HAULLELUJAH! A SOLUTION TO PACKING A U-HAUL!

ECE 445 Design Document

TJ Lambert, Nathaniel Stoll, and Zach Wong
Group 4
TA: Stephanie Jaster
February 27th, 2020

1 Introduction

1.1 Objective

People are constantly moving. Whether it is a short journey from home to school or a long journey from vacationing overseas to flying home, people are always on the move. More specifically, people are always finding a new place to call home during their lifetime. This home could be temporary like a college dorm or permanent like a house to start a family. In America alone, roughly 35.5 million (about 13%) Americans move each year at least once [1]. Reasons like moving/relocating into a new and better home as well as people trying to establish their own household are just the top two reasons in a long list on why people chose to move [1]. These statistics do not even include people moving in and out of the United States as well as people moving worldwide. According to a UN study, in the year 2017 alone, there were a total of 238 million persons migrating from country to country [2]. As you can see, people are always on the move. One of the most difficult tasks in moving is finding a safe and efficient way to store and transport your belongings. In America, people moving their items by renting out moving trucks is the second most popular moving technique right behind people using their own personal vehicle to carry their belongings [1]. Companies like U-Haul offer consumers a variety of different sized trucks to rent to transfer your belongings. However, how do consumers know which truck size is ideal for them given their size and number of belongings? Far too many times people will go for the cheapest size to lower costs or tell themselves they do not have a lot of belongings. Many times, people will end up stuffing their items into the truck or squeeze things into it to ensure everything fits without realizing they could be damaging other items in the process. None of these options work well when you get to your destination and have to deal with the aftermath of having to move out all the cluttered items squeezed into the truck. Wouldn't it be nice if there was a solution that not only tells the consumer what size truck they need, but also gives them step-by-step instruction on what and where their belongings need to be placed in order to efficiently maximize the space?

Our goal for our project is to effectively and efficiently help make the process of moving one’s belongings into and out of the transport vehicle easier. To make this process easier, our project aims to answer the previously mentioned questions of finding a solution that tells consumers exactly what size moving truck they need as well as how they should load their belongings to efficiently maximize the space. To accomplish this feat, it is simply done with a digital tape measure and a mobile application. The tape measure will be used to record and transmit the data containing the dimensions and sizes of the boxes holding the consumer’s belongings. Using bluetooth connectivity, the data from the tape measure will be transferred to our mobile application that will not only calculate the ideal size moving truck needed to hold all the belongings, but also give visual instructions to the user on where and how the boxes should be placed into the truck to efficiently maximize the truck space. Additionally, the mobile application would provide the user a nice interface that allows the user to keep track of which items are in each box as well as if the box contains fragile items or not.

1.2 Background

There are a few solutions that exist in the market today that help minimize and solve the problem proposed above in making the process of packing in items in a moving truck more efficient and easier. These solutions include a digital tape measure, other mobile applications, and professional moving companies. In many popular hardware stores like Menards, Lowes, and Home Depot, consumers can find digital tape measures. As you may have guessed, digital tape measures give consumers the ability to receive digital measurements from the items they are measuring. However, many consumers do not know or have the time to figure out what to do with the received dimensions or how it can be applied in the context of moving. In addition, there are other mobile applications existing in app markets today, but many of them revolve around giving users cosmetics that will make the entire process of moving easier. Specifically, apps like Wunderlist and Sortly give users the ability to make lists and take inventory of what items go in each box, but neither tells the users where they should pack the boxes in the moving truck efficiently nor give ideal rental size truck suggestions [3]. Additionally, other moving labeled apps like TaskRabbit and MagicPlan aim to assist users to decorate and manage their new home upon arrival. Neither of these apps help in the moving process at all [3]. Finally, there are professional moving companies on the market as well. However, this option is both expensive and causes liabilities. According to a report, moving companies charge by the hour for each worker with each hour charge ranging from $25 to $60 an hour [4]. This cost is for one worker alone and without a moving truck too [4]. The total average cost of moving would easily be over $1000 with all things considered. Prices also rise based on the distance needed to travel to get to the destination [4]. As you can see, hiring movers becomes an expensive option fairly quickly. Additionally, some companies charge consumers extra when handling fragile items [4]. This statement insinuates that if consumers did not choose this option, workers may not properly handle a person’s belongings with care which can cause liabilities for the consumer.

1.3 Visual Aid

Figure 1 shows a high-level interaction of our project. The user will measure a box’s three dimensions using the digital tape measure, and the bluetooth module in the tape measure will send these measurements to a connected mobile device through bluetooth. The mobile device will receive these inputs and display them clearly to the user through the application.

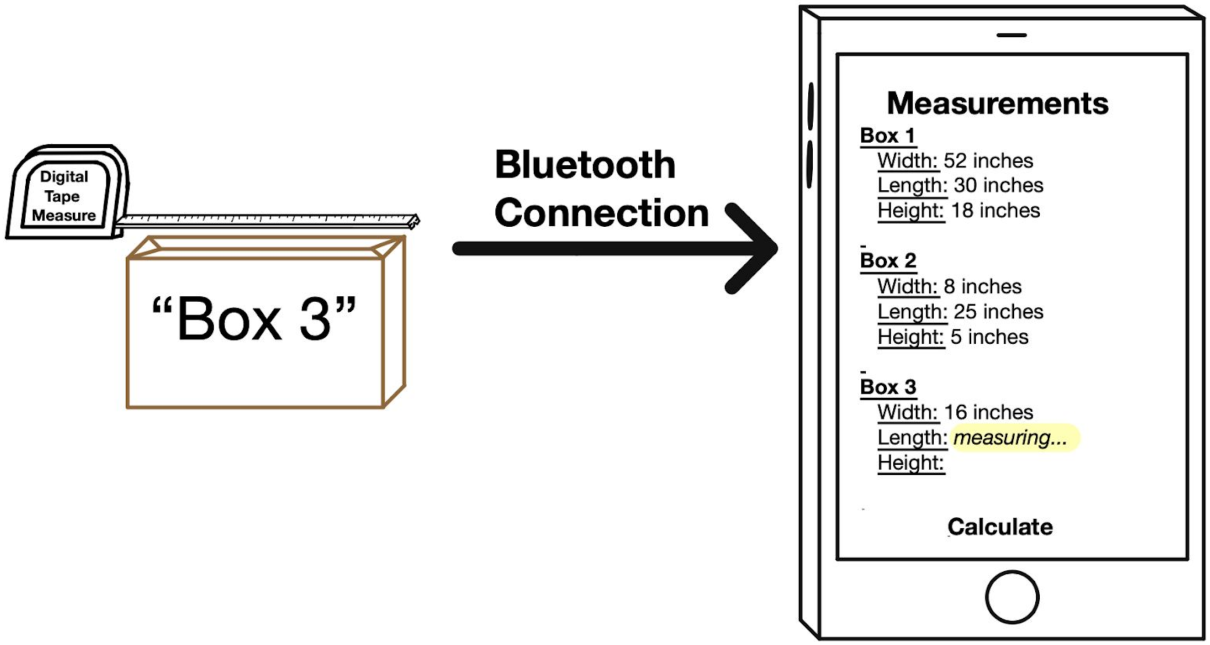

Figure 1. Sending Measurement Visual Aid

1.4 High-Level Requirements List

· Mobile application algorithm must accurately and effectively determine the size of moving truck needed for the user based on incoming user measurement data in under 2 minutes.

