Wander Watch

Senior Design Project Documentation
Spring 2016

Group 16
Sarah Rassel
Jeffrey Rodriguez
Wendelyn Sanabria
Alexis Timms

Table of Content

List of Figures

List of Tables

1. Executive Summary

The Wander Watch is a watch that communicates with a Bluetooth hub and Android application in order to help keep track of the wearer of the watch. The watch communicates with the Hub to show that the watch is within the designated area. If the wearer of the watch leaves the area, the Hub communicates with the Android application to alert the user. The user can then check the application to see the watch-wearer's location via GPS. Using this information, they can easily locate the wearer and return them to their home. This project was originally meant to be used for older patients with dementia, who may wander from their homes. However, it is hoped that the watch can be used to also keep track of children, expanded upon for other uses, and provide an affordable option to keeping track of loved ones.

The watch was designed to be easy to wear, unlike previous products which may have uncomfortable wearables, or small devices that may be easily lost. It is able to display the current time and date, so that it can also act as a functional watch. The Hub was designed to be able to track the watch within a reasonable range. It uses the best technologies that will allow it to track the watch and send alerts efficiently, while also keeping down its cost of production.

One of the major goals of the project was to include an Android application with user-friendly design. The main menus of this application are the map, alerts, and settings. The application allows users to log in with their own account, in order to keep track of their information. The simple menus and features are meant to streamline the project of checking on the watch-wearer and their location. This is explained in more detail later on.

Besides the design of the application, this report will also explain how the watch, application, and Hub function by themselves and with each other. This will demonstrate how each aspect of this project will send the necessary information to each other. Such information includes the watch's location, whether or not the watch is within range of the Hub, and the content of the alerts that the application will receive. In addition to the design and functionality of the three parts of this project, there will also be a brief view of the project's testing phase, budget and finance, milestones and planned timeline, and possible future expansions of the Wander Watch. The latter of these discussions will include extra features for the watch, Hub, and application. There will also be a discussion on improved functionality that would allow the Hub and application to handle tracking more than one watch at a time.

2. Project Description

Designing a small and functional watch to notify a caretaker via mobile application if the wearer leaves a designated area, as well as track the wearer's location. One of the main objectives for this watch was to minimize the amount of power consumed by the components of the watch and to make it lightweight, affordable, and easy to use product.

Previous devices either tracked dementia patients with ankle bracelets or a small device that could be lost or left behind. These products often limited the trackers to communicate with a device that was sold with it, causing the price to increase. The Wander Watch can be comfortably worn and is able to tell time and the date, making it functional for the user. The creation of the application will allow the caretaker to track the user without having to buy a separate device to communicate with the watch. This is not only more convenient, but also more affordable.

2.1. Project Goals

The goal for the watch was to be able to create a product for patients with Alzheimer's that will provide tracking technology in combination with functionality from the watch along with ease of use and designing the watch to be wearable by minimizing the size of the tracking device as much as possible. In addition to having the watch be wearable and functional, the project also includes a phone application for the caretaker to monitor the watch's status and if the patient were to leave their home, to be able to find them on the map.

2.2. Major Technical Objectives

One of the major objectives for the watch was to include a method of tracking for the patient. This required the project to have a full system for tracking in the hardware, as well as the software for the watch.

Another major technical objective for the product was to have a method of communicating information from the watch to the application on the phone. The product is able to maintain communication with the application on the phone regardless of being inside of the home where the watch is able to communicate with the application via Bluetooth or outside where the watch can use the cellular network to communicate with the application the patient's location.

In addition to having all these features on the watch in order to work as intended, the watch was also able to maintain a wearable size to satisfy the wearable goal for the design. In addition to having the watch be wearable for the patient, the watch is able to minimize the amount of power consumed in order to allow the watch to operate. Not only is the power system and power management a major technical objective, the method of recharging was considered in the design process and was addressed appropriately.

The last major technical objective for the design of the product is the application for the phone. The phone application is able to be easy to use, receive information about the status about the watch, and receive GPS information from the watch regardless of how that information is being relayed to the app.

2.3. Requirements and Specifications

The following are some of the quantitative requirements that the project team has taken into account for this project. This list is subject to change as the team continues working on this project throughout Senior Design.

- Screen Size: 128x128 pixels
- Weight: less than 5 ounces
- Cost: $75 or less
- Battery life: 8 hours
- GPS accuracy within 3 meters (approximately 10 feet)
- Notification to the phone application within 1 minute
- Watch detects that it has left the designated area at least 5 seconds after leaving

2.3.1. Design Constraints

The goal of the design constraints was to establish guidelines for both the hardware and software design of both the watch and the Hub that are based on the criterion established on the customer's needs. The design constraints set a criterion for choosing parts for both the watch and the Hub, established criterion for the communication system, and established criterion for the software design of the application and the hardware. The following shows design constraints for hardware, software, and communications for both the watch and the Hub based on the customer's needs for the product.

2.3.1.1. Hardware Constraints

When designing the watch, it needed to have attributes that compared to a traditional watch. Concerning the hardware, the watch maintained a certain size, maintained longevity in battery life, and minimized as much cost as possible in order to make the product affordable for the customer. In addition to constraints about the watch itself, there are also constraints for the Hub. The Hub has fewer constraints considering that the Hub is not a wearable device and is stationed inside the patient's home. However, cost and power were still a concern for the Hub during the design process.

2.3.1.2. Size Constraints

When developing the watch, one of the main objectives was to have all of the hardware for the watch fit onto a board that was as large as the screen. This would leave the team with only having the watch's thickness to worry about. One of the issues that arose during development was the need to increase the size of the battery due to voltage supply constraints from the components. The watch case needed to have a larger perimeter so that only the screen is displayed and the hardware is housed without being exposed. Another issue that came up was the amount of layers the watch hardware needed. The more layers of the watch, the larger the thickness.

2.3.1.3. Power Constraints

One of the most important aspects of the watch is power consumption from the components of the watch with the customer in mind, the design was optimized to minimize as much power consumption as possible to conserve the battery life for as long as possible between each recharge. Another important design that was implemented was an indication of the current status of the battery life and displays the battery life on the screen. In addition to displaying the battery life on the screen, the caretaker is notified that battery life for the watch is in critical condition and is due for a recharge. This is achieved through the software end of the design project through the phone app. When recharging the watch, the method of recharging the watch is simple for caretaker to manage. The caretaker is able to manage the patient's watch and receive any notifications of the watch being at a low power through the application on the smart phone.

2.3.1.4. Cost Constraints

With the future customers in mind, the Wander Watch team worked to minimize the cost as much as possible so that the watch is accessible to as many people as possible. While most components are relatively inexpensive, most of the cost came from the production of the PCB. The team's goal was to have the development of the hardware to be less than $100. The method of reducing the cost of production was to find a PCB vendor that will meet all of the project's requirements at the lowest cost possible. Another approach of reducing cost was to look for vendors that sell the components needed for the watch at the lowest price possible.

3. Existing Projects and Products

Over the course of the project team's research, it was found that there are a lot of existing products and projects that are similar to the idea for the Wander Watch. There are a lot of similarities, but also some major differences that will hopefully show that this project is still worthwhile as a Senior Design project. The following few sections go into detail on some of the other products that were found during research for the project.

One of the first articles found during research was one that listed "10 Lifesaving Location Devices for Dementia Patients" [1]. As can be deduced from the title, the article talks about 10 different devices used to keep track of dementia patients and ensure that they do not get lost. They each have their own features and differences compared to this project. Many of these features were taken into consideration, and the overall project was designed in a way to address both the benefits and pitfalls of some of these previous products.

The first is a program called Project Lifesaver. A person enrolls in the program and is given a transmitter. Once the person gets lost, their caregiver can contact their local Project Lifesaver agency, who will then track down the person via their transmitter. To enroll in the program, an initial $95 fee must be paid, in addition to paying $25 monthly

[2]. Unlike what is planned for the Wander Watch, it seems that caregivers who enroll their patients into the Project Lifesaver program cannot actually track the patient themselves. In addition, they must fill out forms for enrollment, which could take a long time to complete and process. The Wander Watch will instead allow people to purchase the watch and Hub components, set it up, create their own account on the app, and track the watch with no need of a middleman or form processing. The design of the watch itself is also meant to be comfortable to wear. For Project Lifesaver, the patient must wear their transmitter around their ankle. This can be uncomfortable for most people to wear. In particular, a heavy ankle device could also cause difficulty in walking for certain elderly patients.

The second is called Mindme [3]. This is actually two devices made for dementia and elderly patients in mind. One device is called Mindme Alarm, which is used to call emergency services when the user has fallen or gotten hurt. The other device is the Mindme Locate, which uses GPS to locate the user if they have wandered away from where they are supposed to be. A caretaker can even set up a pre-set location that the user should not move out of. The major downside to this product is that it is only available in the European Union. Obviously, a product that is only available in a particular region will limit how many people have access to it. Plus the Mindme devices both come with a rather large initial fee, around £85 or $90, and monthly subscriptions. This makes them an unreasonable option for those who cannot afford the price and subscription. It is also a rather small device that needs to be carried around by the person who is being monitored. Since it cannot be worn, it is possible for the person to lose the device. This can be a risk for people who may wander away from home, or who get lost often.

The third product on the list is the GPS SmartSole [4]. This is a wearable tracking device in the form of a sole that is worn inside a shoe. So the device is placed inside the shoe of the person that needs to be tracked. Their caretaker can keep track of their location using an app. If they become lost, the caretaker can contact emergency services to safely recover them. The SmartSole even allows the user to set up a Geo-Fence for the person wearing the sole. This is a pre-set location that the person needs to stay within, similar to the one for the Mindme device. When the person wearing the sole walks out of the Geo-Fence, the caretaker will receive an alert. This is sent either through the application or by text messages. From there, they can alert emergency services and get the person back home. This is similar to how the project team is planning to handle alerts and messages for the Wander Watch via the application. However, the device itself costs $299, while also having a one-time activation fee of $35 dollars, plus a $24.95 per month subscription. This is incredibly expensive compared to most of the other products and projects that have been researched, and is most certainly unaffordable for the average consumer.

The same company is also planning to sell a Bluetooth version of the SmartSole [5]. This version has a battery life of about a year, which is close to what the project team plans to have for the Wander Watch device. It also comes with a “CUBE gateway”, which is described as an antenna device. This device allows the caretaker to set up a perimeter much like the Geo-Fence. It is plugged into a power outlet, and then the caretaker can set up the perimeter. They can also track the sole using an app, much like the GPS version. Though this Bluetooth version is also much cheaper than the GPS version, it also has a device price, activation fee, and a monthly subscription. Also, due to using Bluetooth, it is meant to only be used indoors and within a limited area, as it is incapable of tracking the sole over a wide range. This is much the same reason why the project team ultimately decided not to use Bluetooth for the outdoor tracking of the Wander Watch. It is much more limited compared to GPS, and can make it difficult to track a person's location past a pre-set location.

The fifth device is called the Safe Link tracker [6]. It is a device that must be carried by the person that needs to be tracked. According to the company's website, it can act as a simple cellphone, and it also comes with an SOS button for emergencies. It sends its current location periodically to the Safe Link servers. The device is associated with a particular account, similar to what is planned for the Wander Watch. Then, caretakers can login to the Safe Link website to check the person's location. If they feel that they have gotten lost or wandered away, the caretaker can then contact emergency services and make sure that the person is safe. However, the device seems to be rather small. In fact, it must be carried by the person being tracked. This can be a problem since it could be lost or misplaced. The device is priced at $169.99 for the device, and $18.97 for the subscription. While this is not as expensive as the GPS SmartSole, it is still probably too much for most people to afford.

The sixth device is called the PocketFinder [7]. This device can be used for tracking elderly people, though the website also has versions of the tracker for children, cars, and pets. The device comes with its own app, through which caretakers can keep track of the user's position via frequent location updates. They can also be sent updates via notifications on their phone, text messages, and e-mail messages. An interesting feature of the PocketFinder is that it will only send updates about its locations when it is in motion. This is meant to conserve battery power. Unfortunately, this device is small in size. In fact, it appears to be the smallest device that has been found during research for this project. Though the PocketFinder website mentions that it can be put inside a pocket or strapped to a belt, it is still possible for the device to be loose and get left behind by the person who needs it.

The seventh device on the list is known as the Revolutionary Tracker [8]. On the website it is stated that this device is not only a GPS tracker, but also functions as a wearable watch and cellphone. This device allows for multiple people to be tracked by a single caretaker, by either their smartphone or computer. Similar to the Safe Link tracker, the Revolutionary Tracker also has an SOS button for emergencies. The watch itself is $199.00, and has a $25 per month subscription. Though this device is a wearable tracking device like the Wander Watch, it is rather expensive by itself. In addition to this, the Revolutionary Tracker does not seem to be available yet, as the website mentions

“(Shipping soon)”. This could be because the product is still being developed, and is only taking pre-orders of the device for now.

The eighth device on the list is actually a program known as the Comfort Zone Check-In [9]. This program offers its own device that can be used to track a person, but it also offers the option to track the person’s own cell phone instead. Whether the caretaker chooses to use the separate device or a cell phone, they can login to a web-based software to track a person’s location. However, the cell phone can only be tracked if it is on the Sprint network. Like with previous devices, having to carry a small device around runs the risk of losing the device. Even if someone were to choose the cell phone option instead, then that would mean make sure the cell phone is being carried by the person being tracked. Much like the other products discussed so far, the Comfort Zone Check-In is not a cheap option. The device is about $100 and its subscription is $14.99 per month. If a caretaker wants to use a cell phone that the person already owns, then they would only have to pay for a $9.99 monthly subscription. The cell phone option seems like a very good idea. There is no need to buy a separate device, and the subscription is cheaper than most of the other devices listed in this section. However, the fact that the cell phone must be running on the Sprint network means that only people who already own a Sprint phone will have the chance to use the Comfort Zone Check-In program.

The ninth device is called the GPS Tracking Locator Watch, and is offered by a company called Bluewater Security [10]. The watch has the same size and appearance as a normal wristwatch, and has a battery life of about 30 days. The watch itself has a panic button for emergencies, and comes with a receiver. If the person with the watch gets a certain distance away from the receiver, it will set off an alarm to show that the person has wandered off. The watch also allows for messaging, so that the caretaker can communicate with the person while trying to track them down. The messages are only accessible on the receiver, and it does not have an option to check the messages via a phone or website. The watch and receiver together are $599.99, which is the most expensive price that has been found during research. The GPS subscription for the device is around $35.00 per month. So not only is the device itself the most expensive device on the list, the subscription is also the most expensive of all the subscriptions that have been seen so far. Some people would not be able to afford the watch and receiver, let alone a $35 per month subscription.

The final item discussed on the list does not involve any technological devices being worn or carried for tracking. It is the MedicAlert Safely Home program [11]. A person is registered into the program and given an ID bracelet with their information. If they get lost, their caretaker can call the police and inform them that the person has been registered with the MedicAlert program. From there, the police can contact MedicAlert directly and get information about “possible locations” that the person could have gone to. This implies that the company does not use GPS to track anyone, nor is there a tracker in the ID bracelet. This seems like it could make it very difficult to actually find the person. In addition, this program only seems to be available in Canada, limiting its use to people within that country. Though the program only has a $60 per year subscription, it does not really differ much from traditional emergency services who do not use advanced technologies to track and recover lost people. Indeed, seeing this program has reaffirmed the team’s belief that using a GPS device like the Wander Watch would make tracking far easier for everyone involved.

Another product that was found during the research for the project was the Keruve Watch [12]. This is a watch that is tracked by a separate device known as the Keruve Receiver. This device can communicate with the watch via a mobile phone network, and track its location via GPS. However, both the Keruve Watch and the receiver have poor battery life compared to what is planned for the Wander Watch. Additionally, the website indicates that people must purchase the watch and receiver. It does not state how much the watch and receiver cost, but it can be assumed that the price is similar to the other products discussed in this section.

An interesting project that was discovered is the F*watch [13]. This is a free source project for designing a watch with GPS tracking capabilities. It is not sold as its own product. Instead, everything that someone would need to build the watch themselves is available for free on the F*watch website. This includes buying parts and building some of them with a 3D printer. The watch does not allow it to be track via other devices. Still, a lot of the hardware and software used in this project is used as a basis for the Wander Watch project.

Overall, the Wander Watch is a viable alternative to these previous products. The aim was to make it cheap enough that most people are able to buy it easily, and be able to set up and use the watch without a need for a middleman or additional services. Ideally, the Hub itself acts as a middleman of sorts for the watch and mobile app. Besides this, most of the products that were researched have a device that the person must carry with them. These devices tend to be small so that they can be carried easily. However, at the same time this makes them easy to lose, since there is no easy way of attaching them to the person that needs to be tracked. This is a big reason as to why it was decided to make this project a watch. It is comfortable to wear while also making sure that it cannot be lost by the user. It is functional, since it can display the current time and date, like a regular watch. The project team was taking into account all the positive features of these products, but also their downsides. It was ensured that the Wander Watch is able to avoid many of the pitfalls that have been observed in these other products. The team made sure that the Wander Watch project is as effective as it is useful, while still being as affordable as possible.

3.1 Benchmark Table

To conclude this section, a table has been made to show the major features of each previous product in comparison to the benchmark that has been planned for the Wander Watch. This is to show a visually convenient comparison between these products and what the project team plan to achieve with the Wander Watch project. This also gives a good way to track what the other products have done, and to check and see how well the project is meeting these benchmarks.

Though many of the previous products have cheaper monthly fees for messages, the project team believes that the reduced price of the Wander Watch will offset the cost of the message alerts. The project team has used this table as a reference when looking at certain parts and considering certain ways of implementing the project's functionalities. This is because it helps to show what has worked with other products, such as GPS versus Wi-Fi, while also showing what should probably be avoided in order for the Wander Watch to function properly while still being affordable. Most of what the team planned to avoid was centered on the cost of the watch and Hub itself. In order to keep costs down, the team made sure that the total price of all the watch and Hub's physical parts was less than most of the products found during research.

One interesting fact about these previous products is that most of them have about the same range. In fact, it is the same range as what is used for the Wander Watch. This proved a point that the decided range for this project was doable, since it had been seen in past products. It also showed that tracking technology has not improved that much since these other products. It is also interesting to see that most previous products don't rely on a Hub to keep track of whether a patient has gone out of range of a certain area. Then again, most of these products are meant to only track the wearer, which does not necessarily include checking to see if the patient is within certain designated boundaries. The Hub for the Wander Watch will aid in keeping track of a patient wearing the watch within a certain distance of the Hub. This should give the Wander Watch a competitive edge compared to products that are unable to track within a certain area. Overall, this table was a great asset to the team in terms of keeping the project's requirements in mind, while also keeping an eye on what previous products have done before.

Comparison Benchmarks Table

Name	**Range**	**Tracking**	**Device fee**	**Monthly fee**	**Hub Included?**	**User can track the patient?**
Wander Watch	10-15 ft.	SMS and GPS	$100 or less	$30 per month, 10 cents per alert	Yes	Yes, on app
Project Lifesaver	10-50 ft.	Radio tracking	$95	$25	No	No
Mindme	About 32 ft.	GPS and SIM	Around $90	Around $18.72	No	Yes, on website
GPS SmartSole	10-50 ft.	GPS	$299 + $35	$24.95	No	Yes, on app
Bluetooth SmartSole	Indoor only	Bluetooth	$49 +	N/A	Yes	Yes, on app
Safe Link Tracker	10-50 ft.	GPS	$169	$18.97	No	Yes, on website
PocketFinder	10-50 ft.	GPS	$129	N/A	No	Yes, site

						and app
Revolutionary Tracker	10-50 ft.	GPS	$199	$25	No	Yes, site and app
Comfort Zone Check-In	10-50 ft.	GPS and cellular	$99.99	$14.99 / $9.99	No	Yes, on website
Locator Watch	10-50 ft.	GPS and GSM	$599	$35	Yes	No
MedicAlert Safely Home	N/A	N/A	None	$60 per year	No	No
Keruve Watch	10-50 ft.	GPS	N/A	N/A	No	Yes, on receiver
F*watch	10-50 ft.	GPS	None	None	No	No

Table 1 Wander Watch Benchmark Comparison Table

4 Introduction to Tracking

A key feature of the Wander Watch is the ability to track the wearer remotely. The group split the tracking into two areas: indoor tracking and outdoor tracking. Both methods of tracking are described in the following sections of the document.

4.1 Indoor Tracking and the Hub

The Hub is located in the watch wearer's home and is responsible for keeping track of the patient while they are inside. The Hub sends a text message over the wireless internet to the caretaker's mobile phone once the watch and the Hub lose contact with one another. The communication between the watch and the Hub was to be set up through either RFID tags or Bluetooth. The group decided that Bluetooth was the best option for price and range of communication.

4.1.1 Radio Frequency Identification (RFID)

One idea for indoor tracking was to use RFID. Data is stored within the RFID tag's microchip and waits to be read. Once read, the tag's antenna receives electromagnetic energy from the RFID reader's antenna creating power which is to be harvested and used to send radio waves back to the reader. The reader would then pick up the tag's wave and interpret the frequencies. Similar to the way RFID tags have been utilized in the transportation industry on highway toll cars and subway passes, the patient would crossover the threshold and an electronic signal would be sent. [14]

A benefit to using RFID tags is that they can store up to 2 kilobytes of data, which would allow the project team to encode information, like the patient's name, home address, or emergency contact number, into the watch. Another benefit is the potential use for indoor localization using a network of tag readers. Unfortunately, the RFID tags limit the patient's movement to inside their house only, which could potentially make them virtual prisoners to the Wander Watch.

4.1.2 Bluetooth

Wireless Personal Area Networks (WPAN), more commonly known as Bluetooth technology, is a wireless communications method that transmits over short distances using low-power radio waves. Bluetooth operates on at frequency range between 2.402 GHz and 2.480 GHz on the unlicensed ISM band. [15]

Bluetooth technology communicates using a master/slave model network, known as a piconet. Within this piconet, eight devices can be connected simultaneously: one master and seven slaves. The master can communicate to any device within the piconet, sending data to the slaves, as well as receiving data from them. Slaves are not allowed to transmit to or receive from each other. Despite the size of the network, there is very little interference from devices within the piconet due to spread-spectrum frequency hopping. Frequency hopping makes it rare for more than one device to be transmitting on the same frequency, but allows for members of the same piconet to randomly hop frequencies in unison. This allows them to stay in touch with one another, while simultaneously avoiding other piconets. The Wander Watch's network consists of two devices. In this case, the Hub is the master and the watch is the slave. The watch is able to transmit its location to the Hub until it is out of range. [16]

As seen in the Table 2 below, there are three different classes of Bluetooth, which vary by range. The group decided that a Class 1 Bluetooth device best fit the needs of the Wander Watch. If the watch is taken outside of the 100 meter perimeter, it disconnects from the network. This action causes a notification message to be sent to the caretaker's phone.

Device Class	Transmit Power	Intended Range
Class 3	1 mW	Less than 10 meters
Class 2	2.5 mW	10 meters (33 feet)
Class 1	100 mW	100 meter (328 feet)

Table 2 Three Types of Bluetooth Classes

4.1.2.1 Bluetooth Candidates

The group reviewed four Bluetooth modules for the Wander Watch. The only requirements for the modules was that it need to be inexpensive, small in size, and reach a minimum distance of 100 meters. To narrow the search, the group only looked into Class 1 devices. The group also narrowed the search even more by looking into dual module Bluetooth devices. This allowed the project team to both transmit and receive if necessary.

4.1.2.1.1 BR-LE4.0-S2A

The BR-LE4.0-S2A costs $19.95 and was the first module the group looked into. This module utilizes Bluetooth Low Energy, having a very low power consumption of 27 milliamps and 19.6 milliamps for transmitting and receiving, respectively. It also has sleep mode with a power consumption of 0.9 microamps, which could be used to save power when the watch moves from indoor tracking to outdoor. The BR-LE4.0-S2A is 11.8 x 12.6 x 1.9 millimeters and has a signal coverage of over 150 meters line of site with an integrated antenna. The chip can be controlled using UART, SPI or USB data interfaces and comes equipped with free iOS and Android libraries and applications, which makes it a very attractive option since a phone application for the Wander Watcher was created.

4.1.2.1.2 Laird BT800

Costing around $10, the Laird BT800 has an operating voltage that ranges from 1.7 to 3.6 Volts. The transmitting max power consumption is less than 80 milliamps and has a sleep mode of 200 micro-amps. The coverage of this module is circa 100 meters and can be interfaced using a USB or 4 configurable general purpose input/output (GPIO) lines. The GPIO will allow more flexibility when integrating the BT800 into the Wander Watch.

4.1.2.1.3 Laird BT740

Another component researched was the Laird BT740, which was priced around $27. This was the largest of the three modules with the dimensions of 15.29 x 28.81 x 3 millimeters. With a max transmitting power consumption of 35 milliamps and an idle mode of 1.25 milliamps, the BT740 has a surprising coverage range of up to 1000 meters. The chip can be interfaced using UART or 8 configurable GPIO lines.

4.1.2.1.4 TI CC2560B RVMR

The final component looked into was Texas Instruments' CC2560B Bluetooth Controller was priced around $5. The module was 7.83 x 8.10 x 0.9 millimeters with a recommended operating voltage of 1.7 to 4.8 Volts. While in sleep mode the current consumption is typically 40 micro-amps. The CC2560B can be interface using UART. The downside of this chip is that it has a transmitting power of Class 1.5. Class 1.5 Bluetooth modules are not officially recognized in any standards. They have a longer range than Class 2 modules, but do not have enough power to be considered a Class 1.

In the case of the Wander Watch, the difference between Class 1 and Class 2 modules, which is 90 meters, will ultimately make all the difference. In order for the group to consider using the CC2560B, the group would need to run tests to examine where within the 90 meters the chip transmits. This is because the point at which the chip transmits a signal could affect the Bluetooth capabilities of the Hub. It could make the transmission faster than expected. However, it could also potentially make the transmission slower.

After looking at the cost of prototyping boards, the group came to the conclusion that buying an expensive board just to test a single chip would not be worth it in the long run. This is especially true if the range of the CC2560B does not meet the original specifications. This would be a waste of money for a single component that may not be used at all. This is something the group made sure to avoid doing at all costs for the Wander Watch project.

4.1.3 Conclusion

The following table was created and compiled according to the research done in the previous sections. This table helps to compare the different Bluetooth components to one another. It also helps to show which components have an advantage in certain areas. This also means that the table shows where each component may have a disadvantage against the other components.

	BR-LE4.0-S2A	**Laird BT800**	**Laird BT740**	**TI CC2560B RVMR**
Operating Voltage	2 - 3.6 V	1.7 - 3.6 V	3.3 – 5.0 V	1.7 - 4.8
Size	11.8 x 12.6 x 1.9 mm	8.5 x 13 x 1.6 mm	15.29 x 28.71 x 3 mm	7.83 x 8.10 x .9 mm
Baud rate	9600 - 460.8k		9600	37.5 - 4000 kbps
Signal coverage	150 m (L.O.S.)	up to 100 m	up to 1000 m	Unknown
Max Data Rate	1 Mbps	3 Mbps	2.1 Mbps	4 Mbps
Price	$19.95	$10.60	$26.65	$4.65
Interface	UART, SPI, USB	USB, GPIO	UART, GPIO	UART

Table 3 Bluetooth Module Comparison

After completing all of the research and doing a side by side comparison, seen above in Table 3, the group decided to create a decision matrix (Table 4) in order to assist in the final selection of the Wander Watch’s Bluetooth device. The weights of the decision matrix were carefully reviewed. Five major features of the Bluetooth modules were being scrutinized and weighted against on another: the operating voltage, the cost, the size of the module, the signal coverage, and the ease of integration.

The Bluetooth module’s operating voltage was decided to be the most important feature when choosing the device and was therefore given the highest weight, a five. Integrating the module into the Wander Watch with ease was considered to be important, but when being compared to the other four factors it was deemed to be more of a desire than a necessity, and was given the lowest rating, a one. Next lowest weight was assigned to the signal coverage. As stated before, the average Class 1 Bluetooth module has a range of 100 meters or more. The range TI CC2560B RVMR was unknown, but it was assumed to have range of at least 100 meters and was given an average rating of a three. The BR-LE4.0-S2A and the Laird BT740 were given the highest rating of because they have signal coverages of 150 meters and 1000 meters, respectively. Although these are higher than the Wander Watch’s required 100 meter minimum, the extended ranges of these modules were considered to be unpredictable until more testing could be done. The final two factors, cost and size, were equally important in the overall scheme of the watch however, cost was given the higher rating. The reasoning behind this was focused primarily on the fact that two Bluetooth modules were purchased. One module is inside of the Wander Watch and the second is located within the Hub. The size of the module was given the next lowest rating of a three. The Bluetooth device with the highest score was the module that best suites the Wander Watch's needs.

	Operating Voltage	**Cost**	**Size**	**Signal Coverage**	**Ease of Integration**	**Total**
Weights:	5	4	3	2	1	
			Bluetooth Modules			
BR-LE4.0-S2A	3	2	2	4	5	42
Laird BT800	5	4	4	3	3	62
Laird BT740	1	2	1	3	5	27
TI CC2560BRVMR	3	5	5	3	3	59

Table 4 Bluetooth Decision Matrix

1	2	3	4	5
Terrible	Bad	Average	Good	Excellent

Table 5 Decision Matrix Key

With the decision matrix complete, the Bluetooth module that received the highest score was the Laird BT800. Not only did the group find the vast coverage range of the BT740 to be unnecessary for the project, but the component was larger and more expensive compared to the other modules. The application friendly capabilities of the BR-LE4.0-S2A made it a notable choice, but in the end it was decided that cost was more important and went with the cheaper BT800. The size and price of the individual TI CC2560BRVMR chip was ideal for this project, but due to the ambiguity of the transmittal range the group decided to go with the BT800.

The final Bluetooth component chosen for the watch was none of the above candidates. In the end it was decided that the best component for the watch is the Sparkfun BlueSMiRF. Though not originally considered as an option for the Bluetooth, it was chosen because it has similar properties that the group was looking for and comparing, as well as a built in antenna, which saved both time and money during the build process. This is a class 1 Bluetooth module that allows for a reasonable range of indoor tracking. There is one module that is located inside the home and plugged into an MCU that is powered by the wall and the other is inside the watch. The group ended up with this choice because of the ease of implementation and connectivity to the application.

4.2 Outdoor Tracking

The group originally discussed tracking the watch outdoors using Bluetooth, but due to the limited range of tracking Bluetooth provides, the group decided to use Global

Positioning System (GPS) to track the wearer of the watch once they are out of sensing range of the Hub. Using GPS as a tracking alternative will provide both accuracy and high reliability.

4.2.1 Global Positioning System (GPS)

The GPS component is lightweight and small; in order to fit into the watch, and has a positional accuracy of at least 3 meters. The group was also looking for a component that can locate the wearer's position within 5 seconds of leaving their designated area. After completing the preliminary research, it was found that there were more requirements that the group needed to set before decided on a GPS module.

4.2.1.1 Geofencing

One GPS feature looked into as a potential substitute for the indoor tracking and a redundant measure implemented for safety is geofencing. A geofence is a virtual barrier that can be programmed and implemented using the global positioning system. Geofencing allows the user to establish triggers that alert them when a device enters or exits the boundaries defined. [17]

The size of the geofence can be determined by the caretaker using specified latitude and longitude coordinates and then choosing a radius. [18] This creates a circular area which encapsulates the watch wearer's house. An example of this can be seen below in Figure 1, where the geofence is represented by the highlighted, red circle. If the Bluetooth connection disconnects and fails to send an alert message to the caretaker, the geofence is used as a backup system. In the case of the Wander Watch, if the watch wearer steps outside of the designated boundaries a text message is sent to the caretaker's mobile device.

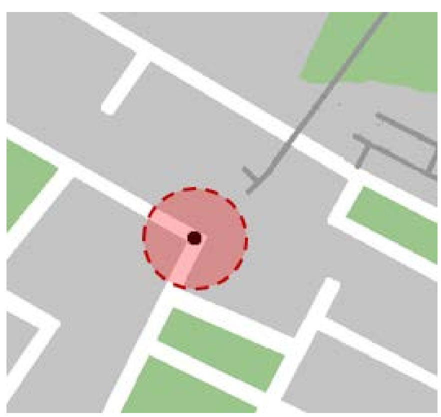


Figure 1 Example of geofencing

4.2.1.2 Active versus Passive GPS

In order to set up the designated area and be able to tell if the watch wearer has entered or exited it, the watch needed an active GPS tracking device, instead of a passive. The data on a passive GPS is stores daily activities throughout the day and then can be uploaded on a docking or charging station at home. The information is made accessible via a home phone line. Passive GPS would save power in the watch, but application would not be able to track the patient in real time. Active GPS would allow the project team to limit the movement to acceptable areas and collect the data in almost real time. The data is bounced down from a satellite and the coordinates are then accessed almost immediately. [19]

4.2.1.3 Data Transmission Protocol

The National Marine Electronics Association (NMEA) has developed a standard that makes interfacing the GPS module with the microcontroller simpler. The NMEA 0183 Standard implements communication between a receiver and a satellite and formats the response in the form of an ASCII character string, called a sentence, at 4800 bits per second. [20]

Each sentence begins with a dollar sign and ends with a carriage return/line feed character. The next two characters identify the talker. In this case, these characters are GP, which represent that it is a GPS position. The talker is followed by the message type, or three characters that will help determine how to interpret the data fields that follow. The Wander Watch is using GGA. GGA provides essential fix data that is essential to 3D location and accuracy data. All of the data fields that follow the message type are delimited by a comma. Any unavailable data is written by leaving a blank space in between the delimiters. If a checksum is supplied, the first character that immediately follows the last data field character is an asterisk. [21] The checksum after the asterisk is a two-digit hexadecimal number. An example of the formatting along with a brief explanation of what the strings mean can be seen in Figure 2 below.

```
$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47
```

Figure 2 GPS data output string

- Global Positioning System Fix Data: GGA
- Time: 12:35:19 Coordinated Universal Time (UTC)
- Latitude: 48 degrees, 07.038’ N
- Longitude: 11 degrees, 31.000’ E
- Number of satellites tracked: 08
- Altitude (Meters, above sea level): 545.4 M

Some of the other values seen above are unnecessary or extraneous. [20]

There are wide ranges of communications that are supported by GPS receivers. The choice between using one of the three serial protocols mentioned in the following sections are primarily based on which microcontroller the group decided to use.

4.2.1.3.1 SPI

Serial Peripheral Interface (SPI) is a synchronous data bus. A synchronous data bus uses separate lines for data and a clock to keep devices on both sides in sync with one another. SPI communicates using the Master/Slave relationship. There can only be one master in the system, usually the microcontroller, but the master can have multiple slaves. The position is of master is determined by whichever side have the clock. There are three forms of communications between them: MOSI, MISO, and SS. MOSI, or Master Out/ Slave In, is used when data is sent from the master to the slave. MISO, or Master In/ Slave Out, is the reverse. The slave sends a response to the master on the next set of clock cycles. The final communication type is called Slave Select (SS). SS is used when multiple slaves are present. The purpose of this communication line is to select which slave to talk and then alerting the slave to prepare to receive or send data. The model below in Figure 3 shows the four lines of communication in a master/slave relationship with multiple slaves. [22]

The group ended up choosing to use a Master/Master relationship rather than the previously mentioned Master/Slave. Both are able to transmit and receive information. This is used as the primary indoor tracking device. The issue of too many slaves is non-existent in the current layout.

