any administrator can access the messages left behind by the user or vice versa.

Being user friendly is extremely important to us. We are catering our product to people with disabilities and don't want to further hinder them with complications. We also intend our product to be usable by anyone else who so chooses to use our product, therefore user friendly UI makes sense. Once our app is opened by the user, a list of locations will become available. For demonstration purposes the user will have to manually click on a location of interest. Each location will come with unique UUID's(Universily Unique Identifiers) which are 128 bit values in length. These UUID's are predefined values that come from us, or administrators, and are given to each iBeacon. They allow the app to search for the constant advertisement of that discretely placed, low energy consumption, iBeacon. Once within range Siri will audibly alert the user that they are within range, fifteen feet at least, of their point of interest.

For anyone who is not familiar with app development I will give a brief overview of the tools necessary to make an app, followed by a technical description for those with a technical background. I'm going to assume that anyone who reads this document knows what a general application or app is. Each app is made by app developers with a purpose in mind. For example, games, flash cards, cloud based services etc. One goes about making these apps with code, the general environment of developing that code varies between IOS and Android markets. Specifically we used the IOS environment tools to make our app.


Figure 1: High Level Diagram.

# Contents

# List of Figures

# List of Tables

# 1 INTRODUCTION

Being a new student on a college campus can be extremely intimidating. Now imagine being a new student with a disability on a college campus. In particular, imagine being a student that is completely blind.

At most universities, the solution for helping a student navigate their campus is to offer a personal attendant that escorts the student from class to class. Often times these attendants are not available for the full day, but only arrive when the student's schedule requires that they transit between classes. This means that when the student doesn't need to come or go from a class but instead needs to visit the library on their free time or needs to travel to the bathroom in the middle of a class, they are on their own. Those students who are extroverts and comfortable with their disability may not mind asking other students for help but others may be too intimidated and may decline to attend school altogether.

We hope to design a personal navigation assistant for the visually impaired. Whenever the wearer of our device passes by discretely placed antenna/beacon on a wall or doorway, they could be notified that they have made it to their destination.

We would also like to add an ability to update the RFID library to enable the user to locate predetermined locations. The RFID would be able to communicate with strategically placed antennas to notify the user that they are in close proximity to their destination.

## 1.1 Problem Statement

We will need to add a low power, light-weight device to a person. When the person walks by one of our RFID/beacons, they should be notified where they are at. In addition, if they walk by a beacon that they have no interest in, they should not be told anything.

Once the customers have seen the utility of our design, can they be sure that the device is trustworthy? It would only take one false notification with an inappropriate room for a customer to demand their money back and never return to our device again. Imagine a woman wandering into a mens bathroom. Our customer base will be only a small fraction of the total world community, so we need to make sure we alienate as few members of that community as possible. In the U.S., there is an estimated 7.3 million persons that are visually impaired. https://nfb.org/blindness-statistics

## 1.2 Customer Base

An assisted living device such as this one will literally be the first of its kind. Other than human services provided by Public resources and other public institutions usually provide some sort of human guide or companion that follow the visually impaired. However, through research we have found that there are no such mobile devices that allow independent guidance without relying on a human guide. That being said we know that by law from the American Disabilities Act, public institutions and other private buildings with public access are required to accommodate people with disabilities. That, in and of itself, has a huge customer base because these two types of establishments exist throughout the United States. Furthermore, for people who may not be able to afford the device we could also target insurance companies. With this new market we would be able to deal out more product while ultimately be able to reach a different customer base as the end result.

These would be our target markets once the development of our device is finished. Specifically targeting those with a visual disability and market our product through ease of use as well as discrete mobility without the need of guidance from other people. We would be creating something that would allow a sense of independence and freedom for people who might feel constricted to certain areas. Ease of mind is something that can be taken for granted by people without a disability our device will be aimed at giving some of that back.

## 1.3 Key Components

Table 1: Parts List.

| Part | Name | Price | Source |
|---|---|---|---|
| Microcontroller | Microchip, PIC32MX270F256b | \$5.00 | Microchip |
| Bluetooth Module | TI, HM-11 | \$8.00 | FastTech |
| Rechargeable Battery | Generic | \$10.00 | Amazon |
| PCB | Built on Campus | unknown | SSU |

## 1.4 Marketing Requirements

- Purpose
- Durability: It will be carried around on the person all day so must be able to survive many bangs and bumps. Should be able to handle a bit of moisture as well.
- Reliability: It should work as specified for a long time.
- Device Life: Device must last a full day on a single charge.
- Mobility: The device must be a manageable size and weight for carrying around all day.
- Accuracy of Location: Person must know pretty accurately that they have arrived. Within approximately 15 feet.
- Device functionality compared to competitors: Must provide a notification in a manner exceeding the functionality of our competitors.

- Next best thing on similar market
- Realistically the amount of people expected to use the product.
- Legal Issues that must be resolved.
- Is it Practical.