· Mobile application must display a visual interface detailing how users should load their belongings into the moving truck to efficiently maximize space through step-by-step instructions.

· Bluetooth module attached to digital tape measure (equipped with at least one button, two status lights, and a power ON/OFF mechanism) must accurately and effectively transmit the digital tape measure measurement data to the mobile application in a timely manner (seconds).

2 Design

2.1 Functional Overview

For our design to be successfully operational in satisfying our project problem, it needs to be comprised of four subsystems: control unit, pre-existing tape measure, power supply, and mobile application. Each subsystem is a vital component to the overall design and when combined together builds a product we believe will solve our problem statement. The power supply contains the necessary items to adequately power our microcontroller, which will in turn power the bluetooth module to transfer the measurement data to our mobile application. The control unit subsystem will house our microcontroller, bluetooth module, and a few cosmetic items like buttons and LEDs. The goal of this subsystem is to receive user inputs from the buttons as well as receive measurement data from the pre-existing digital tape measure. Finally, this subsystem will output the measurements received via bluetooth to the mobile application. The pre-existing tape measure subsystem contains an already existing digital tape measure on the market today, which we will use to “hack” into its digital display to pipeline the measurement data into our control unit. The mobile application subsystem is comprised of a mobile app that will use the phone’s bluetooth connection to receive the measurement data and perform an algorithm that will correctly determine the size of truck need for the user. This subsystem will also provide the user with a visual interface of where they should place their belongings as well.

Figure 2 shows a block diagram containing the physical interactions of the many parts of our project.

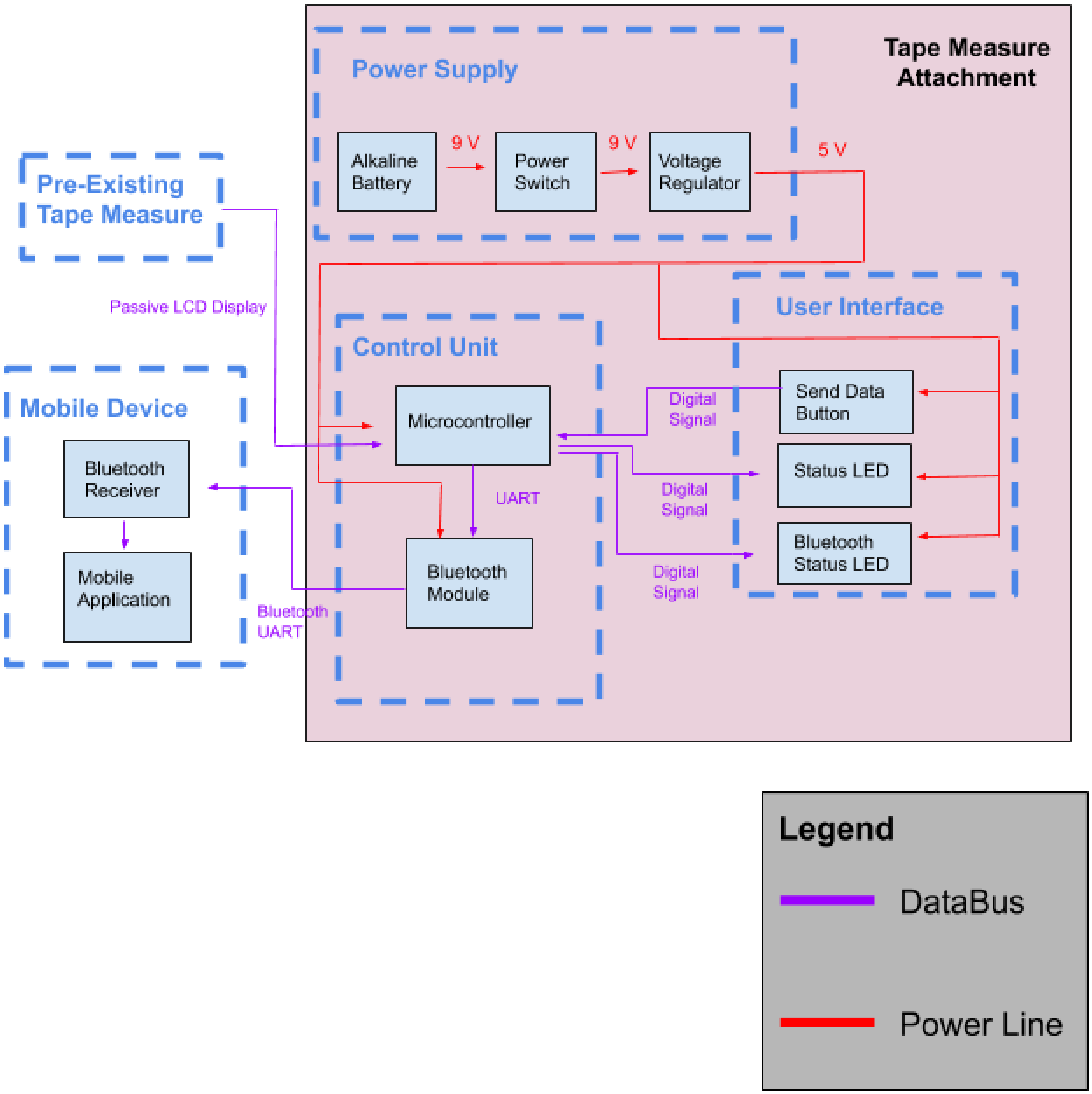

Figure 2. Block Diagram

2.2 Physical Design

Since we need to add electronics (LEDs, a button, a switch, a battery, and a PCB with electrical components) to our tape measure in order to obtain data from it, care should be taken to make sure the electronics do not physically get in the way while the user is working with it. Therefore, ideally, it is best if the form factor of the additions made to the tape measure are kept as small as possible. Additionally, the button we add should be placed in a spot convenient to the user. Accordingly, we decided to put it where one's index finger usually sits on the tape measure. Figure 3 shows a drawing of how we have chosen to map out these components.

