Solar Panel Monitor By MARS

Computer Engineering Technology

Final revision April 14, 2020

Written by: Ramin Kurkeice, Ahmad El-Hajj, Matthew Phillip

Declaration of Joint Authorship

We, Ramin Kurkeice, Ahmad El-Hajj, and Matthew Phillip, we acknowledge that this work is submitted by the group work of all members and is expressed in our own words and work that is paraphrased accordingly. Any uses made within the work of any other author/ authors. Any uses made within it of the works of any other author, in any form (ideas, equations, figures, texts, tables, programs), are properly acknowledged at the point of use. A list of the references used is included. The work breakdown is as follows: Each of us provided functioning, documented hardware for a sensor or effector. Ramin Kurkeice provided the ZPT101B. Ahmad El-Hajj provided the current sensor. Matthew Phillip provided temp sensor. In the integration effort Matthew is the lead for further development of our mobile application, Ahmad is the lead for the Hardware, and Ramin is the lead for connecting the two via the Database.

Proposal

We have created a mobile application, worked with databases, completed a software engineering course, and prototyped a small embedded system with a custom PCB as well as an enclosure (3D printed/laser cut). Our Internet of Things (IoT) capstone project uses a distributed computing model of a smart phone application, a database accessible via the internet, an enterprise wireless (capable of storing certificates) connected embedded system prototype with a custom PCB as well as an enclosure (3D printed/laser cut), and are documented via this technical report targeting OACETT certification guidelines.

Intended project key component descriptions and part numbers

Development platform:

Sensor/Effector 1: Lumosity sensor

Sensor/Effector 2: current sensor

Sensor/Effector 3: power sensor

We will continue to develop skills to configure operating systems, networks, and embedded systems using these key components to create the solar panel sensor and track info from the device and place it within the database. We plan on making the mobile application work with the intended design of the tracker and pull data from the device when needed. and make the device with the skills learned from the previous semesters such as CENG 251, CENG 254, CENG 322, and CENG 317 to effectively create a downsized prototype of the capstone and to make the necessary decisions and teamwork to create the work.

Our project description/specifications will be reviewed by, Dr. Dragos Paraschiv. They will also ideally attend the ICT Capstone Expo to see the outcome and be eligible to apply for NSERC funded extension projects..

The small physical prototypes that we build are to be small and safe enough to be brought to class every week as well as be worked on at home. In alignment with the space below the tray in the Humber North Campus Electronics Parts kit the overall project maximum dimensions are 12 13/16" x 6" x 2 7/8" = 32.5cm x 15.25cm x 7.25cm.

Keeping safety and Z462 in mind, the highest AC voltage that will be used is 16Vrms from a wall adapter from which +/- 15V or as high as 45 VDC can be obtained. Maximum power consumption will not exceed 20 Watts. We are working with prototypes and that prototypes are not to be left powered unattended despite the connectivity that we develop.

Executive Summary

Business Need/Opportunity

Humber College Institute of Technology and Advanced Learning, Has installed 4 solar panels to the gazebo in the campus although unsure of how much power is being collected the MARSINC team as decided to work with Humber to created a prototype that would connect a database with the device and display reading on to the application.

This would bring interest in the Humber community two have visual proof of power collection and further inspire clean energy from the client and hopefully inspire more companies and businesses to implement similar technology.

What we hope to achieve is a framework to be built upon in the field of clean energy recording and data sharing.

Statement of Work

This effort includes the following:

- Created Printed Circuit Boards (PCB) to house the wired components in as small as possible.
- Create and improve an encasing for the device that protects and improves portability.
- Make a mobile application that is simple, easy to use, and free to access.
- Create a database and store readings from the device to be retrieved from the application.