### 1.5 Engineering Requirements

- Functionality: How our system works will be engineered in a way that caters to any type of technically inclined audience.
- Economic: A produce that is easy on any type of wallet would be ideal for marketability. However, we would also have an option for people who wouldnt be able to afford the product, to go to a third party such as insurance companies that could provide the product at a discounted price or even free.
- Energy Efficiency: Discrete electrical components would have to be the makeup of our product. We want to ensure mobility and does not stand out in a crowd. While building the device we would have to incorporate these categories as well as battery life for it. Research on battery types will have to be done.
- Environmentally Friendly: Now in this era it is almost a requirement for most tech devices to be made with eco-friendly components. This means more research on green components.
- Safety: Our device needs to be able to operate under any type of weather, temperature, and needs to have fail safe components build in for emergency shut off.
- Legal: Huge legal implications especially when dealing with new devices that have no previous market. We need to cover our selves in case of any type of legal issue were to arise.
- Maintenance: We need to hire technically savvy people to help fix our product if it were to breakdown. This requires more money that would need to be spend.
- Operational: How it operates is crucial. It needs to be able to operate under the previously stated categories, as well as be easy to use.
- Manufacturability: Since we will target a high demand target we will have to maximize production without decreasing product quality
- Reliability/Accessibility: Our device will have to be able to pass quality inspection at our manufacturing facility before being distributed. While still be accessible to any of our major product customers.

# 2 Challenges

## 2.1 General Challenges

One major challenge in producing our device is in keeping its size and weight down. Any extra weight the product may have, will reduce the chance a customer will be willing to carry it around with them on a regular basis. However, if we make it too small, its battery life may be significantly diminished. Somehow we need to strike a balance between size and utility. This leads to the challenge of power.

Finding a power source that can run the device for a full day is important. Ideally, we would hope that the device could operate for several days, but if we could lighten the weight of the device by sacrificing a day or two, it may be in our best interest to do so.

We will also need to power the beacons that are mounted on the walls. If we can have them running on extremely low power, we could possibly run them on one battery for a year or two, much the same way as smoke detectors are run. If an establishment has access to wall power at their doorway, it could eliminate the need for batteries altogether.

The device must be affordable. The Sense-It could possibly be classified as a medical device, therefore many of them could be purchased through insurance providers at little to no cost to the consumer. Otherwise, if we put the bulk of the cost on the beacons in the doorway, each individual customer will likely have very little cost and the bulk of the cost could be shifted to the few devices that are mounted safely in building.

We will need need beacons that can reach at least 15 feet. To do so, we will likely need a powered device which will add to our power requirements as well as add to the overall size and weight of the product. In addition, we will have to ensure that the door mounts do not interfere with each other. If the beacons are talking over each other, the user may become confused and not know where or if they have arrived.

## 2.2 Research Challenges

When developing a new product a lot of time goes into research. Initially when planning and prioritizing what should be categorized as important and a feature, is done through research. We tackled this project through understanding what roles each individual better suited certain tasks while keeping in mind the overall end result.

During development our team did extensive amounts of research. Ranging from electrical components, to how to develop code for key features needed to write critical functions. The biggest problem with this is that it took time away from actual development. Usually our team did research when we were met with an unforeseen challenge which added to the pressure of meeting deadlines.

## 2.3 Encountered Software Challenges

Our Sense-It system utilizes code written within the Bluetooth modules used to configure as beacons, the IOS application, and the micro-controller. As with anything that requires code needed to operate correctly, one can expect delays due to debugging.

One of the biggest challenges our team met early on was writing the communication code the micro-controller would utilize. We chose to use an MCU that no one on the team had any previous knowledge of. Therefore the overall syntax, and familiarity of registers took time to get used to. In particular the UART code proved a difficult task to complete. This was due to understanding how to use and configure the MCU to do exactly what we needed it to due. In detail we were using a serial terminal to establish a connection from the micro-controller to one of our Bluetooth modules. What proved difficult was that we wanted to write a string of character values to the input of the serial terminal and have it output the same exact values. We successfully managed to input and transmit a full string of character values and receive a response via Google chrome serial terminal. Unfortunately due to time constraints we were unable to fully integrate our Bluetooth communications portion of our code to our iOS application.

Our application was written in Swift. The biggest challenge that our team met during this phase of the project was developing a UI that would demonstrate the whole concept of the idea successfully. We ran into problems early on with being able to successfully implement the right region protocols that would allow the iPhone to scan for our iBeacons successfully. A good amount of time went into researching what apple developer forums had to say about pseudo code that was relevant to us. Excitingly enough our team managed to code our application fairly quickly and works seamlessly with our iBeacons.