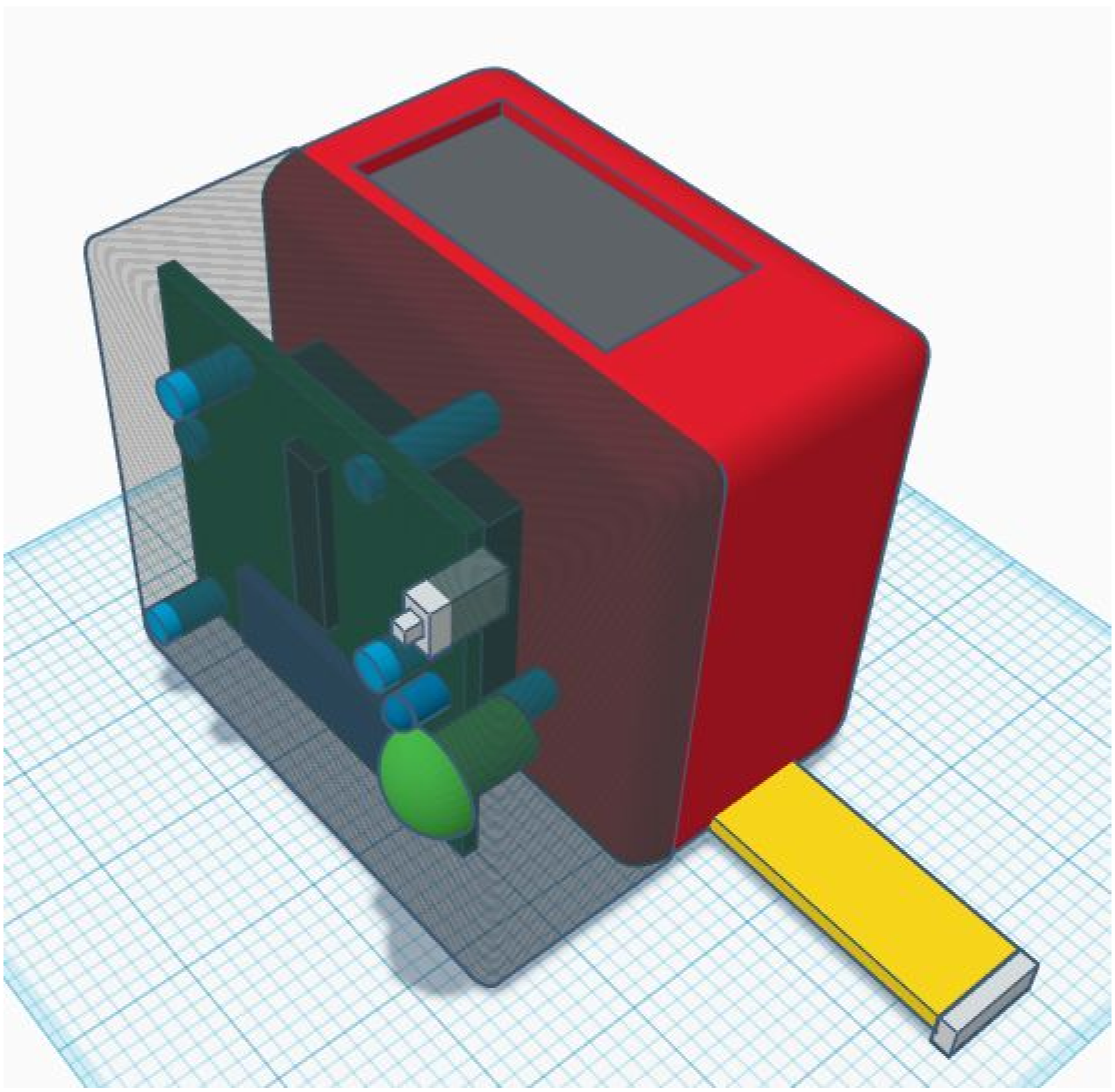

Figure 3. Physical Design

2.3 Power Supply

There are various components being added to the tape measure which require power to operate. These include the microcontroller, the bluetooth module, lights, and buttons. With this being said, it is necessary for us to design a power supply which will power these things. The power supply will be made up of the following: alkaline batteries, power switch, and voltage regulator.

2.3.1 Alkaline Batteries

Ultimately, everything powered in the control unit will be powered by batteries. The batteries we pick need to have a high enough voltage to keep everything powered, and they need to have enough capacity to keep the digital tape measure operating for an extended amount of time. Battery voltage will be applied to a voltage regulator, so it is okay for the total voltage coming from the batteries to exceed the voltage needed for the microcontroller/bluetooth module.

Requirement	**Verification**
9V battery must be able to supply between 8.7-9.3 V and 70-100 mA to the voltage regulator.	A. Using an oscilloscope, set-up a testing environment for the Power Supply subsystem B. Flip the power switch to the ON position. C. Monitor the voltage on the oscilloscope at the input of the voltage regulator. D. Ensure voltage received is between 8.7 V-9.3 V.
Batteries must be able to last more than 1 hour of normal use by the control unit subsystem.	A. Switch the power switch to the ON position B. Ensure Control Unit subsystem is currently functioning by verifying the Status LED is lit up of solid color. C. Begin timer (either on smartphone or stopwatch) D. Begin measuring items and sending data over bluetooth E. When the timer is at 60 minutes (1 hour), stop it and ensure LEDs on microcontroller are still lit up and subsystem is still functioning.

2.3.2 Power Switch

To keep the batteries lasting as long as possible, a small switch (likely, a slide switch) will be used to enable the user to turn on and off power to the voltage regulator, and thus, the control unit.

Requirement	Verification
Must successfully disconnect power from voltage regulator when switched OFF and successfully connect power to the voltage regulator (<0.1V drop) when switched ON.	A. Within the Power Supply subsystem, set-up a testing environment using the oscilloscope. B. Probe the power switch. Initially, set the switch in the OFF position. C. Ensure no voltage is flowing out of the power switch. D. Flip the power switch to the ON position. E. Monitor on the oscilloscope that voltage is being produced from the power switch (<0.1V drop). F. Flip the power switch back to the OFF position. G. Monitor on the oscilloscope that the power switch successfully reduce voltage to the voltage regulator

2.3.3 Voltage Regulator

Sometimes, it is difficult for batteries to maintain a certain voltage over an extended period of time. If certain components require steady voltage levels, it isn't ideal to rely on batteries to accomplish this feat. Additionally, it isn't always possible to pick out battery combinations which meet certain voltage requirements (for example, AA batteries are rated at 1.5V; trying to produce a voltage level which isn't a multiple of 1.5V becomes difficult). Therefore, the concept of voltage regulators becomes useful. In our project, we will be powering a microcontroller and a bluetooth module which will likely require specific voltage levels which are different than the batteries we pick. By using voltage regulators, we can supply the correct voltages to these components even though our batteries may not exactly match these same voltages.

Requirement	Verification
Voltage regulator must be able to supply between 4.5-5.5 V input to the microcontroller and bluetooth module given the incoming 9 Vinput voltage from the 9V battery.	A. Within the Power Supply subsystem, probe the voltage regulator and microcontroller line. B. Specifically, using an oscilloscope measure the voltage output coming the voltage regulator to the microcontroller and ensure it is between 4.5-5.5 Volts.

2.4 Control Unit

The control unit will use an Arduino BT which uses an ATmega328P Microcontroller to receive values from our pre-existing Tape Measure's Passive LCD Display, receive inputs from the user through buttons, display states to the user through LEDs, as well as send data to the mobile device through bluetooth UART.

2.4.1 Microcontroller

We plan on using an ATmega328P Microcontroller. The operating voltage for the microcontroller ranges from 1.8 - 5.5 Volts. We plan on operating it at ~5 Volts for our project design [8]. It has 32kB of flash memory which should be more than enough to store our code [8]. The ATmega328P also contains 14 digital pins and 6 analog pins for us to use for our LEDs and Buttons as well as translating the Passive LCD Display from the Tape Measure [8].

Requirement	Verification
Must be able to accurately reproduce measurement data from the tape measure's display	A. Use the pre-existing tape measure to record measurement data by measuring items B. Record these measurements being display on the tape measure's display on paper C. Press Send Data button D. On mobile device, check to see if the measurements displayed on the notification interface are the same as the ones record on paper

Must be able to transmit measurement data via bluetooth to a paired device.	A. Press the Pair button within the Control Unit subsystem B. On a mobile device, using bluetooth connectivity, pair with the bluetooth module C. Gather measurement data using the pre-existing tape measure D. Press the Send Data button E. Ensure the data was received on the mobile application by viewing the measurement data on the notification interface on the mobile application

2.4.2 Bluetooth Module

We will be using the HM-10 bluetooth module. This bluetooth module will be responsible for receiving data from the ATmega328P Microcontroller and sending it to the Mobile Device. It operates at 2.5 - 3.3V and requires 50mA [9]. The breakout board that it comes on contains a voltage regulator that can accept 3.6 - 6V to bring it down to the usable 2.5 - 3.3V [9]. The HM-10 bluetooth module uses bluetooth 4 BLE [9].