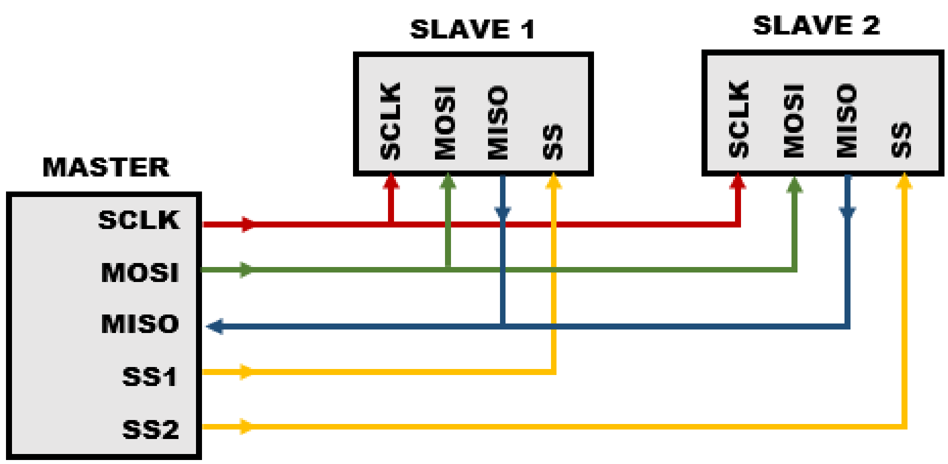


Figure 3 SPI Master/Slave Relationship with Multiple Slaves

Serial Clock (SCLK)	**Master Output/ Slave Input (MOSI)**	**Master Input/ Slave Output (MISO)**	**Slave Select (SS)**
Clock signals determined by the master	Data sent from master to the slaves	Data sent to master from the slaves	Selects which slave will be used

Table 6 Brief Description of the Four Types of SPI Communication

Overall, SPI is faster than asynchronous serial communications and can support multiple slaves. Unfortunately, this requires more signal lines and a separate SS line for each slave. In a large scale production with a vast amount of slaves, this could become an overwhelming problem, especially when trying to route the signals for a PCB layout. A solution to this problem is solved in the Inter-Integrate Circuit communication protocol.

4.2.1.3.2 I^{2}C

Inter-Integrate Circuit (I^{2}C) requires only two wires and not only can it support multiple slaves, but multiple masters as well. It consists of two signals, the SCL and the SDA. The SCL is a clock signal, which is generated by the master. The SDA is a data signal, which carries the data from master to slave and vice versa. I^{2}C has open drain bus drivers. Open drain means that the signal lines can be pulled low, but cannot be driven high. Each signal has a pull-up resistor, depicted in Figure 4, to restore a high signal when no device is pulling it low. The messages in I^{2}C communication protocol are broken into two parts, the address frame and the data frames. To begin a message, the master leaves the SCL high and pulls the SDA low. The address frame is the first part of the message in any communication line. It is made up of 8 bits in total, 7 bits for the address and 1 bit that indicates whether it is a read or write operation. The data frame follows the address frame. Data can be places on the SDA by either master or slave. Once all the data frames have been sent, the master generates a stop condition. The SCL completes a low to high transition and remains high and after the SDA transitions from low to high as well. This results in the termination of data flow. [23]

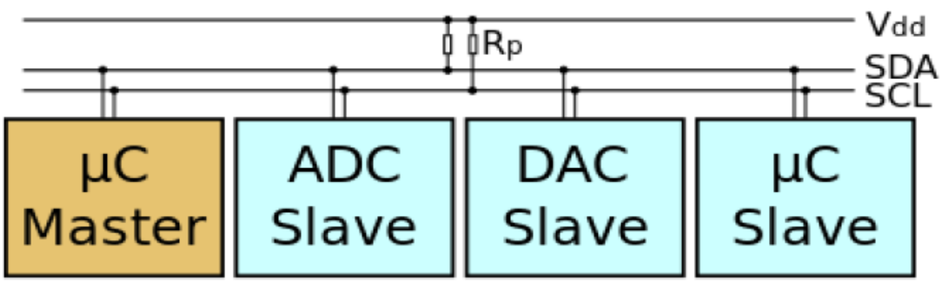


Figure 4 Pull up Resistor for I2C protocol [24]

4.2.1.3.3 UART

Universal Asynchronous Receiver/Transmitter (UART) is an asynchronous communication protocol, meaning two devices do not need to communicate based on an external clock. Instead, the devices use start and stop bits within the data being transmitted. UART acts as an intermediary between parallel and serial interfaces. The parallel interfaces consist of approximately 8 data lines and a few control pins. The serial side is made up of a transmitting line, TX, and a receiving line, RX. A simplified version of the previously described system can be seen below in Figure 5. UART is widely used and a flexible system however, it cannot share communication lines and it is slower than both the SPI and the I^{2}C. [25]

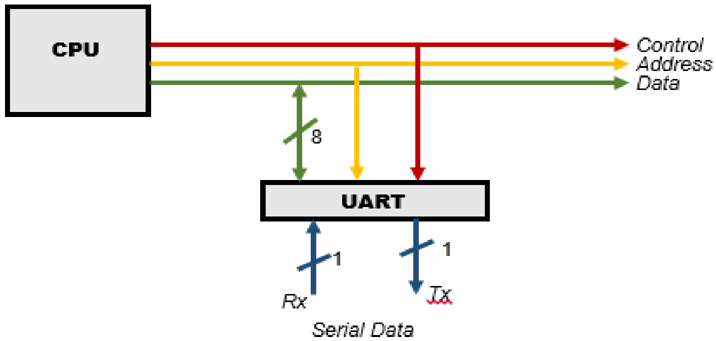


Figure 5 UART serial and parallel control lines

4.2.1.4 GPS Candidates

The group compared five GPS modules with each other: Telit Jupiter JF2, Adafruit MTK MT3339, Copernicus II, Venus638FLPx-L, and the CC4000-TC6000GN from Texas Instruments. Each module supported both active and passive GPS, and communicated using UART, SPI, or I^{2}C.

4.2.1.4.1 Telit Jupiter JF2

The Telit Jupiter JF2 was the first GPS module the group looked into and it had many attractive qualities. It was the cheapest amongst the five modules that were researched, costing only $17.50. At 1 gram, the Jupiter JF2 would add little to the overall weight of the watch and it was small at 11 x 11 x 2.6 mm. The Jupiter JF2 also had low voltage consumption of 1.79 to 1.9 Volts. It also scored high in the field of accuracy with positional accuracy of less than 2.5 meters and speed accuracy of less than .01 meters per second. Unfortunately, the max time to first fix (TTFF) 35 seconds and its update rate is 1 to 5 Hertz.

Pros	Cons
• Low cost	• Low update rate
• Small size	• High TTFF
• Lightweight	
• Low supply voltage	
• Supports UART	
• High positional accuracy	
• High speed accuracy	

Table 7 Telit Jupiter JF2 Pros and Cons

4.2.1.4.2 Adafruit MTK MT3339

The Adafruit MTK MT3339 had plenty of positive and negative features. It was affordably priced, an update rate of 10 Hertz, and accuracy of 3 meters or less. But the module was larger and heavier than desired with dimensions of 16 x 16 x 2.54 mm 4 g weight. Also the high voltage supply of 3 to 4.3 V. Like other modules, the max TTFF cold start was 35 seconds.

Pros	Cons
• Low cost	• Large size
• High update rate	• Heavy
• Supports UART	• High supply voltage
• High positional accuracy	• High TTFF
• High speed accuracy	

Table 8 Adafruit MTK MT3339 Pros and Cons

4.2.1.4.3 Copernicus II

After doing more research the group discovered that the Copernicus was the least attractive GPS module. It was the most expensive module, costing approximately $45 and it weighed 1.7 grams. Its dimensions were 19 x 19 x 2.54 millimeters, which made it the largest amongst all of the other components. The high voltage supply of 2.7 to 3.3 Volts did not seem to be a fair trade off with an update rate of only 1 Hertz and a maximum of 38 seconds for TTFF cold start.

Pros	Cons
• High speed accuracy	• High cost
• High positional accuracy	• Large size
	• High supply voltage
	• Low update rate
	• High TTFF

Table 9 Copernicus II Pros and Cons

4.2.1.4.4 Venus638FLPx-L

The Venus638FLPx-L is the lightest of the modules that was researched for this project, weighing in at approximately 0.03 grams. The dimensions of the chip were 10 x 10 x 1.3 millimeters. The voltage supply ranges from 2.8 to 3.6 Volt. The accuracy of the Venus is similar to the other chips. Two of its most attractive qualities is the update rate, which ranges from 1 to 20 Hertz, and its 29 second TTFF cold start.

Pros	Cons
• Small size	• High cost
• Lightweight	• High supply voltage
• High update rate	
• Supports UART	
• High positional accuracy	
• High speed accuracy	
• Low TTFF	

Table 10 Venus638FLPx-L Pros and Cons

4.2.1.4.5 TI SimpleLink CC4000-TC6000GN

The Texas Instruments SimpleLink is 10 x 9.3 x 2.3 millimeters and weighs 1.195 kilograms. Although the SimpleLink has a low supply voltage of 1.8 Volts, and a similar positional and speed accuracy of the other modules, the update rate is only 1 Hertz. This was not a good option for the group considering how heavy it is and how important a lightweight device is.

Pros	Cons
● Small size	● Heavy
● Supports UART	● Low update rate
● Low supply voltage	● High TTFF
● High positional accuracy	
● Low supply voltage	

Table 11 TI SimpleLink CC4000-TC6000GN Pros and Cons

4.2.1.5 Conclusion

After carefully reviewing the options, the group decided to create a decision matrix (Table 12) to help determine which GPS module best suited the project’s needs. Important features to consider are the power consumption, weight, and size. Although the overall cost of the project was a main concern, the group weighted cost the lowest because compromising the price would be worth it if the module is both small and lightweight. The accuracy of the GPS modules is also important, but it was weighted low because all of the modules have similar positional and speed accuracies. The update was also important, but it does not take priority over the top three. Each GPS module was given a score ranging from one to five for each feature. The key is located below the decision matrix in Table 13.

The Venus638FLPx-L had the highest overall score. Of all of its ratings, it scored the lowest in power consumption. Despite the fact that the group considered power consumption to be the most important of the features, the other aspects of the module make up for it. The Venus is both lightweight and small, so it scored high in both the weight and size categories. It has the highest update rate amongst all of the modules. Compared to other modules, the accuracy was about the same, but the price was considered midrange. Although it was not taken into consideration on the decision matrix, the Venus also had the lowest time to first fix cold start. This GPS module also communicates using UART, which was taken into consideration after selecting the microcontroller.

Though originally the Venus638FLPx-L was the GPS component of choice, the group ended up being able to combine components. The group came to the realization that the chosen GSM device also was a GPS device. This was the best option for the final product because it saved a significant amount of space and power. The final product contains the SIM808 GSM + GPRS + GPS Cellular Module. It has all the necessary GPS specifications needed for proper outdoor tracking combined with the ability to connect to a cellular network.

	Cost	Weight	Size	Power Consumption	Update Rate	Accuracy	Total
Weights:	1	5	4	6	3	2	
GPS Modules							
Jupiter JF2	5	4	3	4	3	4	78
Adafruit	4	1	2	2	3	3	44
Copernicus II	2	3	1	2	2	4	47
Venus638 FLPx-L	3	5	5	2	5	4	83
TI Simple Link	1	1	5	4	2	3	62

Table 12 Bluetooth Decision Matrix

1	2	3	4	5
Terrible	Bad	Average	Good	Excellent

Table 13 Decision Matrix Key

4.3 GSM

There are two types of radio systems that mobile phones use, GSM or Global System for Mobile Communications and CDMA or Code Division Multiple Access. GSM devices are used with SIM cards, and the SIM card is what carries all the information of the customer and helps to verify that it can be used on the network. CDMA devices however are only verified through a network-based list, which means that only phones sold through the carrier can be used. The technologies are both multiple access technologies so that one radio channel can have multiple phone calls or Internet connections on it. In GSM, it uses a time division system and CDMA uses code division. [26]

As stated above the main difference deals with how the data is converted into the radio waves that are received or sent from the cell phone or device. GSM divides the frequency bands into multiple channels. This allows for more than one user to use the tower at the same time. On the other hand CDMA networks use layers and once sent will undo them in the correct order. The way that GSM works allows more than one user to place a call through a tower at the same time. The United States is different than other countries in that there are more CDMA users here than anywhere else, it holds a huge part of the market. [27]

4.3.1 Uses

For the Wander Watch a GSM device was being considered to send the GPS data over a network carrier such as T-Mobile. GSM is what T-Mobile and AT&T use here in the United States. Another reason the group used a GSM device is because of the ability to swap devices easily through the uses of a SIM card that are discussed in the following section. The SIM card is what allows the device to be authorized to use the network. [28]

The reason a GSM device was needed was because once the user is to walk out of the house, they are disconnected from the Bluetooth Hub. This means that there was a need for the watch to be able to communicate with the caretaker and to communicate its location. This is where the GSM device came in because with this technology it had the ability to use the phone carrier towers to send and receive location information. Without the GSM device once the watch is out of range of the home, and therefore out of range of the Bluetooth connection there would be no way for it to communicate. That is why the GSM device was critical for the purpose of the Wander Watch.

There are some issues that were encountered in using a GSM device. One of them is the size of the device alone. The devices found were bigger than what the group had been hoping to use, especially because the type of device wanted is a small, lightweight, wearable device. The power consumption was also an issue because it was hoped the device would have a long battery life so that the user would not need to charge it all the time. However from what has been researched and implemented there is a need to charge the device every night, just like a cellphone or other mobile devices. As a result, this is a factor that was taken into consideration for certain GSM devices in the next section. In the end, it was realized that the GPS and GSM module could be combined, and that most of the devices researched were the size of a breakout board, and what would be used on the PCB would be much more reasonably sized.

4.3.2 Comparison

The following tables were used to compare a few of the devices that were researched for the Wander Watch project. An important factor was that the GSM device needed to be compact and rechargeable. It was also important that the watch had the capability to send messages and track using GPS with the aid of the GSM device built into it. These tables showed how each of these devices had their own advantages, but also some disadvantages that would hamper the project in meeting its goals.

Product	Adafruit FONA 808 Shield - Mini Cellular GSM +GPS for Arduino
Size	2.7" x 2.1" x 0.2"
Weight	19.0 g
Network Capability	2G SIM
Sends Messages	Yes

Receives Messages	Yes
Sends GPRS data	Yes
Receives GPRS data	Yes
Battery	LiPoly battery 500mAh+ LiPoly or Lilon battery
Charger Type	Via Arduino
SIM card	Yes
Accuracy	about 2.5 m
Cost	$49.95

Table 14 Adafruit FONA 808 Specifications

Advantages	Disadvantages
GPS included	Weight
Ability to send messages	Size
Accuracy	Must use Arduino

Table 15 Adafruit FONA 808 Advantages v. Disadvantages

As seen in Table 14, the Adafruit FONA 808 Shield was looked at as a possible GSM device because it has messaging capabilities, as well as GPS tracking capabilities. The GPS is quite accurate too, up to about 2.5 meters. However, the weight was a major concern. The size and battery were a big issue. The more time the watch spends off of the user's wrist and in the charging station, the less time it would spend on actually tracking the person's own location.

In the end, this was not chosen as the GSM device for the Wander Watch. The disadvantages discussed just now would make this device too much of a detriment to the watch's functionality. Table 15 displays the advantages and disadvantages in an easy to read format. [29]

The Adafruit FONA 808 Mini Cellular GSM with GPS Breakout was also considered for use for the Wander Watch. However, this was ultimately not chosen due to some of its disadvantages. Said disadvantages were found to be unacceptable for the Wander Watch project. As shown in the following Table 16, it has messaging capability just like the previous Adafruit component. It also has very accurate GPS capabilities.

Product	Adafruit FONA 808 Mini Cellular GSM +GPS Breakout
Size	1.7" x 1.7" x 0.3"
Weight	12.3 g
Network Capability	2G SIM
Sends Messages	Yes
Receives Messages	Yes
Sends GPRS data	Yes
Receives GPRS data	Yes
Battery	LiPoly battery 500mAh+ LiPoly or Lilon battery

Charger Type	MicroUSB
SIM card	Yes
Accuracy	About 2.5 m
Cost	$49.95

Table 16 Adafruit 808 GPS Breakout Specifications

Advantages	Disadvantages
GPS included	Medium weight
Ability to send messages	
Micro USB charger	
Accuracy	
Small size	

Table 17 Adafruit 808 GPS Breakout Advantages v. Disadvantages

This one is a better option overall because it is smaller, lighter, and has a more universal way to charge. Its charging method is via micro USB rather than by Arduino. More users are able to charge the device via micro USB, rather than having to deal with a method that they may not be familiar with. It is the same price as the previous component. Thus, the price of this component is not a major factor in comparing these two components. Still, the project team found other GSM devices that had better performance than this one. This is why it was not chosen for the project in the end. The following tables show how this device was still a relatively good option for other possible projects. Table 17 shows the disadvantages and advantages of this particular GSM device. [30]

The following Table 18 shows the third device that was considered for the project. This is the Adafruit FONA with mini cellular GSM Breakout uFL V1. It was very similar to the other two Adafruit devices looked at in the previous paragraphs and table. It can send messages over 2G SIM and has a micro USB charger. However, this device did not come with its own built-in GPS component. It was cheaper because of this lack of GPS component. This also makes its weight difference noticeably smaller.

Although the Adacruit FONA was a considerable option to use for the design, it was not the best option for the watch. This is due to the fact that the GPS component is very important to the final product. If the watch needs to have a separate GPS component, this would counteract the fact that this is lighter and cheaper by increasing the overall cost, which evened out its advantages in the end.

Thus, any GSM device that needs a separate GPS component to do any tracking will not be used for the project. This is why the FONA was not chosen for the Wander Watch project. Table 19 shows the advantages and disadvantages of this device. [31]

Product	Adafruit FONA - Mini Cellular GSM Breakout uFL Ver. 1
Size	1.75" x 1.25" x 0.3"

Weight	9.0 g
Network Capability	2G SIM
Sends Messages	Yes
Receives Messages	Yes
Sends GPRS data	Yes
Receives GPRS data	Yes
Battery	LiPoly battery 500mAh+ LiPoly or Lilon battery
Charger Type	MicroUSB
SIM card	Yes
Accuracy	N/A
Cost	$39.95

Table 18 Adafruit FONA uFL Version Specifications

Advantages	**Disadvantages**
Ability to send messages	GPS not included
Micro USB charger	
Small size	
Lightweight	

Table 19 Adafruit FONA uFL Version Advantages v. Disadvantages

The SM5100B by Sparkfun is a great option as the GSM device of the watch. This product is lightweight. It also has complete capabilities in regards to SMS text messaging. It is low power and quite comparable in size to the other products. It can hold a SIM card which is of course important and necessary for any GSM device. All things considered it is also quite affordable, though it is one of the more expensive devices needed for this project with a price of $39.95. [32]

Product	SM5100B
Size	35.0 x 39.0 x 2.9 mm
Weight	< 9 g
Network Capability	2G
Sends Messages	Yes
Receives Messages	Yes
Sends GPRS data	Yes
Receives GPRS data	Yes
Battery	Power supply Vbat = 3.3 V to 4.2V, Typical 3.6 V
Charger Type	Unknown
SIM card	Standard
Accuracy	Unknown
Cost	$39.95

Table 20 SM5100B Specifications

Advantages	Disadvantages
Ability to send messages	Unknown accuracy
Low power consumption	
Lightweight	

Table 21 SM5100B Advantages v. Disadvantages

The Quectel M66, as shown in Table 22 is another great option. The reason it is the best is because it is almost 10g lighter than most of the other options, as well as thinner and smaller. This will allow for a more wearable device for the end user. It includes a Bluetooth option, which the other products did not have. It is capable of sending messages which is of course one of the most important features considering that is the main uses of the device.

The Quectel M66 is also able to perform tracking using GPS. This was another one of the most important features of the Wander Watch. Thus, this component met a lot of the needs and requirements of the overall project.

Unfortunately, even though this seems to be the best option there was difficulty finding a supplier. There was also no pricing information out there that could be found. The manufacturer was contacted but there wasn't any response. Therefore this was not be the best part, because it was be too difficult to obtain for the project. [33]

Table 23 shows the advantages and the disadvantages of the Quectel component. This shows how useful this component could be, but its unknowns proved to be a detriment to the project.

Product	Quectel M66

Size	17.7 x 15.8 x 2.3 mm
Weight	1.3 g
Network Capability	2G
Sends Messages	Yes
Receives Messages	Yes
Sends GPRS data	Yes
Receives GPRS data	Yes
Unknown	Unknown
Charger Type	Unknown
SIM card	Yes
Accuracy	Unknown
Cost	Unknown

Table 22 Quectel M66 Specifications

Advantages	Disadvantages
Sends messages	Unable to find pricing
Smallest	Unknown accuracy
Most lightweight	
Low power consumption	

Table 23 Quectel M66 Advantages v. Disadvantages

The following Figure 7 shows the pin layout of the Quectel M66. These pins allow the Quectel to function with other components. This needed to be taken into consideration when planning out the connections between the different components. Of course, this would only apply if the project team was able to obtain this part at all.

Unfortunately, the Quectel manufacturers never responded to the inquiries and therefore the group had to choose another GSM component to use.

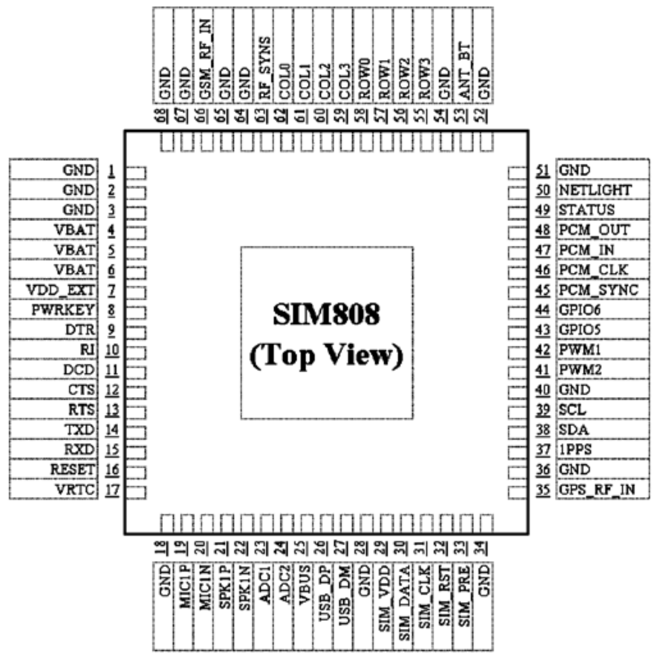


Figure 6 SIM808 PIN Layout (Courtesy of Adafruit)

The Table 24 below the figure is a list of the pins that are needed for the purposes of the Wander Watch. The table shows the PIN name, PIN number, and a short description of its use.

PIN Name	PIN Number	Description
VBAT	4,5,6	Main power supply
PWR KEY	8	Power on/off key
GND	1,2,3,18,28,34,36, 40,51,52,54,64,65, 67,68	Ground
TXD	14	Transmit data UART
RXD	15	Receive data UART
SIM_VDD	29	Power supply for SIM card
SIM_CLK	31	SIM clock
SIM_DATA	30	SIM data
SIM_RST	32	SIM reset
SIM_PRESENCE	33	SIM card detection
GPS_RF_IN	35	Connect GPS antenna

Table 24 Quectel M66 PIN Names and Descriptions

To decide the best option for the GSM device a decision matrix was used as show in Table 25. The weights are shown at the top of the table and the scale is based on a 1-5 scale with 1 being the lowest, and least likely option and 5 being the highest and best option. Power consumed was the most important factor because the group needed to conserve as much battery power as possible.

	Cost	Weight	Size	Power Consumed	Totals
Weight	1	3	2	4	
GSM Devices					
Mini Cellular GSM +GPS for Arduino	1	2	2	3	23
FONA 808	1	2	3	3	25
FONA GSM Breakout uFL v1	2	3	3	3	29
SM5100B	3	4	5	3	37
Quectel M66	2	5	5	3	39

Table 25 GSM Devices Decision Matrix

As the table shows the Quectel is the best option, in regards to size and weight is surpasses the other options easily. However, there is limited information about the cost and how to order the part. The manufacturer was not easily contacted and therefore the next best option according to the decision matrix was the GSM that was used for the project. The SM5100B is still comparable to the Quectel in regards to size and weight and is a good price. It is also easily ordered through Sparkfun and was the top pick.

Though originally the Sparkfun SM5100B was the top pick for the product the group ended up changing this in favor of another device. After careful consideration the group ended up deciding to use the SIM808 GSM + GPRS + GPS Cellular Module. This is the module that is on the few Adafruit breakout boards that are in the comparison tables above. The main reason for not using the Quectel device is that the manufacturer was unable to be contacted. But besides that, the SIM808 is a great choice for the Wander Watch because it combines the GSM and GPS components into one module. As mentioned above in the GPS section this was a way to save space on the final product. This module, in combination with a SIM card holder allows the watch to connect to the T-Mobile cellular network and gives the watch the ability to receive and send messages of notification of wandering as well as messages containing the coordinates of the wearer.

4.4 SIM Card

Subscriber Identity Module [34], or SIM, cards are necessary for GSM devices. The SIM card is inserted into the GSM device such as the Adafruit or Quectel before they will work. Without it, the device is not able to tap into any mobile networks, and it holds information for the phone or device. The SIM card is how the network knows that this device is authorized to use the network otherwise it will not work. For the device the group is using a SIM card so that the watch has the ability to be connected to a mobile network once the user is out of range of the Bluetooth in the house. It is important that it has network capabilities because the watch is able to text the location of the wearer to the caretaker so that the wearer can be located. [35]

SIM cards are very useful when used in a phone that needs to switch phones or plans. The SIM card has all the necessary information on it to make the switch very easy. This is a useful application for the Wander Watch if say an upgraded device was made and the caretaker purchased it for the wearer. The SIM card could be traded from the original device to the new device without the need to change plans or add contacts in again or activate it to connect to the network again.

4.4.1 Purpose

The reason that a SIM card was needed was because the watch had to have a GSM device in it. The SIM card, as stated above, is how the watch is allowed to use the network of whichever service provider chosen. In the United States the carriers that use GSM are T-Mobile and AT&T, and T-Mobile were considered to use for the mobile plan. The SIM card is the only way for the device to connect to the network, and without it, there is no way for the device to make calls, send SMS messages, or connect to mobile Internet services.

Using a SIM card caused no issues because depending on the device chosen for the end product, there is already a spot made for it in the device. This means that the size is not an issue. The other perk of a SIM card is that contacts can be stored onto it. This is important because the contact of the caretaker will need to be stored so that it can be easily accessed once the watch is programmed to send the message out.

As mentioned earlier many of the GSM devices researched were breakout boards. These boards already contained an area to insert a SIM card and therefore size was not considered. Also the majority of the SIM cards researched were able to be uses as standard, mini, micro, or nano size. However for the purposes of the project the group used a module and not the breakout board. This caused a need for a SIM card holder to be used in the design of the final product. The SIM card holder picked was standard size and therefore took up a large area.

4.4.2 Comparison

The following tables compare a few of the SIM cards looked at for use with the Wander Watch.

Table 26 shows the GSM 2G SIM Card from Ting and Adafruit. The perks of using this product was that the size of it is adaptable and can be changed to fit the needs of the GSM device chosen. Also it is a pay-as-you-go plan that runs on T-Mobile’s network. This was a good option because then the owner of the watch would only pay for what is used and hopefully there is not much use for the messaging system. Though it was one of the more expensive SIM card options up front, the pay-as-you-go and the use of the T-Mobile network balances out the cost. [36]

Product	GSM 2G SIM Card from Ting & Adafruit
Carrier	T-Mobile
Size	Standard/Mini/Micro/Nano
Text	Yes
Voice	Yes
Data	Yes
Type	Pay as you go
Cost	$9.00

Table 26 Ting & Adafruit SIM Card Specifications

Table 27 shows the ATT Wireless GO Phone SIM Card option. This was the cheapest option looked at. It also uses a pay-as-you-go plan which is the type that is best for the Wander Watch application, but it can only be used as a standard size SIM card. Also there is some information saying that AT&T may discontinue their 2G networks by 2017, so this might not be wise to use. [37]

Product	ATT Wireless GO Phone SIM Card 3G 2G/ EDGE
Carrier	AT&T
Size	Standard
Text	Yes
Voice	Yes
Data	Yes
Type	Pay as you go
Cost	$2.18

Table 27 ATT Wireless SIM Card Specifications

Table 28 shows the Simple Mobile – SIM Card. The Simple Mobile network has good coverage in Florida, which is of course where the group tested the Wander Watch. [38] This is important so that the product can be used anywhere throughout the state without worry of network issues. This SIM card has options for changing sizes as well in case the device chosen has a different size requirement. The cost was about mid-range in comparison to the other products and it has a pre-pay option, which was not the best option for the Wander Watch because it was not guaranteed how many messages would be used. [39]

Product	Simple Mobile - SIM Card
Carrier	Simple Mobile – T-Mobile
Size	Standard/Micro/Nano
Text	Yes
Voice	Yes
Data	No
Type	Pre-pay
Cost	$6.99

Table 28 Simple Mobile SIM Card Specifications

Table 29 displays the T-Mobile 3-in-1 SIM Starter kit. This option was the most expensive of the products compared at $15 dollars. There was a way to get the card for free with purchase of a plan. This was not the best option because a plan isn't really needed especially because the Wander Watch does not need voice or data options, though most of the SIM cards do provide the capability. [40] Also the upfront cost is a lot assuming that the group is not going to be sending too many messages. And the fact that there is no pay as you go option means it was not the best choice for the group.

Product	T-Mobile 3-in-1 SIM Starter Kit
Carrier	T-Mobile
Size	Standard/Micro/Nano
Text	Yes

Voice	Yes
Data	Yes
Type	Plan
Cost	$15/free with code

Table 29 T-Mobile SIM Card Specifications

The following Table 30 shows the decision matrix that was used to decide which option was best for the SIM card. The scale used is from 1-5 and 5 means that it is the better choice.

	Cost	**Plan**	**Size**	**Carrier**	**Totals**
Weight	3	1	2	4	
SIM Cards					
GSM 2G SIM Card from Ting & Adafruit	4	4	5	5	46
ATT Wireless GO Phone SIM w/EDGE	5	4	4	1	31
Simple Mobile	4	3	4	2	31
T-Mobile 3-in-1 SIM starter kit	1	3	4	4	30

Table 30 SIM Card Decision Matrix

As seen in the decision matrix, the SIM card from Ting and Adafruit was the best option. This was partly because of the carrier, because it runs on T-Mobile’s network. This SIM card also runs on a “pay-as-you-go” plan. The “pay-as-you-go” type plan was the best option for what the group was looking for the project because it is unnecessary to pay for too many messages, considering that hopefully there is no need for a huge number of messages sent because hopefully the patient will not be wandering too often.

5 Android Application

Though there are many different kinds of mobile applications, it was decided that creating an Android application was the best option for this project. There were many factors that led to this decision. These included the fact that it is cheaper overall to develop Android applications, the majority of these apps are made using Java, Android offers a lot of tools and features that are helpful for the project, most of the team members already had experience with Android application development, and there were already plenty of tutorials and guides for working with Android that helped with development.

5.1 Android Application

The first factor was that it was ultimately cheaper to develop an Android application than it was to develop an application for other mobile devices. One of the competitors against

Android are the mobile devices developed by Apple. For both Android and Apple, it is free to develop applications with either of their SDKs and code. Apple uses Xcode to develop their apps. However, to distribute these applications through the App Store, a yearly membership is required. [41] The cheapest membership option is $99, which can be a very significant chunk of the project's budget. In comparison, Android requires only a onetime $25 fee to distribute an application on the Google Play store. [42] Additionally, developing an Android application will ensure that it is accessible to a larger number of people. As of the second quarter of 2015, Android smartphones make up 82.8% of the market share of smartphones. This means that Android smartphones are the most widely available phone on the market, and thus it is the one that people are most likely to buy and own. In comparison, iOS smartphones only took up 13.9% of the market during the same time period. [43] Since it is less costly in the long run, and Android smartphones are more widely available, this was the first major factor in deciding to create an Android application for the project.

Overall, the project team was not concerned with having the Wander Watch application be available on devices besides Android. While having compatibility across multiple devices would have been a great thing to have, it was also very expensive. As mentioned above, there are fees that need to be paid before a developer is allowed to distribute their apps on a certain device. Larger companies with bigger budgets are able to release their applications across multiple devices. Examples of this are video game companies who can release their games on multiple consoles. They have enough money available to pay for both development of the game itself, and for the fees and licenses needed to publish the game on different consoles. As a college project, the Wander Watch did not have enough of a budget to justify the cost of developing its application for more than just Android. Besides the cost, there would have also needed to be a lot of time spent coding the application for multiple devices. Earlier it was explained that Apple uses Xcode for their applications. So then if the project's application was needed to be compatible with Apple devices, it would have needed to be rewritten in Xcode. Given how Android devices are much more common in today's market, and this project is working with a rather small budget, there was not much justification for developing the application for anything other than Android. It may be a future consideration to make other versions for future expansions of this project, but for now the project is only focusing on the Android market.

Many of the Android apps available today are made using the Java programming language. It is official supported for Android, and it is one of the top languages used for application development [44]. Java is taught at many universities for computer programming. In fact, most of the project team had some level of experience with Java thanks to some of the courses that were taken at the University of Central Florida. This included courses such as Object Oriented Programming, Computer Science II, and Processes for Object-Oriented Software. Additionally, the user interface of the application is created using XML. This language has some similarities with HTML, which most of the team were already decently familiar with.

If the project team had decided to create the application for other devices, such as Apple devices, it would have required learning completely new programming languages in a small amount of time. In the case of Apple, this would mean trying to learn Xcode. While Apple does provide plenty of resources and tutorials for Xcode, it would still take a considerable amount of time to learn what is needed for the project. In contrast, the project team was already familiar with the programming languages used most commonly for Android application development. So sticking with Android for the creation of the application goes well with the team's current skill sets.

Android also has a lot of features and components that are useful for parts of this project. As an example, it has the ability to add maps into an application for various functions [45]. This was used to set up the map in the application, which helps track the watch. The map is similar to Google Maps, so for a lot of users it should be familiar to work with. Having a map that users should already know how to use will help with the application's usability. Android also offers support for Bluetooth [46] and other location strategies [47]. Though the project team ultimately chose not to use Bluetooth, it was still great to see that Android has compatibility with Bluetooth. Besides this, some of the other location strategies are helpful in allowing the application to receive and properly display the watch's location.

Another helpful component is the ability to let the application give notifications. These notifications are the messages that an application can display on the user's phone, outside of the application itself [48]. This is useful in letting the user know when new alerts have been sent from the Wander Watch Hub to their app. Of course, Android also has the ability to allow apps to send and receive text messages. This was part of the alert system for the application, to ensure that the application is able to receive and display alerts from the Hub. All in all, Android already has a lot of pre-built components and features that are needed for the project's application, particularly for its major functions. By choosing Android, it helped ensure that the app's development was quick to complete.

In addition to Android and Java's useful features, at least one of the team members was already experienced with developing Android applications used Android Studio. This is the official IDE for Android applications. In COP 4331 Processes for Object-Oriented Software, they were a part of a group that developed a small application for the class using Android Studio. If the project team were developing the application for other devices, then it would be necessary to learn how to use other IDEs for different kinds of apps. For example, Apple has an entire toolset and IDE exclusively for Xcode [49]. Much like the concern with spending too much time learning a new programming language, it was decided that trying to learn a new IDE for application development would be more trouble that it could be worth. Using an IDE that the project team already had familiarity with allowed for the application to be completed in a timely manner.