## 2.4 Encountered Circuit Board Fabrication Challenges

No one on the team had any previous circuit board fabrication experience. We knew early on that we would need to familiarize ourselves with the process. The challenges that came from this particular phase were vast. I'll start by saying that the biggest challenge was the necessary preperation needed to even start the process. It takes about two days for the photosensitive ink to cure correctly onto the copper board. Therefore the best solution we had for this problem was to prepare multiple boards in bulk for future use. Once the copper boards were ready, the process itself was relatively simple. After we became familiar with the process we minimized errors that could potentially disqualify the board as usable and increased our success per board yield.

Once we applied the wanted circuit traces onto the PCB. we needed to check the continuity of the board. We needed to refer to the wanted eagle schematic of the board and make sure none of our traces were being shorted out. Over time as we got better initially making the boards we eliminated most of these problems however not completely. Scrapping away any unwanted copper from the board using a microscope took time and patience. Many of our circuit designs were to keep our product as small as possible. Therefore, when quality checking the circuit traces, we were working at a microscopic level that needed a microscope with intense resolution to successfully assure trace continuity. Again this took time and had to be done in compliance with our time line.

## 2.5 The BM77

We dedicated a whole section to a Bluetooth module from Microchip called "BM77". Initially when our group first pitched our concept to the department we used readily available standard five pin Bluetooth modules. Afterwards our group decided to improve upon the design of our product through research for a chip that had all the features of a Bluetooth beacon as well as built in Bluetooth 4.0 capabilities, all in one. We came across the BM77 as a perfect option because of all the features that came with it and also because it was documented that it would work seamlessly with the line of microcontrollar's we have become familiar with. Unfortunately tapping into and utilizing the chip was unsuccessful. It proved extremely difficult to be able to interface with. Our group contacted and spoke to one of the companies engineer's and even they advised to use a different chip because the documentation behind the chip itself was lackluster at best. Notably we were also informed that the BM77 was a chip that was just hard to interface with in general. Their weren't very many forums that we could use to reference off of, even though the chip itself had been released for years now. Regardless our group tackled the task as best we could and purchased equipment, "Picktail" that would help us configure the chip to our liking. The BM77 as is measures in length and width in centimeters "picture". Therefore at first our group had to become familiar with making a schematic that would allow for a PCB to be made in order for the chip to be configured. Fortunately the BM77 had a schematic readily available. We have become experts in the PCB fabrication process because of it. However still after weeks of trying we were unsuccessful configuring the chip. We moved on to purchasing a "Pictail" that had an already mounted BM77 on the board itself. Their were three ways to configure the BM77 through the "Pictail". Unfortunately the easiest way still proved unsuccessful. It was not because our group couldn't do it. Rather what we encountered was a Half completed GUI that the engineers at Microchip had made. This GUI would not allow for the BM77 to be configured specifically the way our project required it to be done. Our last option was to manually connect the "Pictail" via a DB9 RS232 connector. This proved unsuccessful as well. At this point in time our group was juggling various tasks of the project that were equally important and we had to make a decision to stop trying to develop our product around the BM77. Thankfully we were working on a plan B in parallel while trying to configure the BM77. We ended up using the Bluetooth modules that we had successfully configured prior to the BM77, however they were the raw chips themselves not the already configured ready to go counterparts. It was a little past the midway mark that we decided to drop the BM77, in respects to our May 6 deadline. Through the countless lessons that we learned because of the BM77. We were able to construct PCB boards, configure, and mount the boards successfully.

# 3 Implementation

## 3.1 Time Chart

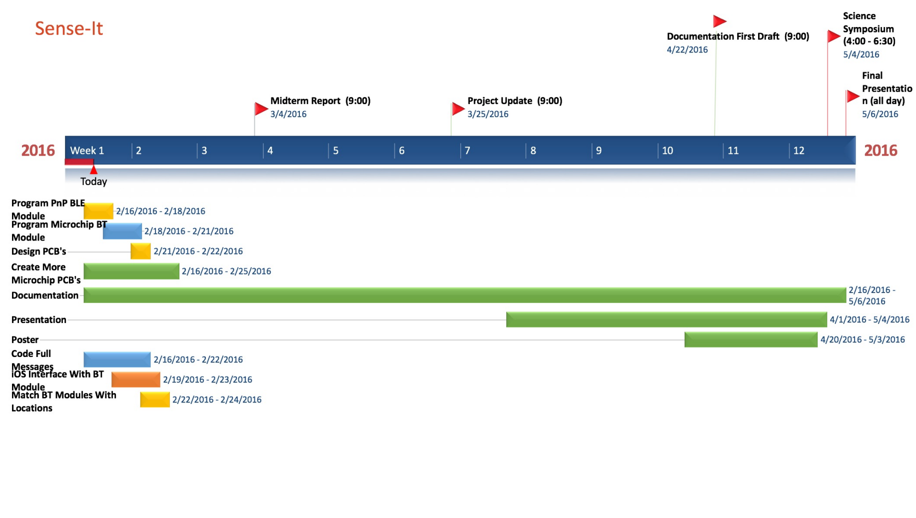


Figure 2: Gantt Chart.