Requirement	Verification
Bluetooth module must be able to send measurements to the smartphone accurately with a transfer speed of at least 2 Mbps.	A. Design and implement a test program using the Arduino software that will print out bluetooth transfer rate to serial monitor B. Run the test program C. Ensure the transfer rate from the bluetooth module is of at least 2 Mbps

2.4.3 Send Data Button

This button will be responsible for grabbing the current measurement from the pre-existing tape measure, and when pressed will signal to the bluetooth module to send measurements to the smartphone.

Requirement	Verification
When pressed, the send data button must signal the bluetooth module to send data, and the bluetooth module must not send data otherwise.	A. Press Send Data button. Ensure it does not get stuck. B. On a mobile device, check to see if notification interface appears asking the user to fill in details in regards to that specific box with the sent over data dimensions. C. Do not press data button D. On mobile device, ensure no notification interface does not appear to user

2.4.4 Status LED

This LED will be lit while the device is powered on and not lit while the device is turned off.

Requirement	Verification
Must be lit while the device is on and not lit while the device is off. Must be visible from 1 meter away	A. Measure out 1 meter away from LED B. Ensure device is off and ensure LED is not on C. Turn device on D. Ensure LED is of solid color and visible

2.4.5 Bluetooth Status LED

This LED will be solid lit while the bluetooth module is connected to a mobile device, not lit while the bluetooth module is not connected to a mobile device and not looking for a mobile device, and blinking while the bluetooth module is waiting to be paired to a device (after we press the pair button, but before we connect it to a mobile device).

Requirement	Verification
Must be lit with a solid color while bluetooth module is connected, blinking while	A. Do not pair bluetooth module with mobile device. B. Measure out 1 meter away from the

bluetooth module is searching for a mobile device, and not lit otherwise. Must be visible from 1 meter away.	LED C. Ensure LED is blinking visibly D. Pair bluetooth module with mobile device E. Ensure LED is of solid color and visible

2.5 Mobile Device

The control unit will send data through bluetooth UART to the mobile device. This device will receive values, store them in a mobile application, input them into an algorithm, and display them in a visual for the user to know where to store their items and how much space will be required.

2.5.1 Bluetooth Receiver

The user will have an existing smartphone that will be able to pair to our bluetooth module in our control unit.

Requirement	Verification
Bluetooth receiver on user's smartphone must be able to connect to bluetooth module via bluetooth connectivity	A. On a mobile device, open bluetooth connection settings interface B. Find the corresponding bluetooth module device name on the connections list C. On the mobile device, tap to connect/pair with the bluetooth module D. Ensure connection is made through notifications on the smartphone's interface.

2.5.2 Mobile Application

Our mobile application will receive values from the bluetooth receiver, store them in the app, run an algorithm that looks for the best places to store the boxes in the truck, tell the user what size truck is required, and show a visual of where to store the items in the truck.

Requirement	Verification
The algorithm needs to decide where items must be placed and take less than 2 minutes to compute.	A. On mobile device, ensure all data measurement is sent by pressing the Send Data button B. Run the algorithm C. Use a stopwatch and start it D. Stop the stopwatch when the notification interface displays on the mobile device stating algorithm is complete E. Ensure stopwatch time is less than 2 minutes
The algorithm must be able to report the smallest possible size truck needed for the user to store their items.	A. On the mobile device, notification should appear that the algorithm is complete. B. Users should then be able to see results of the algorithm C. View result D. Look on the packing truck website (e.g. U-Haul) to ensure the result received was not the biggest truck size.
The algorithm must display one solution for the most efficient way to pack the truck.	A. On the mobile device, users should receive notification stating the algorithm is complete. B. Upon completion of the algorithm, mobile application should display to the user a list of efficient packing solutions C. Users will then have the option to view all possible solutions and choose accordingly the best solution they see fit

2.6 Pre-existing Tape Measure

Our application needs objects' dimensions in order to calculate the best way to pack items in a confined space. These dimensions, ultimately, will be entered in by the user of our application. The quicker the user can figure out dimensions of the things he/she is moving, the more useful our application will be to him/her. Our solution to minimize the amount of time for this process is to use a pre-existing digital tape measure, route the measurement readings from it, and send this data over bluetooth to our application.

Requirement	Verification
Tape measure must have accessible and usable data signals which can be used by our control unit.	A. Deconstruct pre-existing tape measure B. View existing electronics of tape measure, specifically datalines that route to display on tape measure C. Check these signals to see if we are able to reroute them to our bluetooth module
The number of data signals used by the tape measure should be less than or equal to the number of inputs available on our microcontroller.	A. Deconstruct pre-existing tape measure B. Check if existing data signals on tape measure are less than number of inputs on microcontroller by counting the datalines

2.7 Schematics

Perhaps one of the more challenging parts of this project is the fact that there are a number of circuit elements which must be connected together in a small area. In an effort to show how these elements are connected to one another, the schematics corresponding to each module can be found below.

2.7.1 Power Module Schematic

Figure 4 below outlines the Power Module Schematic containing the 9V battery, the slide switch, and the 5V linear voltage regulator. The battery is used as the power source for all of the added electronics, the slide switch is used to turn on and off power to the added electronics, and the voltage regulator steps down the voltage level from 9V to 5V.

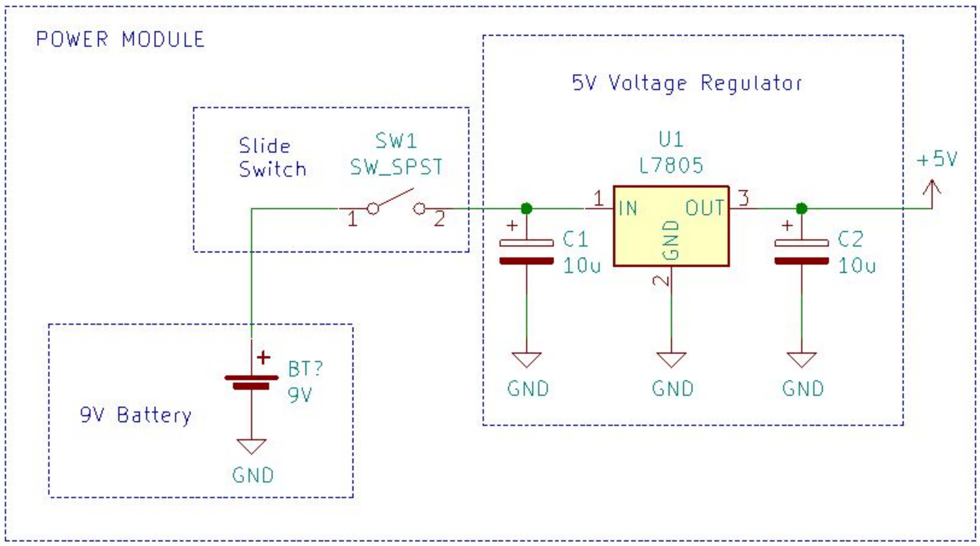

Figure 4. Power Module Schematic

2.7.2 Control Unit Schematic

Figure 5 below outlines the Control Unit Schematic containing the microcontroller, the bluetooth module, and the data signals coming from the tape measure. The microcontroller serves as a middle-man between the data signals and the bluetooth module. More specifically, it interprets the data coming from the tape measure, and uses the bluetooth module as a means to communicate with the paired bluetooth device.