The last major factor was that both Android and Java have a lot of tutorials available freely online. While many of the project team members have learned a lot about Android and Java programming through their classes, there are still plenty of topics that the team does not know a lot about. Fortunately, the tutorials for both Android and Java helped fill in the holes for this project's success. These tutorials cover topics such as working with Android Studio, how to include components such as maps and alerts into an application, and how to build a basic application that can serve as the foundation for more advanced apps. Through their research for this semester, the Wander Watch team had already taken note of several tutorials and articles that are helpful for this project.

For Android, Google has an extensive website dedicated to helping people with Android development. It even features a video course for getting started with building an application [50]. Meanwhile, the Java programming language has a large amount of documentation available on Oracle's website [51], in addition to the various tutorials that are available elsewhere online. Since Android is the most widely used smartphone OS at the time of this writing, it makes sense that there would be plenty of resources like this for all the facets of Android application development. In comparison, other devices have their own sets of resources and tutorials as well.

However, since they are not as widely used as Android, these resources are not as extensive or widespread for devices such as Apple's iOS devices. Having a wide range of resources and tutorials allowed the team to learn more about Android and Java, and how to better apply many tools and features to improve the application for this project. This is why Android and Java are very easy to get started with for school projects such as this, and how Android and Java can allow people to become better application developers.

Ultimately, it was found that choosing to develop an Android application for the Wander Watch project is the best choice. Besides the cheaper cost, Android is the most popular smartphone OS in today's market. This will ensure that the Wander Watch can be used by a large range of users. Focusing only on Android will help the team avoid spending unnecessary cost and time on developing different versions of the application for devices that do not have as large of a user base as Android.

Not to mention, most of this project's team are already familiar to Java programming and Android development. If there is something that is not known, the project team can easily refer to the many tutorials and resources available online for help. These reasons are why Android was ultimately chosen as the development platform of the Wander Watch's application.

5.2 Features

One of the main features of the application is that users can log in with their own account. This account is associated with their particular watch and Hub. While setting up the account on the application, the user needs to have the phone close to the watch that they want to use. While users are setting up their account on the application, they will be asked to enter their phone number, and the number of the watch. Each watch will come with a unique string of numbers, which ideally would be found inside of a user manual that comes with the watch. Thus, a message will be sent to the watch to tell it to save the user's phone number. This way, the watch will know where to send messages.

After this, the user is able to log into the account to check the status of the watch. They are able to view the watch's location, check any alerts they have received from the Hub, and change settings associated with the watch. These features are key parts of the project because all of the useful information in regards to setup and checking location are located in one convenient place. This will give the caretaker peace of mind knowing they can check on the patient at any time and know that they will be alerted very quickly if there are ever any issues. The following pages discuss the appearance of the application and how it will work.

5.2.1 Menus

When the application is opened, for the first time, it displays the login screen. They will have to create their account first. By clicking on Register a New Account, they will be taken to a screen that will help the user through the process of setting up their account. The user has to associate their account with a certain e-mail address, and create a password for their account. Then, the application tells the user how to connect their particular application account to their watch. The watch must know the user's phone number.

Once it knows which account to connect to, it sets up its connection to the watch. The application sends a message to the watch containing the phone's number, and the watch sends a test alert to make sure that the user can receive alerts on their phone. When the user confirms that they can see the alert, the application then allows the user to set up the geofence for the watch to use. As stated previously, the geofence is what the watch uses to tell when it is wandering. Once this is complete, the application lets the user know that the setup process is complete, and displays the main menu.

When the user opens the application after the setup process, it will display the login screen. Once the user logs in using their account email and password, they can see the main screen of the application. There are three main options on this screen: the Map, Alerts, and the Settings button. Clicking on each button will take the user to their respective menus. The following diagram shows how the user is able to navigate between the application's different menus. The map of course allows the user to view the location of the wearer. The alerts area has all of the messages in one area for easy viewing. The settings menu has options to customize the application to the users' needs.

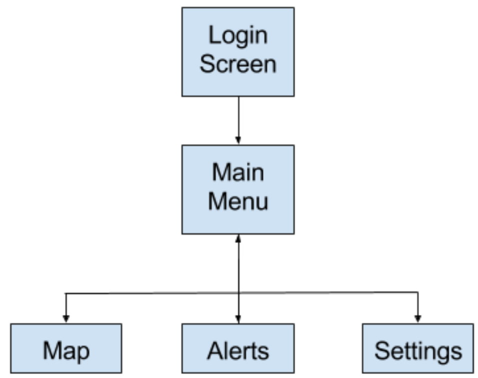


Figure 7 Navigation diagram of the application's menus

These menus are designed to be user-friendly. This includes them being easy to understand and visually appealing. The following diagram shows a mockup of what each menu was going to look like in the planning stage of the Wander Watch.

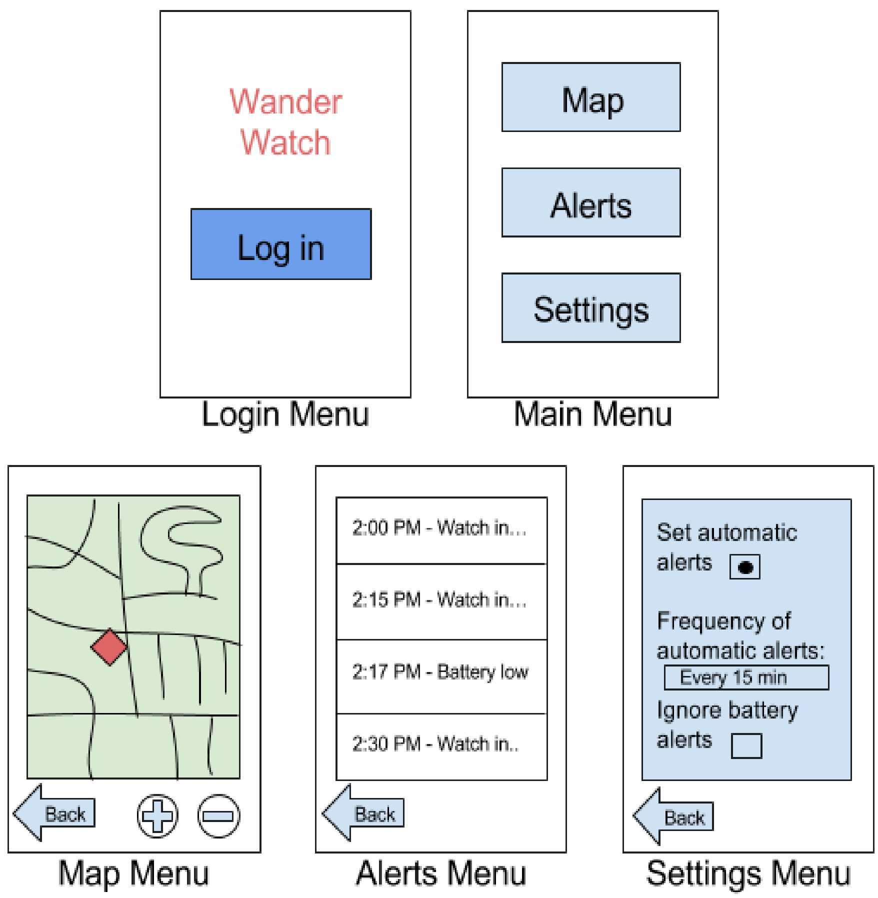
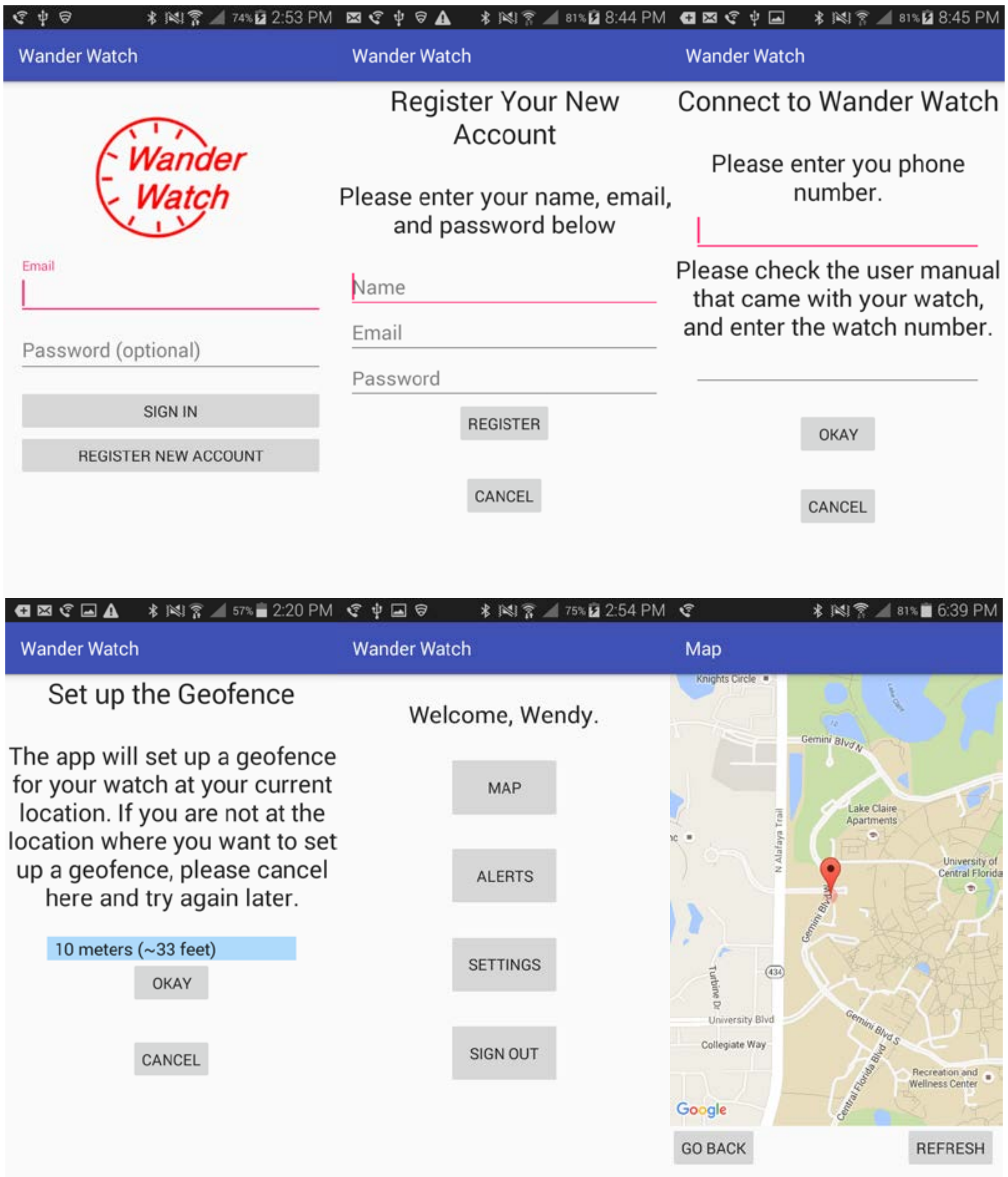


Figure 8 Mockups of the application menus

The final appearance of the menus is different from what is shown here, but the menus are still simple and easy to use. Through the Settings menu, the user can change settings related to the alerts, geofence, and their account. This is discussed in more detail in the next section.

The following figures show the final screens of the app. These screens include the screens for login, account set up, saving the phone's number for the watch, setting up the geofence, main menu, map, alerts, and settings.



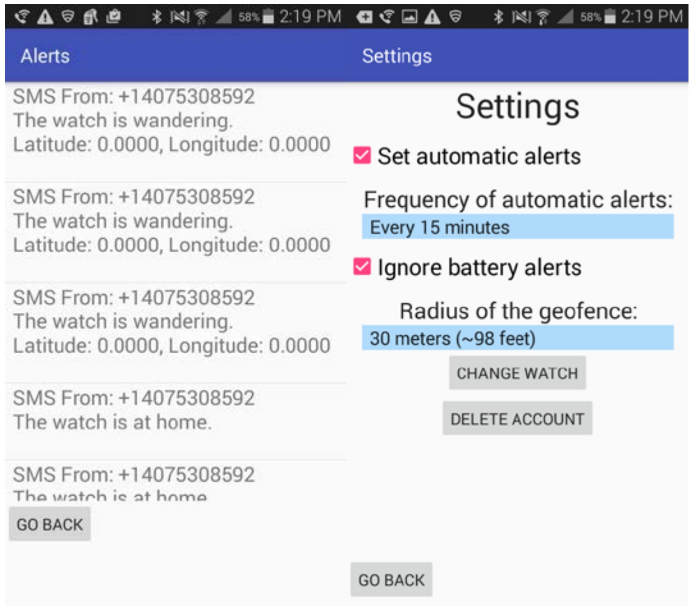


Figure 9 Final screens of the application menus

The choice to have these three menus was made because the app's functions could be organized into three easy-to-understand categories. It has been determined that these categories cover all of the application's needed functions. All of the currently planned functions fit into one of these three menus.

At this point in time, the application does not need more menus or content than the menus discussed here. However, it is possible that future expansions could add more functions that would require more menus to keep it properly organized. For the time being, these menus are sufficient for helping the user within the current scope of the project: keeping track of just one watch and one Hub at a time.

5.2.2 Functions

Each menu is associated with a particular set of features and functions. They are named such that the user is able to tell which menu is related to a certain activity. So then, the Map menu allows the user to view the app's GPS map and see the watch's location, the Alerts menu displays alerts from the Hub, and the Settings menu contains a list of settings that the user can change to meet their needs.

The Map menu shows the map which will track where the person wearing the watch is located. As mentioned earlier, this is displayed as a standard Google Map implemented into the Wander Watch application. The user will also be able to zoom in and move the map around, much like the standalone Google Map app. They will also be able to see the geofence on their home location. Changes in the geofence's radius from the Settings menu will be reflected in the map on this screen. When the watch is at home, it will show the marker in the middle of the geofence by default. This is because caretakers usually do not need to track patients within a home. When the watch is wandering, the application will check the alerts and get the watch’s location. With this information, the application will then update the marker to the watch’s current location, based on the messages sent from the watch.

Like the standard Google Maps application, the map that is in the Wander Watch application should be accurate within 10 to 30 feet. This will ensure that tracking down the person wearing the watch should be as effective and accurate as possible. At this point in time, the Wander Watch team does not plan to add anything extra to the Map menu. There may be more advanced or improved map features that can be added in future versions of the application.

The Alerts menu will show the user any alerts they have received from the Wander Watch. These alerts can tell the user if the watch has left the Hub radius, and at what time. This type of alert will inform the user to check the map for the person’s location, and to take steps to relocate the person. Each wandering alert will contain coordinates of the watch. On the map screen, the application extracts the coordinates from the latest message and displays the marker on the map accordingly. They will also be able to inform the user when the watch needs to be recharged. The Alerts menu will show up to the last 10 or 20 alerts.

The alerts themselves will also show a notification outside of the application itself. Many Android apps are able to display a small notification to inform the user that something has changed in the app. This happens even if the application is closed. These notifications we planned to be enabled for the Wander Watch app. This way, the user will know when they have gotten a new alert, and they do not have to keep checking the application themselves. However, since our initial research, we decided to send messages to the application via SMS text messages. Thus, most phones show alerts for new text messages, so there was no need to enable notifications specifically for the app.

The Settings menu will allow the user to change certain aspects of the application. Here, the user can choose if they wish to receive automatic alerts. These alerts are sent to the user at certain intervals. The user can also set when they want to receive the alerts. This could be every 15 minutes, every half hour, every hour, and so on. In addition, the user can chose to ignore alerts about the watch’s battery getting low.

Besides changing the settings related to the alerts, the Settings menu was originally intended to also let the user change the color scheme of the application. This is a relatively simple function to program into an Android application, though we were unable to have this fully implemented in the final prototype. The options for the color scheme change would have been such that the colors are not too harsh on the eyes. So this would mean no extremely bright colors, or color combinations that would cause the text to be hard to read. It is hoped that this feature might be added into future versions of this project.

The Settings menu will also give the user the option to change which watch is associated with the account. For the scope of this project, each account will only be able to handle one watch. The user can also delete their account, which will remove all information related to their account, and watch. They can create another account, but would have to redo the process of associating the watch with their application account.

These functions cover everything the user would need to utilize the application and be able to keep track of their loved ones with the Wander Watch. Future expansions of this project could include more functions, such as the ability to keep track of more than one watch. This is further discussed in a later section.

6 Hardware

After initializing a diagram of how the hardware of the watch is set up, the stages of research commenced. The goal of the research was to establish the type of parts needed for the watch's hardware, research and compare components available in the market today, develop a final parts list and draw up a price list for the development of the watch.

6.1 Watch Hardware Design

The hardware for the watch begins with the research stage, establishing design goals based on the design constraints set by the customer, and drafting a block diagram of the systems needed inside of the watch. Following the block diagram, research for the parts consisted of characteristics each component has and comparing those characteristics to the design constraints for the watch. This research has been shown in previous sections, and is discussed in further sections as well.

A select number of different components from different manufacturers were then compared to one another and selected for the final parts list based on meeting the design constraints needs. This is based on what the Wander Watch is expected to do, while also staying within a relatively low budget to ensure ease of purchase for most consumers.

6.1.1. Block Diagram

For the Wander Watch project, the team split the important sections of the watch into four categories: Power System, Display, Communication/GPS, and Applications. All of these sections are connected to the center, which is the microcontroller unit.

Inside the power system, the battery is the main source for power. Connected to the battery is the fuel gauge to calculate the amount of power remaining in the battery and displays that data as a percentage. Also connected to the battery is the battery charger which recharges the battery overtime with a micro USB port.

The power from the battery is regulated with two linear regulators. One for the MCU and one for the LED display. Going into the center of the watch is the microcontroller unit. The microcontroller unit houses the program for the GPS module and applications to be ran on the watch. The MCU also is linked to the display as far as controlling the backlight when the button is pressed. The LCD display shows the time, date, and battery percentage left with a battery symbol beside it.

For the communication section of the block diagram, the main component is the GPS module that communicates back and forth with the Hub. The Bluetooth portion of the project is what is in connection with the Hub. It is used as a backup to the geofence tracking. The Bluetooth contributes to the indoor tracking portion of the project.

The following figure, Figure 10, shows the block diagram of the watch hardware. It shows how each section has been divided between the group members for this project.

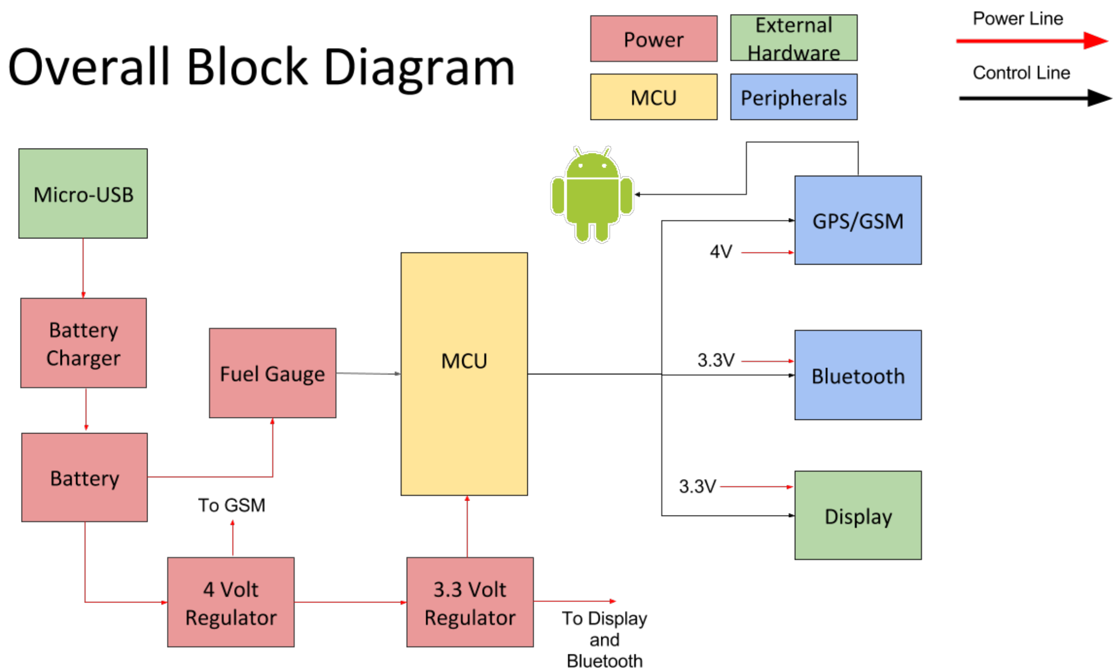


Figure 10 Block Diagram

6.1.2 Processors

The main processor choices that are best suited for this project is between the FPGA and the MCU. Where the FPGA excels in is its ability to change based on the user's needs, ability to process large amounts of data at a quicker rate compared to the MCU, and flexibility of operations that it can perform. The MCU excels in its low cost and low power usage. A MCU excels in low power consumption, ease of soldering, and the options available in today's market at a low cost.

Due to the scale of the project and the cost efficiency initiative, the MCU is a better choice over the FPGA. What makes it the better option in this application is the low cost of the chip and the wide variety of options available and ease of soldering onto a PCB.

Also, this design does not need the reprogrammable ability of the FPGA because the user does not need to change anything in the processor over time.

6.1.2.1 MCU

One of the important features the MCU must have for the watch is the ability to operate with the least amount of power possible to allow the watch to operate for as long as it can. Another important aspect for the MCU is the compatibility of the GPS module.

Not only is compatibility an important factor, but also the amount of power is needed to power the GPS module. One aspect that the MCU needs that could change the entire design approach of the watch is having Wi-Fi capability on the MCU. This allows for the MCU to operate as a hotspot when the watch leaves the designated area and triggers the alert for the caretaker. The watch inside the home can use the Wi-Fi provided in the house to conserve power.

One of the MCUs that the project group considered for the watch was the Texas Instrument MSP430FR5969, also known as Wolverine. The Wolverine is an ultra-low powered MCU that runs on an FRAM platform that utilizes an ultra-low power architecture along with the FRAM architecture that maximizes in speed at the lowest amount of power needed to operate. This MCU is best used in small devices that require low power operation. [52] Below, Table 31, is the specifications table for the Wolverine.

Specifications	MSP430FR5969
Voltage Range	1.8V to 3.6V
Active Mode	100 µA/MHz
Standby Range	0.4 µA
Off Mode	0.02 µA
RAM	64KB
RTC	0.25 µA (RTC is clocked by a 3.7-pF Crystal.)
Communication	UART, IrDa, SPI, I2C,

Table 31 Wolverine specifications

Another option from the ultra-low power range from Texas Instruments is the MSP430FG479. This MCU is also designed to be an optimal choice when developing products that need as much battery life as possible. This MCU comes with five low power modes that can be chosen via programming. However, this MCU, along with the previous MSP430 MCU, does not support Wi-Fi communications. [53]

As a result, if the team focuses on Wi-Fi communications between the parts of the project, choosing these MCUs would not be the best choice for the project’s requirements. It has even been found that there are other Wi-Fi capable components that would perform better than these two. However, most of those components are well outside of the project's intended budget and therefore were not the best choice for the group since keeping cost low is key. Below, Table 32, is a summary of the MCU.

Specifications	MSP430FG479
Voltage Range	1.8V to 3.6V
Active Mode	262 µA
Standby Mode	1.1 µA
Off Mode	0.1 µA
RAM	60KB
RTC	(No information available but included)
Communication	UART, IrDA, SPI, I2C

Table 32 MSP430FG479 specifications

Another MCU that has been considered is the ATmega328P. The ATmega chip is a low powered MCU that runs on a RISC based architecture that includes a 32KB flash RAM along with read and write capabilities. The MCU also includes five programmable power saving modes. However, this MCU does not support Wi-Fi communications. [54] Below, Table 33, is a summary of the MCU.

Specifications	ATmega328p
Voltage Range	1.8V to 5.5V
Active Mode	0.2 mA/MHz
Standby Range	0.75 µA (Typically)
Off Mode	0.1 µA (Typically)
RAM	32KB
RTC	32KHz
Communication	UART, I2C, SPI

Table 33 ATmega328p specifications

Similar to the previous MCU, another option for the MCU is the ATmega1284p. The ATmega1284p chip is a low powered MCU that falls in the same family as the previous chip. The ATmega1284p is a low powered MCU that runs on a RISC architecture that includes a 128KB flash RAM along with read and write capabilities. The ATmega1284p includes 3 SPI, 1 I2C, and 2 UART peripherals for other devices. Below, table 33 is a summary of the MCU.

Specifications	ATmega1284p
Voltage Range	1.8V to 5.5V
Active Mode	0.2 mA/MHz
Standby Range	0.75 µA (Typically)
Off Mode	0.1 µA (Typically)
RAM	128KB
RTC	32KHz
Communication	2 UART, 1 I2C, 3 SPI

Table 33 ATmega1284p specifications

If the Wi-Fi hotspot approach is chosen, then the Texas Instruments CC3200 is another option to consider for the Wander Watch project. The CC3200 is the first certified MCU with Wi-Fi connectivity. Thus it is able to perfectly handle Wi-Fi communications. The CC3200 is a wireless MCU that integrates a high performance ARM Cortex-M4 MCU running at 80MHz. The MCU is also UART compatible which is important for the Bluetooth component. [55] Below, Table 34, is a summary of the MCU.

Specifications	CC3200
Voltage Range	2.1V to 3.6V
Active Mode	59mA to 225mA
Standby Mode	825 µA
Off Mode	250 µA
RAM	256KB
RTC	32.768KHz Crystal
Communication	I2C, UART, SPI, GPIO multiplexing

Table 34 CC3200 specifications

Table 35 is a summary of all of the MCUs mentioned about with advantages and disadvantages compared side by side.

MCU	Advantages	Disadvantages
MSP430FR5969	Lowest in power consumption, variety of low power modes	Does not support Wi-Fi
MSP430FG479	Low power consumption, variety of low power modes	Does not support Wi-Fi
ATmega328p	Available open source material	Does not support Wi-Fi
ATmega1284p	Available open source material, more peripherals for devices	Does not support Wi-Fi
CC3200	Supports Wi-Fi communication	Highest in power consumption

Table 35 MCU advantages and disadvantages

6.1.3 Display

For the display, there was the option of choosing an LCD screen with the option of adding a backlight feature and the OLED screen where the brightness can be altered by pushing the button on the side of the watch. Due to the nature of how OLED technology works, a backlight won't be needed for the OLED screen.

6.1.3.1 Controlling Brightness

The LCD's brightness is coming from the backlight that was added to the display. The display will already be drawing current from the battery to be powered, the backlight will also need to have power being supplied from the battery as well. The result is the display consuming more power.

If the OLED display is used, the brightness can be controlled through the software. Ideally, the user can push a button on the side and temporarily increase the screen's brightness for about 3 seconds and the screen will return to its natural dim settings. Current draw is dependent on the brightness of screen over time. With the button plan, this will minimize the amount of time the screen is bright, thus reducing the amount of current draw. In the end the group opted to not include a button. Instead the watch display will always be on.

6.1.3.2 OLED vs. LCD

The reason why the Wander Watch team decided against the LCD screen and chose the OLED screen was the fact that the OLED does not require a backlight like the LCD screen needs so this in turn, less power consumption from the OLED screen vs. the LCD screen. The OLED has a better contrast in comparison to the LCD which will improve the quality of what is displayed onto the screen.

6.1.3.3 128 x 64 OLED

The first display option is the Adafruit Monochrome 1.3" 128x64 OLED graphic display. The display requires a 3.3V supply to turn on. For current draw, it depends on how bright and how long the OLEDs are lit. Typically, the OLED screen draws about 25mA on average. A nice addition to this OLED display is the PCB that the display includes. This display's PCB will include the communication port needed for the display to work in conjunction with the display. In addition, the display also comes with a regulator that will step down the voltage from the battery to the required 3.3V in order to power the display. Below, Table 36, is a summary Adafruit Monochrome 1.3" OLED display.

Specifications	**Adafruit Monochrome 1.3" display**
Size	128 x 64 px
Voltage	3.3V
Current Draw	~25mA
Color Depth	Monochrome
Communication	I^2C

Table 36 Adafruit Monochrome 1.3" display specifications

Another option in the market for an OLED display is the 128x64 OLED graphic display in white. This display comes as itself and not with the PCB like the Adafruit display. This display is more affordable compared to the OLED display kit from Adafruit. However, the OLED display will not include the PCB that has additional components to power the OLED display. Table 37 is a summary of generic OLED display.

Specifications	**Generic 128x64 OLED display**
Size	128 x 64 px
Voltage	3.3V
Current Draw	~25mA
Color Depth	Monochrome
Communication	I^2C

Table 37 Generic OLED display specifications

6.1.4 Power Supply

For the battery, what is important for the watch is the ability to power all of the components, have a large current draw, and the ability for the battery to recharge via a micro USB. Another factor to consider is the voltages needed for all of the components in the watch and to consider a range that can be used for the device. This battery is the main source of power and regulating that voltage for the components will come from regulators.

One option is to choose a disposable battery route, specifically, alkaline batteries. What is nice about disposable batteries is that there are a wide variety in the market, they're easily accessible, and are an affordable option when designing the power system for the watch. A con for the disposable battery is the fact they are in fact, disposable. Once the power has been drained, there is no method of recharging that battery, thus increasing cost and maintenance for the watch. Another situation is that the battery could almost be out of power at any time. Another con is the possible size of the batteries. Considering that the smaller options for the batteries are already thick in size, this can drastically increase the size of the watch which is not ideal for the customer and for the overall design of the watch.

Another option is a Lithium Polymer (Li-Po) battery. This is a very common approach in battery choices and is commonly found in electronic devices such as laptops and tablets. One advantage is that Li-Po batteries are generally much lighter while still holding their charge compared to other batteries on the market. Not only are they much lighter, Li-Po batteries hold a lot more charge compared to other batteries due to lithium being a highly reactive element. In addition to holding a lot more charge, Li-Po batteries hold their charge a lot longer compared to other batteries. One big disadvantage with the Li-Po battery is that the battery will degrade overtime.

Due to the design constraint of the watch, the Li-Po battery was the best choice for the project's design. The main reason why this was the best choice for the watch is the recharging capabilities that the battery is capable of doing. Another reason why is the longevity of the battery's life. This allows the project's product to last for a few years without replacing components. Compared to the alkaline batteries, the battery would have to be replaced a lot more often compared to only having to recharge the battery via micro USB.

6.1.4.1 How Much Voltage

Most of the devices needed for the project's watch can be powered by a 3.7V battery that have voltage regulators regulating the power needed for the components. One possibility that can change the battery that the project needs is the inclusion of the GSM/GPS. Typically GSMs needs more than 3.7V to power the component. If such a case occurs, then the battery will have to be changed to a 7.4V Li-Po battery. A con for this change is that it can increase the size and weight of the battery. However, adjustments were made for the GSM to be included in these changes.

6.1.4.2 Capacity

For the battery mentioned above, the differences in the batteries are the current draw of each battery. The differences in these batteries are size, weight, and price. The main aspect of the battery to consider is cost vs. power of each battery. Typically, more power would be a better choice for the overall design due to longevity of battery life in between each recharge.

The first battery being considered for the design is the 3.7V Li-Po battery. As all devices can be powered by this battery, with the exception of the GSM, this would be a good choice due to size, weight, and price being relatively low. The first battery in the market is the 3.7V battery with a capacity of 500mAh. While relatively small in terms of capacity, this battery meets the size constraints by being only 29mm x 36mm x 4.75mm and weighing 10.5g. This battery is priced at $7.95 on Adafruit. Table 38, is a summary of the pros and cons of this battery.

Pros and Cons of the 3.7V 500mAh Li-Po battery
Pros: • Small in terms of size and weight • Meets size constraints based on the project specifications • Low price
Cons: • Very limited capacity

Table 38 3.7V 500mAh Li-Po battery pros and cons

Another option to consider when choosing a battery for the watch is to increase the capacity of the battery by going up to 2000mAh. This 3.7V battery is larger, but will last a lot longer compared to the 500mAh battery. This battery is sized at 100mm x 76mm x 7mm and weighs 54g. This battery is priced at $12.50 on Adafruit. Table 39 is a summary of the pros and cons of this battery.

Pros and Cons of the 3.7V 2000mAh Li-Po battery
Pros: • Much larger capacity compared to the 500mAh battery • Relatively low price • Better suited for power constraints based on specifications
Cons: • Larger in size, needs to be considered during case design • Considerably heavier compared to the others

Table 39 3.7V 2000mAh Li-Po battery pros and cons

One other option available for the Li-Po battery is the 3.7V 1200mAh capacity battery. Being the middle of the two previous options, it can sustain a healthy battery life for the watch which will help meet the longevity specification of the watch. The battery is 63mm x 37mm x 7mm and weighs 25g. Below, Table 40, is a summary of the pros and cons of this battery.

Pros and Cons of the 3.7V 1200mAh Li-Po battery
Pros: • Much larger capacity compared to the 500mAh battery • Relatively low price • Better suited for power constraints based on specifications
Cons: • Larger in size, needs to be considered during case design

Table 40 3.7V 1200mAh Li-Po battery pros and cons

While the 7.4V battery is an excellent option for the watch, the size constraints on the watch limit the size and weight of the battery considerably due to the watch needing to be a wearable device. Thus, the 7.4V battery is not the best option due to the size constraints of the watch.

6.1.4.3 Recharging Capability

With the power system for the watch, it must have a method of recharging the Li-Po battery. A battery charging IC will allow the battery inside the watch to be rechargeable via a micro USB connection. Ideally, this component is small in size, with minimal leakage current, and retains high voltage and current accuracy.

One battery charging IC that is considered for the watch’s hardware is the Texas Instrument BQ24232 USB friendly battery charger. This series of battery charging ICs specializes in linear chargers and power path management for small portable devices. This IC is USB compliant and can operate with a USB connection. [56] Table 42 is a summary of the features in the BQ24232.

Specification	BQ24232
#Series Cells	Single Cell
Battery Charge Voltage	4.2V
Current Input Range	100mA – 500mA
Size	3mm x 3mm
Special Features	Over Voltage Protection, Power Path

Table 41 BQ24232 features

Another battery charging IC in the market is the Microchip MCP73831. Another battery charging IC similar to the Texas Instruments IC where the chip specializes in performing battery charges in small devices. In addition, this IC is also USB friendly where the battery can be recharged through the USB. [57] Table 42 is a summary of the main features in the MCP73831.

Specification	MCP73831
#Series Cells	Single Cell
Battery Charge Voltage	3.75V to 6V
Current Input Range	15mA to 500mA
Size	2mm x 3mm
Special Features	Under Voltage lockout, reverse discharge protection

Table 42 MCP73831 features

Table 43, which is shown below, is a summary of the advantages and disadvantages between the two batteries charging ICs. This shows how choosing one or the other could have major effects on the Wander Watch's ability to carry out its intended functionality.

Component	Advantages	Disadvantages
BQ24232	Over Voltage Protection, Fixed voltage charge	Limited options in Current Input
MCP73831	Fewer components needed	No overvoltage protection

Table 43 BQ24232 advantages and disadvantages

6.1.4.4 Fuel Gauge

The fuel gauge needs to accurately determine the amount of battery remaining inside the watch while minimizing the amount of current draw needed for the component to run at all. It also needs to be as small as possible to minimize the space needed to take up on the PCB. Size is an important factor in keeping the watch to a certain size and weight, and thus making sure that it is comfortable to wear.