## 3.2 Preliminary Testing

Our Sense-It is comprised of many key components: Power Supply, Mini-Microcontroller, Bluetooth 4.0 Modules, a coded iOS application by our team, and Siri from an iPhone. At the heart of our design it is crucial for our team to be able to establish connection between the micro-controller and the actual iBeacon itself. Without this connection we wont be able to address the key issue, which is to build an assisted living device for the visually impaired. Moving forward with the design once proof of concept was established the team needs to keep engineering requirements in mind at all times. This assisted living device is being built to help and not hinder users, therefore design needs to be discrete and mobile. The housing by default also needs to meet these requirements as well as provide protection for the electronics inside.

Initially we wanted to interface with an RFID tag using a pic and various components associated with an RF-Reader. However, to be able to establish a communication to a RF tag requires the purchase of components that are really not cost effective. In practicality my partner and I need to make sure that the scope of this project is within the bounds of our budget and understanding. Therefore, we are configuring two Bluetooth devices to be able to talk to one another with UART configuration. Without the use of any terminals we want our devices to be able to recognize one another within relevant proximity and automatically establish connection. This will be the start of our prototyping moving forward with our RFID project.

## 3.3 Prototype Board Fabrication

Before we could build the final product, we needed to test the components on our prototyping boards. Unfortunately, the Bluetooth module that we had planned to use was a surface mount component that could not be easily used on a bread board. To integrate the module into our testing we need to to fabricate a breakout board for it and solder it to the board.

To create our boards, we used a FeCl3 etching process. To do so, we required the use of copper clad, FR-4, FeCl3, Sodium Carbonate, UV lights, soldering tools and PCB modelling software.

The first step in the process required developing a photo mask in a PCB modelling software. The software we chose to use was EagleCad due to its large library of components and popularity among classmates and many other designers we came across on the internet. EagleCads popularity ensured that there was plenty of documentation and tutorials available. Our Bluetooth module, the BM77 by Microchip, is a fairly new device with a unique foot print so we were unable to find any prebuilt libraries for it so we had to create our own.

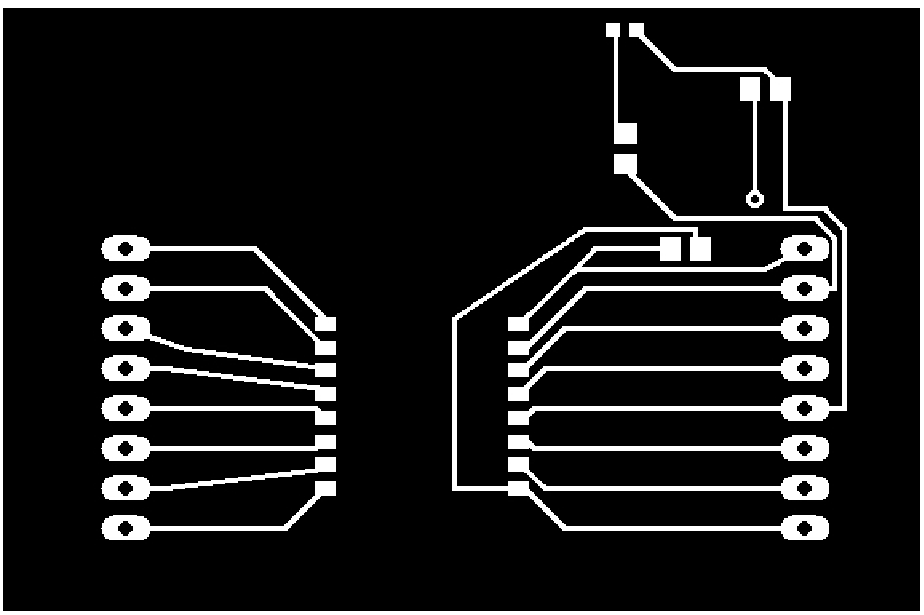

Figure 3: Eagle Mask.

After we created our mask, we prepared our copper by coating it with a photoresist and let it rest for two days. When it was ready, we applied our mask to the copper and placed it in our UV light bed. The mask protected the areas that we would need to remove from the UV light while the other areas were hardened.

The board was then submerged in a 0.08% Sodium Bicarbonate aqueous solution and lightly agitated. The sodium bicarbonate works to remove the photoresist that had not been exposed to the UV light. When it is done, the copper is revealed in all areas that were masked. If any areas that were not supposed to be exposed are revealed, they can be touched up with a permanent marker.

Once the board has been dried and touched up, it must be submerged in the FeCl3. FeCl3 is a strong acid so care should be taken while using it. We recommend safety goggles and latex gloves. You may also want to wear clothing that you are not worried about discoloring. The board in the acid requires regular agitation so we used a circular agitator and let it ride for approximately thirty minutes. If the board requires much longer than thirty minutes, there is a good chance that the acid will eat away at the adhesive holding the copper to the board and the board may need to be discarded. The end product should be a board with copper removed everywhere except where photoresist and marker had rested. When done, the board should be washed with water and lightly scrubbed to remove the photoresist.