Contents

List of Figures

1.0 Introduction

Mars is a small organization of students that are taking on this capstone project for Humber college institute of advanced learning and technology on the Solar panel monitoring of the solar panels on the gazebo. We are omitting the website application as we believe that the mobile application is more compact easy to access and can be ported to a website in the future. We are including a Lumosity sensor as well as a current and voltage sensor to calculate power and lux of the solar panel. We also are making a database that collects this information and adds it to the mobile application. We undertook this assignment as a way to help improve our skills and inspire companies to maintain solar panels. The problems we have are Humber is unable to access the database of the solar panels from the original installers and we must make one from scratch we believe this will set back our project but we are confidant in finishing the product.

1.1 Scope and Requirements

The solar monitoring system is an Internet of Things (IoT) capstone project that uses a distributed computing model of a smart phone application, a web application, a database accessible via the internet, an enterprise wireless (capable of storing certificates) connected embedded system prototype with a custom PCB as well as a laser cut enclosure and 3d print, and is documented via an OACETT certification acceptable technical report. The capstone project must be able to fit in the space below the tray in the Humber NorthCampus Electronics Parts kit the overall project maximum dimensions are 12 13/16" x 6" x 2 7/8" = 32.5cm x 15.25cm x 7.25cm. Keeping safety and Z462 in mind, the highestAC voltage that will be used is 16Vrms from a wall adapter from which +/- 15V or as high as 45 VDC can be obtained. Maximum power consumption will not exceed 20 Watts. Android device must have a data visualization activity and action control activity. CSA testing will not be done in this project. Database must collect data at a frequent time (every 30 minutes) and must be accessible through mobile application.

2.0 Background

We would like to thank our sponsor, Dr. Dragos Paraschiv from Humber institute of advance learning and technology for supporting this project. Solar panels monitoring has been a part of the process for solar panel instillation for years. We believe that the system in place for solar panel monitoring is help with companies who are carbon footprint conscience to manage the steady supply of power and Lumosity to properly understand and quality check the solar panels that were installed. According to the Ontario solar installers they state that when a solar panel is installed it isn't ideal to have it running and move on with your day as it is important to have monitoring of the solar panel because it can help with checking of quality of energy from time to time as well is making sure everything is working correctly. (Ontario Solar Installers, 2019). Thus we use monitoring when solar panels are installed to check the proper Lumosity input on a certain day for example a sunny day should yield more power although we check the monitoring software and find that the Lumosity is lower than expected we can then cross examine the sensors and fix the problem before a more expensive solution is reached. We are doing this capstone in response to Humber college institute of advanced learning and technology due to their mission statement that they would like to reduce carbon emission by a considerable amount every year. Humber college installed solar panels to the gazebo in the college grounds over the summer of 2019. Their goal was shaped by their recognition of clean energy and have won awards for their determination according to MediaCorp Canada Inc. They state that Humber college is one of four colleges that won the 2019 greenest employers award as they showcase a number of green features like water bottle stations, bio walls, PV solar panels, etc.

(Yerema & Kristina , 2019). with this information Humber wanting to further green energy we are tasked to monitor these new solar panels. As stated above the importance of monitoring these solar panels is to make sure that clean energy is maintained for as long as possible. This can inspire more businesses to follow green energy as it holds a lot of benefits. RecSolar states that many expanding businesses invest in solar power as they take advantage of a new facility and infostructure this investment can also provide branding and community goodwill with the public for reducing carbon footprints and help with increase your total savings in energy storage. (RecSolar, 2017). These and more are the reason why we plan to take this capstone project and as we see the world is more conscious of green energy and reducing carbon footprints it is also very important that companies are responsible to monitor it as well to ensure quality in power management. These can benefit business in the future on power consumption, energy storage, and reduce cost. Thus our capstone's goal is to help inspire organizations to maintain quality of solar panels.

3.0 Methodology

Our methodology of the Solar panel senor for Humber institute of technology and advanced learning was to use the agile method for our project management we created Gantt charts for both the software and hardware components we in the middle of the work had problems that delayed our progress for example pcb were faulty on the first few attempts and major bugs in our code that took longer to find and fix than expected.

Although using the agile method was the most efficient way of getting this capstone project done.