One of the fuel gauges being considered is Texas Instruments BQ28Z610 Battery Fuel Gauge with Integrated Protector for 1-2 Series Packs. This fuel gauge is a highly accurate, enables charge control and cell balancing. The cell balancing has an internal bypass that allows to optimize the battery life. [58] The table below, Table 44, summarizes the main features of the fuel gauge.

Specification	BQ28Z610
#Series Cells	Single Cell, 2-4 Cells
Battery Capacity (Min)/(Max) (mAh)	100/14000
Communication Interface	I^2C
Size	4mm x 2.5mm
Special Features	Battery Management Unit Cell Balancing

Table 44 BQ28Z610 features

Another fuel gauge that is available in the market is the Texas Instruments BQ27542-G1. This fuel gauge is a microcontroller peripheral that is able to provide fuel gauging for single cell lithium ion batteries. This fuel gauge does not require much development in firmware in order for the component to function properly. [59] Below, Table 45, is a summary of the main features of the BQ27542-G1 fuel gauge. This gives a good idea of how the features of this gauge could work with the rest of the watch.

Specification	BQ27542-G1
#Series Cells	Single Cell
Battery Capacity (Min)/(Max) (mAh)	N/A /14500
Communication Interface	HDQ and I^2C
Size	2.5mm x 4mm
Special Features	Provides information on remaining battery capacity, run time remaining, and battery voltage

Table 45 BQ27542-G1 fuel gauge features

Another option for the fuel gauge is the MAX17043 from Maxim Integrated. This standalone fuel gauge measures Li-Po battery capacity with small devices in mind. The MAX17043 uses an I2C interface to communicate with the MCU. Below, Table 46, is a summary of the MAX17043 fuel gauge.

Specification	MAX17043
#Series Cells	Single Cell
Battery Capacity (Min)/(Max) (mAh)	N/A /6000
Communication Interface	I^2C
Size	2mm x 3mm
Special Features	Provides information on remaining battery capacity, run time remaining, and battery voltage

Table 46 MAX17043 fuel gauge features

Another fuel gauge being considered for the watch design is the Texas Instruments BQ27010. This stand-alone fuel gauge measures Li-Po battery capacity built with small portable devices in mind. [60] Table 47, is a summary of the main features of the BQ27010 fuel gauge.

Specification	BQ27010
#Series Cells	Single Cell
Battery Capacity (Min)/(Max) (mAh)	300 / 6000
Communication Interface	HDQ and I^2C
Size	3mm x 4mm
Special Features	Five low power operating modes, Automatic capacity reduction with age

Table 47 BQ27010 fuel gauge features

6.1.4.5 Voltage Regulator

An important parameter considered for the power system for the watch was choosing what type of voltage regulator to choose for the power system. While the switching regulator is the first option to consider when regulating the power from the battery, the linear regulator is the better option for the two because the battery being used for the watch is close to the regulated voltage the team is seeking. With a small difference in voltages, the low dropout regulator excels in this application.

One design approach for regulating the voltage for various watch components is to run regulators in parallel from the battery to distribute current evenly for various components that need various voltage levels for the different components inside the watch. A good reason for this design approach is the fact that all of the components will have its own voltage level satisfied based on the datasheets of the components. A con for this design approach is overworking the battery inside of the watch.

Another option is to have one regulator inside of the watch that will regulate the supply voltage in a range that satisfies all of the components inside of the watch. What is good about this design approach is the overall all hardware design not needing as many components to power the system. A con for this design approach is the fact that some of the components may not work optimally due to the voltage supply not being at the suggested level based on datasheets for each component.

If the watch will use a 7.4V battery to satisfy the voltage requirement of the GSM, then the watch will need regulators that will drop down to the required voltage that the GSM needs, and a larger voltage drop for both the MCU and the display. This makes the LDO regulator an unfavorable choice because the voltage drop for all the components is a lot higher.

Another option to consider is to include a switching regulator to have a boost option for the GSM. A good reason for including a boosting regulator for the GSM is the option of having the 3.7V battery for the watch. A con for this regulator is the noise that is produced from the switching regulator.

One part that can be considered for regulating the battery supply for the components inside the watch is the Texas Instruments LM3480. This linear regulator specializes in working with low power devices and small devices. [61] Table 48 is a summary of the LM3480 linear regulator.

Specification	LM3480
Input Voltage	Up to 30V
Output Voltage	3.3V to 15V
Output Current	100mA
Noise	100 µVrms
Size	2.92mm x 1.30mm
Special Features	Overcurrent Protection, Thermal Shutdown

Table 48 LM3480 linear regulator specifications

Another linear regulator for the watch is the Texas Instruments TPS799. It is an easy-to-use power management IC. The TPS799 uses minimal amount of components in its design which helps reduce the total cost of the design. Below, table 49, is a summary of the TPS799 linear regulator.

Specification	TPS799
Input Voltage	2.7V to 6.5V
Output Voltage	1.2V to 6.5V
Output Current	200mA
Noise	29 µVrms
Size	3mm x 3mm
Special Features	Minimal design

Table 49 TPS799 linear regulator specifications

Another linear regulator for the watch is the Texas Instruments TPS74701. It is an easy-to-use power management IC. The user-programmable soft-start minimizes stress on the input power source by reducing capacitive inrush of current on start up. This regulator is ideal for devices or components that have a special start-up sequence. [62] Below, Table 50, is a summary of the Texas Instruments TPS74701.

Specification	TPS74701
Input Voltage	0.8V to 5.5V
Output Voltage	0.8V to 3.6V
Output Current	500mA
Noise	20 µVrms
Size	3mm x 3mm
Special Features	Soft Start, Overcurrent Protection, Thermal Shutdown

Table 50 Texas Instruments TPS74701 specifications

A switching regulator that can be used for boosting the voltage for the GSM is the Texas Instruments TPS61088. The TPS61088 is a high power density, fully synchronous boost converter with an 11-mΩ power switch and a 13-mΩ rectifier switch. This switch provides a high efficiency and small size solution in portable devices. This boost converter also comes with an adjustable switching frequency that can be changed by the user. The TPS61088 has efficiency as high as 91% at 3.3V input voltage, 9V output voltage, and 3A output current. However, it is unfortunately rather larger in size. [63] Table 51 is a summary of the TPS61088 boost converter. This shows specifically why the size is an issue with this component.

Specification	TPS61088
Input Voltage	2.7V to 12V
Output Voltage	4.5V to 12.6V
Output Current	3A
Switching Frequency	200KHz to 2.2MHz
Efficiency	91%
Size	4.5mm x 3.5mm
Special Features	Adjustable current limit, Light load efficiency, synchronous rectification

Table 51 TPS61088 boost converter specifications

Another boost converter available in the market to boost the input voltage of the GSM is the Texas Instruments TPS61093-Q1. The TPS61093-Q1 is a fixed-frequency boost converter designed for high integration and high reliability. When the output current exceeds the overload limit, the isolation switch of the IC opens up to disconnect the output from the input. This disconnection protects the IC and the input supply.

However, this component also has its own set of disadvantages. It is less power efficient compared to the other components. It also has fixed frequency switching, which could become an issue for the Wander Watch’s expected functionality. [64] Below, Table 52, is a summary of the TPS61093-Q1 boost converter.

Specification	TPS61093-Q1
Input Voltage	1.6V to 16V
Output Voltage	1.7V to 17V
Output Current	1.1A
Switching Frequency	1.2MHz
Efficiency	Up to 88%
Size	2.5mm x 2.5mm
Special Features	Output Discharge

Table 52 TPS61093-Q1 boost converter summary

Another switching regulator to consider for the watch design is the TPS63050 from Texas Instruments. The TPS63050 is a switching regulator that can operate as a low drop out regulator and a boost converter based on the status of the Li-Po battery. Below, Table 53, is a summary of the TPS63050.

Specification	TPS63050
Input Voltage	2.5V to 5.5V
Output Voltage	2.5V to 5.5V
Output Current	1.0A
Switching Frequency	2.5MHz
Efficiency	Up to 94%
Size	2.5mm x 2.5mm
Special Features	Both a linear regulator and boost converter

Table 53 TPS63050 switching regulator summary

Table 54, which is shown below, is a summary of the advantages and disadvantages for the regulators mentioned above. This table shows how each regulator could benefit the project. Still, it also shows how a particular component could hamper the watch's functionality, or otherwise not allow it to properly function at all.

Component	**Advantage**	**Disadvantage**
LM3480	Overcurrent Protection, Wide range of voltage outputs	High noise output
TPS799	Minimal in design	High noise output
TPS74701	Overcurrent Protection, low noise output	Larger in size
TPS61088	Adjustable current output, high efficiency	Larger in size
TPS61093-Q1	Wide voltage output, smaller in size	less efficient, fixed switching frequency
TPS63050	Both a linear and boost regulator	Low output current

Table 54 Advantages and disadvantages of regulators

6.1.5 Final Parts List

Originally, the group decided to use the CC3200 due to the MCU and the Wi-Fi module being included onto one chip making the number of chips on the PCB fewer than the other options. However, the implementation of Wi-Fi was much more difficult compared to Bluetooth. Also, the amount of components that are needed for the CC3200 were also more expensive than using a hub for indoor tracking. The group decided to switch to the

ATmega1284p for the amount of peripheral needed that the chip can support, the cost being reduced overall by switching to this chip.

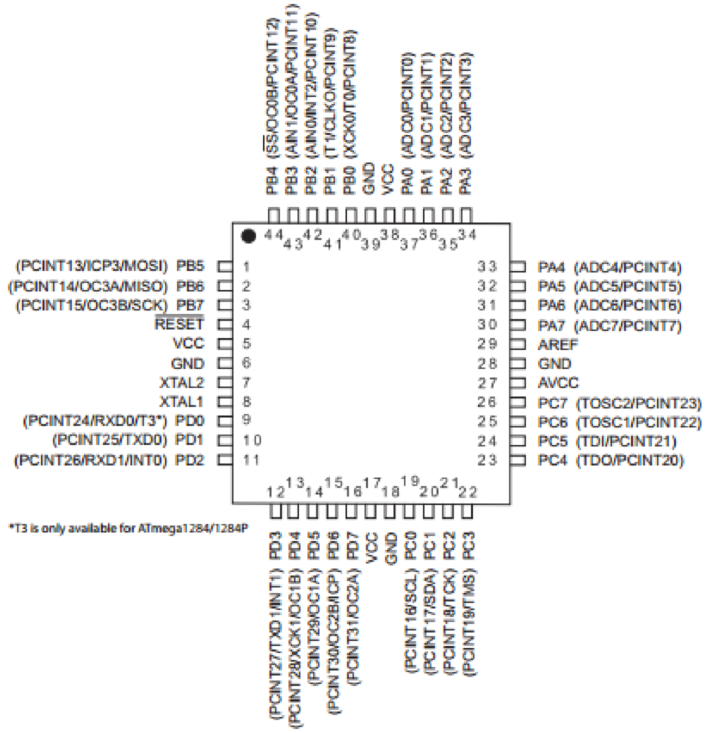


Figure 11 ATmega1284p pin layout (Courtesy of Texas Instruments)

The communication peripherals that the ATmega1284p MCU hosts are the two UART pins, one I2C pin, one SPI pin, and up to thirty-two programmable GPIO pins. Table 55 is a summary of the components and the communication peripherals assigned to the components.

Component	Communication Peripheral
Display	SPI
GPS/GSM	UART
Bluetooth	UART
Boot Mode	JTAG
Fuel Gauge	I^2C

Table 55 MCU communication protocols

When the group uploaded the software for the chip, the method done was using an Arduino UNO as an FTDI uploader and then uploading the code onto the blank chip with the Arduino IDE and then the software for the watch.

For the display, the overall design and product benefited more from the generic OLED display versus the Adafruit OLED display because the cost of the Adafruit OLED display is much higher compared to the generic OLED display with a developed PCB and components which reduced the overall cost of the developed watch.

For the battery, the group originally decided on the 3.7V Li-Po battery with 2000mAh capacity due to the large capacity available to allow the watch to operate for a longer period of time. In the end, the group decided to switch to the 1200mAh Li-Po battery due the size and weight constraints on the watch specifications. While this is a significant decrease in capacity, the 1200mAh battery is still capable of maintaining a battery life to last the required amount of time while reducing the size and weight of the overall watch. This also allowed for the final PCB to be smaller in size.

The battery charging IC that was originally selected for the watch design was the BQ24232 due to the USB friendly features and the charge current available for the battery. However, the group changed the battery charging IC to the MCP73831. The MCP73831 required fewer components compared to the BQ24232 which reduced the overall cost of the watch and reduced the amount of parts needed on the PCB. Below, Figure 12, is an example of how the battery charging IC is designed.

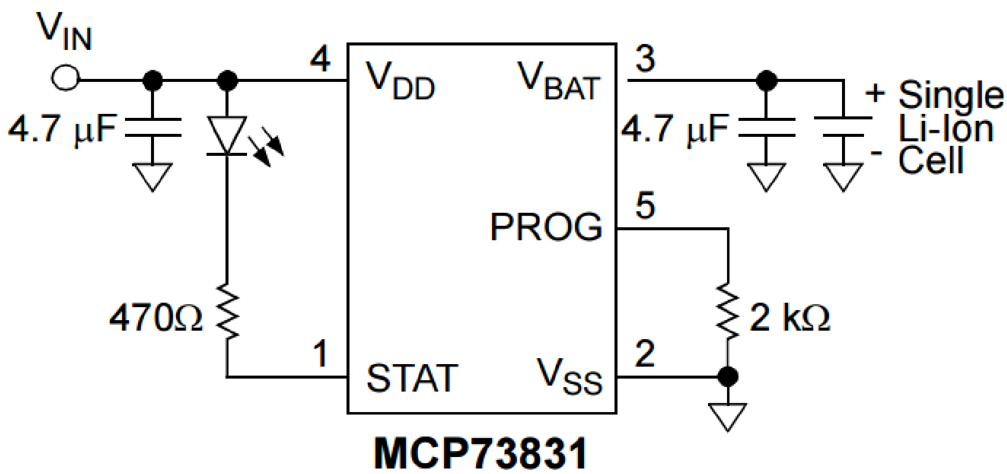


Figure 12 MCP73831 functional diagram (Courtesy of Microchip)

To regulate the voltages from the battery to the components in the watch, there needed to be both a LDO regulator and a switching regulator. The switching regulator was needed due to how the Li-Po battery operates. The Li-Po battery fully charged is at 4.2V. Thus needing a regulator that can maintain a voltage necessary for the GPS/GSM at any voltage. The overall design has one of both the LDO regulator and switching regulator that runs in parallel so the battery won't be overworked and be damaged in the process. Each regulator works around the typical voltages suggested in the components' datasheets.

The linear regulator that was chosen for the watch design was the TPS74701. However, the group changed the linear regulator to the TPS799. The reasons for this change was the need for fewer components on the final design and the need for higher accuracy when regulating the battery voltage down to 3.3V. This change allowed the watch to require fewer components on the final PCB and reduced the total cost of production. Table 56 is a list of operating voltages.

Component	Typical Voltage Supply
GPS	3.7V to 4.4V
Bluetooth	1.7V to 3.6V
MCU	1.8V to 5.5V
Display	3.3V to 5V

Table 56 Watch component voltage distribution

Below, Figure 13, is an example of a typical application circuit of the TPS799 linear regulator that illustrates all the pin connections for the regulator.

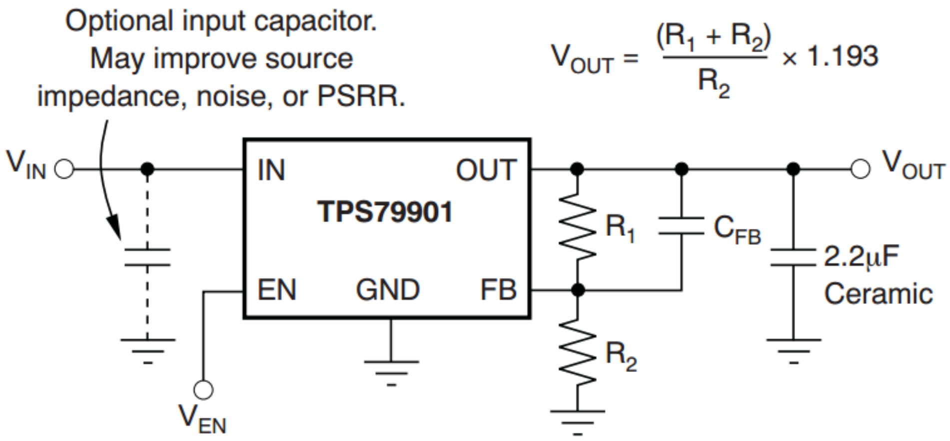


Figure 13 TPS799 pin connections (Courtesy of Texas Instruments)

Based on the typical operating voltages for all of the devices inside of the watch, the overall voltage that is supplied to those components is 3.3V. The components mentioned in the above table have their supply voltage regulated by one LDO regulator to reduce the amount of components needed for the overall design and to simplify the power system for the watch.

The GSM/GPS has a higher typical voltage supply range, typically 4V according to the datasheet. Since the battery can only supply 3.7V, the voltage from the battery needed to be boosted in order for the GSM/GPS to operate properly. The boost converter regulator that was previously chosen for the GSM/GPS is the Texas Instruments TPS61093-Q1. However, after further research on Li-Po batteries, the group made the change to the TPS63050. This is due to how Li-Po batteries topology works. Therefore, the watch needed a regulator that can reduce the voltage and boost the voltage at 4.0V. The switching regulator also has an adjustable setup that allows for the output voltage to setup at any output voltage range.

Figure 14 is an example of a typical application circuit of the TPS63050 switching regulator that illustrates all the pin connections for the switching regulator.

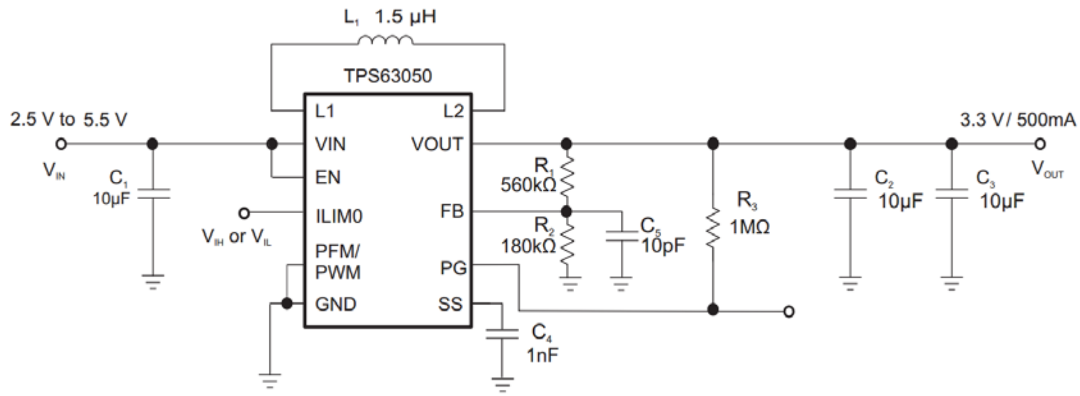


Figure 14 TPS63050 pin connections (Courtesy of Texas Instruments)

Originally, the group decided on implementing the BQ27010 into the watch design. However, due to the amount of components that were needed to support the chip, the group decided to switch to the MAX17043. This change in design allowed the final board design to use fewer components and also reduced the final cost of the product. Below, Figure 15, is a typical circuit application for the MAX17043 fuel gauge.

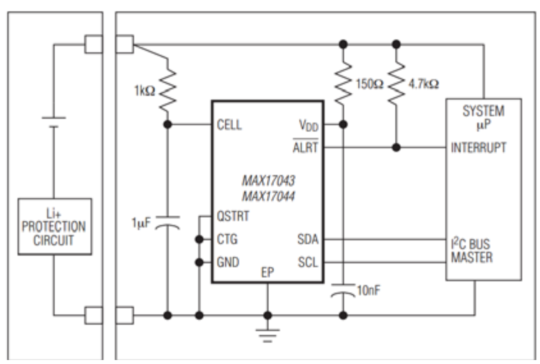


Figure 15 MAX17043 fuel gauge application (Courtesy of Maxim Integrated)

Figure 16, is a final block diagram that lists all of the components being used in the watch hardware design along with the communication peripherals assigned to the components. A more detailed design of the hardware for the watch will be explained further into the document.

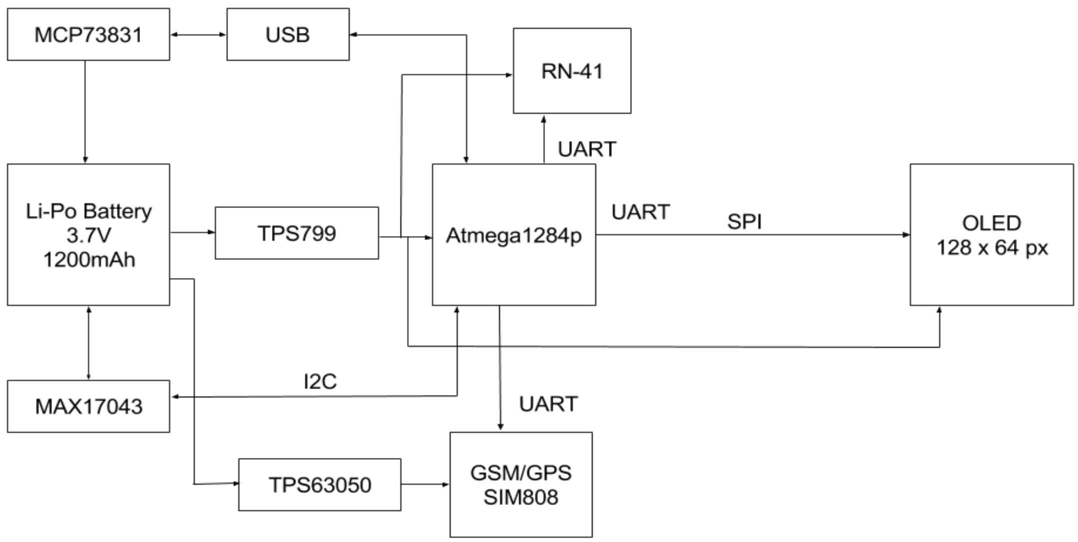


Figure 16 Final Watch Block Diagram

The final design of the watch had components that were selected based on the specifications set by the group for size, weight, and final production cost of the final product.

6.1.5.1 GPS/GSM Hardware Design

The SIM808 module is crucial to the watch hardware design as it is the main component for outdoor tracking, as well as a key communication component between the watch and the application. The pin assignment for the SIM808 can be seen below in Figure 17.

The main voltage supply input is pin 4 VBAT and can take a supply range from 3.4 to 4.4 Volts. Pin 8 PWRKEY, is used to power the device on and off and should be pulled low for at least one second and then released. The transmit output pin, pin 14 TXD, is used to output the NMEA 0183 standard message or the text messages. Pin 15 RXD is a received input of asynchronous UART port and is used to input binary commands to the GPS receiver and text messages.

The pins shown in the following figure will need to be taken into carefully consideration during the development of the Wander Watch project. This is because this could have a very noticeable impact on how the different components will be able to interact and communicate with one another. This in turn can have a negative impact on how well it performs its functions.

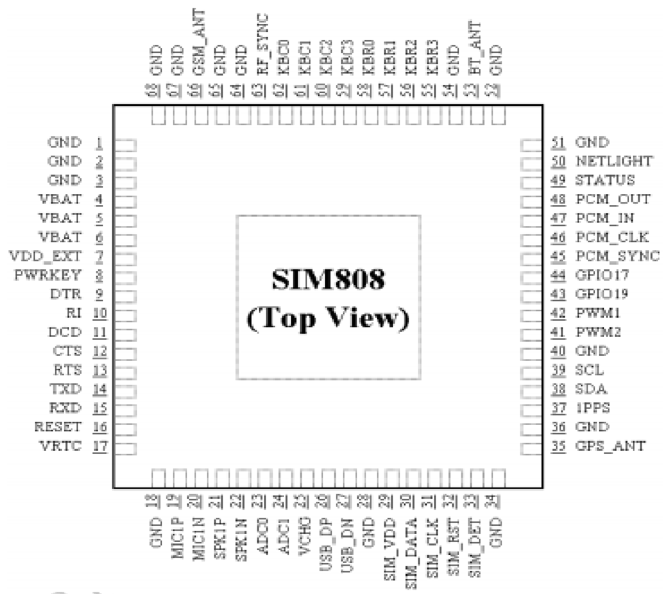


Figure 17 Sim808 pin layout

Pin Number	Signal Name	Type	Description
1	GND	Ground	System ground
4	VBAT	Power Input	Main voltage supply input, 3.6V – 4.4V
8	PWRKEY	Input	Internally pulled to VBAT. Power on/off
14	TXD	Output	Transmit data
15	RXD	Input	Receives data

Table 57 Sim808 pin assignments

6.2 Hub Hardware Design

The Hub is the communication component that connects to the home Wi-Fi network. A microcontroller is the base of the Hub. It also has a Bluetooth component that is discussed in the Indoor Tracking section. And the other main component of the Hub is the Wi-Fi Module that is discussed in more depth in the following section. The reason for the Hub is so that the wearer can be tracked while in the home. A boundary was set because the connection can only go so far. Once the wearer leaves said boundary, then the Hub knows that the watch is disconnected. The main goal is to be a centralized point in the home for the watch to connect with and a secondary goal of the Hub is to be able to alert the caretaker once the wearer has left the vicinity of the home. The watch also has messaging capability, but there is an initial message alerts the caretaker as soon as the watch disconnects.

Some of the following components have a hibernation module option. In the future, this will be a good power saving option for the Hub. The Hub will not need to constantly check in with the watch every second, like it does now. There can be an option to check in maybe every 10 minutes, maybe more. That way in the down time the Hub can save power by hibernating. Of course if it only checks in periodically then there will also need to be a portion of code written in that will allow for an override of the timed check in, in case the wearer wanders out of range before the next check. This override will wake the Hub and allow for the messaging to begin.

6.2.1 Hardware

The main component of the Hub is a microcontroller. The Hub was to be built using a microcontroller, Wi-Fi module component, and a Bluetooth component. The following sections show the block diagram of the Hub and component comparisons made while making the final decision of which part to use.

The final product of the Hub does not contain a Wi-Fi module as previously chosen. The group decided that it was too difficult to implement and ultimately not necessary for the final product. However the Hub still contains a Bluetooth module to allow for the indoor tracking.

6.2.1.1 Block Diagram

The following block diagram will show how the Hub and its major components will be put together. As discussed previously, its components are the Wi-Fi module, the MCU device, the Bluetooth component, and the power supply from a home outlet.

This diagram also shows the major communications that the Hub deals with. One is communications with a home router in order to handle messages from the Hub to the Android application. The other is communications with the Wander Watch itself. This is so that the Hub can track the watch's location, and detect when the watch is out of range.

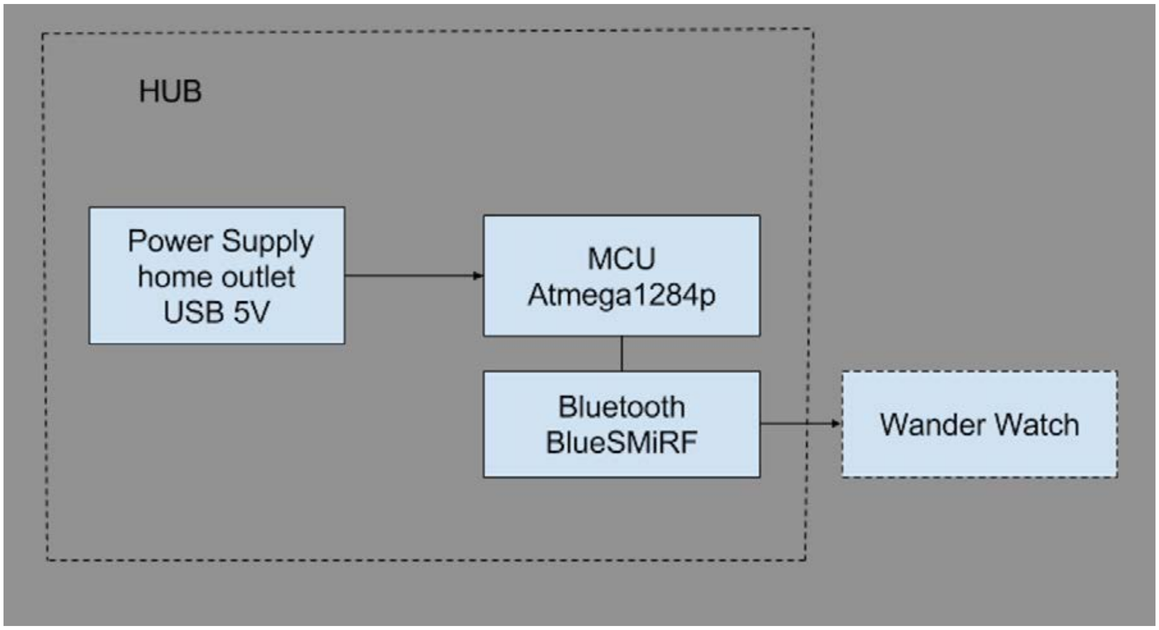


Figure 19 Hub Block Diagram

6.2.1.2 Processor Comparison

The following table, Table 54, shows some of the parameters of the TI Tiva C Series MCU TM4C123x with ARM Cortex-M4F. A few perks of this MCU are that there is a module that allows for battery-backed hibernation. This will allow for a power saving option. Another great feature of this device is that it has plenty of general-purpose input/output (GPIO) lines. This is needed because the device that has been chosen for the Bluetooth component needs to use 4 GPIO lines.

There are a few other options for communication between chipsets. The MCU has UART capabilities with 8 lines and UART is the chip with programming that controls a computer's interface to its attached devices [65]. Inter-Integrated Circuit (I^2C), which is a bus that allows for easy communication between components that reside on the same circuit board, is another form of communication it can use. (I2C) The last option for communication is Synchronous Serial Interface (SSI), this is a widely used serial interface between an absolute position sensor and a controller. This interface uses a clock pulse train from a controller to initiate an output from the sensor [66]. The power options for the board are and ICDI USB cable, which is the default option or a USB OTG, cable. There is a moveable jumper shunt located on the POWER SELECT headers that is used to select one of the two sources. The typical range of power needed is 5 V and has a max of 5.25 V. The breakout voltage is 3.3 V with a max of 260 mA [67].

Product	TI TM4C123GE6PM
Series	Tiva C
CPU	ARM Cortex-MF4
Dimensions	6.0" x 2.25" x 0.65"

Pin & Package	64LQFP
Flash (KB)	128
DMA Channels	32
EEPROM (kB)	2
Capture Pins	24
Battery-Backed Hibernation Module	Yes
IEEE 1588	No
Boot Loader in ROM	Yes
Digital Comparators	16
SRAM (kB)	32
Max Speed (MHz)	80
GPIOs	43
UART	8
LCD Controller	No
External Peripheral Interface	No
Power Requirements	Min: 4.75V Typical: 5V Max: 5.25 V
Price	$11.41

Table 58 TM4C123 Specifications

The following table, Table 59, shows some of the parameters of the TI Tiva C Series MCU TM4C129x with ARM Cortex-M4F. This MCU also has the battery-backed hibernation module option. Again, a good idea for power saving. This MCU has even more options for communication protocols with UART, even more GPIOs than the MCU listed above, 4 SSI, and 10 I^2C which are all the same as the one above. But the C1294 has a couple other options with the Ethernet PHY with IEEE 1588 hardware support and a USB 2.0 interface option that supports Link Power Management (LPM).

Ethernet, which is discussed later on in the standards section, could be useful way to connect the Hub to the home network. The IEEE 1588 is also discussed later in the paper in the Standards section. The C1294 has 3 power options, unlike the C123G which only has 2 options. The first option is the default ICDI USB cable, the second option is the target USB cable, and the third option is the BoosterPack or Breadboard adapter connection option. It is still similar to the C123G because the typical voltage that it takes it 5 V with a max of 5.25 V. This MCU also has a 3.3 V breakout voltage with a maximum total output power limits of 1 amp [68].

Product	TI TM4C1294NCPDT
Series	Tiva C
CPU	ARM Cortex-M4F
Dimensions	4.9" x 2.2" x 0.425"
Pin & Package	128TQFP
Flash (KB)	1024
DMA Channels	32

EEPROM (kB)	6
Capture Pins	24
Battery-Backed Hibernation Module	1
IEEE 1588	Yes
Boot Loader in ROM	Yes
Digital Comparators	16
SRAM (kB)	256
Max Speed (MHz)	120
GPIOs	90
UART	8
LCD Controller	No
External Peripheral Interface	Yes
Power Requirements	Min: 4.75V Typical: 5V Max: 5.25 V
Price	$16.95

Table 59 TM4C1294 Specifications

A little background on the IEEE 1588 protocol is as follows. This protocol was released in 2002 because of the need for devices with clocks that are networked within the same system to be able to be synchronized. Even if clocks are set at the same rate there is no guarantee that they will stay synched. That is why there is still need for synchronization. IEEE 1588 uses precision time protocol or PTP to provide a fault tolerant synchronization for different clocks along the same network. The way that it works is to adjust all the clocks of a network to the highest quality clock. It will define value ranges for the standard set of clock characteristics. The protocol has a Best Master Clock (BMC) that is determined to be the highest quality clock within the network, and this clock is used to synchronize all the other clocks (slave clocks) of the network. If something changes like the BMC is no longer the highest quality or if it is removed from the network then the algorithm will redefine the BMC and make adjustments to the slave clocks.

To make sure that all the clocks of the network are synchronized the clocks will use what is called Bidirectional Multicast Communication. How this works is to send a sync packet that has a timestamp from the master clock. The grandmaster clock and the slave clock trade these time-stamped packets back and forth and they timestamp the packets again upon receiving them. The difference in the departure and arrival times can be calculated. By using the offset measured the clocks are able to readjust so that there is only propagation delay, which also means that this protocol operates with the assumption that the propagation delay is symmetrical. Because of this assumption the slave clock is able to determine and readjust for the delay.

Table 56 shows some of the parameters of the TI CC3200MOD SimpleLink Wi-Fi MCU. This chip combines an ARM Cortex-M4 processor with a built-in Wi-Fi connectivity module. Since the Bluetooth module that was chosen only needs 4 GPIO pins, even this small MCU has enough. This results in 25 total GPIO pins available. This can be very helpful in allowing a decent among of working room for putting the Hub together.

While it is not necessary to choose a device this small for the application of the Hub, it could still be useful for the functionality of the Hub itself. Unlike the watch, where size is a huge constraint, this device would be a useful choice for the MCU because it will not require a separate Wi-Fi module to be chosen and programmed to work with an MCU.

This chip also comes with integrated crystal. This allows the chip to have better capabilities than other chips, and allows it to perform at its peak for many kinds of processed. This would make it an excellent choice for the Wander Watch Hub. Similar to the options above, it has I^2C and UART capabilities. Though due to size there are only two UART and only one I^2C on this chip. The following table lists some of the features of this chip. The following table, Table 60, is a summary of the CC3200MOD device.