To mount the Bluetooth module, we required instruments and skills a bit more precise than what we had available. The BM77s contacts are very small and close together. Our first couple attempts at soldering them to a board resulted in several shorts that could only be seen through a microscope. Through some casual research, we discovered that solder paste may make the task a bit easier. By carefully applying the solder paste to our board and positioning the BM77 on top of it we succeeded in properly soldering our chip to our prototype board. The board headers were then applied by using a common iron and solder.

### 3.4 Wall Mount

At each point of interest, our wall unit can be disguised behind a typical sign that you would find in any common hallway. Our prototypes have been mounted behind ten-inch square bathroom placards. The circuit board itself is only two inches by two and a half inches and less than a quarter inch thick.

In the above figure, Figure 1, we see our two Bluetooth modules and PIC Microcontroller. The rightmost module performs the communication link for sending messages to iOS devices when requested. The module on the left performs the beacon duties. By running the duties on two separate modules ensures that the beacon is always available. This version of Bluetooth requires that if the module has paired, its beacon signal will drop. Dropping the beacon every time a user requests information from a beacon would disrupt the service for every other user.

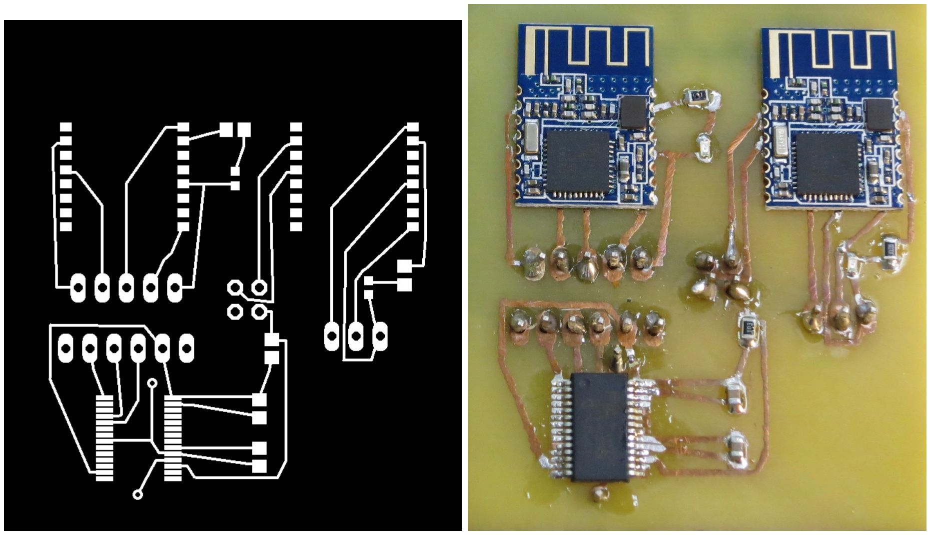

Figure 4: Wall Mount Footprint and Board.

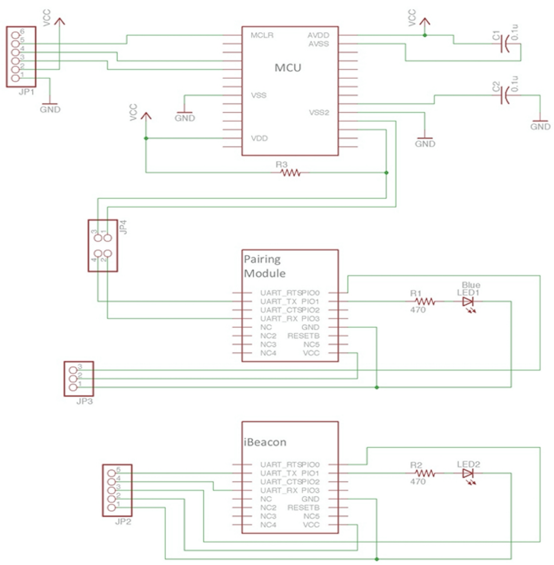


Figure 5: Wall Mount Schematic.

### 3.5 IOS Application Developement

In this section we will provide a detailed overview of our app creation. Following the description of our app we will also provide our code in our "Source code" section. Please keep in mind that we use terminology needed to describe app creation specifically in XCode. For a higher level description please refer to our "System Overview Section". We used Xcode to generate and compile the source code. Specifically we used the language of "Swift" to code our app. Swift is a multi-paradigm, compiled programming language created for iOS, OS X, watch OS, tv OS and Linux development by Apple Inc. Swift is designed to work with Apple's Cocoa and Cocoa Touch frameworks and the large body of existing Objective-C code written for Apple products. Swift is a high level language that allows for "safer" code to be run. By safer we mean apple developed it to be more resilient to erroneous code compared to objective-c.