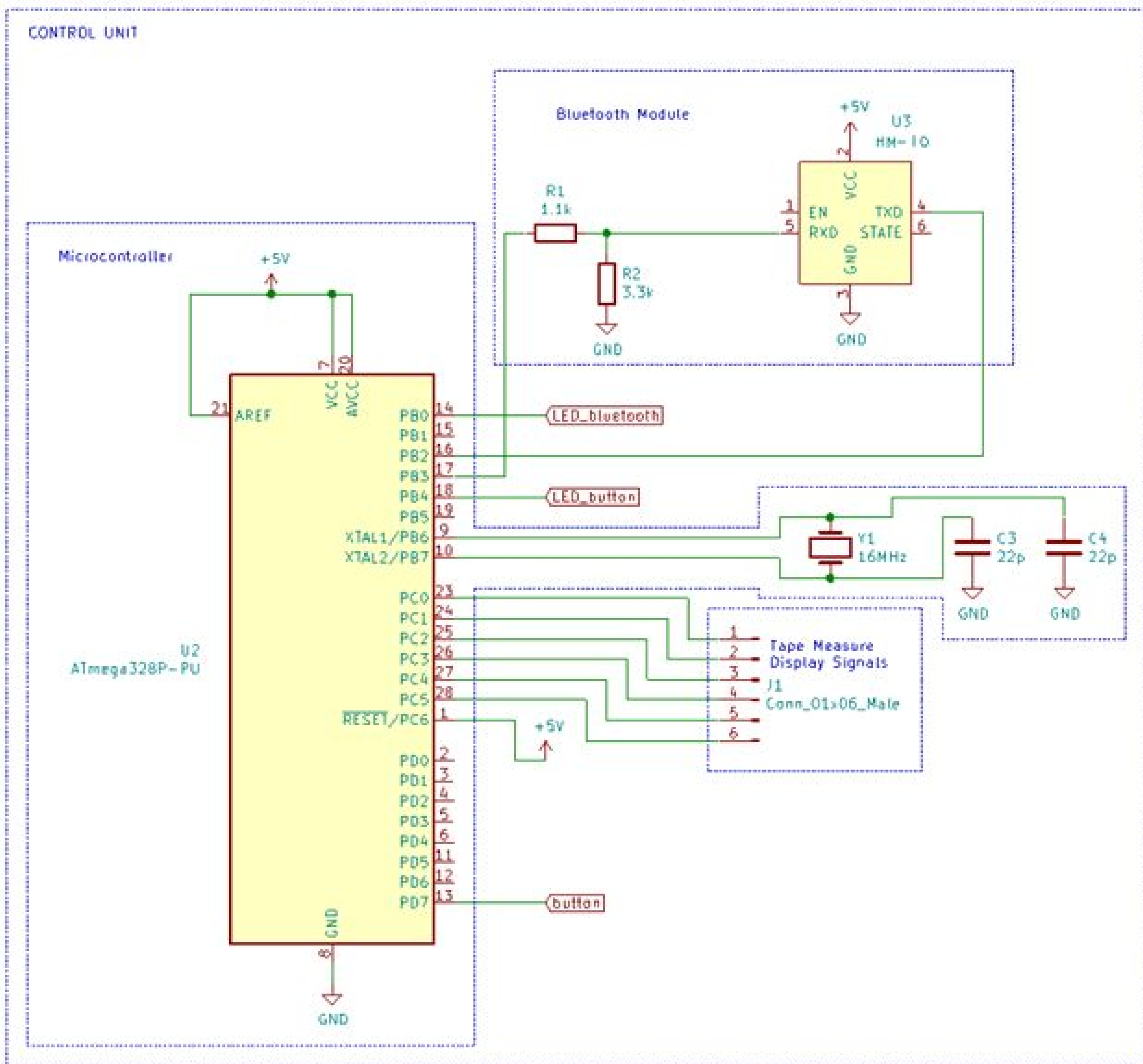

Figure 5. Control Unit Schematic

2.7.3 User Interface Schematic

Figure 6 below outlines the User Interface Schematic containing the LEDs and the button. The LEDs are used to provide visual information to the user regarding ON/OFF status and bluetooth connection status. The button will be used to signal to the microcontroller when to send data to the paired mobile device. Note, all of these components are connected to digital pins of the ATMEGA328P.

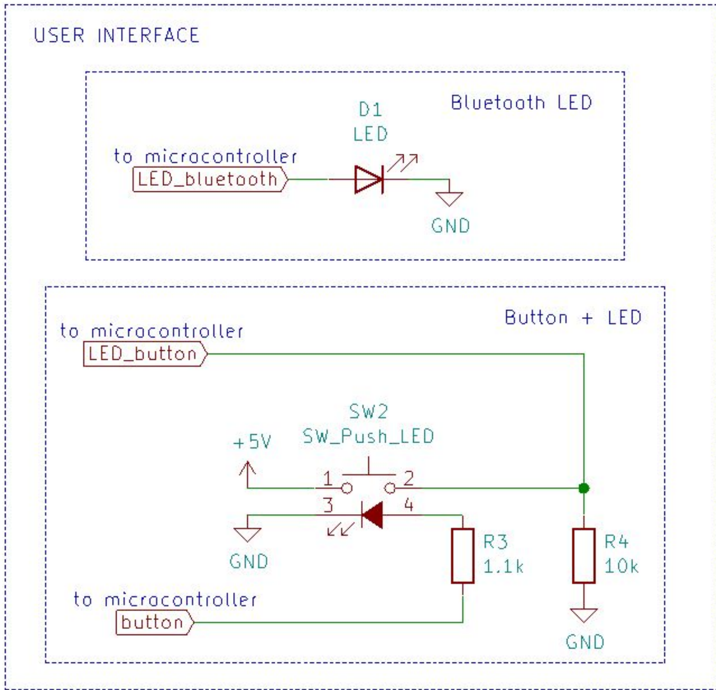

Figure 6. User Interface Schematic

2.8 Software

The tape measure interacts with the mobile application through the bluetooth module in our control unit. A high-level overview of this interaction is displayed in figure 4.