Product	TI CC3200MOD
Series	SimpleLink
CPU	ARM Cortex-M4
Dimensions	1.27 mm x 20.5 mm x 17.5 mm
Pin & Package	63 Pin
RAM	256 KB
DMA Channels	32 μDMA
Hibernation option	Yes
Wi-Fi Module included	Yes
Security	WPA2 Personal and Enterprise
Watchdog Timer Module	1
Inter-Integrated Circuit	1
GPIOs	25
UART	2
Integrated Bluetooth Controller	No
External Peripheral Interface	2
Power Requirements	Vbat = 2.3 to 3.6 V
Price	$24.03

Table 60 CC3200MOD Specifications

According to the decision matrix in Table 61 below, the TM4C129 is the best choice for the project. The amount of peripheral connections able to be used was a big decision for the MCU for the Hub because there is going to need to be peripherals for the Bluetooth module as well as the Wi-Fi module. The Tm4C1249 has plenty of compatibility in regards to that.

Cost was another important factor for this component. The TM4C1249 was about mid-range for the MCU comparisons in terms of price. However, this module has its own Wi-Fi component. This makes it more expensive than other modules that have been researched for the Wander Watch project. However, in the end this Wi-Fi component is not necessary. This is because comparisons have been made for the Wi-Fi modules, and a better component with the necessary capabilities has been chosen. As a result, the expensiveness of this component could not be fully justified. The following table shows the pros and cons of each of these components when weighed against one another.

	Cost	**Peripherals**	**Size**	**Power**	**Totals**
Weights	3	4	1	2	
		MCU			
TM4C123G	5	3	4	3	37
TM4C1249	4	4	4	3	38
CC3200MOD	3	4	4	4	37

Table 61 Hub MCU Decision Matrix

The power to the device chosen for the Hub is going to need a voltage converter to use the wall outlet. For a typical home the output is 120 V, but the MCU can only take up to 5.25 V. For this reason the group will need an adapter to ensure that the MCU is not going to get damaged.

There are plenty to choose from since the typical output for a device that connects via USB is 5 V. Any phone charger outlet converter can be used. The ones for iPhones takes in 100 V-240 V and outputs 5 V. This makes it a good option for the group to use since the group already has one to use.

6.2.1.3 Bluetooth Hardware Design

The BlueSMiRF Gold has a power supply of 3.3 Volts located on pin 11. It can be interfaced using UART pins located on pins 13 and 14. The pin locations and further explanations can be seen in Figure 21 and Table 62, respectively. The figure shows the layout of the pins. Meanwhile, the table helps to explain each of the pins in terms of I/O, power supply, and a short description of the function of each of the pins.

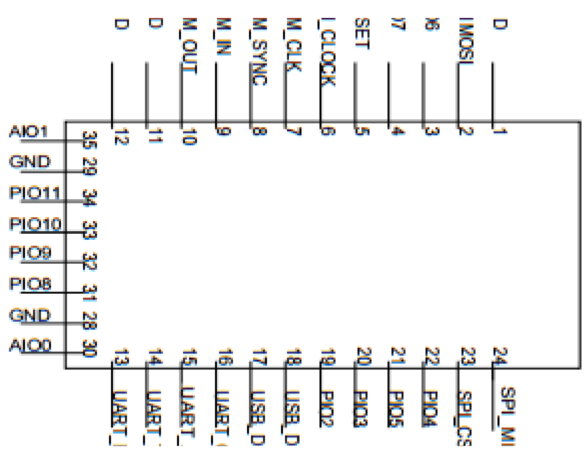


Figure 21 BlueSMiRF Gold pin layout

Pin Number	**Pin Name**	**I/O**	**Supply Domain**	**Description**

1	GND	GND	-	Ground
11	VDD	Power Supply	3.3V	Regulated power input
13	UART_RX	Bidirectional	VDD	UART receive input
14	UART_TX	Bidirectional	VDD	UART transmit output

Table 62 BlueSMiRF Gold pin assignment

6.2.1.4 Wi-Fi Module

The Wi-Fi Module is a needed part because the Hub, which is going to be the main component in the home that keeps track of the wearer, needs to be able to connect with the watch and track whether the wearer is within range or not. The Hub is connected to the watch via Bluetooth because while it is in the home there is no need to it to have to use the phone carrier which will cost more because if it used the carrier, then data would be being used which will cost much more than connecting via the home Wi-Fi network and Bluetooth. For this to be feasible the Hub must have the capability to connect to Wi-Fi and this is where the Wi-Fi module comes in.

6.2.1.4.1 Background

All wireless devices have a radio frequency transceiver. This allows the device to send and receive messages. The physical components of the wireless device are integrated onto a circuit board. They typically can transmit on 2.4 GHz or 5 GHz frequency band using 20 MHz or 40 MHz wide channels depending on the standard that is being used.

The software portion of the Wi-Fi module contained the network driver that is installed or provided with the device. It can change the mode of operation that the transceiver module would normally use.

Both of these are integrated to make the complete device and it uses a radio frequency spectrum to do the wireless communicating. The signals that have larger wavelengths are more easily able to pass through solid objects and travel further distances, but the data will transfer more slowly. Whereas of course the opposite is true and the signals with smaller wavelengths are useful when there are fewer obstacles and a closer range. With these parameters the data will travel very quickly between devices. If there are obstacles in the way, the data is likely to have to be resent more often because it will break up into packets upon passing through an object and will leave room for more error once they try to reconnect on the other side. [70]

Transmission Control Protocol or TCP and User Datagram Protocol or UDP will be mentioned in the comparison section so here is a bit of background on it. TCP is connection oriented, so once a connection is made the data can be sent both directions.

UDP is a simpler connectionless Internet protocol. For UDP multiple messages are sent as packets in chunks. The major differences are that TCP ensures a reliable and ordered delivery of the information while UDP just sends information and does not check the readiness of the receiver. TCP is also much more reliable because it acknowledges and will retransmit data, therefore there is never any missing data. UDP on the other hand does not check that the communication was received and does not retransmit data. TCP is also sequenced and if there is ever any data that is received in the wrong order the TCP protocol with reorder and deliver it again. With UDP the sequence may not be maintained so there is no way of predicting the order the message will be received. A couple other differences are that TCP requires three packets for socket connection and is able to handle controlling congestion as well as reliability. But UDP is quite lightweight in comparison and there is no capability to track or order messages. Also TCP will read data as a byte stream and the messages are transmitted to segment boundaries. While with UDP the messages are sent in packages, which are of course sent individually and aren't checked for integrity until they arrive at the other end, and packets have defined boundary, where as a data stream has no boundary. [71]

6.2.1.4.2 Purpose

For the project the group is looking to make the watch Wi-Fi compatible so that it can connect to the home network while the user is in the house. Once they walk out of range, the Wi-Fi network will notify the caretaker that they are no longer connected on the home network. The router will be used to send the message and this way will eliminate the need for a GSM device in the home because the message can be sent over the Wi-Fi network in the home. Not needing to use a phone carrier subscription for the entirety is ideal to keep costs of the project low.

Some trouble with using a Wi-Fi module as a part of the watch is the size. While the length and width are seemingly not going to be a problem there are some issues with the height of it in the products researched so far. There will be a comparison chart on the following page of a few of the different products available for use.

As stated above, the original plan of the Wi-Fi module was to use the watch as a Wi-Fi compatible device so that it is connected to the Hub via Wi-Fi. However upon closer inspection it was decided that the watch will be connected via Bluetooth. The Hub will still need to be Wi-Fi compatible, but the watch will no longer need a Wi-Fi module of its own. Therefore the main purpose of the Wi-Fi module is now going to be making the Hub, which is going to be a microcontroller of some kind, Wi-Fi capable. This will allow the Hub to connect to the home Wi-Fi network, which will then be able to communicate to the watch. The Hub will still be able to send a message to the caretaker via Wi-Fi once the Bluetooth module notifies the Hub that the wearer is out of range.

6.2.1.4.3 Messaging System

In researching a messaging system for the Hub to send messages over Wi-Fi, not much was found. Much of the information suggests using an already built application to send messages over Wi-Fi such as Google Voice, Tango, WhatsApp, or TextFree. Users pointed out that the problem with these applications are that a separate account is needed or a separate phone number as with Google Voice in particular (Android Central).

Users also kept mentioning the Apple way of messaging, which is called iMessage. These messages are sent over Wi-Fi and just happen to be handled in the same application that text messages on iPhones are. The way that iMessage works is to send the message over Wi-Fi or over cellular data, in which fees may apply. If the iOS device does not have a cellular data plan, then it is unable to send messages when not connected to Wi-Fi [72].

Different messaging systems were looked at to decide how to send a message. The watch needs to be able to send a message to the caretaker when it is out of range of the house. The thought was that the Hub could be used to send the message over Wi-Fi, which would cut down on the cost of sending a message over a carrier's network. The following paragraph shows messaging systems and applications researched.

Messaging systems comparable to iMessage include BBM, WhatsApp, Facebook Messenger, Google Hangouts, or Viber. BBM or Blackberry Messenger is an option for a messaging application that can be used. Unfortunately this application still needs an active data plan to be used. It will work over a Wi-Fi connection but only if there is also an active data plan attached to it [73]. WhatsApp is another option for a messaging system but has the same issues as BBM, it requires a data capable device. There are ways to get around it and install it onto a Wi-Fi only device but it will only mirror what is already on your data capable device [74].

Other messaging systems are as follows, Facebook Messenger application is another researched application. It can send messages over Wi-Fi, and will use the mobile data when there is no Wi-Fi available. However when sending text messages over the application, it will cause regular texting rates and data charges to apply. The application does warn the user in such events. [75] Google Hangouts is a messaging system that also works over Wi-Fi. The main downfall of Google Hangouts and Google Voice is that there is a need for a separate phone number. This can be confusing and is unnecessary and unwanted for an application like the Wander Watch. [76] Viber is yet another messaging application that can be compared to iMessage. It touts free text, calling, photo messages and location sharing. The calls and messaging system of Viber do however use the data plan of the device [77].

While all of these messaging systems are able to work over a Wi-Fi connection, it does not eliminate the need for the device to be able to be connected to a mobile network. The point of trying to use the Hub for the messaging system for the Wander Watch was to allow for the users to cut costs of messaging over a mobile network and use an already existing Wi-Fi home network to alert the caretaker of the wearer leaving the premises. However upon researching it does not seem that this is easily done and that is the reason that a different method of messaging was found.

6.2.1.4.4 Wi-Fi Comparison

The following is a comparison of a few different Wi-Fi module options. The TI CC3000 is a self-contained wireless network processor that will help to simplify the implementation of Internet connectivity. This device helps to minimize the software requirements of the host MCU and that's why it is ideal for a low-cost, low-power device. The processor uses IEEE 802.11 b/g and has an embedded IPv4 TCP/IP stack. It contains an integrated crystal and power management and the dimensions of the product allow for space saving, coming in at 16.6 mm x 13.5 mm x 2 mm. While this product is still in production and available for use TI does not recommend it because there is a newer and better product available from them that is discussed below. All of these specifications and more is shown in the following, Table 63 [78].

Product	**TI CC3000**
Network	IEEE 802.11 b/g
Stack	IPv4 TCP/IP
Antenna	FCC, IC, and CE certified
Integrated Crystal	Yes
Dimensions	16.3 mm x 13.5 mm x 2 mm
Price	$38.00

Table 63 CC3000 Specification

The TI CC3100 SimpleLink is a device that is a part of a new family of products that helps to simplify the implementation of Internet connectivity. It integrates all Wi-Fi protocols, which helps to minimize the MCU requirements. The system includes 802.11 b/g/n radio, baseband, and MAC to allow for fast and secure internet connections and also as 256-bit encryption capabilities. The power options for the chip include integrated DC-Dc converters that can support a wide range of supply voltages.

There is also a subsystem on board that enables it to have low-power consumption modes and a hibernation mode that only requires 4 µA of current. This device can connect to any 8, 16, or 32-bit MCU over SPI or UART interface. This ability is important because the MCU will likely be an 8-bit UART capable device.

There are many options available for this module that makes it the better choice by far as compared with the CC3000. The 802.11 n capabilities which will allow for faster network connections is reason enough. It also has a hibernation mode that will go well with the hibernation mode of the MCU chosen. The security options of WPA2 Personal and Enterprise are another perk of the CC3100 [79].

Product	**TI CC3100**
Network	IEEE 802.11 b/g/n
Stack	TCP/IP or TLS/SSL
Security	WPA2 Personal and Enterprise
Hibernation	Yes
Dimensions	0.5 mmx 9 mm x 9 mm
Supply Voltage	2.1 to 3.6 V
Pre-regulated mode	1.85 V
Host Interface	UART 8, 16, or 32
Price	$22.00

Table 64 CC3100 Specifications

The TI CC3200MOD is shown in the Table 65 below. The integration of both into one chip is a great option because it allows for one less product to be purchased and programmed. The Wi-Fi capabilities of the CC3200MOD are also quite good. It uses 802.11 b/g/n and a TCP/IP stack that has 8 simultaneous TCP, UDP, or RAW sockets and contains 2 Simultaneous TLS v1.2 or SSL 3.0 Sockets. As a result, this device can interface through UART, I^2C, SSI, and ADC. The UART in particular will most likely be used for the project at hand. Also there are 25 GPIO pins, which is important for the Bluetooth module chosen [80]. The following table shows some further information for this component.

Product	CC3200MOD
Network	IEEE 802.11 b/g/n
Stack	TCP/IP, TLS/SSL
Security	WPA2 Personal and Enterprise
Hibernation	Yes
Dimensions	1.27 x 20.5 x 17.5 mm
Supply Voltage	Vbat = 2.3 to 3.6 V
Host Interface	UART, I2C, SPI, ADC
Price	$24.03

Table 65 CC32000MOD Specifications

TCP/IP is an architectural model that is based off of the OSI model. It has 4 layers that approximately match the 7 layers of the OSI model. The layers are hardware, network interface, Internet, transport, and application [81].

The TX Power of the device stands at 17 dBm at 1 DSSS, 17.25 dBm at 11 CCK, and 13.5 dBm at 54 OFDM. The TX Power is the broadcasting power of your transmitting antenna. 70 is a typical value of the TX power for a router, but of course a device this size will need to be at a much smaller range [82]. The RX Sensitivity of the chip comes in at -94.7 dBm at 1 DSSS, -87 dBm at 11 CCK, and -73 dBm at 54 OFDM. The RX sensitivity deals with the range of the product and the values stated previously equate to a range of about 100-1000 meters [83].

The application throughput comes in at 16 Mbps for UDP and 13 Mbps for TCP. In Table 66 below, the decision matrix is shown for choosing the Wi-Fi Module. The scale is 1-5 with 5 being the best option.

	Cost	**Network**	**Power**	**Size**	**Totals**
Weight	4	3	1	2	
Wi-Fi Modules					
CC3000	1	2	1	3	17
CC3100	5	5	4	4	47
CC3200MOD	3	5	4	4	39

Table 66 Wi-Fi Module Decision Matrix

As stated above the group decided to eliminate the Wi-Fi module from the Hub design and chose to go with Bluetooth only. This had to do with ease of implementation, cost, and necessity.

6.2.1.5 Hub Final Block Diagram

The following Figure 22 shows the final block diagram for the Hub design. This diagram shows how the modules will be connected to the microcontroller and the necessary parts to get the microcontroller up and running. The Wi-Fi module will be connected by a UART serial connection and the Bluetooth module will be connected by SSI. The power supply is also shown and the parts needed to step down and convert from 120 Volts to something usable by the microcontroller. This is to ensure the voltage compatibility between components.

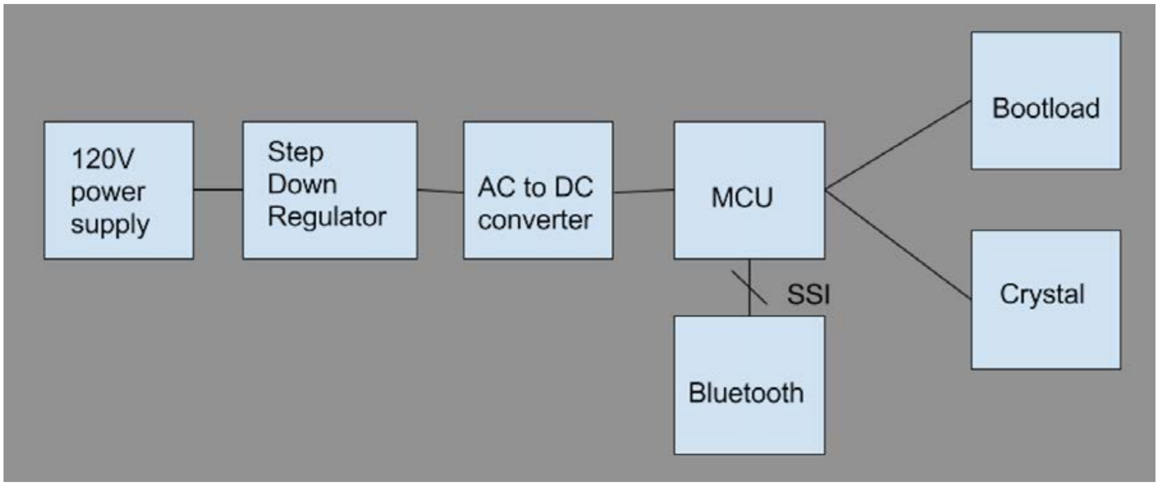


Figure 22 Final Block Diagram

The final hub design does not include a Wi-Fi module because it was decided that it was unnecessary to include. The Bluetooth still stands as an added backup for tracking purposes.

6.3 Application Design

The following sections will be going into greater detail on the construction of the Android application itself. Here, the discussion will be focused on how it will be built, how it will be able to communicate with the Hub, how the navigation of the application with work, and explain the parts that will be used to build up the application. No code will actually be shown here, but these sections will be going into details on the functions and implementations that will be utilized to give the application certain features.

6.3.1 Software Goals for the Application

The software goals for the application are meant to be used as a sort of guideline to making sure that the application is capable of certain functions. Besides its major functions, which were discussed in a previous section, this section will go over other goals that the Wander Watch application is able to meet.

For one, the application is able to be built, tested, and made fully functional within the time given for this Senior Design project. The Wander Watch team has done research into the feasibility of the application, and has determined that it is indeed doable for this project. In fact, it has been found that past groups in this course have built similar projects. Thus, if other people were able to complete projects similar to the Wander Watch, then the project team should be able to finish the application along with the rest of the project on time. A more in depth discussion of the project's timelines and milestones are discussed in the respective sections near the end of this report.

A second goal is that the application should have no problems communicating with the other hardware that will be used for the watch and Hub. Through the research for this project, it has been determined that the Android application should be compatible with the watch and Hub in terms of sending messages and alerts between each other. The Hub is able to send alerts via SMS messages, and the application can be designed to receive these alerts as SMS messages as well. Using SMS messages on a cellular network instead of other kinds of communication has been determined to be the best option in order to keep the cost of this project as low as possible.

Another goal is that it should be buildable using mainly Android Studio. Given that Android and the Java programming language has a lot of implementations and features that will be useful for this project, Android Studio already has practically everything that is needed to build the functions of the application. At this point in time, the project team's research has not shown that something that does not already exist as an Android implementation or Android tutorial would be needed for this project. Something may happen down the line that would require use of other tools besides Android Studio.

However, the team has not found anything that would require relying on another program, nor has the research up to this point suggest that the project will need something outside of the team's chosen programs and parts. So for the time being, the Wander Watch application seems to be perfectly doable with what the project team has decided to use so far.

A goal that has been discussed previously is that the interface of the application should be user friendly. It should be easy to understand and use. For now, mock-ups of the screens of the application have been created, as shown in a previous section. The screens will be modified slightly once the application is actually built, but the project team will be making sure that the final application is still visually appealing and easy to understand. If it is found that some parts of the application are confusing to use, they should be changed accordingly. The project team will use certain guidelines and benchmarks while testing the application to make sure that it is indeed user-friendly for most users. This is discussed further in the Application Testing section.

One final goal is that the application should not take up too much battery power on an Android phone. According to the team's research, it is believed that battery power will not be too much of an issue for this project. The Wander Watch application is relatively simple in design and function. Thus, it should not take up too much of the phone's power.

Regardless, the project team will still be aware of possible battery problems. The application will be tested for any such problems, and part of the testing phase will be used find ways to optimize the application so that it doesn't use a lot of power. Taking time to further optimize the application's power consumption will only be done if it is found to be necessary. The project team's research on this matter indicates that this should not be too much of an issue for the application.

Now that the second semester of our project has ended, the application has been built and tested. It works as intended, and it meets all the goals and guidelines that have been detailed above. The following sections will have notes at the end of each, which will explain any changes that were made in the final app.

6.3.2 Communication with the Hub

The Hub and application communicate with each other to share information on the watch's location, and on the user's settings for which alerts can be sent. Based on the user's settings, the Hub will send automatic alerts for letting the user know whether or not the watch is still within range, or to let them know when the watch is low on battery power and needs to be recharged.

These alerts will be sent to the application via SMS messages. Android applications are capable of receiving text messages via SMS communications, and there are plenty of texting apps that work by sending messages this way [84]. Thus, it has already been proven that an application can send and receive these SMS messages. So as long as the Hub is connected to the right cellular network, and it is associated with the right account, the user should be able to receive the alerts from the Hub without any major problems. The following diagram shows these communications for the project.

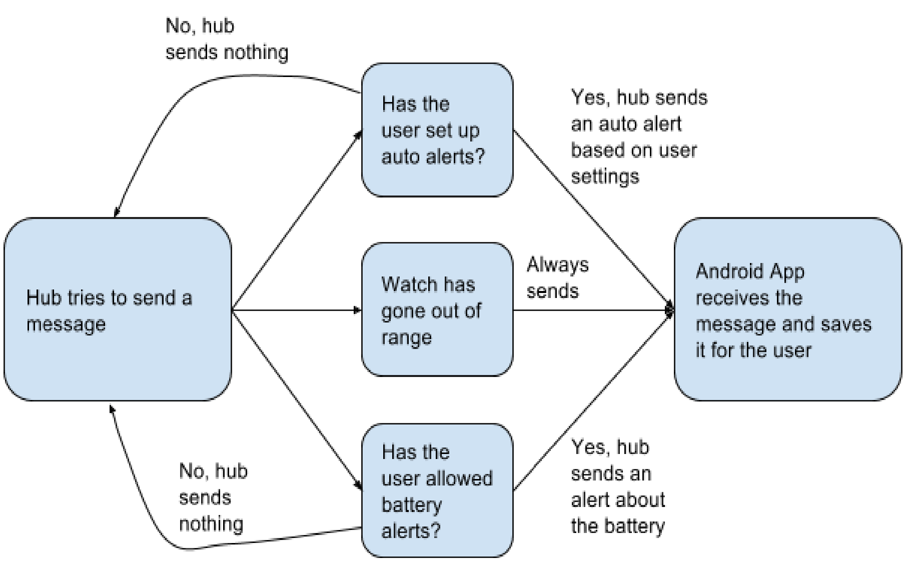


Figure 23 Communication flow diagram

As discussed in the Functions section, the application will only receive messages based on how the user has changed their settings. If the user has not set up automatic alerts, the Hub will be notified that it should not send these kinds of alerts. Likewise, the Hub will not send alerts about low battery power if the user does not allow it in the settings. The only types of alerts that will always be sent are those that indicate when the watch has gone out of range. By default, when the user sets up a new account for the application, the account will only receive alerts for the watch being out of range of the Hub. The user can change this by changes the related options in the settings menu of the application.

The need for a cellular network connection does present a problem, though. The Hub needs to always have a strong connection to a cellular network, and the phone that has the application also needs to be able to access such a network, or otherwise be able to receive SMS messages via a different type of connection. However, there is a large availability of cellular networks today, and these networks are continuing to grow in availability. Any phone should be able to connect to the cellular network needed for the application, and receive messages via this connection without needing additional hardware or software. Indeed, more and more users are relying on cellular networks as they continue to grow in availability in many locations. Even though SMS messages have a cost attached to them, most of them offer a fairly minimal cost for most users. In fact, for this project it has been estimated that each alert should only cost about ten cents. Thus, the project team feels that using these types of networks for communication will not be that much of an issue for the average user.

The Wander Watch team had originally considered simply making it so that the Hub sends a text message to the user's phone, so that the user does not have the need to use a separate app. However, these text messages are sent through the phone's carrier network. This could involve needing permission from various carriers in order to authorize the message. This in turn could have limited how many people would be able to use the Wander Watch, depending on what carrier would let the project team use their networks for the alerts. This is a similar case to one of the previous products found during the team's research, the Comfort Zone Check-In. That product could track a person via their cell phone, but only if it was already on the Sprint network. Besides needing permission from various cellular networks, this could also incur a cost for being able to send the alerts through these networks. This could very well put a large cost on this project's users. The team wants the Wander Watch application to be able to work without being limited to certain networks like this. The particular cellular network that was chosen for the project was determined to be the cheapest per message sent, while also been one of the largest networks available. It was decided that this was the best option to make sure that this project will work in a wide range of locations, while still making it affordable to build and use.

Unfortunately, this whole section became obsolete once we decided to not use a hub after all. The watch now sends SMS messages itself to the phone's app. So the above diagram and other information related to the hub work more or less the same, but using the watch instead of the originally planned hub.

6.3.3 Code Flow

Code flow refers to how a user moves through the program while said program is running. For this project, the code flow describes the possible ways the user can navigate through the app. In the Menus section, it was discussed how the user can move from one main menu to another. The code flow is essentially a more in depth version of the menu navigation diagram shown in the Menus section of this report. Code flow analyzes how different decisions can be made based on the information the user have given to the application. These decisions can determine what kind of information the user will be able to see, and where they can go within a menu. Figure 24 shows the possible ways that the user can move from one part of the application to another, including within a menu.

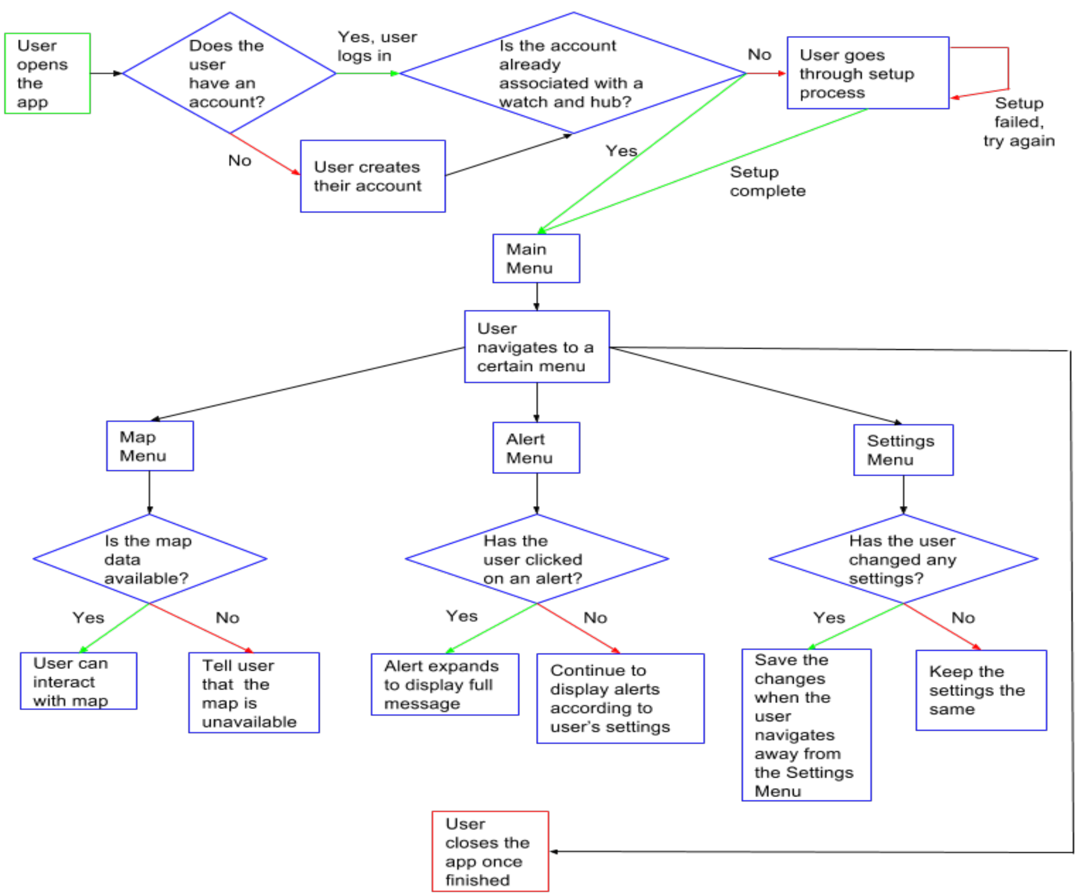


Figure 24 Control flow diagram for the Wander Watch application

The control flow diagram shows other processes that the application will need to handle for navigation. Overall, the application will be able to check five major interactions. First, if the user has an account. Second, if the user has gone through the setup process to associate their account with a specific Hub and watch. Third, if the map data is available to be displayed within the Map Menu. Fourth, if the user has clicked on an alert in the Alert Menu. Fifth, if the user has changed any settings in the Settings Menu.

This diagram also includes situations where the application may fail a certain process, or simple not complete a process due to the user changing the settings for their account. For example, if the set-up process fails for whatever reason, the application will prompt the user to try the process again. Also, the application may be unable to retrieve the map data for the Map Menu. This could be due to a bad connection, either from the phone or the watch for whatever reason. If this happens, the application will tell the user that the data is currently unavailable, and will not display a full map.

The application will also check for changes in the Settings Menu. Instead of having a dedicated button for saving, the application will check for changes when the user navigates away from the menu. If the user has made changes, the application will automatically save the changes before navigating back to the main menu. The user will be able to tell that the changes have been saved by seeing a small "Saving" prompt before the navigation is complete. If the user has not made any changes, the application will continue to the main menu without trying to make any changes to the user's settings.

The user should be able to exit the application from any of the menus, even from the login screen. This control flow diagram shows the navigation and if-else decisions that will be taken into consideration for the application's actual code. This also helps the team figure out the kinds of Java functions and code that will be needed to ensure that the application can navigate properly between the menus, and complete certain actions based on what the user does to their account.

6.3.4 Diagrams and Structures

The following two sections will discuss some of the inner workings of the application. The first section will discuss the application's class diagrams. This shows some of the actual Java functions and code that will be expected to be useful in building the application's functionality. The second section will cover the data structures that will be used for this project, and their relation to the application. This means relating the different pieces of data that the applications will utilize with the parts of the application itself.

Both of these diagrams may be modified as work continues towards the final product of this project. Still, these diagrams can represent, based on the research for this project and progress on the project so far, an estimate of what can be expected to find within the final application for the Wander Watch project.

6.3.4.1 Class Diagrams

There are many different functions that will be programmed in Java to ensure that the Wander Watch application will work correctly. The following class diagram shows the classes and Java functions that are expected to be utilized in the application. This may change in the future, either by adding new classes and functions, removing those that are found to be ultimately not needed, or replacing an old class with an entirely new class. Thus, this diagram is not final, but based on what is needed and what the application needs to do, the project teams has estimated that it is a decent representation of the classes and functions that will be found in the final application.

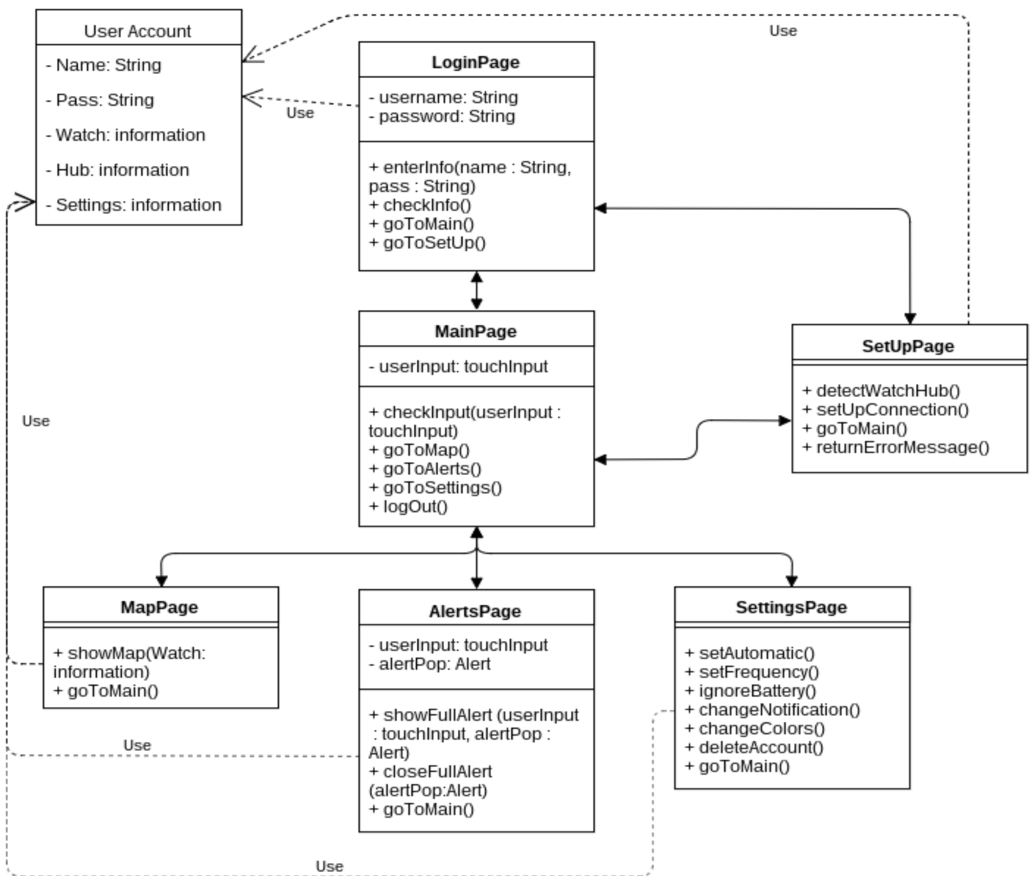


Figure 25 UML class diagram for the Wander Watch application

The user's account acts as an overall container of information. It has info on the user's username, password, the watch associated with their account, the Hub associated with their account, and the settings for their account. The pages for login, setup, map, alerts, and settings all use information based on the data saved in the user's account. Starting at the login screen, the user inputs their username and password. This is processed by the enterInfo function, which in turn will let this information be checked by the checkInfo function. The application checks the inputs using the checkInfo function, and navigates to the main screen if the information matches with what is in the user's account.

If the user does not have an account already, they are redirected to the setup page, to set up their account and the connection with their watch and Hub. On that page, the detectWatchHub and setUpConnection functions are used to find the user's watch and Hub and establish a connection between them and the application. If an error occurs, the application uses returnErrorMessage to let the user know that something has gone wrong, and will prompt them to try again. Otherwise, the user will be sent to the main page via the goToMain function once the setup is completed.

The main page contains functions to help the user navigate to other pages. It collects information on where the user has clicked, hence the userInput variable. From there, the user can navigate between the map, alerts, and settings screens. This is based on which button they have clicked on. Each navigation function corresponds to their respective page. For example, goToMap will go to the map page. The user can also log out of their account from the main page, since the logOut function is contained on the main page.

On the map page, it uses the function showMap to get information about the watch's location, and display an accurate map of where the watch is. The watch information is stored in the user's account information. So this function will get the necessary location data from the user's account. On the alerts page, it collects information about the user's alert from their account and displays these alerts. The userInput function is again used to display the right data based on where the user clicks on. For this screen, the user taps on a particular alert to display its full information, or to close an already opened alert. It has the showFullAlert function to allow the user to show a full alert, and the closeFullAlert function to close it. The alertPop variable holds the full message of the chosen alert. Thus, alertPop will be used to display the full message when the user chooses to open a full alert.