We currently have a simply coded one-page app. We utilize only two view-controllers, where the first view-controller is in charge of keeping track of points of interest. The second view-controller, directly linked to the first, is where we can establish a link between the iBeacon and our app. Once that is done Siri audibly gives the user a notification, proving concept.

While brainstorming our team knew that we needed a library or directories that allowed for our UUID's to be labeled behind each classroom. We created a table listing classrooms that pertained to unique UUID's, which are 128 bit in length, iBeacon Identifiers. We had to make sure that this table was editable. Therefore we added three main features on this part of our storyboard's "SenseItViewController". We added a push button feature that allowed to toggle between deleting any classrooms that were no longer of interest. We also added another push button feature that allowed to add more classrooms or other points of interest. Most notably we added the feature of clicking on a Point of Interest(PoI). In future installments we would like to have this an audible feature, but for conceptual purposes it is push button. Following the next viewcontroller titled "ClassroomViewController". We have to click on a PoI to get here and we see various things. We see two distinct areas where a user can relay notes to an administrator and display any notes that an administrator has left behind. We would like these notes to be audibly given to the user, if any, via Siri. This communication between user and administrator and user will be utilized via UART as specified above, however currently our team is still working on this interface. The most important part of this page is the "Link" button, currently located on the upper left hand side of the App. Once this push-button is pressed it initializes the app to start looking for that specific UUID. If the phone is within range of the iBeacon a pop-up message is displayed on the phone. Siri audibly reads the message telling the user that they have arrived at their PoI. Therefore providing a real world demonstration of our proof of concept.

The following is a description of our most important functions that allow us to effectively demonstrate our concept using our app. Our code Utilizes four main functions that give life to our two viewcontrollers. The first is our bluetoothManager function which is just calling on the CBCentralManagerDelegat. This protocol allows the delegate to monitor the discovery,connectivity,and retrieval of peripheral devices. In our code we utilize our function, "bluetoothManager", and the CBCentralManager to check that all the different states that a peripheral can be put in is actually accounted for. We also use our bluetoothManager function as an overide for our viewdidload() function. We use this function on our main viewcontroller to set the environment for bluetooth connectivity, once the app is loaded up for the first time.

In order to give our app peripherals to look for we needed specific 128 bit identifiers pertaining to different iBeacons,UUIDS(Universally Unique Identifiers). We made use of the NSUUID class function to store uniqe addresses to their specific variables. Once an iBeacon is within range of the app. Which is executed by the "CLRegion" function, it starts the process of evaluating proximity, signal strength, peripheral count and accuracy. We store these UUID's to specific classrooms that are listed on the first page of our viewcontroller. Once the user clicks on an indicated point of interest the app moves over to the next viewcontroller with that prexisting UUID already stored. Once the link button is pushed it will actively look for that specific UUID.

Initializing our CLLocationManagerDelegate allows us to create a central point for configuring the delivery of location and heading related events to our app. Most notably the delegate will be used to report range to nearby iBeacons and audibly voice specific messages that correspond to specific cases. We created a "locationManager" function that allowed us to use a subclass identifier "CLRegion", to determine the state of region we entered. While creating different instances of this subclass, CLRegion function, the app is able to read the location and beacon strength. We went even further to determine accuracy and strength using an array "beacons" that compared the indexes of our array to the actual CLBeacon objects and wrote specific algorithms determining: Number of beacons found, proximity, received signal strength, and voice alerts pertaining to proximity cases.

Initializing the AVSpeechSynthesizer is key to allowing Siri to audibly voice messages. We used this function to voice messages that were critical to proximity. As mentioned above once our proximity,count,accuracy,and signal strength algorithms had a values to display. We ran our synthesizer function to alert user's of the critical data via Siri. You do this by feeding the function a string of the message you want to hear. Once that is done the next line is followed by a line of code that grabs that message in the form now as an object denoted by the speechSynthesizer.speakUtterance(message) form.

# 4 Final Testing

## 4.1 Battery Test

In order to test the performance of our potential batteries, we set up a small circuit containing a PIC microcontroller and a BM77 Bluetooth module set up as an iBeacon.

Our microcontroller was set up to toggle power to a port every second and our Bluetooth module was set to advertise five times per second. Our battery was a 3V lithium pill battery by Energizer. The batteries initial measured voltage was 3.3V but after only thirty minutes of running the circuit, the voltage of the battery had dropped to 3.0 volts. After one and a half hours, the voltage had dropped to 2.7V and was no longer powering our devices.

These 3V lithium pill batteries are completely incapable of powering our device for any appropriate length of time. Our end product would be much more efficient than this setup for we would not need the microcontroller to constantly stay active but we would need more than thirty minutes of active time per charge.

We have begun a secondary test on the battery, hooking up just the Bluetooth device to it. After the first ten hours of testing the battery dropped from 3.2V to 3.0V. If the battery has settled into this voltage and can operate the device for several days this is a good sign that a more robust battery should be able to handle a pick operating sparingly throughout the day while simultaneously operating a Bluetooth module.