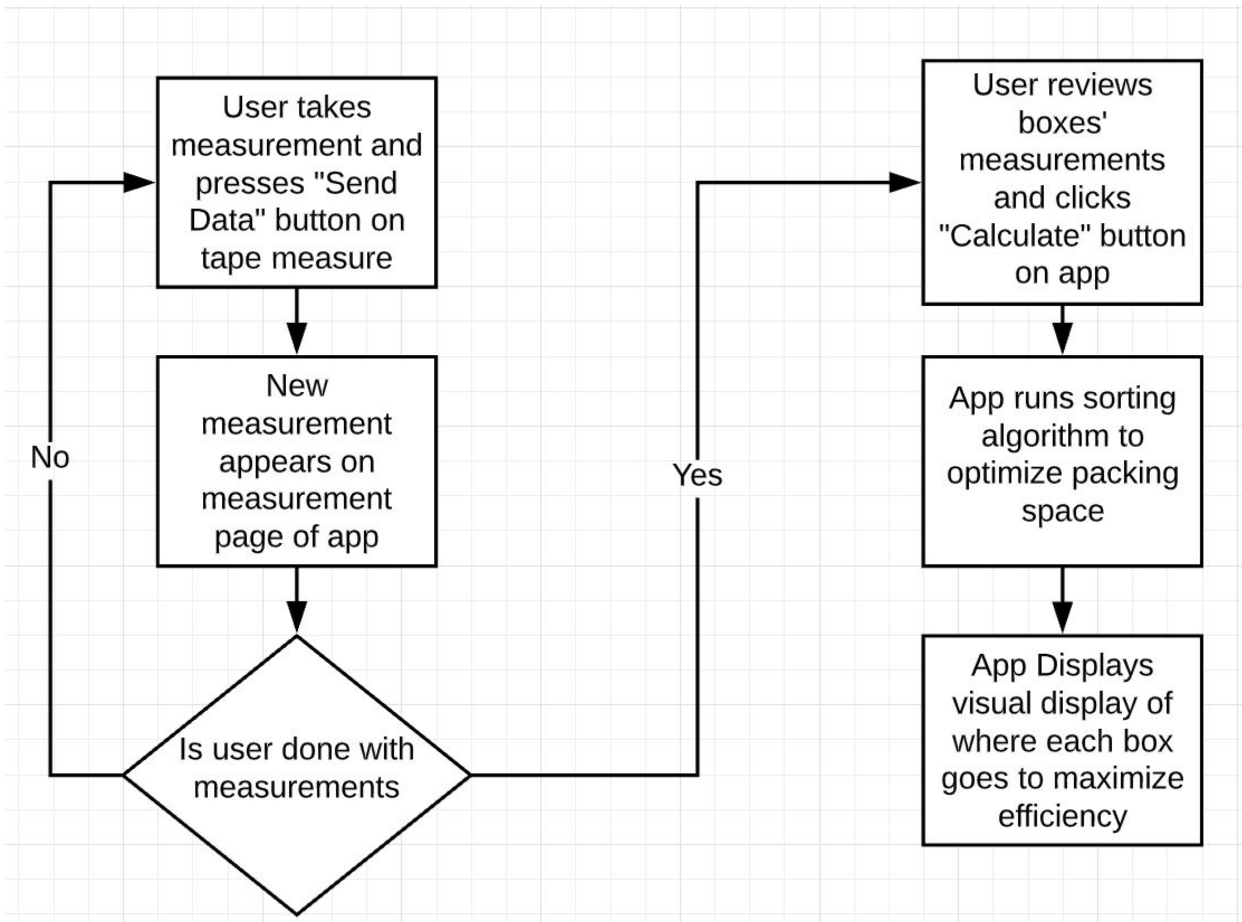

Figure 7. High-level Tape Measure and App Interaction

2.9 Fault Tolerance

Perhaps one of the biggest obstacles which will be faced in this project is decoding the measurement signals which are sent to the display of the tape measure. There are multiple reasons why this could be a challenge. First, since we have not yet had the opportunity to look inside the tape measure we plan to use (the eTape16 [13]), there is a lot of uncertainty with what electronics we will find inside. With this being said, though, we *have* had the opportunity to open up a different tape measure which has a digital display, giving us a rough idea of what we can likely expect to find inside the one we hope to use. For reference, pictures of this tape measure can be found in Appendix A (Figures 8-10). In the tape measure we were able to disassemble, we found the display is connected to the main PCB via a 14 pin ribbon cable of which only 8 pins appear to be used. It can be assumed that two of these pins are likely used to power the display leaving 6 pins to be used for communication purposes. Looking at Equation 1, there aren't many numbers which can be represented using only 6 bits:

$$Largest\ Decimal\ Representation\ =\ 2^{\#\ bits} = 2^6 = 64 \qquad \text{Eq. 1}$$

Therefore, it can be assumed parallel communication is not the primary method with which the on-board microcontroller talks to the display, but rather, serial communication must be incorporated somehow. This leads into the second potential problem: figuring out which type of serial communication is used. There are many common serial communication protocols which exist. These include RS-232, SPI, I^2C, and UART, to name a few [14]. There is no guarantee, however, that any of these are used by the digital tape measure. Therefore, we will likely need to probe the individual pins of the ribbon cable to figure out how each signal impacts the output on the display. More specifically, we will be looking at the signals' voltage levels, the apparent baud rate (determined by Equation 2), start/stop bit patterns, and framing patterns/lengths to try to characterize which protocol might be used between the tape measure's microcontroller and it's display.

$$Baud\ Rate = \frac{\#\ symbols}{1\ second} = \frac{1}{period},\ assuming\ (1\ bit\ per\ symbol) \qquad \text{Eq. 2}$$

One final concern to note here is the physical interaction between the tape measure's display signals and the ATMEGA328P's inputs. There is a good chance the voltage the tape measure uses for communication is not 5V logic like the inputs to the ATMEGA328P. One possible workaround for this would be to use the analog inputs on the ATMEGA328P instead of the digital ones and map the voltage range to something more desirable. The issue with this, however, is performing an analogRead() command (from the Arduino Analog I/O library [15]) takes a fairly significant amount of time. According to [15], it takes about 100 microseconds to call analogRead() once. Seeing as there may be more than one signal which would need to be analyzed, the total amount of time it would likely take to analyze X signals is given in Equation 3:

$$Total\ Read\ Time = \frac{100\ \mu s}{1\ signal} \times (\#\ signals) \qquad \text{Eq. 3}$$

Taking the inverse of this result produces the maximum baud rate (again, assuming 1 bit per symbol) which the ATMEGA328P could theoretically interpret, given in Equation 4:

$$Max\ Baud\ Rate = \frac{1}{Total\ Read\ Time} = \frac{1\ signal}{100\ \mu s \times (\#\ signals)} \qquad \text{Eq. 4}$$

In the event this sampling frequency isn't quick enough, it may be best to try amplifying the signals coming from the tape measure so they become compatible with the ATMEGA328P's digital inputs which are considerably faster than the analog inputs. This, however, would likely require the use of a level shifter. Level shifters have limits on how quickly they can be switched. Generally speaking, though, the frequency with which digital inputs can scan the signals—even with the switching limits of the level shifters—still would provide better performance (in terms of sampling rates) than simply using the analog pins. Therefore, we should be able to read signals coming from the tape measure for different voltage levels and/or standard baud rates.

With all this being said, even in the worst case scenario where we would be unable to read data from the tape measure, this would not necessarily break our project. Ultimately, we need to get measurement data somehow, and using a digital tape measure is one of many ways to do this. We could, instead, use cameras, encoders, and/or other distance sensors to perform the same task in a different way. Ultimately, we believe the tape measure would be the best option for the consumer, but there are other options out there which we could use as substitutes in our project in the event we are unable to get the tape measure to work.

3 Costs

Our development costs come from three people working 10 hours a week at $50 an hour per person. Taking this value for the 16 weeks that we are working on this project gives us $60,000 from Equation 5.

$$3 * \$50/hour * 10\ hours/week * 16\ weeks * 2.5 = \$60,000 \qquad \text{Eq. 5}$$

The cost of our prototype will be $81.39. If we purchase our parts in bulk (by the 100 parts) our total will be $60.48. The breakdown of these costs are listed below.