On the settings page, there are functions to change a particular settings based on how the user interacts with its corresponding input. The setAutomatic function will set automatic functions for the user. The setFrequency function is used to set up the frequency of which the automatic functions will be sent. So if the setAutomatic function is changed so that the user will not receive automatic alerts, then the setFrequency function will do nothing.

The changeNotification function is related to the notifications that will appear outside of the app. The user can choose to allow or prevent these application notifications. The changeColors function will change the color scheme of the application itself, and is purely a cosmetic change.

The deleteAccount function will of course delete the user’s account information from the application. If the user chooses to delete their account, then all of their information will be wiped, and they will be sent back to the log in page to make a new account.

All three of these pages have the goToMain function to allow the user to navigate back to the main screen. Since only the main page has the logOut function, the user can only log out from there. However, they will be able to close the application on any screen, just like any other Android application. There may be more functions or classes added to the application, but for the time being this class diagram covers most of the functionality that is needed for this project.

The functions themselves will be built in Android Studio using mostly Java. The code that will be used within the functions has not been specified, since at this point it is not known what code will be needed for the functions to properly work. However, given what is known to be necessary for the application’s success, it should be simple to understand what kind of code, loops, and other functions may be needed to ensure that the application will function properly and efficiently.

In addition, it has been discussed that there are various tutorials and resources available freely online that will help for working with Android application development. These resources will be utilized once the functions and code are being fleshed out in the actual Wander Watch application for this project.

The following figure is a more updated version of the class diagram, based on how the final application was built for this project. The functionality is still the same compared to what has been stated so far. The only difference is in the names of some of the functions and pages of the application, there is no longer a need to store hub information, and there are more variables being taken into account for certain values in the settings. Still, it is easy to tell which page or function corresponds to which features.

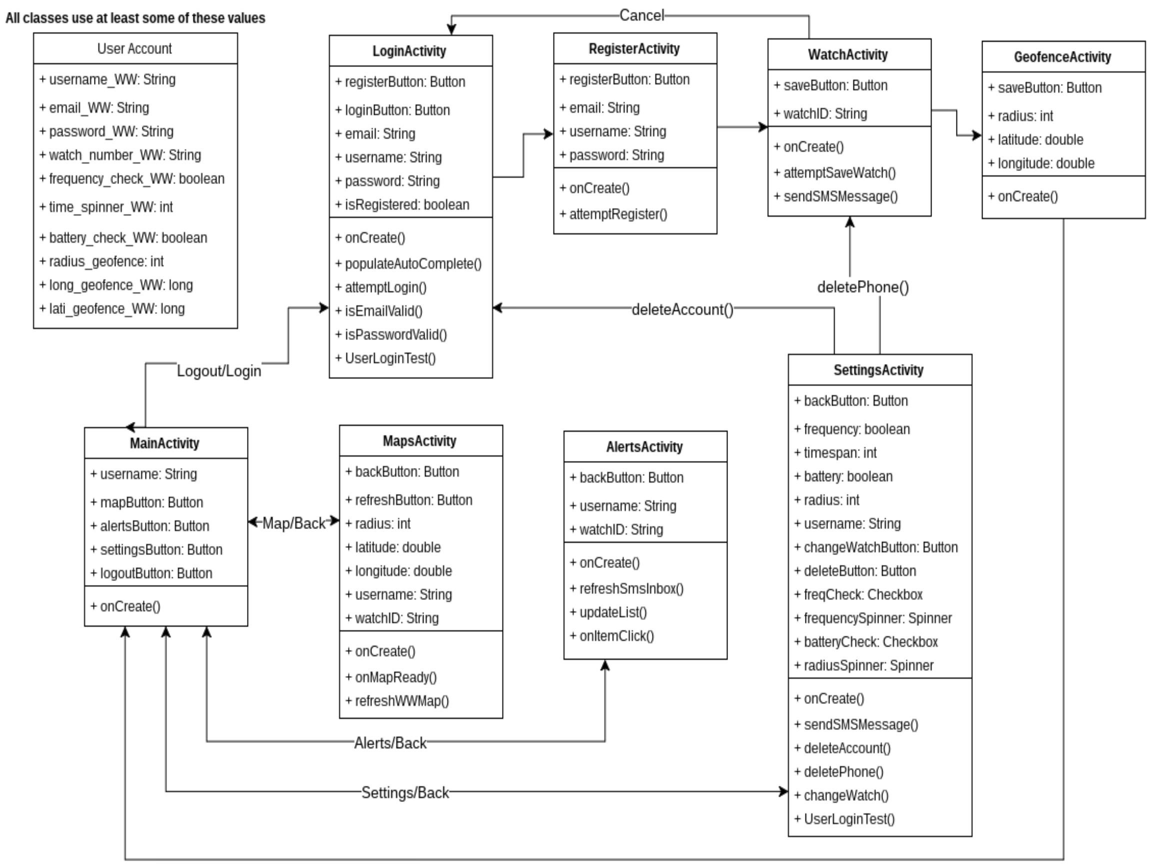


Figure 26 Final UML class diagram for the Wander Watch application

RegisterActivity, WatchActivity, and GeofenceActivity are the three screens related to the setup of the account. The other activity pages correspond with their respective screens from before.

6.3.4.2 Data Structures

There is a lot of data that the Wander Watch application will work with in order to function properly. The most important data would be the GPS location data. Without it, the application would not be able to display accurate tracking information on the map. However, a lot of the data that will be sent to and kept by the application will be important for other vital functions. This not only includes the GPS location of the actual watch, but also information regarding the user's account.

Each account will have its own unique information attached to it. As a result, it is important for the application to have clear boundaries about what data it can use for which processes, and how it should be able to store this data. Otherwise, it could misplace data, use the wrong data for a different function than intended, or display the wrong data in the wrong menu altogether.

The following data structure diagram shows a visual representation of the kinds of the data that the application will have to deal with. It also shows what parts of the application will have to process which types of data from the user's account.

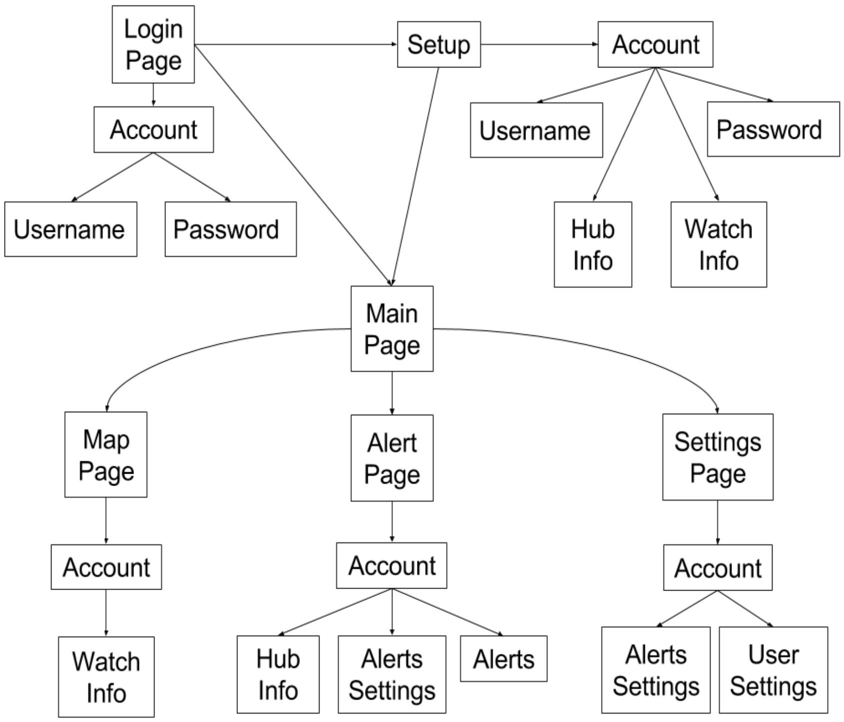


Figure 27 Data structure diagram for the Wander Watch application

As stated in the Class Diagram section, all the information that the application needs is associated with the user's account. So in order to get the right information, each part of the application needs to access the user's account and make sure to retrieve the right information.

Note that the alerts settings and user settings are not altered during the setup process. These settings will be set to their default values on a new account. The user can go to the settings menu to change these settings to their own preferences. Similarly, the user's account will not have any alerts associated with it when it is first created. Once the user starts receiving alerts from the Hub, these messages will be stored alongside their account information.

The login page checks and verifies the user's username and password. The setup page will populate the information for the user's watch and Hub during the setup process. From there, the other pages use parts of the account to perform their functions. For example, the map uses the watch information from the account to show the user an accurate map.

It is a good idea to keep all of the information associated with a single account together like this, because it helps to keep this information organized and easy to find. The information is stored on the user's phone, rather than stored on a separate server or hard drive. This way, the user can change settings without needing a connection to a server. This also allows the application to store information about the watch and Hub in an easy to access place. This in turn will allow it to get the right info it needs to display the right map, and the correct alerts from the Hub.

If the Wander Watch has a larger group of users in the future, it could be necessary to store account information on a server for management and to back up account information. However, for the current scope of this project, keeping the account information on the phone will be enough to allow the application to fulfill its intended functions.

For the final version of the application, there is no hub information being stored. Instead, it is storing the watch's phone number in order to keep track of messages from the watch. Thus, in place of Hub Info in the data structure diagram, it would instead just be Watch Info, if Watch Info is not already there. Meanwhile, Alerts Settings and User Settings are combined into just one set of settings. So any changes made to the geofence radius or the alerts are saved to the user's settings. These settings are saved as SharedPreferences in the application itself. This allows them to be 'shared' across the entire app. This way, for example, users can change the radius of the geofence, and then view the map to see the updated geofence.

6.4 Case Design

The case for the watch blocks out dust and water from entering inside the electronics. The watch casing also has access for the micro USB and for recharging the device. The watch case also compensates the size and thickness of both the battery and the final PCB. The top of the case also has a section that exposes the 1.3" OLED display so that the user can check the time and date. In addition to the display being shown on the top, the watch case also has the internal hardware unexposed for environmental protection.

The case has been 3D printed due to ease of production and the low cost of 3D printing. The case was designed on SolidWorks due to the program's ease of use and how easily the design of the case can be sent to the printer. The 3D printing was done at the university for no charge.

6.5 Schematic Diagrams

As part of the development stage of the Wander Watch project, the schematics and PCB layout of the watch have been developed before the prototyping stages. This is so that the project team has an understanding of how the components will be connected to one another. After the schematics have been developed, the group simulated the systems in a lab to verify the designs for the watch.

Once the schematics have been developed and simulated, the PCB layout for the watch was developed. The schematic illustrated how all of the connections were made between the various components inside of the watch including the communication peripherals made between the components and the MCU. The schematic also established the network of the power system. This detailed the power circuit and how power is distributed throughout the design.

The schematic does not illustrate how the components are distributed onto the PCB. Routing and layout of the PCB has been handled within the PCB section of the development stages. The schematic process is more for implementing the circuits based on the information given with the figures and tables provided in the components' datasheets. The next page shows Figure 27, which is a draft of the schematic for the watch. Due to the size of schematic and the amount of systems inside of the watch, the watch schematic is broken off into sections to better illustrate the designs.

Below, Figure 28, is the circuit for the battery charging IC inside of the watch.

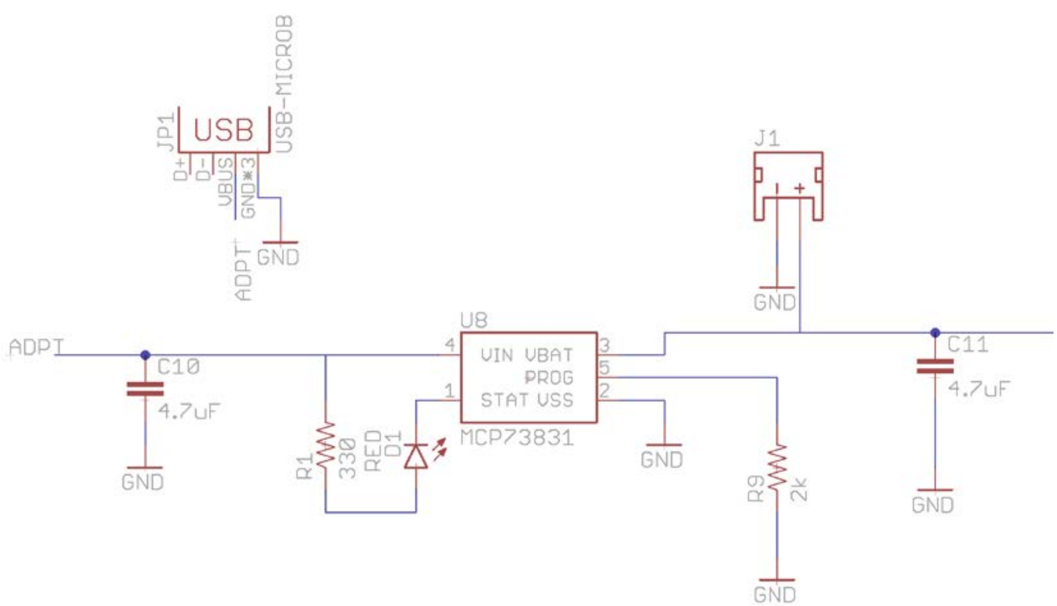


Figure 28 Charging Circuit for Watch Design

Below, Figure 29, is the circuit for the fuel gauge inside of the watch.

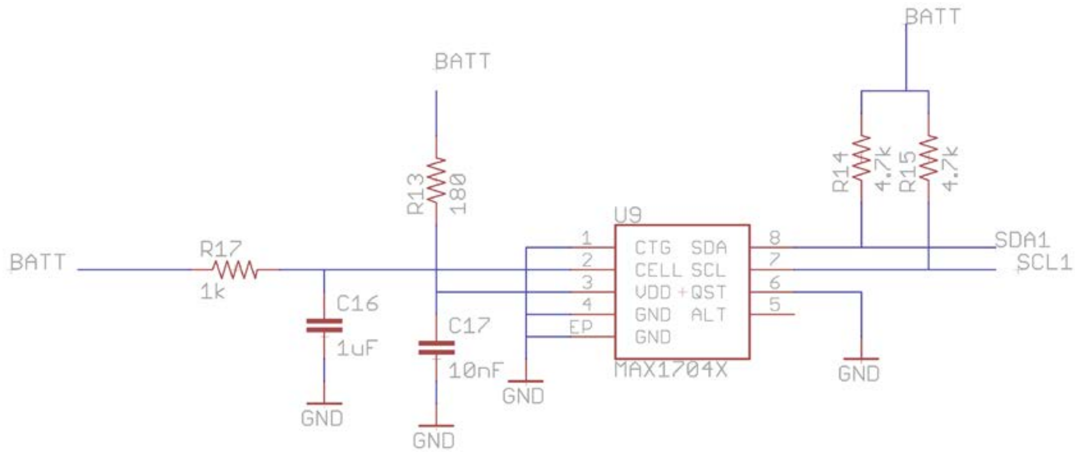


Figure 29 Fuel Gauge circuit for Watch Design

Below, Figure 30, is the circuit for the linear regulator inside of the watch.

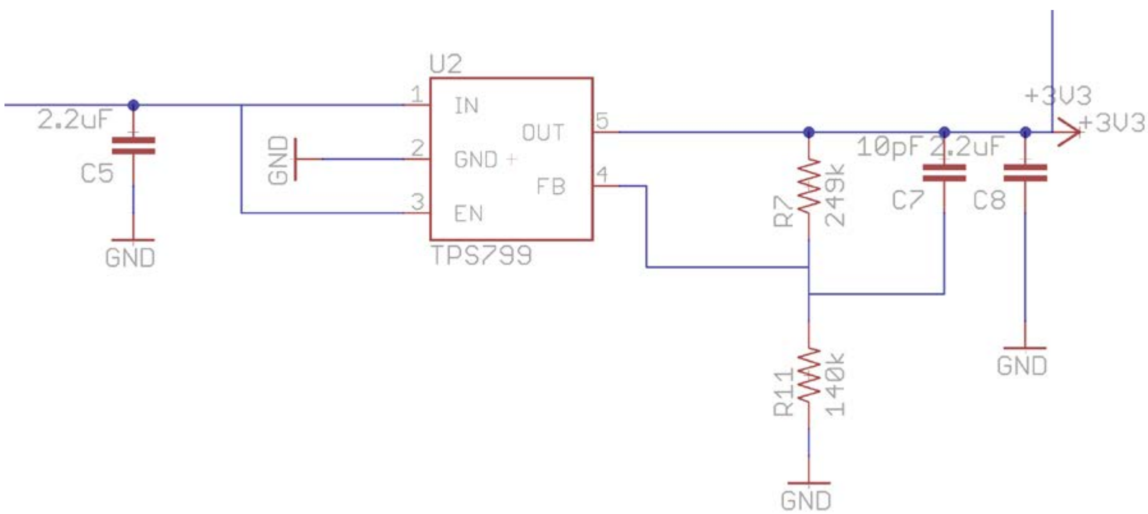


Figure 30 Circuit for the Linear Regulator for Watch Design

Below, Figure 31, is the circuit for the switching regulator inside of the watch.

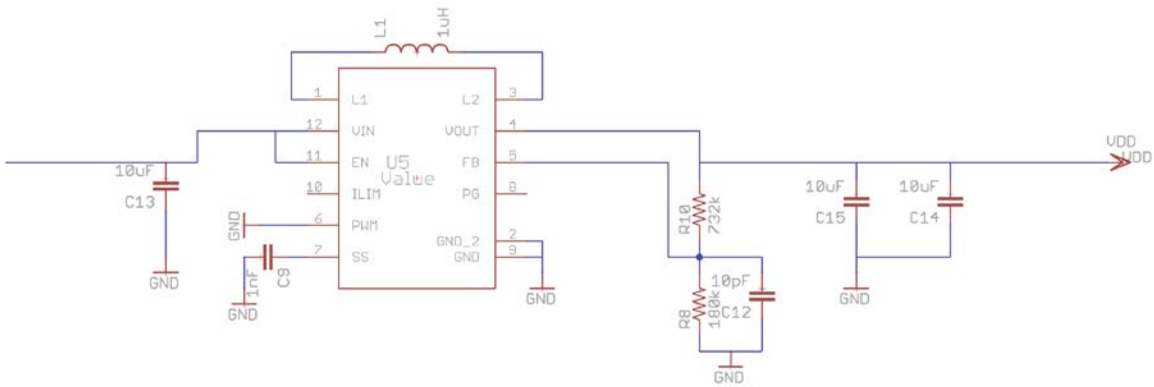


Figure 31 Circuit for the Switching Regulator for Watch Design

Below, Figure 32, is the circuit for the GPS/GSM inside of the watch.

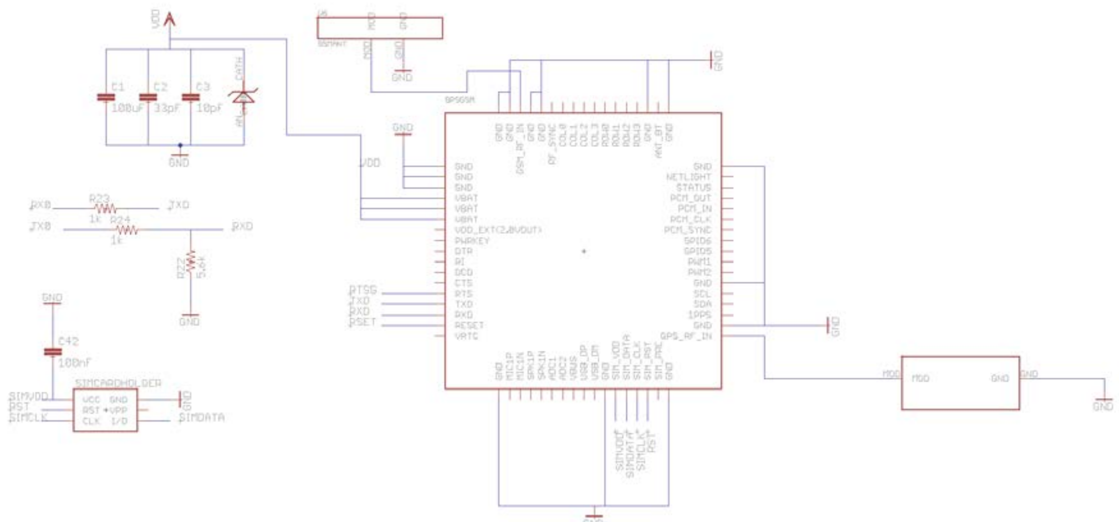


Figure 32 Circuit for GPS/GSM for Watch Design

Below, Figure 33, is the circuit for the Bluetooth inside of the watch.

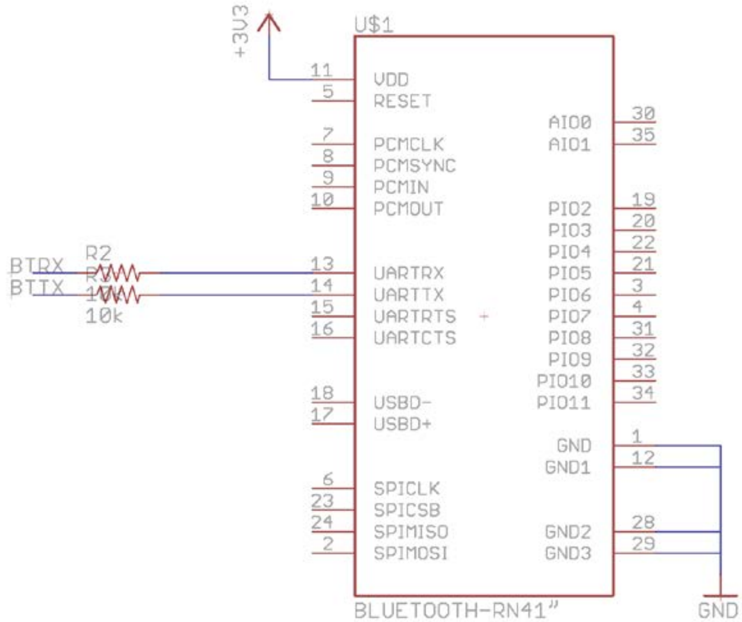


Figure 33 Circuit for the Bluetooth for Watch Design

Below, Figure 34, is the circuit for the display inside the watch.

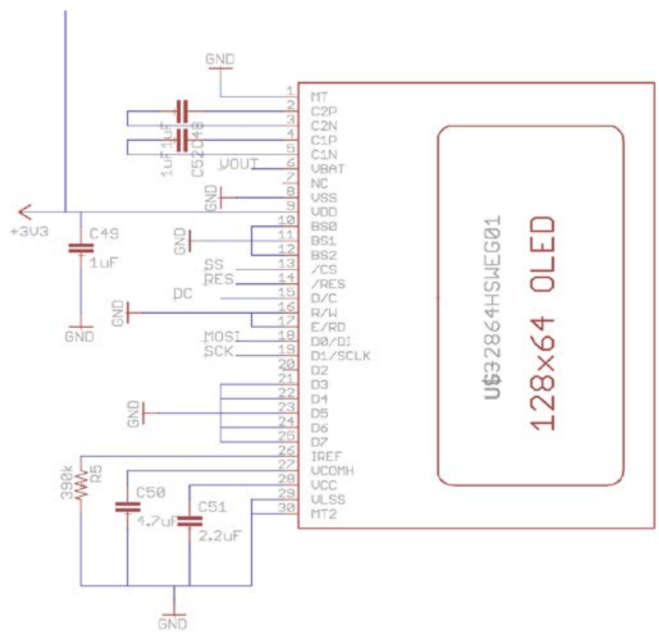


Figure 34 Display Circuit for Watch Design

Below, Figure 35, is the circuit for the MCU inside of the watch.

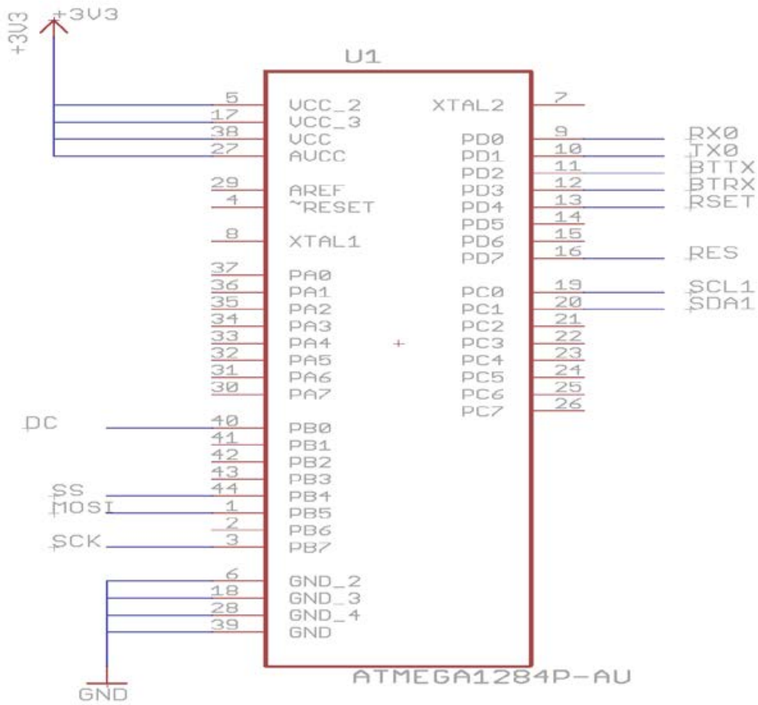


Figure 35 MCU Circuit for Watch Design

Below, Table 67, is a summary of the communication peripherals from the MCU to the assigned to the components inside of the watch's hardware.

Component	Peripheral	Pin Number
Display	SPI	Pin 44, Pin 1, Pin 3, Pin 16
GPS/GSM	UART	Pin 9, Pin 10, Pin 13
Bluetooth	UART	Pin 11, Pin 12
Fuel Gauge	I^2C	Pin 19, Pin 20

Table 67 Watch component communication peripherals

Referring to the table above, Table 67, each of the methods of communicating between the device and the MCU has a set amount of pins that are dedicated to that device. Each device has an amount of pins that are based on the communication protocol being used by that device.

6.5.1 PCB Specifications

Due to the size constraints set by the specifications, the watch's final PCB could be no larger than 63mm x 37mm. That is based on the largest device inside the watch, the battery. The design utilized both sides of the PCB to place all the components necessary for the design.

Below, Figure 36, is the final PCB design for the watch, the watch was designed to be the same size of the battery, which was 63mm x 37mm. The red represents the top layer of the PCB which includes the tracking portion and the display. The blue represents the bottom layer which includes the power system and the MCU.

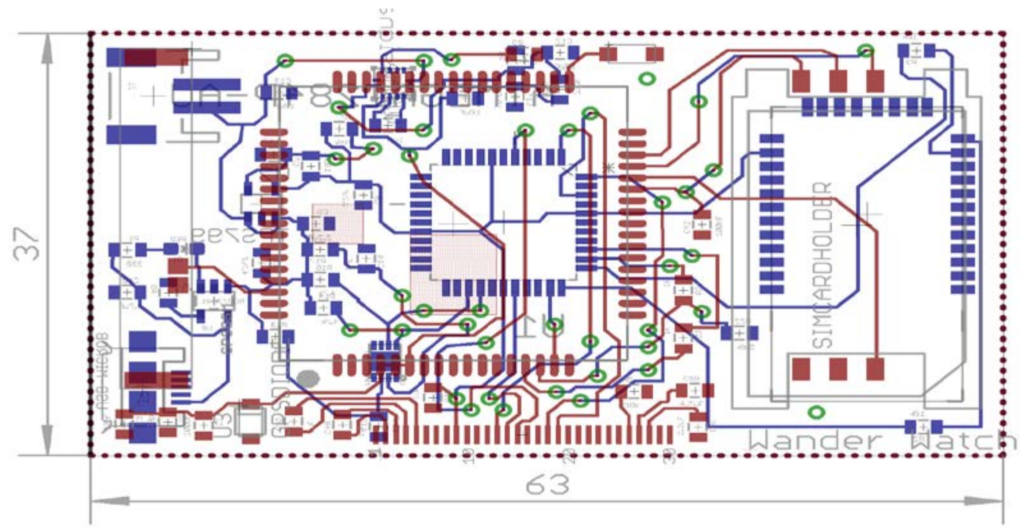


Figure 36 Final PCB for Watch Design

The options for the PCB vendors are 4PCB, PCBCart, OSH Park, Flex PCB, and PCB Way. Each vendor has their own cost of producing the PCBs for the watch. Cost can be based on how many layers the board will have, the size of the board, and the quantity of boards ordered. Below, Table 68, is a summary of the PCB vendors and the cost of producing PCBs based on the amount of layers being produced.

Vendor	Two Layer Cost	Four Layer Cost	Size Cost
4PCB	$33	$66	Fixed
PCB Cart	$61.20	$144.57	Variable
OSH Park	$5	$10	Variable
Flex PCB	$42.61	$93.48	Variable
PCB Way	$12	$49	Variable

Table 68 PCB price summary

The final board design was made using only two layers and the group decided to use PCB Way for manufacturing the PCBs for the watch.

7 Software

The following sections discusses the software design that was used for the physical components of the Wander Watch Project. This section focuses on the watch and Hub, since the Android application is discussed in detail in other sections.

7.1 Watch Software Design

The software design for the watch was focused mostly on its GPS and GSM capabilities. This involves discussion of the software involved with the GPS tracking, and their capabilities in relation to the project. The following section discusses how the chosen GPS/GSM component is able to handle all of this in terms of its software implementation.

7.1.1 GPS/GSM Software Design

The SIM808 communicates by using AT commands, which are instructions used to control a modem. If an AT command is received by the module correctly, the module responds "OK", however if there is an unexpected result the module responds "It outputs at a default rate of 9600 bits per second, but is adjustable to 115200. It has an update rate of up to 5 Hertz.

There were several AT commands that could have been implemented when programming the Wander Watch, but the group focused on receiving GPS data, receiving text message, and sending text messages. The tables below are demonstrations of AT commands that when combined implement the previously mentioned goals.

When GPS data is need an AT command is sent to power on the GPS portion of the SIM808. The next step is to define the NMEA sentence structure which, in the case of the Wander Watch, is GGA. The GPS is told to read the data and report every two fixes it receives. It the sends the formatted NMEA data to the serial monitor. Table 69 demonstrates the previously explained process.

Demonstration	Syntax	Expect Result
Turn on GNSS power	AT+CGNSPWR =1	OK
Define the last NMEA sentence that parsed	AT+CGNSSEQ="GGA"	OK
Read GNSS navigation information	AT+CGNSINF	+CGNSINF: 1,1,20150327014838.000,31.2 21783,121.354528,114.600,0. 28,0.0,1,,1.9,2.2,1.0,,8,4,,,42,, OK
Set URC reporting every 2(1-255) GNSS fix	AT+CGNSURC=2	OK
Turn off URC reporting	AT+CGNSURC=0	OK
Send Command to GNSS	AT+CGNSCMD=0,"$ PMTK000*32"	OK
Send NMEA data to AT UART	AT+CGNSTST=1	OK

Table 69 Receiving GPS Data

Receiving a text message uses only one command after text mode is enabled. The command for receiving text messages can be seen below in Table 70.

Demonstration	Syntax	Expect Result
Enable the use of text mode parameters	AT+CSDH=1	OK
Set text mode parameter	AT+CSMP=17,167	OK
Switch to text mode	AT+CMGF=1	OK
Reading an SMS	AT+CMGR=2	+CMGR: "REC UNREAD","+5551234567","Joe L","02/11/19,09:57:28+00",145,36,0,0,"+447785016005",145,8 Test sms OK

Table 70 Receiving SMS

Table 71 below shows the AT commands needed to send text messages to the user's phone. The process begins by enabling and setting the module to text mode. The phone number and message are sent in the following AT command. The module looks for a control-Z character which marks the end of a message, so remembering to send this character was a vital step.

Demonstration	Syntax	Expect Result
Enable the use of text mode parameters	AT+CSDH=1	OK
Set text mode parameter	AT+CSMP=17,167	OK
Switch to text mode	AT+CMGF=1	OK
Sending an SMS	AT+CMGS="5551234567" >Test SMS	+CMGS: 204 OK

Table 71 Sending SMS

7.2 Hub Software Design

The next section will discuss the software design for the Hub. More specifically, it will be focused on the usage of Bluetooth for the Hub's functionality. This will be used for communicating with the watch, and thus making sure that the watch is still within the Hub's range.

Understanding the software related to Bluetooth is important for allowing the Wander Watch project to perform its functions according to the project requirements. Without this knowledge, the project team would not be able to tell when the watch is within or outside of the intended range.

Thus, the next section discusses Bluetooth on a software level, showing how the pins of the Bluetooth component relate to the software design of the Hub itself. This will also prove how the particular Bluetooth component chosen for the project will be able to meet the project expectations and needs.

7.2.1 Bluetooth Software Design

The BlueSMiRF Gold serial interface uses UART to communicate to the microcontroller. The serial data goes into the module through the RX-I pin and is passed to out the Bluetooth connection. Any data coming into the module is passed through the TX-O pin. The Bluetooth module has a default baud rate of 115200 bps, but it was lowered because the other devices have trouble processing data that quickly. The Bluetooth module has two modes of operation: data mode and command mode. The command mode was used because the Wander Watch does not send any data over Bluetooth. Command mode is enter by sending the string “$$$” to the module. At this point there are several potential commands to send. Some commands set the SMiRF to specific settings, while others get information or are action commands. A few of these commands can in Table 72.

Command	Description	Factory Settings
SM,<0,1,2,3,4,5>	Set Mode (0 = Slave, 1 = Master, 2 = Trigger, 3 = Auto, 4 = DTR, 5 = Any).	4, DTR
SP, <string>	Set pin code	1234
GK	Get connection status	-
C	Connect immediately to the stored remote address	-

Table 72 BlueSMiRF Gold interface pin assignments

8 Development

Following the research of all the components, the development of the product begins for both the hardware and the software. Development starts with schematics and simulations of the hardware for the watch and mock ups of the software that can be ran on an environment without the need of the watch. Following the simulations and development of software, prototyping and product development began. The goal of development is to design the product based on the research of the parts chosen off the design constraints.

8.1 Building the Watch Prototype

When building the prototype for the watch, the goal of the prototype is to have the power system tuned to power both the MCU and the display and use power as efficiently as possible. To achieve this, the project team will build the prototype on the breadboard and test to see if the display will power up with the power system on. Any adjustments that need to be made to have the MCU and display turn on and use power efficiently will ideally be made during this stage of development. Once the ideal setup is established for the watch, then the development of the PCB will begin.

In the lab, development of the watch was done with breakout boards and all of the components sampled from the manufacturers. This process was ideal during the development stages because the components can be evaluated individually to check if the component will work properly with the overall design. Not only can an individual part be tested, a subsystem for the watch can easily be tested due to the ease of isolating the other systems from one another versus testing with a PCB. If any components that need to be changed, or if a subsystem is not working properly with the system as a whole, it can go through the redesigning stage of the product development.

After the initial development and evaluation of the hardware design of the watch has been completed in the lab, the system can then be developed onto the PCB. The goal of the PCB development is to develop the PCB using minimal space for the hardware. This will in turn reduce the overall size of the watch. The development of the PCB will also go through testing the system as a whole and any changes to the PCB that are needed will go through the redesigning stage during this time.

8.2 Development of Hub

For testing and prototyping the hub, the prototyping was done using an Arduino UNO that had a Bluetooth module attached onto a breadboard that communicated with another Arduino UNO that also has another Bluetooth module on a breadboard.