## 4.2 iBeacon Signal Tests

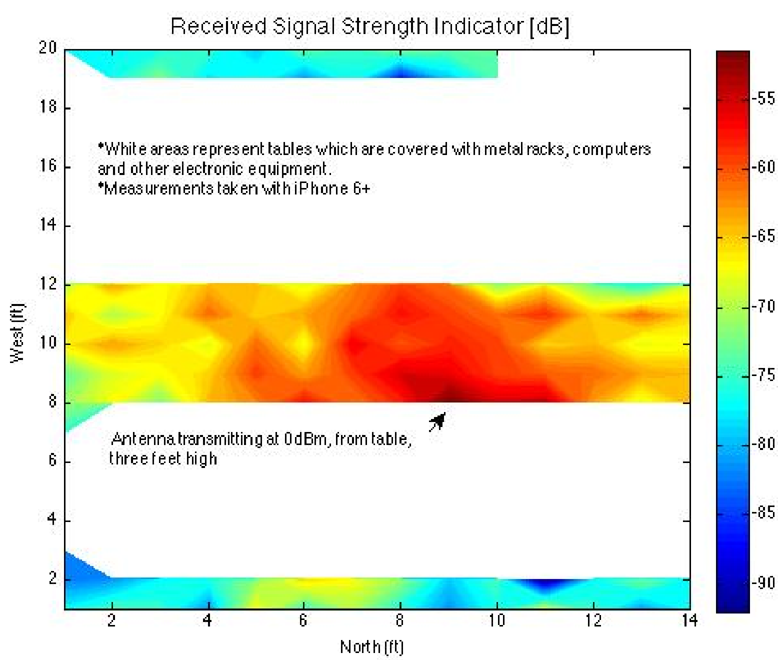


Figure 6: iBeacon dB Signal Graph.

Several tests were required to develop an understanding of the Bluetooth signal characteristics. We tested: range with Line of Sight, range through walls, floors and around corners, range through human bodies, and tested for destructive interference.

To test Line of Sight range, we found an eight feet wide hallway that was 275 feet long. At one end of the hallway, we placed a wall mount circuit board on an upturned trash can. With an iPhone 6+ in hand, we walked the hallway until we lost sight of the signal. When the module was configured for 0dBm transmission power, we made it all the way to the other end of the hallway without losing the signal. Reprogramming the module to its lowest power setting, -24dBm, the Line of Sight range was reduced to approximately 225 feet. This reduction was not significant and may mean that running the modules at their lowest power settings will not affect the performance of our network in any material way.

In the middle of our lab, we have two rows of tables that are covered with computers, tools, metal racks and countless junk. In an attempt to test for destructive interference due to multipathing, we set up our wall mount board in the center of the room on a table. In between the crowded tables, we measured a standard RF distribution pattern that diminished with range and did not see any significant drops in signal with the module in our line of sight. Carrying the module around the other sides of the table, we saw a significant drop in signal strength but it was consistent with loss of Line of sight.

As you can see in figure 6. We conducted a test evaluating the signal strength of our iBeacons (dB). We placed them outside a doorway and measured the strength while staying still for a fixed interval of time (10 seconds) in various positions through out the hallway, max being ten feet away.

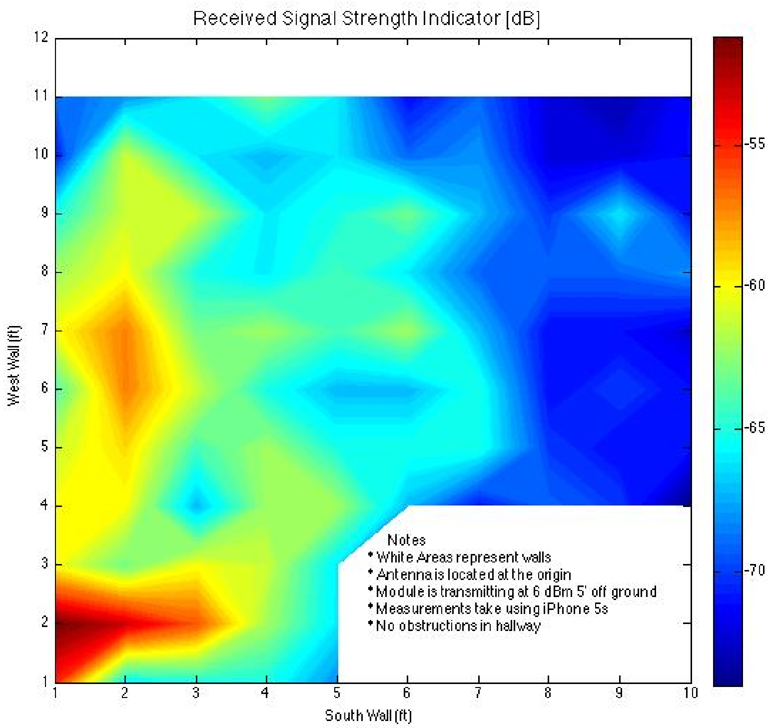


Figure 7: iBeacon Heat Signal Graph. iBeacon located at the lower left hand conrner, [1,1]

In this particular figure 7. We use the exact data from figure 6 to generate the graph, however we show the strength of our signal relating it to heat. What we a looking for are "Nulls" or "Cold" spots. We wanted to see if there were any dropped signals close to the door were the beacon was located. This would mean that any users carrying our device might encounter some connection problems related to this particular spot. We see in the data that there is in fact a cold spot relatively close to the beacon itself.