Part	Cost (Prototype)	Cost (Bulk - 100 parts)
Microcontroller (ATMEGA328P-PU - Digikey)	$2.08	$1.73
28 Pin Socket (ED281DT - Digikey)	$0.33	$0.22
5V Voltage Regulator (UA7805CKCS - Digikey)	$0.79	$0.49
10µF Capacitor (x2) (P828-ND - Digikey)	$0.48	$0.17
10k Resistor (10KQBK-ND -	$0.10	$0.03

Digikey)		
220 Resistor (x2) (220GBK-ND - Digikey)	$0.20	$0.05
1.1k Resistor (1.1KQBK-ND - Digikey)	$0.10	$0.03
3.3k Resistor (3.3KQBK-ND - Digikey)	$0.10	$0.03
22pF Capacitor (x2) (399-1926-ND - Digikey)	$0.88	$0.36
16MHz Clock Crystal (300-6034-ND - Digikey)	$0.54	$0.36
Pushbutton with Green LED (450-2758-ND - Digikey)	$6.10	$5.13
Blue LED (C503B-BCN-CV0Z0461-ND - Digikey)	$0.21	$0.16
Slide Switch (CKN10666-ND - Digikey)	$5.82	$4.74
9V Battery (Amazon Basics)	$3.49	$0.92
Bluetooth Module (HM-10 - amazon.com)	$9.99	$7.13
Digital Tape Measure (eTape16 ET16.75-db-RP - amazon.com)	$29.99	$29.99
PCB (PCBWay)	$5	$1.16
Casing (3D-print estimate)	$3	$3
Other Resistors, Transistors, and General Electrical Components (Digikey estimate)	$10	$0.50
Total Cost	**$81.39**	**$60.48**

4 Schedule

Week	Nathaniel	TJ	Zach
Feb. 24, 2020	Talk with machine shop about recommendations for physical design. Submit order for tape measure and initial components.	Research Packing Problem Algorithms & plan the cross platform app on Visual Studio	Research 3D visual interfaces and structures to embed for mobile application development.
Mar. 2, 2020	Once tape measure is received, begin figuring out how signals to the display work.	Begin connecting the app to bluetooth and displaying to the user if the phone is connected to the bluetooth module.	Begin construction and design of 3D model algorithm based on given results from sorting algorithm
Mar. 9, 2020	Finalize plan for collecting signals from the display, order extra parts as needed. Submit proposal/order for PCB.	Work on the page where the measurement values will be added and allow the user to input values manually and start getting measurements from bluetooth module.	Program bluetooth module to send incoming data from the microcontroller/tape measure combination to our mobile application when Send Data button is pressed. Ensure all data is being sent over from module to mobile app.
Mar. 16, 2020	Try to incorporate microcontroller to read data from display.	Continue getting measurements from tape measure and save these values in Box class while parsing and making sure the inputs are doubles. Also start algorithm.	Begin adding cosmetic features to the mobile application like labeling, inventory, and classification of items. Users can label their boxes and specify if they are light, heavy, fragile,

			etc. They can also take inventory of what items are in each box
Mar. 23, 2020	Solder components onto PCB.	Continue to work on the algorithm that can place the boxes in 3D array	Create notification interfaces for users when the algorithm is complete, so they can view possible solutions. Additionally create a home UI for mobile application.
Mar. 30, 2020	Design exterior casing for tape measure.	Work on the box display page and create 3D image	Begin testing of the sorting algorithm and 3D model algorithm with the bluetooth module
Apr. 6, 2020	Assemble tape measure and make modifications as needed.	Continue working on the display page	Update and refine algorithm based off of testing results and analysis
Apr. 13, 2020	Design and Catch up week	Design and Catch up week	Design and Catch up week
Apr. 20, 2020	Design and Catch up week	Design and Catch up week	Design and Catch up week
Apr. 27, 2020	Demo	Demo	Demo
May 4, 2020	Presentation	Presentation	Presentation

5 Safety and Ethics

There are a few safety and ethical concerns that reside with our project. One of the main safety concerns comes from the lithium-ion batteries used by our digital tape measure. These batteries can explode if they become overcharged or reach an extreme heat temperature [5]. Additionally, these batteries should not be charged in extreme cold temperatures either as they can deteriorate and leak chemical acid [5]. To ensure safety, we will adequately test and monitor the battery cell temperature to ensure its overall quality and performance following the guidelines laid out in this battery lab safety guide [10].

Another safety concern stems from the enclosed bluetooth module attached onto the digital tape measure. Since the tape measure can be used in both indoor and outdoor environments, the bluetooth module should be encased in material in such a way no moisture or water can short the circuit. For our specification, the casing material will adhere to IP67 standards ensuring it will withstand up to 1 meter of water following the guidelines of the Ingress Protection standard [11].

The bluetooth module itself also raises a safety concern due to the exposure of wires that the module carries. Exposed wires causes issues for both the user and the device itself. For the device, the exposed wires can be damaged and/or short-circuited through a variety of means including human induced. However, the bigger safety issue comes from the user end as exposed wires can cause harm to the user using the device. Electrical shocks and burns are just some of the damage that users can receive due to the interaction from exposed wires. To address this safety issue, we will first ensure the correct amount of voltage and current is flowing through each electrical component in our bluetooth module by probing each part and verifying it with our requirements. Next, we will then insulate the wires to prevent any of them from being exposed to the user. We plan on insulating the wires using electrical or thermal insulating tape on our wires. To ensure the tape has a definitive seal on the wires, we will test and make sure it does not conduct any electrical as well as heat up. We will test by following the student manual for electrical safety [12].

An ethical concern our project raises involves the containment of private user data. Within our mobile application, users will be able to store their own private data that is kept hidden from other users. This issue raises the concern of possible data leaks or piracy that can take place through malicious software attacks focused on app or phone data. These issues go against the IEEE Code of Ethics #9 - to avoid injuring others, their property, reputation, or employment by false or malicious action [6]. To mitigate this issue, we will have a notification appear to users recommending popular security applications to install if they are not installed already.

Since the mobile application calculates the needed amount of space to hold one's belongings, our project raises concern of taking advantage of the given measurement data and suggesting the biggest size truck possible. This concern violates the IEEE Code of Ethics #3 - to be honest and realistic in stating claims or estimates based on available data [6]. To avoid violation, we want to construct and develop an algorithm that does not just estimate the space from the given user measurement data, but one that calculates the entire dataset and concludes a total amount of space needed for the user. Additionally, this means that the data received from the pre-existing tape measure must be accurate and precise when sent over bluetooth connection.

We must ensure that the measurement data is accurate as can be in order for our algorithm to calculate the ideal and optimal solution for the user. The algorithm, given the accurate and precise measurement data, would then perform over the entire dataset and recommend to the user what size truck they should get given the user measured measurement data.

Finally, to avoid violation of IEEE Code of Ethics #1 - hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or the environment [6]; we accept responsibility for our design. Our project aims to ensure the safety of its users and while we hope to test all possible scenarios there are a billion more that may occur. Therefore, we accept responsibility for faults in our design and ensure we take appropriate action to ensure user safety when these faults occur.

References:

[1] C. Holmes, “2018 Stats and Facts of the American Mover,” *Move.org*, 13-May-2019. [Online]. Available: https://www.move.org/moving-stats-facts/. [Accessed: 07-Feb-2020].