3.1 Required Resources

Below we discuss the required resources in this capstone project. We will explain the parts components and materials that were used and their specifications. We will later talk about manufacturing where we discuss the PCB (Printed Circuit Board) and the Encasement of the devices. Next, we will mention tools and facilities which were the tools, resources, software's, and facilities we worked with and in with how each one was used for in our project. Later we will explain the shipping duty and taxes that came with the production period of the devices will we used 3rd party manufactures' we explain the cost and taxes and the explanation of possible duty fees that could arise. Finally we talk about time expenditure which will discuss the leading time and work time in our project.

3.1.1 Parts, Components, Materials

This capstone project requires key components to produce such as (1) Raspberry pi 3 model B, (2) ACS 712 current sensor, (3) ZMPT101B voltage sensor, and (4) LM 35 temperature sensor. This project will run on a Raspberry pi 3 model B which is a microprocessor. The Raspberry pi contains a total of 40 pins to operate with. The pins we need are the 3v3 pin, GPIO2 and GPIO3, and the ground pin. The ACS 712 is a current sensor that measures the current in amperes of the following input and exports it in an analog output, its max readings are 5A. Since the we need to be able to read the data, we will need an Analog to Digital Converter (ADC). The type of ADC used in this project is the ADS1115. The ADS1115 contains 4 analog channels to convert, an alert/ready, address pin, an I2C SDA pin, I2C SCL pin, ground pin and a voltage pin that needs 2.0 to 5.5 volts. The ZMPT101B is a voltage sensor that measures the inputs voltage. The supply voltage ranges between 5 to 30 volts and can measure AC voltage within 250 volts. The ZMPT101B has a total of 6 pins. Two for the input which is the Neutral and Phase. It contains a voltage power source and an analog output pin and 2 ground pins to displace the voltage. The output of the produces and analog output which will require an analog to digital converter as well. The previous ADC used will work on a different analog channel pin since it reads a total of 4. The Lm35 is a temperature sensor that will measure the temperature of the sensor in Celsius. The LM35 consists a voltage pin and a ground pin and a Vout pin which outputs the readings in an analog form. PCB sheets are required to create an interrogated PCB with the following sensors. PCB sockets are needed to help connect all sensors and parts to each other under one PCB, making it safer and easier to enclose. The following socket pins needed is a 6pin 3 by 2 for the raspberry pi pins, 10 pin 10x1 for the ads1115 analog to digital converter, two 3 pin 3x1 for the ACS 712 current sensor and the LM 35 temperature sensor, and a 4 pin 4x1 for the ZMPT101B voltage sensor. Soldering wires are necessary to successfully connect pins to the PCB boards. A Velamen Solar Cell is needed to create power so we can read and measure it. It contains a max of 1 volt and 200 milliamperes. We will be switching from the LM35 to the Lumosity sensor of a mini solar panel this is done due to an agreement that temperature will be found through the weather apk in our mobile app instead from an analog to digital reading this will reduce the PCB wiring cost as we already are supplied a Lumosity sensor and reduce the size of the PCB This will also reduce the power consumption of the Raspberry pi 3 b.