The main reason that the through-hole method can be used is because there are not size restrictions for the Hub. There is plenty of space for all the components and unlike the watch, which needs to be a small, lightweight, wearable device; the Hub is just going to be in the home as a separate entity. Even with 3 different devices needed, the MCU, the Wi-Fi module, and the Bluetooth module, it will not end up being too big [85].

Another reason to use through-hole is because it is best for prototyping, which is what is going to be needed for the Wander Watch Hub component. Also soldering and de-soldering is much easier with through-hole as compared to surface mounting. This makes replacement of components much easier. The pin count is greatly increased when using surface mount technology; so using through-hole can help to keep that down [86].

9 Testing

For the final stage of the product development cycle, the watch must go through a series of tests to ensure that both the hardware and software are working both together and meet the criterion of the design specifications and standards set at the beginning of the product life cycle. Not only is the watch going through a series of tests to meet design specifications, but also to measure how the product performs in various settings to elaborate in product details. Each test will evaluate what tests need to be conducted, the criterion for each of those tests, and the procedure of how the test should be performed and detail what needs to be measured.

9.1 Evaluation Plan

The tests for the watch are established to make sure that the watch's hardware is working with each other and that the components are working efficiently. The tests will also show if any changes need to be made to the hardware during the development stages. The tests will also show how the hardware is performing during the GPS tracking stages along with how the hardware performs with the software being developed for the watch. How the evaluation is going to be executed is to check if each test meets their respective criterion, how well the watch performs during the tests, and evaluate if any changes or improvements need to be made to the watch so that the watch will function properly.

9.2 Hardware Specific Testing

The goal and expected results the project team has for the testing stages of the Wander Watch is to see if all the hardware is powered on and functioning with limited software on the MCU. This will illustrate that all the components are functioning properly. This will also ease the transition in the communications and software end of the product's life cycle. Any changes that may need to be made during the testing stages should take into consideration how these changes could possibly affect the software of the watch.

9.2.1 Watch Component Testing

Important tests that need to be performed for the hardware include testing if the display will turn on, testing the recharging capabilities of the power system making sure that the battery will recharge, testing how long the battery will last in various settings and environments. All of the tests listed must be ran during the prototype stage and the final stages with the PCB developed to assure that the design decisions were optimal based on the design constraints.

9.2.1.1 Testing the Display

For the first test, the team will want to make sure that the display will turn on with the current set up of both the prototype boards and the PCBs. If there are any changes to the design that need to be made to make sure that the display will turn on, those changes will be made during the prototype stage. For the PCB test, this is to make sure all the components are soldered correctly so that the display will work. Another item to check on this test is to see if the date and time are displaying properly on the display. If not, the changes must be made in the software of the device.

Criterion for testing the display is to check and see if the display does power up when the watch is powered on. Following that test, the brightness settings must be checked. The brightness settings will be controlled by a button on the side of the watch. One test will be to check and see if the brightness on the display is controlled by pressing the button on the side of the watch. Another aspect of the watch that must be checked is the display showing the date, time and battery life correctly.

The procedure for testing the display is first to power the watch on and watch the OLED screen illuminate all of the LEDs on the screen. If all of the LEDs illuminate, then the screen has passed the criterion for that test. Following the power on sequence, the screen should be displaying the date and time along with the battery life percentage. If all of those elements are on the screen, then this test passes the criterion. The last test is to check the brightness settings. Initially, the screen should be at a low brightness setting in order to conserve power. To increase the brightness on the screen, the button on the side needs to be pressed in order for the LEDs to increase brightness for three seconds. If the button on the side increases the brightness on the screen for three seconds and then returns to its native state, then the test passes its criterion.

9.2.1.2 Testing Recharging

The second test will be once the battery is at low power and will need to be recharged. What is important during this stage is to check that the battery will recharge during both the prototype and final product stages. Not only will the project team need to check if the battery is recharged, but also check if the battery is displaying an accurate percentage of battery life and to see how long it will take for the watch to be fully recharged.

Criterion for the battery recharging test are checking if the battery has recharged, the accuracy of how much battery life remains, how much time did it take for the battery to recharge, and check to see if the battery has stopped drawing power from the outlet while the watch is running. The battery charging IC has more detail about overdrawing power inside of the datasheet provided by the manufacturer. Timing data will be recorded for final product details.

The procedure for checking if the battery has recharged is first depleting majority of the power at a low power level. Then, start the recharging process by connecting the battery with the micro USB. Over time, periodically check on the status of the battery life. If the battery has been fully recharged, then it has passed the criterion. Once the battery has been fully charged, the display must show and accurate rating of how much battery life is on the watch in order for it to meet the criterion. When the battery is fully charged, it should now be drawing more power for charging in order for the watch to pass the test.

9.2.1.3 Testing Battery Life

Another series of tests that needs to be ran is how long the battery will last in various settings and environments. The settings and environments that need to be checking are when the wearer is in the home while being connected to Wi-Fi, while being in home connected to Wi-Fi while the brightness on the OLED screen is set to a high brightness level, and when the wearer is outside the bounded area, ideally the home, and the watch will be in its hotspot mode. These tests will establish the product’s battery life in various settings for product details. The wearer will not be in these exact situations. However, this will give the customer a baseline for how long the watch’s battery will last on average. Ideally, this test will be conducted during the prototype stage so that any changes the project team would like to make to the watch that will improve battery efficiency can easily be made in lieu of the PCB stage where it will be seen as the final stage of development.

The criterion for this test is to check how long the battery will last in a low power mode with the OLED display on a low brightness setting, low power mode with the OLED display on a high brightness setting, both of which will be inside the home where the Wi-Fi is accessible. The second set of criterion for testing is outside of the home where the watch will be operating with the GSM active. One test will be to see how long the battery will last when the GSM is active outside of the home, sending coordinates to the application while the OLED display is on the low brightness setting. Another test will be to see how long the battery will last when the GSM is active outside of the home, sending coordinates to the application while the OLED display is on the high brightness setting.

The procedure for testing the battery life starts with the watch being connected the home Wi-Fi and having the screen on a low brightness setting. The test will begin when the watch is powered on or has been disconnected from the recharging USB. The test can be timed with any stopwatch application on a smartphone. The second test has the same procedure, except the brightness settings on the screen will be at a high setting. The third test is the same procedure, except the test will be outside of Wi-Fi range where the watch will be communicating with the GSM component. The last test will be the same set up as the test before, except the screen will be on a high brightness setting. These series of tests don’t have criterion to pass, but all times must be recorded for product detail.

9.2.2 GPS Connectivity Test

Before any other types of test are done on the GPS device, the group must ensure that the basic functionality is working. To do so, the group checked to see if the GPS is able to connect to the satellites and receive the correct coordinates. During this test, the GPS module will remain stationary. The test will consist of connecting the GPS device to a laptop where it will feed the longitude and latitude coordinates in order to confirm whether or not the GPS is connected and receiving.

9.2.3 GPS Location and Proximity Test

The GPS module testing will begin by setting up the GPS device. Two locations will be set up with two group members at each location. The first location will be called “Home Base” and the other location will be referred to as “The Wearer”. Home Base is a fixed location that will represent the caretaker. The Wearer represents the location of the watch wearer and consists of eight locations, A through H. The first four positions (A, B, C, and D) will be the control group and will be predetermined by the group members at Home Base. To ensure that the GPS device is working before the group fan out to the other locations, Location A will be Home Base. The other four locations (E, F, G, and H) will be selected by the Wearer group members. As they travel, the GPS will transmit their coordinates to members at Home Base. Table 73, located below, explains the overall set up for the test. The actual GPS coordinates will be looked up on Google Maps and the received GPS coordinates will be the incoming data from the GPS module. The two columns will be compared to one another to see how accurate the proximity of the GPS device is.

<table>
<tr><th>Location</th><th>Distance Status</th><th>Actual GPS Coordinate</th><th>Received GPS Coordinate</th></tr>
<tr><td>A</td><td>Known, Home Base</td><td rowspan="8">This section will be filled with the GPS coordinates found using Google Maps for Locations A – H.</td><td rowspan="8">This section will be the coordinates received from the GPS module as it travels to Locations A-H.</td></tr>
<tr><td>B</td><td>Known</td></tr>
<tr><td>C</td><td>Known</td></tr>
<tr><td>D</td><td>Known</td></tr>
<tr><td>E</td><td>Unknown</td></tr>
<tr><td>F</td><td>Unknown</td></tr>
<tr><td>G</td><td>Unknown</td></tr>
<tr><td>H</td><td>Unknown</td></tr>
</table>

Table 73 GPS Location and Proximity Test data entry log

9.2.4 Arduino/Bluetooth Connectivity Test

To ensure that the SMiRF is capable of pairing with other Bluetooth devices, a connectivity test will be conducted. An Arduino board with a USB Host will be connected to the Bluetooth module to assist in its testing. The Arduino will be connected to a host computer which will be receiving the messages during this test.

Once Android device's Bluetooth is enabled and the Bluetooth emulator on the device is opened, scan for Bluetooth devices in the area. The host computer should be on the list of located device and will be selected. If functioning correctly, the host computer and the Android device will become paired.

9.2.5 Bluetooth Range Test

The Bluetooth range test will help determine the range that the watch wearer will be able to travel from the Hub. Conducting this test will also help determine the reliability and accuracy of the chip. To test the range, the user will stand at a specified distance from the module. A character will be sent from the Bluetooth emulator on the Android device to the computer screen. The tabulated results for the various range tests will be filled in Table 74.

Distance (m.)	**Character Sent**	**Character Received**	**In range?**
1	'a'	'a'	Yes
2	'b'	'b'	Yes
5	'c'	'c'	Yes
10	'X'	'X'	Yes
20	'Y'	'Y'	Yes
50	'Z'	'Z'	Yes
100	'1'	N/A	No Signal
120	'2'	N/A	No Signal
150	'3'	N/A	No Signal
200	'?'	N/A	No Signal

Table 74 Bluetooth Range Test data entry log

9.3 Software Specific Testing

The following sections will discuss the specifications for software testing in greater detail. This will mostly be focused on testing related to the GPS, geofencing, and Bluetooth aspects of the Wander Watch Project.

Each section goes over how each of these topics will be tested in detail, and how each one relates to the operation of the overall project. This is important because improper communications could lead to the wrong data being shared between components. Worse yet, it could cause no data at all to be exchanged between the project's different components.

9.3.1 GPS Module Software Testing

In order to test the SIM808 connection, the group will connect it to the computer, or host, and utilize its serial communication protocol. Once the serial communication is set up an AT command will be sent to the GPS module. This AT command is “AT”. If the module is working, then it will respond “OK”. Any other messages received outside the previously specified will be deemed an error.

9.3.2 Geofence Test

The Geofence Test will be conducted in order to test the boundaries of the geofence, as well as measure the response time of the alert message. To do so, a geofence will be established around a specific building. Group members who represent the watch wearer will begin inside of the designated area with the GPS module and proceed to exit the geofence. If functioning correctly, the alert message should be sent to the caretaker’s phone. While the test is being conducted, the time it takes from when the watch wearer exits the geofence to when the caretaker receives the alert will be recorded in Table 75. The test will be conducted in an areas with both high and low mobile signal in order for the test to be unbiased.

Signal Strength	**Trial #**	**Alert Message Received?**	**Time (seconds)**
High	1	Yes: The caretaker was notified of the watch leaving the geofence. No: The caretaker was not notified of the watch leaving the geofence.	Time it took to receive alert message.
	2		
	3		
	4		
Low	1		
	2		
	3		
	4		

Table 75 Geofence Test data entry log

9.3.3 Bluetooth Communication Test

This test will be conducted in order to determine whether or not the Bluetooth module is receiving messages. To test this, the string ‘THIS IS A TEST’ will be sent to the receiver from an Android device Bluetooth emulator. If the is functioning correctly, the same message will appear on the computer screen.

9.4 Watch and Hub Communication

This section deals with testing of the communication between the watch and the Hub. The Hub has to know whether the watch is in range or not and it will be connected via Bluetooth while it is in range. Once the wearer wanders out of range of the Hub the watch will be using the GSM and GPS to communicate to the caretaker. Both of these communications will need to be tested to verify that the group has made a workable product

Note that, since the project has changed to use a geofence instead of a hub, the following section is no longer applicable to the project. Still, it is kept here to show what we had planned.

9.4.1 In Range

When the watch is in range of the Hub it is important that it periodically checks to verify that it is still in range. A time of 10 minutes in between checks is reasonable. Therefore, the watch can save power by hibernating and the Hub can also save power by using its hibernation module. If the watch were to check constantly back and forth with the Hub, the already limited battery life would be depleted far too quickly and the group does not want the wearer and caretaker to have to worry too much about charging. The goal is to only have the watch need to be charged at most once a night.

Bringing the charging up, a possible solution could be to have 2 watches. The wearer could still have tendency to wander at night, whether sleepwalking or not. And another watch to have while they sleep could give peace of mind to the caretaker so that hopefully they can get a good night's rest as well. Since the watch is going to need to charge, two separate devices to use one while the other is charging is not too out of the question.
Back to testing in range:

1. Set watch to check in with the Hub every 10 minutes.
2. Allow both the Hub and the watch to hibernate in between checks.
3. When checking in, have the Hub make note of the check by a message sent over Wi-Fi. Even an email could be a good check.
4. Only for the testing stage should the message be sent with every check in.
5. Once it is verified that the checks work and the watch and Hub are communicating properly it is appropriate to allow the Hub and watch to communicate freely without needing to send a message with each check.
6. Make sure to have a safety set, in case the wearer wanders in between checks.
7. In which case, the Hub will wake up, check, see that the wearer is gone. The message can be sent through the Hub, and the watch will then turn on the GSM component to take care of the out of range communication and the GPS tracking.

9.4.2 Out of Range

Once the watch is out of range the GSM device will be activated and the watch will be able to connect to phone towers in the area over a phone carrier. Before final presentation the device will need to be tested to ensure that the watch is actually able to communicate and send the location information and text message to the caretaker. This can be tested by:

1. Take watch out of range of the Hub. This will ensure that the Bluetooth is disconnected and it will be programmed to know when the Bluetooth connection is lost so that the GSM device can take over.
2. Once out or range just keep traveling with the watch so that many data points can be collected.
3. The watch should send the text message to the phone number that is listed for the caretaker.
4. Check the text message.
5. Verify that the coordinates on the message correspond to the correct location of the watch.
6. Once this is checked that means that the watch is working correctly out of range of the Hub.

9.5 Application Testing

Since there will be testing phases for the watch, there will also be one phase exclusively for the Wander Watch application. The application testing will be focused on ensuring that the application works as intended, and that it is easy for most users to understand and use.

The first phase of testing will be centered on the application's features and functions. The major goal is to make sure that the application can track the watch via GPS, and that the application can also receive and display alerts from the Hub. Since these are the two major features of the application, it is vital that these work before any superficial or cosmetic features that the project team want to add in afterwards.

First, the tracking feature will be tested. Once the map extension has been added to the map, it will be ensured that it actually displays a map, and that the map itself displays accurate location information for the phone itself. If the map is not displaying properly, then the project team should be able to look into it and figure out how to fix it. From there, the team will work on implementing the tracking function so that the map will display the location of the watch rather than the phone that has the application. The map extension that is available from Google will usually display the phone's own location by default. The project team will need to ensure that the application can receive the information it needs about the watch's location from the watch itself.

So the team will be testing not only the application's ability to receive the location data, but also the watch's ability to send the data to the application in the first place. It is expected that this will provide some trouble in getting it right, but the project team has made sure that there will be help available for the project. This help will be expected to come from professors here at the University of Central Florida who are experts in device to application communication.

Second, the project team will be testing the application such that it can properly receive and display alerts from the watch. Before the team tests it with actual watch alerts, the application will be tested with regular text messages. The team's research has found tutorials about making text messaging Android applications that are able to send and receive messages from one another. This tutorial will be the basis for something similar to test the application's ability to receive messages. At the same time, the application's ability to display notifications will also be tested. Notifications are alerts or updates that are displayed on the phone, but outside of the application. This is so that the user can know that something has happened without needing to constantly have the application open.

Once it has been determined that the application can both receive messages and display notifications, the project team will start testing it with actual messages from the watch. Since Android applications can already be built to receive text messages as SMS messages, the project team will be handling the watch messages in a similar manner. The watch will be tested to send the alerts as SMS messages, and the applications will be tested to receive them. As with the GPS testing, the project team has already kept a few professors in mind to ask for help if problems with the watch alerts occur during testing.

Overall, it is expected that this part of the testing will take at least 2 weeks to complete. This testing will be performed while testing the watch and Hub as well. It has been determined that this would not only save time, but also help the project team find problems or bugs faster. In addition, since the watch, Hub, and application depend on each other for information, it is important that all of three of them are tested together to ensure smooth operation and to fix any problems as soon as possible.

The next part of the application testing will be focused on making the application user friendly. For this part, the project team has decided to have some outside testers try out the Wander Watch application. The team will be observing how the testers get used to the application, and taking note of any problems or confusion they may experience. This also includes keeping track of any bugs or crashing that may occur while the application is in use. If any of these occur, the Wander Watch team will take steps to look at the code and fix these bugs as soon as possible.

Though user friendliness tends to be an objective part of any application, the project team wants to make sure that the application is easy to use and understand for most users. This may not be applicable to every possible user, but the team will make sure that the appearance and organization of the application does not impede its functionalities. The team will measure the user friendliness of the project's application by asking the testers how they feel about the application, using a scale of one to five, with five being the best experience. This will mostly be concerned with whether or not the testers had any trouble logging into and navigating the application, if they had any problems receive alerts, if they had any trouble viewing and zooming into and out of the map, and if they found the settings to be usable.

The settings themselves will also be tested for any bugs or problems. The team needs to make sure that turning battery alerts on and off still works, and that automatic alerts can be properly set up and received. The project team will also test if changing the colors of the application works, and if the available color combinations are not eye straining for most users. However, this particular feature is one that is low on the project's priority list, seeing as it is superficial and does not significantly impact any of the application's more vital functions.

Once all of this has been tested and ensured that it works as intended, the project team would like to possibly add in and test new features for the application, if there is enough time to do so. Such features include the ability to connect the application to more than one watch and receive alerts from more than one Hub. If the team works to implement these features, then said features will have to be tested to receive alerts and location information from all of the new watches and Hubs. For location data, the application would need to have the ability to switch between different maps, with one map being for one particular watch.

For the alerts, the application would need to have a good organizational scheme for the alerts. On the alert screen, the application could have it so that the user can see the alerts for each Hub or watch, and slide left or right to view alerts from a different Hub or watch. In turn, new features and functionality like these will have to be tested extensively for bugs. This is especially important because, since the application will be receive even more information and data, it could be difficult to manage and properly display everything.

However, this may be beyond the time and scope the Wander Watch team has for this Senior Design project. As a result, any further features such as these may not be worked on during the time for this course. It is still good to think about future expansions in case the team gets the opportunity to further improve the project. Even so, the team plans to focus on testing the GPS map, receiving alerts, and usability of the application. Anything else would be great to add on in addition to the major features, but the project team would not consider them necessary for the completion of the project’s current scope.

With the conclusion of the project, the application has been fully tested on all of the features discussed above. Any issues that came up during testing have been corrected and retested, and in the end the application is able to work successfully with the watch to properly track it. There was not enough time to implement additional features, but at the least the application works within the minimal intended scope of the project.

10 Administrative Content

The following sections will go into detail on non-technical parts of the project, such as how the project will handle its budget and the time spent on the project. Specifically, the first few sections will go over the various standards used as a part of this project. This is to show that the project will be complying with necessary and relevant standards. The next section after that will go over the budget and finance of this project, while the second will go into detail about the project’s expected milestones and timelines.

The budget section is meant to show the research made on the amount of money needed to buy the right parts for this project. It will also go over other finances that need to be taken into consideration for the project. The second section is to show how far the project team has planned ahead, and how the team has organized the time that is expected for building the watch, Hub, and application. It will show how the team plans to complete the Wander Watch project in a timely manner, and devote enough time to finish each major section of it.

10.1 Standards

Various standards have been reviewed for the project. These standards are ones that have been taken into account over the course of researching for the Wander Watch project. These standards are PCB, GPS, Bluetooth, Wi-Fi, IEEE 1588, Ethernet, SIM card, GPRS, and GSM standards. All of these standards were found to be relevant to this project. Thus, their relevance to the various parts of the project are discussed in the following sections.

10.1.1 PCB Standards

According to the IPC Association of Connecting Electronics Industries standards documentation for printed circuit boards, section J001E 3.2 states that all solder alloys must be Sn60Pb40, Sn62Pb38, or Sn96.3Ag3.7 in accordance with J-STD-006. J001E 3.3 also states that Flux shall also be in accordance with J-STD-004. Flux shall conform to flux activity levels L0 or L1 of flux materials rosin (RO) or resin (RE). [87]

10.1.2 GPS Standards

The Global Positioning System is operated and maintained by the U.S. Department of Defense (DoD). The GPS is a radio-navigation system consisting of 32 satellites orbiting the Earth at an altitude of approximately 12,550 miles. The satellite network, or constellation, works with a network of ground stations used for monitoring and control in order to provide the GPS user with accurate positional, velocity and time information.

Of the 32 operational GPS satellites, a standard of at least 24 are operational 95% of the time. These satellites are located in six equidistant planes and four slots. Each plane is tilted at 55 degrees relative to the equator. Every satellite can be identified by a two character code: a letter which represents the orbital plane and a number which identifies the satellite number in the plane.

The standards for the GPS position and geometric range computations are based on the World Geodetic System 1984 (WGS 84) Earth-Centered, Earth-Fixed (ECEF) coordinate system. The time standards of the systems are set to Coordinated Universal Time (UTC). There are different standards for civilian and the U.S. military performance specifications.

For example, the horizontal and vertical positional error for civilian users is 100 meters and 156 meters, respectively, 95% of the time, whereas the U.S. military has a 16 meter 3-D positional error 50% of the time. The 95 percentile timing accuracy for civilians is 340 nanoseconds and for the military it is 200 nanoseconds. [88]

10.1.3 Bluetooth Standards

The Laird BT800 Bluetooth module is based on the Bluetooth Core Version 4.0, also known as Bluetooth Smart. This version of the Bluetooth standard combines both Version 1 and Version 3, focusing on the low power applications and faster transmission speed. The combine technology opened up many doors for the latest version of Bluetooth.

It also costs less and had more flexible development capabilities. The design of the Bluetooth Version 4.0 can implement both dual mode and single mode. Only the low energy protocol stack is implemented in a single mode implementation, however in the dual mode implementation, Bluetooth Smart is integrated into an existing Classic Bluetooth controller. More information is shown in the table below (Table 76).

Operating Frequency	2.4 GHz
Min. Output	Approx. 0.01 mW (-20 dBm)

Max. Output	Approx. 10 mW (+10dBm)
Peak current consumption	<15 mA
Over the air data rate	1 Mbit/s
Application throughput	0.27 Mbit/s
Latency	6 ms
Topology	Piconet
Security	128 bits AES CCM
Modes	Broadcast, Connection, Event Data Models Reads, Writes

Table 76 Bluetooth Standards

Every Bluetooth device operates at a frequency of 2.4 gigahertz. The standard established the minimum output power level for a transmitter to be approximately 0.01 milliwatts, and set the maximum output power level at approximately 10 milliwatts. Bluetooth sends data over the air at a rate of 1 megabit per second and has an application throughput of 0.27 megabits per second.

Bluetooth networks, also known as, piconets use a master/slave relationship to communicate. In Classic Bluetooth technology, there can be up to seven active slaves, whereas Bluetooth Smart technology does not have a defined set number of slave and depends on the implementation. These networks are vital for proper communications between different parts of the network.

10.1.4 Wi-Fi Standards

Wi-Fi uses 802.11 networking standards and of those there are a few different kinds:

- 802.11a can transmit at 5 GHz that will transfer up to 54 megabits of data per second. This kind uses orthogonal frequency-division multiplexing (OFDM). OFDM is a technique of coding that is more efficient because the way it works is to split a radio signal into sub-signals before the reach a receiver. This split allows for a reduction in interference.
- 802.11b is the cheapest standard available, but it is also the slowest. Its popularity has declined as faster standards have become more affordable. The transmission rate is 2.4 GHz frequency band of the radio spectrum. It is able to handle up to 11 megabits of data per second, and it uses complementary code keying (CCK) modulation, which is helpful for improving speeds.
- 801.11g has a transmission rate of 2.4 GHz but with an ability to handle up to 54 megabits of data per second it is much faster than 802.11b. The reason that it is faster is because it uses the OFDM coding technique that 802.11a uses.
- 802.11n is most widely available of all the standards. It is also backward compatible, meaning it can use earlier versions of data and interfaces [89], with a, b, and g. 802.11n has been reported to achieve speeds of up to 140 megabits per second. It can also transmit 4 streams of data, with each transmitting 150

megabits per second. Though it has the capability to transmit 4, most routers are not able to handle that many and typically can only allow for 2 or 3 streams.

- 802.11ac is the most recent standard available as of early in 2013. At IEEE it is still in its draft form but there are devices on the market that will support it. 802.11ac is backward compatible with standard n, and therefore also a, b, and g. This standard is less prone to interference and is much faster than the standards before it. It can push up to 450 megabits per second when using a single stream. Though real-world applications most likely have less speed. It allows for multiple streams with an option for up to 8. A few other names for this standard are 5G Wi-Fi because it uses the 5 GHz frequency band, Gigabit Wi-Fi because it has potential to exceed a gigabit per second when using multiple streams, or Very High Throughput (VHT) again because of the potential to exceed a gigabit per second [90].

10.1.5 IEEE 1588 Standards

As far as standards go there are not standard methods of clock adjustment outlined by the IEEE 1588. There is only a standard protocol for the exchange of messages between clocks. This is a huge benefit of the protocol because it means that clocks from different manufacturers are able to synchronize with each other [69].

10.1.6 Ethernet Standards

Ethernet has a simple set of rules, but first here is some background of how Ethernet works:

- Medium – devices attached to a common medium that will provide a path along which the electronic signals can travel as shown in Figure 37 below. In the past it was typically a coaxial copper cable, but nowadays it is more common to have a twisted pair or fiber optic cabling.

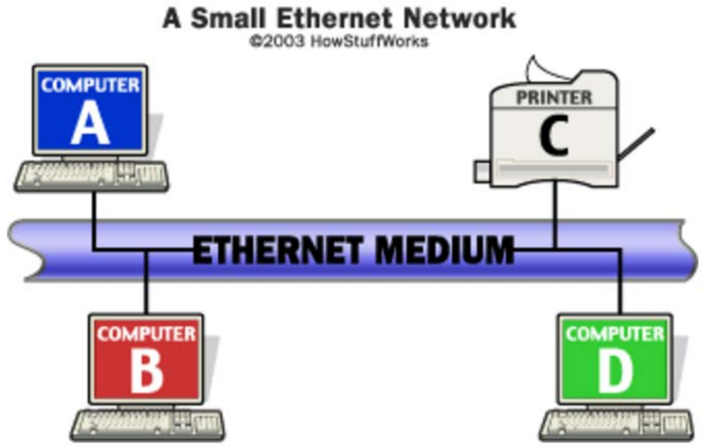


Figure 37 Ethernet Medium (Courtesy of How Stuff Works)

- Segment – this is a single shared medium
- Node – devices that are attached to the Ethernet segment are referred to as nodes or stations
- Frame – a frame is a variably sized chunk of information, and this is how the nodes communicate in short messages
- If a device on the network receives a frame it checks to verify if the frame is intended for itself and if it isn't it discards it without even examining the contents of the frame
- A broadcast address is a bit different in that its destination address is equal to the broadcast address, because it is intended to be sent to all nodes on the network. Every node is able to receive and process the frame
- Network diameter is the practical distance between two devices on an Ethernet network. The network cable must be short enough that devices on opposite ends of the network can receive messages from each other with minimal delay and clearly.
- Network diameter can be increased with a repeater, which is a way to connect multiple Ethernet segments. The repeater listens to each segment and will repeat the signal heard on one segment onto each of the other segments that it is connected to.
- Segmentation of an Ethernet network can help with congestion issues by allowing a single segment to be split into multiple segments. This creates multiple collision domains, but also creates another problem of separate segments not being able to share information with each other.
- Bridges help eliminate the problems caused by segmentation. Bridges are used to connect two or more network segments as shown in Figure 38 below, which allows an increase in network diameter like a repeater does, but bridges can help

to regulate traffic. They receive and send transmissions just as any other node does, but do not function in the exact same way. The bridge will still transmit what it receives to the other segment of the network, but before it does this it examines the destination address before deciding how to handle it. By doing that the bridge is able to reduce unnecessary traffic because it decides whether it needs to transmit to the other segment of the network or not.

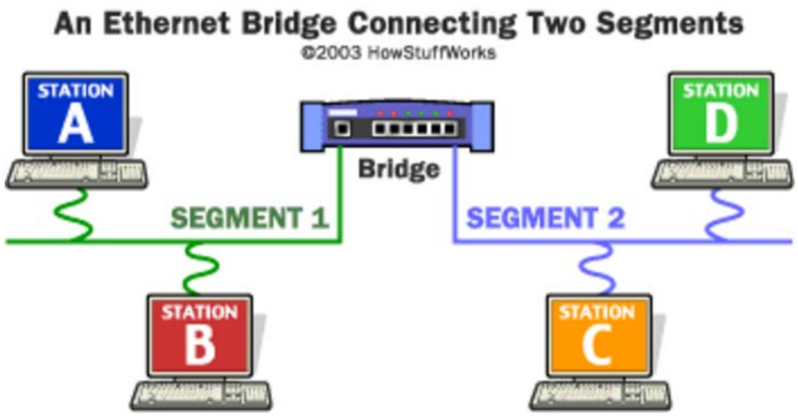


Figure 38 Ethernet Bridge (Courtesy of How Stuff Works)

Ethernet protocol calls for:

- Explicit minimum and maximum lengths for frames.
- Set of required pieces of information must appear in these frames.
- Ex: each frame is required to contain a destination address and a source address, which will identify the recipient and the sender of the message
- No two Ethernet devices should ever have the same address, because the address is how the device/node is uniquely identified.
- Because the medium is attached to every device on the network it is critical that the address of the destination is on there because that is how it will get to the intended recipient.
- Regulates communication among nodes using carrier-sense multiple access with collision detection or CSMA/CD. This protocol means that when one station is transmitting, all the other stations are listening. If another station has something to transmit while the network is currently in use, it will wait and that is the carrier sense part of the protocol. In essence the stations "listen" in to determine if another station is transmitting before beginning their own transmission.
- Another important part of the protocol deals with collision. If there is silence and two stations begin to transmit at about the same time then a collision occurs and their transmission comes back distorted. So they know a collision has happened and will cease their transmission. The protocol calls for the station to wait a

random amount of time before checking again that the medium is silent. This random amount of time helps to ensure that it is unlikely that they will collide again and again indefinitely.

- CSMA/CD causes slight limitations on the number of devices that can be connected to one Ethernet cable. This is because only one station can transmit at one time, that if there are too many devices on the medium then each will have to wait too long of a time to transmit any data and it is not practical [91].

10.1.7 SIM Card Standards

The standards for SIM cards were developed by ETSI and are as follows:

- Specification of cordless telephone system subscriber identity module for both fixed part and mobile station
- SIM conformance test specification
- Security aspects: SIM – Mobile equipment interface
- Test specification
- Test specification for SIM application Programming Interface for Java card
- Mobile Station conformance specification; Part 4 SIM application toolkit conformance test specification
- Universal Subscriber Identity Module (USIM) conformance test specification
- USIM Application Toolkit
- Characteristics of the USIM application
- Part 3 Universal Integrated Circuit Card (UICC)/ Characteristics of the TSIM application [92]

10.1.8 GPRS Standards

The standards GPRS were developed by ETSI and are as follows:

- Mobile System, Base Station System interface, Radio Link Control/Medium Access Control (RLC/MAC) protocol
- GPRS Tunneling Protocol (GTP) across the Gn and Gp Interface
- GPRS Service description; Stage 2
- GPRS Service description; Stage 1
- Network Integration Testing between GPRS and Internet Protocol Networks; Part 1: Test Suite Structure and Test Purposes (TSS&TP)
- Guide to TETRA Advanced Packet Service (TAPS)
- GEO-Mobile Radio Interface Specifications (Release 2) GPRS [93]

10.1.9 GSM Standards

The standards for GSM are as follows:

- Frequency band in United States: 1900 MHz, devices are called tri-band because the devices can operate in Europe and the United States

- European standard is 900 MHz to 1800 MHz, devices are called bi-band if they can only be used in Europe
- Maximum throughput of 9.6 kbps, this allows for transmission of voice and low-volume digital data [94].

The standards defined by ETSI are:

- Physical layer on the radio path; General description
- Multiplexing and multiple access on the radio path
- Radio transmission and reception
- Network architecture
- Technical Specifications and Technical Reports for a GERAN-based 3GPP system [95].

10.2 Budget and Finance

Having a good budget is an important aspect of any good project. It is necessary to keep track of how much money is being spent on what parts or services. This section will explain how much money the project team expects to spend on parts and other items needed for the project.

It will also help show how the project team plans to make it so that this project is easy to build and produce. Though the final price will be shown to be more than originally planned, it has been discussed in other sections why these particular parts are deemed to be necessary for the project's overall success.

Extensive research has been made by each team member to put together a final table of parts that will be needed for the Wander Watch. One positive thing is that development of the Android application should not incur a cost by itself. The cost mainly lies in the physical parts.

In many cases, the cheapest parts could not be chosen. This is mainly because the cheapest parts did not always meet the needs and requirements of the project. Thus, for some parts of the project, a more expensive option was chosen instead because it meets more of the project requirements.

The following table, has been made based on the project team's research into how much each component of the Wander Watch should cost. Each part shown here has been determined to be necessary for the project to function as planned. Note that the final price is for the components discussed individually. It does not take into account the possibility of buying extra or replacement parts during development.

List of Final Parts

Part Type	Name	Manufacturer	Qty.	Price
Bluetooth	BlueSMiRF	Sparkfun	2	$29.95
MCU	CC3200	Texas Instruments	1	$16.78
Fuel Gauge	BQ27010	Texas Instruments	1	$3.63
Switching Converter	TPS61093-Q1	Texas Instruments	1	$4.21
LDO Regulator	TPS74701	Texas Instruments	1	$2.10
Battery Charging IC	BQ24232	Texas Instruments	1	$2.88
Battery	3.7V 2000mAh Li-Po Battery	Sparkfun	1	$12.95
Display	OLED 128 x 64 px Display	Generic	1	$10.45
USB	USB Micro-B Breakout Board	Adafruit	1	$1.50
GSM/GPS	SIM808	Adafruit	1	$29.95
SIM Card	2G SIM Card from Ting	Ting	1	$9.00
MCU	TM4C1294NCPDT	Texas Instruments	1	$16.95

Table 77 Project Parts List

It should also be noted that the messages will be sent over SMS, so this will incur a cost depending on the network. However, as stated in previous sections, this will only be a very minimal cost, up to roughly ten cents per message. This is based on the $3 a month fee for thirty messages, which the team found was the best pricing for messages that could be found for SMS messages.

Overall, the project team expects the Wander Watch project to cost roughly $170.30 for just the parts. Including about a $12 for the cost of messaging fees the total cost comes to around $182.30. This does go over the intended goal of $100 or less, but the project team has determined that these parts were the best options for this project.