[2] “Population Facts,” *United Nations Population Division*, vol. 5, no. 2017, pp. 1–5, Dec-2017.

[3] S. Dwyer, “Your Move Simplified: The 9 Best Apps for Moving,” *RISMedia*, 12-Jul-2017. [Online]. Available: https://rismedia.com/2017/07/12/your-move-simplified-9-best-apps-moving/. [Accessed: 07-Feb-2020].

[4] J. Green, “How Much Does It Cost To Hire Movers - Complete Guide,” *My Moving Reviews*, 15-Nov-2019. [Online]. Available: https://www.mymovingreviews.com/move/how-much-does-it-cost-to-hire-movers/. [Accessed: 07-Feb-2020].

[5] batteryuniversity.com,“Lithium-ion Safety Concerns,” *Lithium-ion Safety Concerns – Battery University*. [Online]. Available: https://batteryuniversity.com/learn/archive/lithium_ion_safety_concerns. [Accessed: 07-Feb-2020].

[6] Ieee.org, "IEEE IEEE Code of Ethics", 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 03- Feb- 2020].

[7] A. Harris, “Best Digital Tape Measures 2020 - Reviews & Top Picks,” *HealthyHandyman*, 05-Jan-2020. [Online]. Available: https://healthyhandyman.com/best-digital-tape-measures/. [Accessed: 07-Feb-2020].

[8] “Arduino BT (Bluetooth),” *Arduino*. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoardBT?from=Main.ArduinoBoardBluetooth. [Accessed: 10-Feb-2020].

[9] M. Currey, “HM-10 Bluetooth 4 BLE Modules,” Martyn Currey, 05-Jan-2017. [Online]. Available: http://www.martyncurrey.com/hm-10-bluetooth-4ble-modules/. [Accessed: 02-Mar-2020].

[10] “Division of Research Safety,” Illinois. [Online]. Available: https://drs.illinois.edu/Page/SafetyLibrary/BatterySafety. [Accessed: 23-Feb-2020].

[11] “Ingress Protection & IP Testing: NTS,” National Technical Systems. [Online]. Available: https://www.nts.com/services/testing/environmental/ingress-protection/. [Accessed: 23-Feb-2020].

[12] “Electrical Safety: Safety & Health for Electrical Trades (Student Manual),” elcosh. [Online]. Available: http://www.elcosh.org/document/1624/891/d000543/section5.html. [Accessed: 23-Feb-2020].

[13] *eTape16*. [Online]. Available: https://etape16.com/shop/etape16-digital-tape-measure/. [Accessed: 20-Feb-2020].

[14] J. Patrick, “Serial Protocols Compared,” *embedded.com*, 31-May-2002. [Online]. Available: https://www.embedded.com/serial-protocols-compared/ [Accessed: 27-Feb-2020].

[15] “analogRead(),” *Arduino*, 7-Sep-2019. [Online]. Available: https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/ [Accessed: 27-Feb-2020].

Appendix A: Digital Tape Measure Example

Figure 8. Digital Laser Tape Measure

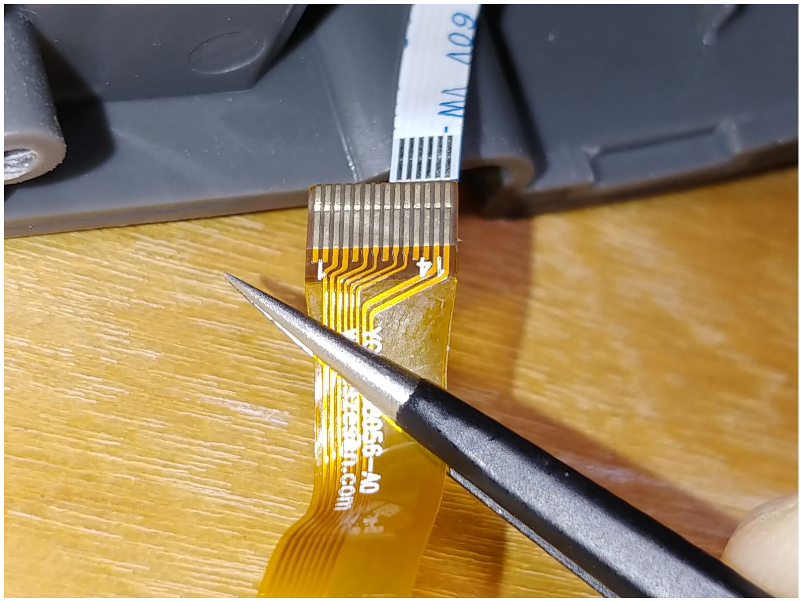

Figure 9. Ribbon Cable Going to Display

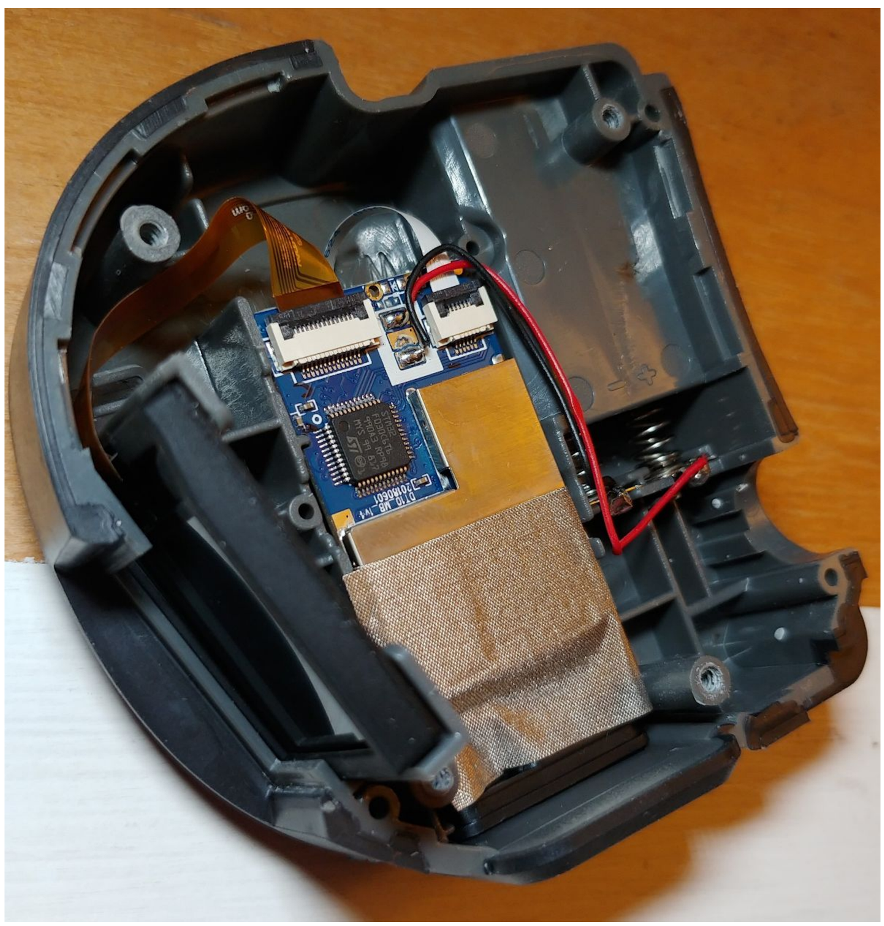

Figure 10. Electronics Inside Tape Measure