# 5 Wrap Up

As electrical engineering students, we take great pride in what opportunities our education has afforded us. We would love it if the first product that we have designed and produced was something that could help someones life be a little easier. Sense-It called on our ability to design circuits involving power management and signal processing. We were also required to use our knowledge of microcontroller design to set up an interface between all of the devices involved in the project. Our knowledge of antennas/beacons and RF was necessary to ensure that the devices could communicate reliably and efficiently. We also had an opportunity to take advantage of the PCB fabrication materials in our lab, an experience very few graduating engineers ever get. In the end, this project put our skills as engineers to their first true test.

Throughout this process we were fortunate to have the assistance of our Professors and fellow engineering students. We would like to give special thanks to those that had a direct significant impact on our ability to complete our product.

*Dr. Don Estreich*: Our faculty adviser was there for us every week, from design, production and implementation. When we needed assistance over the Winter Break, Dr. Estreich took time from his family to meet with us at his office. We had many weeks where we felt we were falling behind and we not sure we would ever get back on track. During our weekly status meetings, he assured us that we were on the correct path and gave us the confidence to muscle through the challenges.

*Nick Alvarez*: We were lucky enough to find a student from the local junior college, Santa Rosa Junior College, that was participating in the Mathematics, Engineering and Science Achievement (MESA) internship program that was interested in our product. When Nick arrived, we we not sure how to utilize him so we dumped a Bluetooth module on his lap and told him to go home and make it do something. With very little instruction from us, he managed to recreate a portion of our early work. Seeing him accomplish this on his own proved to us that he would be an asset. Nick was trusted with cleaning up our microcontroller code, performing RF testing, fabricating PCBs and countless other tasks.

*Aaron Marquez*: Our product relies heavily on the existence of Apple iPhone to relay messages to the user. We had very limited experience with coding iOS devices. When we found out Aaron was trying to teach himself how to code in SWIFT, we took advantage of his efforts and recruited him to help build our application. If we had attempted to code the application ourselves, we would have had to sacrifice time spent in other areas of our project. In the end, we were very pleased with how the application turned out and it would have been nowhere as functional if not for Aaron's involvement.

In addition assisting with our application, Aaron was also kind enough to print the enclosure for our hand-held device.

*Graham (Mack) Blacksmith*: During Winter Break, we were stuck on campus attempting and failing at creating our first PCBs. Fortunately, the recently

graduated Mack was still in the local area. After a couple of phone calls, he came out on a Saturday to help us build our first board. With his assistance we were able to define a reliable process that we have left to future students.

## References


[1] National Federation of the Blind, Webpage, retrieved on October 07, 2015 https://nfb.org/blindness-statistics

[2] Americans With Disabilities Act Webpage 2015 ada.gov

[3] Microchip PIC18F45K20 Webpage http://ww1.microchip.com/downloads/en/DeviceDoc/41303G.pdf

[4] *TI Navigating Your Way Through the RF Jungle Webpage* http://www.ti.com/lit/wp/slyy056/slyy056.pdf

[5] The Basics of an RFID System Webpage http://rfid.atlasrfidstore.com/hs-fs/hub/300870/file-252314647-pdf/Content/basics-of-an-rfid-system-atlasrfidstore.pdf

[6] Bluetooth Transceiver RF Module Wireless Serial TTL V1.05 Manual http://www.ram-e-shop.com/ds/general/Bluetooth_TRx_Module_New.pdf

[7] *Bluetooth to Serial Port Module HC*05 *Webpage Datasheet* : http://www.electronica60norte.com/mwfls/pdf/newBluetooth.pdf

[8] Eagle CAD Tutorial

http://www.cadsoftusa.com/training-service/tutorials/

[9] Pew Internet http://www.pewinternet.org/2015/04/01/us-smartphone-use-in-2015/

[10] *JNHuaMao Technology Company Bluetooth* 4.0 *Module* Datasheet:https://wiki.microduino.cc/images/f/fc/Bluetooth40_en.pdf

[11] Swift Coding:http://developer.apple.com

[12] *Localization and Triangulation Research* http://www.intechopen.com/books/emerging-communications-for-wireless-sensor-networks/indoor-location-tracking-using-received-signal-strength-indicator

[13] RSSI Signal Strength and Research http://standards.ieee.org/about/get/802/802.11.html