The final price does not include the estimated cost of the casings for the watch and Hub. These outer casings will be made using a 3D printer at the University of Central Florida. The decision to use a 3D printer was made because the project team has found that this is the cheapest way to obtain custom casings at an affordable price for the project. It is assumed that the cost of the cases will be roughly about $5 for each of the casings, for a total of around $10. This is such that they should be easy and affordable to produce for future instances of the Wander Watch. The price for the cases is based on the pricing for

3D printing at the university. [96] The group was able to print the case for free so there ended up begin no cost associated with this portion of the project.

Unfortunately, the Wander Watch team has been unable to secure outside funding for this project. Originally, Texas Instruments was considered as a major sponsor, but the team could not show how the project could be accepted for Texas Instruments' funding program. There are strict guidelines that must be followed in order for a project to be funded by this company. Since the team was unable to change parts of the project to fit the requirements, the Wander Watch project was not able to be accepted for funding.

The team has been unable to obtain funding from other groups well. Mainly this was due to how the purpose of the Wander Watch relates mostly to taking care of dementia patients and the elderly. This project has a useful application, but it does not fit the interests of most Senior Design sponsors. Many of the sponsors the Wander Watch team has spoken to were mostly interested in projects that could be used for various military applications. Others were looking for projects that could be utilized for expanding their own existing products and projects. Some groups had already given their sponsorship to other Senior Design teams by the time the Wander Watch team tried to make contact with them. As a result of this lack of sponsorship, this project will be funded with the team member's own set of funds.

Since a major goal of the project is to make the Wander Watch affordable to produce and purchase, the team will be using this opportunity to show that this project can be doable on a low student budget. Other projects, such as previous products discussed near the beginning of this report, may take hundreds of dollars more for development than what is planned for the Wander Watch project. Some projects may even cost thousands of dollars. The team has been focused on keep the cost of this project down as much as possible. This is why the final list of parts has parts that are low in cost but will still satisfactorily meet the requirements for the Wander Watch.

10.3 Milestones and Timeline

This section will discuss how the project team has planned out the timeline for building and testing the watch, Hub, and application for this project. It is important to budget both money and time wisely. Otherwise, the Wander Watch project could run out of time for development, or run over the expected budget for physical parts.

If everything is completed early enough, the project team may have extra time for additional testing and bug fixing. This extra time could also possibly be enough to add in extra features. However, even if the team does not have enough time to add more features, it is expected that the team will have enough time to at least finish what the project needs to function correctly.

10.3.1 Major Milestones

For this project, a milestone is considered to be a point at which the project team completes a significant step in developing and building this Senior Design project. The first major milestone will be done by the end of the fall semester. This will be the conclusion of the team's research, which will show that the project team is now prepared to move on to building the watch, Hub, and application for the Wander Watch project.

The project team has also begun collecting the physical parts for build the watch and Hub. All of the parts that are needed will be collected before the beginning of the spring semester of Senior Design. This way, the team can have plenty of time to begin putting together and testing these parts. This will also give the project team enough time to change parts or obtain more parts in case something goes wrong during testing or building the watch and Hub.

Concurrently with the collection of the physical components, the project team will also make sure that the basic skeleton of the application is developed before the beginning of the spring semester. This skeleton will have the basic features of the apps. These include: a working login screen, users have the ability to create an account and log in with it, and having the main screens, which are Main, Map, Alerts, and Settings. These screens will be fairly basic.

For example, the Alerts screen will not have any actual alerts. Instead, it will probably have some test messages in place of actual alerts. Likewise, the map will not display the location of the watch, since the watch will not be built and tested at that point. Instead, the map will just display the current location of the user's phone. In essence, this will be similar to the expected final application, simply without the correct map, ability to receive alerts, and ability to change settings related to the alerts and watch.

So by the beginning of the spring semester, the Wander Watch team will have three major milestones completed. Once the spring semester starts, the team plans to build and test the three aspects of this project as soon as possible. This will be done so that this project can be finished as soon as possible, while also giving the team plenty of time to test everything and fix any problems.

The building of the watch, Hub, and application are each their own milestone. The building of the watch includes its electronic components, but also includes the casing needed to house these components, as well as the strap by which the watch can be attached to a person’s wrist. Likewise, the building of the Hub includes the physical casing of the Hub, and making sure its electronic components will be protected by the casing.

The testing of each will also be separate milestones. Ideally, the building and testing of each of the three aspects will happen simultaneously. Since each aspect relies on some features of the others, having them worked on at the same time will help with testing communication between the three. This not only saves time for building and testing, but will also help the team find any bugs or problems as soon as possible.

The final planned milestone will be the presentation and demonstration of this project at the end of the spring semester. This will show that the project team has completed a successful project for Senior Design. It will be ensured that the building and testing of this project will be completed well before the demonstration. If the team spend too much time testing or building, the project could end up being rushed, thus finishing the Wander Watch project with a lackluster project.

In order, the project team has 10 major milestones for this Senior Design project. These are finishing research by the end of the fall semester, collecting all the necessary physical parts, building the basic skeleton of the application, building the watch, building the Hub, adding the required features for the application, testing the watch, testing the Hub, testing the application, and giving the final presentation and demonstration of the project.

Preferably, the Wander Watch team will be able to reach these milestones on time. To aid the team in keeping track of what they should work on, an estimated timeline of the project has been created. This timeline is shown in the next section. It will help the team stay organized and focused on what is needed to complete each major step of the overall project. Completing each step will help the team reach the final goal of a completed Senior Design project that will work and function as planned.

In addition to preserving the success of the project, it will help the team keep track of how much time is spent on certain steps. This will help the project team to determine whether or not the team can afford to spend more time on later steps, or have to speed up in order to finish everything by the end of the spring semester.

10.3.2 Planned Timeline

The following timeline is only a rough estimate of how the Wander Watch team expects to spend time for the Wander Watch project. Milestones that are closer to the current date are more concrete, but those further into the spring semester may be subject to change.

This will depend on how much time is spent on reaching a particular milestone, or if certain problems and delays occur along the way.

The later dates, such as those for building and testing each aspect of the project, are the furthest dates that the project team expects to finish with a particular step. In essence, each milestone should be completed before the date that is listed here in this timeline.

The final date for the presentation is subject to change. This is because the final date for presentation is not yet known. It will be based on the actual date that is set during the spring semester of Senior Design. The data listed on the timeline is based on the last possible date of the official final examination week for the University of Central Florida for this semester. Overall, the plan is to finish all the necessary building, implementation, and testing during the spring semester. The best case scenario is that the Wander Watch team will able to complete these steps well before the final presentation. This is without focusing on additional features, and only on the features that have been discussed and planned so far.

Besides allowing for more time for testing or implementing additional features, this extra time should also allow the team to properly prepare the final presentation and demonstration. It is important to do so as soon as possible since the presentation will not only show how much the team has learned over the course of this project, but also whether or not the project was successfully completed. The project timeline is shown below.

Project Timeline		
Date	**Title**	**Description**
11/10/15	Research Completion	Finishing this essay
1/11/16	Complete Parts Collection	Collect and purchase parts for the watch.
1/11/16	Complete Skeleton App	Finishing the basic skeleton of the app.
2/29/16	Prototyping	Developing systems for the watch and Hub.
4/1/16	Testing	Test the watch and verify functionality.
4/21/16	Final Project Presentation	Presenting and demonstrating the final product.

Table 78 Project Timeline

It is hoped that this timeline will help serve as a reminder about what the project team should be focusing on. It should also aid them in keeping track of what needs to be done, and how much time can be afforded to the completion of a certain milestone. Though the timeline is still subject to change, it will still hopefully serve as a useful guideline for the success of the Wander Watch project.

11 Future Expansions

With technology changing as rapidly as it is the future of the Wander Watch is a boundless. One aspect of the Wander Watch that could be developed in the future will be integrating multiple watches into a single network. Another future venue is to add sensors to the watch. These sensors will monitor basic vital signs such as heart rate and blood pressure. In the case of watch owners who have preexisting heart conditions, or high or low blood pressure, a medical alert will be sent when the patient’s heart rate exceeds normality or if their blood pressure rises or falls above or below its normal level.

The implementation of RFID tags would provide the Wander Watch with two new, useful features: indoor tracking and storing information. In its current state, the watch can tell when the patient leaves the house but cannot tell where inside the house the wearer is. Installing RFID readers in strategic locations around the house can create an indoor positioning system. The RFID tags can also store information on them, which in the case of the Wander Watch, will be used to store a brief medical history of the patient. Examples of this information include allergies, existing heart conditions, medicine they need, or their home address.

Another feature that could be added to the Wander Watch is a panic button. If the wearer falls down in their home or injures themselves in any type of way, they could press the panic button and an alert message will not only be sent to their caretaker, but also to an emergency dispatcher. When the ambulance arrives all the emergency responders need to do is scan the watch to receive the patient’s medical history located in the RFID tag inside of the Wander Watch.

These new features will make the Wander Watch not only useful for everyday users, but it will also expand marketability to hospitals and nursing homes. Multiple patients can have their vital signs monitored, movement tracked, and medical history on hand within a matter of seconds all through the Wander Watch.

12 Conclusion

The Wander Watch has allowed the group members to experience what it is like to thoroughly research and design a project. As in a real world application, the team members collaborated and joined with other members to make the idea a reality. The task of taking the products already on the market and trying to make a product of similar function but with some important changes was challenging. This senior design project has opened the eyes of the members to how much has been learned, but also how much there is that is unknown. Luckily a good base of knowledge has been earned throughout the student careers of the members. This project has allowed that knowledge base to be used and implemented and expanded through real world application. This project has also allowed the group members to gain expertise in several areas of engineering because while each member focused on a specific area of the project, there is still need for each person to have at least a base of knowledge of the overall project and each subsequent part.

Throughout the research process the group realized why the products out there now are so large, but took on the task of making a device that is much smaller and yet still functional. It was learned that finding products to implement into one device can be challenging. Many different products were looked at and decisions such as size, cost, and power consumption were used to decide what will be used in the final product. Compatibility was a slight constraint because finding products that have the right components to communicate with each other was difficult especially because often times the products deemed best for each component were not from the same manufacturer.

The main purpose of the Wander Watch is to track Alzheimer and dementia patients that stay home alone and may have a tendency to wander. The final product is going to be a watch sized device that is worn on the wrist, it is important that it also function as a watch because it should not feel like a tracking device, but rather something the patient will want to wear. The watch will have a GPS component that allows the wearer to be tracked once they wander out of the home, and it also has a GSM device that allows it to communicate through phone towers to send the message to the caretaker that they are out of the home and also will send their location to the caretaker so that they can be easily found. Though power was a big constraint on the project, through all the research, the group believes that the right components and devices have been found that will allow for a decent battery life of the final product. Also an application is being developed that will allow for the real-time tracking of the wearer.

Appendix A: References

[1] A. Sauer. (2014, August 8). 10 Lifesaving Location Devices for Dementia Patients. [Online]. Available: http://www.alzheimers.net/8-8-14-location-devices-dementia

[2] Project Lifesaver International: Choices to Enroll My Loved One. [Online]. Available: http://www.projectlifesaver.org/enroll-a-loved-one/

[3] MindMe: Personal Alarms for the elderly, dementia sufferers & lone workers. [Online]. Available: http://www.mindme.care/

[4] GPS SmartSole GPS Tracking Device – Wearable Technology that keeps you connected to the ones you love. [Online]. Available: http://www.gpssmartsole.com/

[5] Bluetooth Low Energy SmartSole – Wearable Tracking Technology – keeping you connected to the ones you love. [Online]. Available: http://www.gpssmartsole.com/ble-smart-sole.php

[6] Safe Link: How Does It Work? [Online]. Available: http://www.safelinkgps.com/index.php/get-started/how-it-works

[7] PocketFinder GPS Senior Tracker. [Online]. Available: http://pocketfinder.com/gpsseniortracker/

[8] Revolutionary Tracker. [Online]. Available: http://www.revtrackbuy.com/

[9] Alzheimer's Association | GPS Tracking | Comfort Zone Check-In. [Online]. Available: http://www.comfortzonecheckin.com/default.aspx

[10] Elderly and Alzheimer's GPS Tracking Locator Watch with Bluewater Security Professionals. [Online]. Available: http://www.bluewatersecurityprofessionals.com/elderlytracking.htm

[11] Medic Alert Safely Home | Alzheimer Society of Canada. [Online]. Available: http://www.alzheimer.ca/en/Living-with-dementia/Day-to-day-living/Safety/Safely-Home

[12] Keruve Best locator Alzheimer. [Online]. Available: http://www.keruve.com/best-personal-locator/

[13] F*watch Project Description. [Online]. Available: http://www.ohwr.org/projects/f-watch/wiki

[14] K. Bonsor and W. Fenlon, "How RFID Works," HowStuffWorks.com, 5 November 2007. [Online]. Available: http://electronics.howstuffworks.com/gadgets/high-tech-gadgets/rfid.htm. [Accessed November 2015].

[15] J. Wright, "Security Laboratory: Wireless Security," SANS Technology Institute, [Online]. Available: http://www.sans.edu/research/security-laboratory/article/bluetooth. [Accessed October 2015].

[16] C. Franklin and J. Layton, "How Bluetooth Works," HowStuffWorks.com, 28 June 2000. [Online]. Available: http://electronics.howstuffworks.com/bluetooth2.htm. [Accessed October 2015].

[17] M. Rouse, "Geo-fencing (geofencing)," WhatIs.com, September 2015. [Online]. Available: http://whatis.techtarget.com/definition/geofencing. [Accessed December 2015].
[18] Android Developers, "Creating and Monitoring Geofences," Android Developers, [Online]. Available: https://developer.android.com/training/location/geofencing.html. [Accessed December 2015].
[19] G. Barlett, "The Difference Between Active and Passive GPS Tracking Devices," Rocky Mountain Tracking, Inc., 25 May 2009. [Online]. Available: http://www.rmtracking.com/blog/2009/05/25/the-difference-between-active-and-passive-gps-tracking-devices/. [Accessed September 2015].
[20] D. DePriest, "NMEA Data," [Online]. Available: http://www.gpsinformation.org/dale/nmea.htm. [Accessed November 2015].
[21] Wikipedia contributors, "NMEA 0183," Wikipedia, The Free Encyclopedia, 17 Novemeber 2015. [Online]. Available: https://en.wikipedia.org/wiki/NMEA_0183. [Accessed November 2015].
[22] M. Grusin, "Serial Peripheral Interface (SPI)," Sparkfun, [Online]. Available: https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi. [Accessed November 2015].
[23] Sparkfun, "I2C," Sparkfun, [Online]. Available: https://learn.sparkfun.com/tutorials/i2c. [Accessed November 2015].
[24] W. contributors, "I2C," Wikipedia, The Free Encyclopedia., 28 November 2015. [Online]. Available: https://en.wikipedia.org/w/index.php?title=I%C2%B2C&oldid=692767205. [Accessed November 2015].
[25] Sparkfun, "Serial Communication," Sparkfun, [Online]. Available: https://learn.sparkfun.com/tutorials/serial-communication#uarts. [Accessed November 2015].
[26] S. Segan, "CDMA vs. GSM: What's te Differece?," 6 February 2015. [Online]. Available: http://www.pcmag.com/article2/0,2817,2407896,00.asp. [Accessed 7 December 2015].
[27] J. Herrman, "Gizmodo," 14 September 2010. [Online]. Available: http://gizmodo.com/5637136/giz-explains-gsm-vs-cdma. [Accessed 7 December 2015].
[28] P. Nickinson, "AndroidCentral," 17 January 2012. [Online]. Available: http://www.androidcentral.com/android-z-what-gsm. [Accessed 7 December 2015].
[29] Adafruit, "Adafruit FONA 808 Shield - Mini Cellular GSM + GPS for Arduino," Adafruit, [Online]. Available: https://www.adafruit.com/product/2636. [Accessed 7 December 2015].
[30] Adafruit, "Adafruit FONA 808 - Mini Cellular GSM + GPS Breakout," Adafruit, [Online]. Available: http://www.adafruit.com/products/2542?gclid=CMqAu6Cv0cgCFdYUHwodZjsCSA. [Accessed 7 December 2015].

[31] Adafruit, "Adafruit FONA - Mini Cellular GSM Breakout uFL Version - v1," Adafruit, [Online]. Available: http://www.adafruit.com/product/1946. [Accessed 7 December 2015].

[32] Sparkfun, "GSM/GPRS Module - SM5100B," Sparkfun, [Online]. Available: https://www.sparkfun.com/products/9533. [Accessed 7 December 2015].

[33] Quectel, "GSM/GPRS M66," Quectel, [Online]. Available: http://www.quectel.com/product/prodetail.aspx?id=73. [Accessed 7 December 2015].

[34] R. Ware, "What is a SIM card? And why do we need one?," About Tech, 2015. [Online]. Available: http://cellphones.about.com/od/faqs/f/What-Is-A-Sim-Card-And-Why-Do-We-Need-One.htm. [Accessed 7 December 2015].

[35] J. Lee, "Why Do Cellphones Need a SIM Card?," Make Use Of, 6 December 2013. [Online]. Available: http://www.makeuseof.com/tag/why-do-cellphones-need-a-sim-card/. [Accessed 7 December 2015].

[36] Adafruit, "GSM 2G SIM Card from Ting & Adafruit - Data/Voice/Text USA Only," Adafruit, [Online]. Available: http://www.adafruit.com/products/2505. [Accessed 7 December 2015].

[37] G. Kohl, "AT&T makes it official: 2G service will end," Security InfoWatch, 6 August 2012. [Online]. Available: http://www.securityinfowatch.com/article/10756141/att-officially-announces-plan-to-end-2g-service-in-filings-with-the-securities-and-exchange-commission. [Accessed 7 December 2015].

[38] SIMPLE Mobile, "Service Area Map," SIMPLE Mobile, [Online]. Available: https://www.simplemobile.com/wps/portal/home/support/coverage/!ut/p/a1/fc5LC4JAFAXg39LC7dzbqDPaTowyoZcF2WxCxUbxMaKW9O8zaBNUZ3fgO3BAQAiiju65jPpc1VH56oJd5uiv9b3j0cDjJq52nntY8g0aRwNOIP4Sl_8H2wV9A_wRB8EHkccVGZKKILHZ1LSoYTNKOePIxo9np451S4Jo02vapi25teP1rO-. [Accessed 7 December 2015].

[39] Best Buy, "Simple Mobile - SIM Card," Best Buy, [Online]. Available: http://www.bestbuy.com/site/simple-mobile-sim-card/7725001.p?id=1219286764319&skuId=7725001. [Accessed 7 December 2015].

[40] T-Mobile, "T-Mobile 3-in-1 SIM Starter Kit," T-Mobile, [Online]. Available: http://www.t-mobile.com/cell-phones/t-mobile-3-in-1-sim-starter-kit.html#reviews. [Accessed 7 December 2015].

[41] Choose a Membership – Support – Apple Developer. [Online]. Available: https://developer.apple.com/support/compare-memberships/

[42] How to use the Google Play Developer Console. [Online]. Available: https://support.google.com/googleplay/android-developer/answer/6112435?hl=en&rd=1

[43] R. Llamas, R. Reith, and K. Nagamine. (2015). IDC: Smartphone OS Market Share 2015, 2014, 2013, and 2012. [Online]. Available: http://www.idc.com/prodserv/smartphone-os-market-share.jsp

[44] N. Diakopoulos and S. Cass. (2015, July 20). Interactive: The Top Programming Languages of 2015. [Online]. Available: http://spectrum.ieee.org/static/interactive-the-top-programming-languages-2015#index/2015/0/1/0/0/1/50/1/50/1/50/1/30/1/30/1/30/1/20/1/20/1/5/1/5/1/20/1/100/

[45] Android Developers: Adding Maps. [Online]. Available: https://developer.android.com/training/maps/index.html#start

[46] Android Developers: Bluetooth. [Online]. Available: http://developer.android.com/guide/topics/connectivity/bluetooth.html

[47] Android Developers: Location Strategies. [Online]. Available: http://developer.android.com/guide/topics/location/strategies.html

[48] Android Developers: Notifications. [Online]. Available: http://developer.android.com/guide/topics/ui/notifiers/notifications.html

[49] Xcode – Downloads – Apple Developer. [Online]. Available: https://developer.apple.com/xcode/download/

[50] Android Developers: Getting Started. [Online]. Available: https://developer.android.com/training/index.html

[51] The Java Tutorials. [Online]. Available: https://docs.oracle.com/javase/tutorial/

[52] Texas Instruments, MSP430FR5969 16 MHz Ultra-Low-Power Microcontroller featuring 64 KB FRAM, 2 KB SRAM, 40 IO, Dallas: Texas Instruments, 2012.

[53] Texas Instruments, MSP430FG47x, Dallas: Texas Instruments, 2008.

[54] Atmel Corporation, ATMEL 8-BIT MICROCONTROLLER WITH 4/8/16/32KBYTES IN-SYSTEM PROGRAMMABLE FLASH, San Jose: Atmel Corporation, 2015.

[55] Texas Instruments, CC3200 SimpleLink Wi-Fi and Internet-of-Things Solution, a Single-Chip Wireless MCU (Rev. F), Dallas: Texas Instruments, 2015.

[56] Texas Instruments, bq2423x USB-Friendly Lithium-Ion Battery Charger And Power-Path Management IC (Rev. I), Dallas: Texas Instruments, 2015.

[57] Microchip Technology Inc., Miniature Single-Cell, Fully Integrated Li-Ion, Li-Polymer Charge Management Controllers, Chandler: Microchip Technology Inc., 2014.

[58] Texas Instruments, Li-Ion AND Li-Pol BATTERY GAS GAUGE IC FOR PORTABLE APPLICATIONS, Dallas: Texas Instruments, 2007.

[59] Texas Instruments, bq27542-G1 Single Cell Li-Ion Battery Fuel Gauge for Battery Pack Integration, Dallas: Texas Instruments, 2015.

[60] Texas Instruments, bq28z610 Impedance Track™ Gas Gauge and Protection Solution for 1-Series to 2-Series Cell Li-Ion Battery Packs (Rev. B), Dallas: Texas Instruments, 2015.

[61] Texas Instruments, LM3480 100-mA, SOT-23, Quasi Low-Dropout Linear Voltage Regulator (Rev. H), Dallas: Texas Instruments, 2015.

[62] Texas Instruments, TPS74701 500-mA Low-Dropout Linear Regulator With Programmable Soft-Start (Rev. G), Dallas: Texas Instruments, 2015.

[63] Texas Instruments, TPS61088 10-A Fully-Integrated Synchronous Boost Converter (Rev. A), Dallas: Texas Instruments, 2015.

[64] Texas Instruments, TPS61093-Q1 Low Input Boost Converter With Integrated Power Diode and Input/Output Isolation, Dallas: Texas Instruments, 2015.

[65] K. C. a. M. DeHann, "UART (Universal Asynchronous Receive/Transitter)," WhatIs, February 2011. [Online]. Available: http://whatis.techtarget.com/definition/UART-Universal-Asynchronous-Receiver-Transmitter. [Accessed 7 December 2015].

[66] National Instruments, "What is Serial Synchronous Interface (SSI)?," National Instruments, [Online]. Available: http://digital.ni.com/public.nsf/allkb/862567530005F09C862566BE004E469D . [Accessed 7 December 2015].

[67] Texas Instruments, "TM4C123GE6PM High performance 32-bit ARM Cortex-M4F based MCU," Texas Instruments, [Online]. Available: http://www.ti.com/product/tm4c123ge6pm. [Accessed 7 December 2015].

[68] Texas Instruments, "TM4C1294NCPDT IoT enabled High performance 32-bit ARM Cortex-M4F based MCU," Texas Instruments, [Online]. Available: http://www.ti.com/product/tm4c1294ncpdt. [Accessed 7 December 2015].

[69] Real Time Automation, Inc., "IEEE 1588 Unplugged - An introduction to IEEE 1588," Real Time Automation, Inc., [Online]. Available: http://www.rtaautomation.com/technologies/ieee-1588/. [Accessed 7 December 2015].

[70] SiteSell Inc., "How WiFi Works to Transmit From Your Wireless Enabled Devices," SiteSell Inc., 2009. [Online]. Available: http://www.wireless-home-network-made-easy.com/how-does-wifi-work.html. [Accessed 7 December 2015].

[71] Diffen, "TCP vs. UDP," Diffen, [Online]. Available: http://www.diffen.com/difference/TCP_vs_UDP. [Accessed 7 December 2015].

[72] Apple Inc., "Use messages wit your iPhone, iPad, or iPod touch," Apple Inc., 16 September 2015. [Online]. Available: https://support.apple.com/en-us/HT201287. [Accessed 7 December 2015].

[73] M. Arrambide, "Using BBM With WIFI," Wandering Trader, [Online]. Available: http://wanderingtrader.com/travel-tips/using-bbm-with-wifi/. [Accessed 7 December 2015].

[74] M. Brewis, "How to install WhatsApp on iPad or Android tablet: use WhatsApp on a Wi-Fi-only tablet," TechAdvisor, 7 July 2015. [Online]. Available: http://www.pcadvisor.co.uk/how-to/google-android/how-install-whatsapp-on-tablet-no-3g-ipad-app-3517280/. [Accessed 7 December 2015].

[75] Facebook, "Does the Messenger app cost anything?," Facebook, 2014. [Online]. Available: https://www.facebook.com/help/148968815185634. [Accessed 7 December 2015].

[76] P. Nickinson, "What you need to know about the new Google Hangouts, Hangouts Dialer, and the old Google Voice," Android Central, 10 September 2014. [Online]. Available: http://www.androidcentral.com/what-you-need-know-about-new-google-hangouts-and-google-voice. [Accessed 7 December 2015].

[77] Viber, "Viber for IOS," Viber, [Online]. Available: https://www.viber.com/en/products/iphone. [Accessed 7 December 2015].

[78] Texas Instruments, "CC3000," Texas Instruments, [Online]. Available: http://www.ti.com/product/cc3000. [Accessed 7 December 2015].

[79] Texas Instruments, "CC3100," Texas Instruments, [Online]. Available: http://www.ti.com/product/cc3100/description. [Accessed 7 December 2015].

[80] Texas Instruments, "CC3200MOD," Texas Instruments, [Online]. Available: http://www.ti.com/product/cc3200MOD. [Accessed 7 December 2015].

[81] The TCP/IP Guide, "TCP/IP Architecture and the TCP/IP Model," The TCP/IP Guide, 20 September 2005. [Online]. Available: http://www.tcpipguide.com/free/t_TCPIPArchitectureandtheTCPIPModel-2.htm. [Accessed 7 December 2015].

[82] How-To Geek, "How to Boost Your Wi-Fi Network Signal and Increase Range with DD-WRT," How-To Geek, [Online]. Available: http://www.howtogeek.com/56810/how-to-extend-your-wireless-network-with-dd-wrt/. [Accessed 7 December 2015].

[83] Digi, "Receiver Sensitivity," Digi, [Online]. Available: http://www.digi.com/technology/rf-articles/receiver-sensitivity. [Accessed 7 December 2015].

[84] Sending & Receiving SMS messages via WiFi. [Online]. Available: http://www.androidauthority.com/qa/question/sending-receiving-sms-messages-via-wifi/

[85] Optimum Design Associates, "Through-Hole vs. Surface Mount," Optimum Design Associates, [Online]. Available: http://blog.optimumdesign.com/through-hole-vs-surface-mount. [Accessed 7 December 2015].

[86] Engineers Garage, "Difference between through hole and Surface mount components," Engineers Garage, [Online]. Available: http://www.engineersgarage.com/contribution/anjali/difference-between-through-hole-and-surface-mount-components. [Accessed 7 December 2015].

[87] Standard, Joint Industry, Space Applications Electronic Hardware Addendum to IPC J-STD-001E Requirements for Soldered Electrical and Electronic Assemblies, Bannockburn: IPC, 2010.

[88] Department of Defense, "Global Positioning System Standard Positioning Service Performance Standard," no. 4, p. 160, September 2008.

[89] M. Rouse, "Backward Compatible (backward compatibility) definition," WhatIs, [Online]. Available: http://searchenterpriselinux.techtarget.com/definition/backward-compatible . [Accessed 7 December 2015].

[90] T. V. W. B. J. Marshall Brain, "How WiFi Works," HowStuffWorks, [Online]. Available: http://computer.howstuffworks.com/wireless-network1.htm. [Accessed 7 December 2015].

[91] N. Pidgeon, "How Ethernet Works," HowStuffWorks, [Online]. Available: http://computer.howstuffworks.com/ethernet.htm. [Accessed 7 December 2015].

[92] ETSI, "SIM," ETSI, [Online]. Available: http://www.etsi.org/technologies-clusters/technologies/smart-cards/sim. [Accessed 7 December 2015].
[93] ETSI, "General Packet Radio Service, GPRS," ETSI, [Online]. Available: http://www.etsi.org/index.php/technologies-clusters/technologies/mobile/gprs. [Accessed 7 December 2015].
[94] CCM, "The GSM Standard," CCM, [Online]. Available: http://ccm.net/contents/694-the-gsm-standard. [Accessed 7 December 2015].
[95] ETSI, "Mobile Technologies GSM," ETSI, [Online]. Available: http://www.etsi.org/index.php/technologies-clusters/technologies/mobile/gsm. [Accessed 7 December 2015].
[96] 3D Printer – ASME@UCF. [Online]. Available: http://www.asmeatucf.com/3d-printer.html

Appendix B: Photo Permission

Venus638PLx Datasheet Photo Permission

IS Info - SkyTraq <info@skytraq.com.tw>
To: alexis.timms;
Tue 12/8/2015 4:25 PM

No problem.

Best Regards,

Oliver Huang
SkyTraq Technology, Inc.
4F, No.26, Minsiang Street, Hsinchu City 300, Taiwan
TEL: +886 3 5678650 ext 400
FAX: +886 3 5678680
Website: www.skytraq.com.tw

AL alexis.timms
To: info@skytraq.com.tw;
Reply all
Tue 12/8/2015 3:37 PM

Sent Items

To whom it may concern:

I am a student at the University of Central Florida and I am using the Venus638PLx in my senior design project. I am required to write a paper and would like to ask permission to use images from the datasheet in my term paper. Please let me know if this would be acceptable at your earliest convenience.

Figure 36 GPS module photo permission

4. Copyright and Use of Material

1 - We recognise that when you obtain access to this website, your computer downloads a copy of the materials on it. Unless stated otherwise on, or in respect of, any given page of this website, you are hereby permitted to so download and display the content of this website and print, or store in electronic form (but not on a server or other network enabled device), individual pages from it for your personal use only, and not for any commercial purpose whatsoever. You are responsible for acting in compliance at all times with all applicable copyright laws.

Figure 37 Laird Technology's Copyright Policy

No problem

Question: Hi my name is Sarah and I am a student at the University of Central Florida. I am just looking to get permission to use a photo that is in the IEEE 1588 Protocol Overview article, (http://www.rtaautomation.com/technologies/ieee-1588/). It is for my senior design paper and is for educational purposes only. Please let me know as soon as possible. Thank you.

Let me know if you have any other questions or concerns.

All the best,

Drew Baryenbruch
Real Time Automation, Inc.
1-262-439-4999
1-262-439-4961 (direct)
150 S. Sunny Slope Rd.
Suite 130
Brookfield, Wi 53005

Figure 38 Real Time Automation Permission

I am a student and would like to use one of your articles in my research project. Can I have your permission to do so?

We are happy to grant permission to use our material for noncommercial, nonprofit, educational purposes. We ask that you not alter the images or content from the way they appear on our site, and that you cite HowStuffWorks as your source.

Figure 39 How Stuff Works Citation Information

Use of Site. National Instruments ("NI") authorizes you to view and download the materials at this Web site ("Site") only for your personal, non-commercial use in accordance with the terms and conditions set forth herein, including without limitation the following restrictions: (a) you must retain all copyright and other proprietary notices contained in the original materials on any copies of the materials; (b) you may not modify the materials at this Site in any way or reproduce or publicly display, perform, or distribute or otherwise use them for any public or commercial purpose; and (c) you may not use any of these materials on any other Web site or networked computer environment for any purpose. Unauthorized use of any materials at this Site may violate copyright, trademark, and other laws. If you breach any of these Terms, your authorization to use this Site automatically terminates and you must immediately destroy any downloaded or printed materials. Any rights not expressly granted herein are reserved to NI.

Figure 40 National Instruments Citation Information

4. Use Restrictions and Termination of Access to TI Services

TI Services are protected by copyright laws, international copyright treaties, and other intellectual property laws and treaties. Except as stated herein, no TI Service, nor any part of any TI Service, may be reproduced, duplicated, mirrored, modified, displayed, distributed, copied, sold, resold, visited, or otherwise exploited for any purpose without express prior written consent of TI.

You agree not to use TI Services in a manner that violates any applicable law or regulation; to stalk, harass, or harm another individual; to impersonate any person or entity or otherwise misrepresent your affiliation with a person or entity; to interfere with or disrupt TI Services or servers or networks connected to TI Services; use any data mining, robots, or similar data gathering or extraction methods in connection with TI Services; frame or utilize framing techniques to enclose any trademark, logo, proprietary, or other information (including datasheets, images, text, page layout, or form); and attempt to gain unauthorized access to any portion of TI Services or any other accounts, computer systems, or networks connected to TI Services, whether through hacking, password mining, or any other means.

Subject to any Service Terms that may apply, TI grants you permission to download, reproduce, display, and distribute TI Services solely for non-commercial or personal use, provided that you do not modify such TI Services, and provided further that you retain all copyright and proprietary notices as they appear in such TI Services.

TI further grants to K-12 educational institutions, universities, and community colleges permission to download, reproduce, display, and distribute TI Services solely for use in the classroom, provided that such institutions identify TI as the source of TI Services and include the following credit line: "Courtesy of Texas Instruments." Unauthorized use of any TI Service is expressly prohibited by law, and may result in civil and criminal penalties. This grant of permission terminates if you breach any provision in these Terms of Use or Service Terms. Upon termination, you agree to destroy any materials relating to TI Services.

TI reserves the right, in its sole discretion, to terminate, suspend, or modify your registration with, or access to, all or any part of TI Services, without notice, at any time and for any reason.

Figure 41 Texas Instruments documentation on Copyrighted material.
http://www.ti.com/corp/docs/legal/copyright.shtml

Appendix C: Datasheets

Component	Part Name	Datasheet Links
GPS	Venus638FLPx-L	http://cdn.sparkfun.com/datasheets/Sensors/GPS/Venus638FLPx.pdf
Bluetooth	BT800	http://www.lairdtech.com/products/bt800
MCU	CC3200	http://www.ti.com/lit/ds/symlink/cc3200.pdf
Fuel Gauge	BQ27010	http://www.ti.com.cn/cn/lit/ds/symlink/bq27010.pdf
LDO Regulator	TPS74701	http://www.ti.com/lit/ds/symlink/tps74701-q1.pdf
Battery Charging IC	BQ24232	http://www.ti.com/lit/ds/symlink/bq24232.pdf
Battery	3.7V 2000mAh Li-Po Battery	https://www.sparkfun.com/datasheets/Batteries/UnionBattery-2000mAh.pdf
Display	OLED 128 x 64 px Display	https://www.adafruit.com/datasheets/UG-2864HSWEG01.pdf
GSM	SM5100B	https://www.sparkfun.com/datasheets/CellularShield/SM5100B%20AT%20Command%20Set.pdf
Wi-Fi Module	CC3100	http://www.ti.com/lit/ds/symlink/cc3100.pdf
SIM Card	2G SIM Card from Ting & Adafruit	https://www.adafruit.com/product/2505
MCU	TM4C1294NCPDT	http://www.ti.com/lit/ds/symlink/tm4c1294ncpdt.pdf