3.1.2 Manufacturing

Ramin Kurkeice worked on the PCB (Printed Circuit Board) for the sensor zmpt101b the PCB was designed using a software called fritzing it was made after a breadboard wiring he made and once it was created digitally the computer generates a schematic and a PCB design he then spent most of his time fixing the schematics to be more readable and follow the guidelines. Also there was extra time spent for the redesigning the PCB as there were some shorts and how to incorporate PCB sockets to reduce space. The files used were then printed to have 3 sockets a 10 pin for i2c, 6 pin for the pi, and a 4 pin for the voltage sensor. They were wired with a resistor in mind. Ramin Has also used Inkscape software for SVG editing to create an encapsulation for the pi He would then save the scalable vector image as a pdf for laser cutting the software used for laser cutting was Corel draw that pulled the pdf image and follows it with the laser the case was clear acrylic and was designed with green outline for pieces and red outline for cutting holes into those pieces. There was a total of 6 pieces which all connected like a puzzle. Ahmad used a software called fritzing to create his PCB with the following breadboard image , design schematic and PCB design was rechecked for the proper design found used 3 sockets for the pi i2c and the current sensor. The case was made with open AutoCAD and cura to make the 3d model of the raspberry pi 3 b case. It was later printed and took about 8 hours of inconsecutive prints. And used PLA grey as the printing material. Matthew Phillip worked on the PCB and case design for the DS18B20 temperature sensor, which was simple design because there was not too much material to insert to the PCB board. To create the breadboard, schematic, and PCB design the platform that was used is called fritzing. On the PCB board Matthew Phillip made sure the dimensions of the via were 1.0mm for hole diameter and the ring thickness was 0.5mm. Then Matthew added the DS18B20 temperature sensor with three appropriate vias so the sensor can fit into the PCB board. After that a 4.7k resistor was added with two appropriate via holes so the resistor can fit snuggly in the PCB. Finally, five vias was added in the suitable position and size so the raspberry pi can be inserted into the PCB board. Matthew Phillip designed the case by using a laser cutter in the prototype lab in humber college. The case was based on a design on a web application called Thingiverse with modifications using the platform called CorelDRAW X7. The modifications were made to make sure that the raspberry pi, the PCB, and the temperature sensor can all fit conformable in the case.

3.1.3 Tools and Facilities

During our 4 months of development on the prototype of this capstone we used a series of tools that were instrumental in the project success and development. We started working on both the hardware and software of the project immediately although we did not start working with hardware until a few weeks of planning and recording data in the first month we used Android Studio for the production of our mobile application we used the java programming language and used SQLite for the database as it is an easy to understand and pull from program. We also used Microsoft word for the writing of our proposals and reports, software such as Microsoft excel for our budget, and Gantt chart for the project planning in the agile method where had goals to reach each week in the semester. We later used the software Fritzing for PCB (Printed Circuit Board) planning and developed a connected circuit diagram, a schematic, and a PCB that is to be sent to print. The facilities we used other than our own homes were the Humber labs we had access to the workshops and were able to send our PCB to be printed there. We also used their labs write our code and learn new method to enhance the project. We used two different software for the encapsulation be the prototype members we used both Cura (3D printing design software) and Inkscape (Scalable Vector Image editing software) for the development of a 3D printed case and a laser cut case. Each case came with pros and cons from the software used. We then GitHub Desktop for easy push and pull from our project to be displayed for the public viewing. We also used GitHub's markdown language to write a page for our project progress and a readme to describe the step by step instructions to create the device yourself. We developed in a secure location using Humber Institute of Learning and Advanced Technology internet environment as it requires valid certificate before having access to the Wi-Fi this ensures a protected device during development to avoid any malicious code from being implemented and having less of a security breach through the Bluetooth devices we used. We used the idea lab in Humber to use their 3d printers for our cases, we also used the laser cutter from the Humber workshop to make the other case. We used soldering irons and reverse soldering pump from the Humber labs to solder our PCB's. we used PCB sockets to be solder onto the PCB so the connection to the sensors will not be permanent. We used power supplies and a frequency display to test our sensors we also used a raspberry pi 3 B as the microcontroller of the project. The minor tools used were our breadboard, wires, resistors, they were used for testing our connections and sensors, we used analog to digital converters for the converting of the analog voltage to digital data. For minor software used we used a Weather online apk that was used in our mobile app to display the weather. We also used WhatsApp to communicate with team members and emailed each other for documents needed and such.

3.1.4 Shipping, duty, taxes

The first sensor that was shipped, was the DS18B20, which is a sensor used to measure temperature. Matthew Phillip bought the SunFounder DS18B20 Temperature sensor Module for the raspberry pi and r3 from the online web application amazon and amazon got it from a company called SunFounder CA. The total cost for the product shipping and handling came up to $10.68 and the product took two days to ship. the GST/HST was included with the price. The duty for this product was not applicable as it was from a local retailer.

The second sensor that was shipped, was the ACS 712, and this sensor is used to measure current. Ahmad El-Hajj bought the RoboJax ACS712 Hall Affect 30A Current Sensor for Arduino from the Sayal store and other parts from online web application amazon. The cost of the sensor is $14.50, another product was the mini solar panel which costed $7.95, and the soldering kit 34.98. The total cost for the product shipping and handling came up to $0.00 as amazon has free delivery for orders over $35.00 and the product took one day to ship through amazon prime which came unprotected outside till I had to pick it up also the sensors costed around $5.00 worth of gas as I purchased it from Sayal . the GST/HST also came up to $7.57. The duty for this product came up to $0.00 due to not having any tariffs on the products I ordered.

The third sensor that was shipped, was the ZMPT101B, and this is a sensor used to measure voltage. Ramin Kurkeice bought the Voltage Transformer Module AC Sensor ZMPT101B J1 for the Arduino from the online web application eBay and for a low price of $3.60. I also bought the i2c digital coveter which was $3.45 and for testing I bought male to female wires $11.95 and PCB sockets $9.99 The total cost for the product shipping and handling came up to $20.00 and the product took seven days to ship. The total with GST/HST also came up to $29.07. The duty for this product came up to $0.00 due to no tariffs on these products and coming from Canadian companies.

3.1.5 Time expenditure

The lead time for our parts where not that large as the products were already in stock as we ordered we do acknowledge that the lead time for our PCB prints and 3d prints were large and the total was 8 days and 8 hours of lead time for the product shipped and product created. The work time to complete the whole project should take up to 4 months maximum because that's when the project is due and we have to work on things like parts, components, materials, tools, facilities, and shipping and to collect all the parts, components, and materials should spawn of 30 minutes to 2 hours maximum but the lead time should take up to three days because of shipping. The PCB should be done with a spawn of one to two weeks because we as a group have to make a PCB design that fits all of our components on the board with no problem. We also have to trim any extra wire and solder down the components. The case will have a lead time of one week because we have to first design an appropriate case that can fit all the necessary parts, components, and materials in it. We also have to use a 3D printer or laser cutter to make the case, so the work time will take about one to 3 days. For the Tools the work time and lead time will be very low because we can borrow all the tools we need from the parts crib. The facilities should also have a low lead and work time because we can use most facilities like the parts crib, and the prototype lab for laser cutting, 3D printing, and for tools. The work time will also increase because we have to have team meets discuss about how the project is going and what's the next step.

3.2 Development Platform

The development platform will include an in-depth analyzation of our mobile application, which will discuss the data visualization, action control, and login activity. Next, we will talk about the image and firmware, discussing how the code can be accessed and run, if the hardware is wireless, and code distributed in the repository, finally we will talk about the breadboard / PCB design, pictures, and enclosure.

3.2.1 Mobile Application

The mobile application was made using the program android studio, which at this current stage in our development it is only available on android devices using software marshmallow and above. This means worldwide we have a huge market to reach instead of the other platforms we could have chosen. This app was programmed in the java language. We split this app in 3 parts. The front end, which was the layout of the app, the design of the colors, and the icon shown. The back end, which was the database used, and pulling and pushing records. The functionality, which was the code used to have navigation, code used to alter the front end. This was done with all members of this group participating in all three parts with leads for each part to help with the process to be developed more smoothly. The current status of the app is still requiring a better database as we used firebase to create a temporary database, what we discovered was that what we wanted to use for creating graphed data from records of certain times was much more difficult to create using the JSON format firebase uses. Thus we plan on going to finish the database with SQLite as this program uses the table format for holding records which we find will fit the functionality we are trying to achieve more efficiently. The app works using a login activity which was made using serial number tracking as a traditional login was not necessary for the scope of this project. We decided a serial number was best as if there were multiple devices the user would only want data from there device a not from all devices in the database. This was done by having the main page having a nine-digit password asking for the serial number of the device. (see Figure 2). Once entered there will be a toast alert confirming or denying the serial number. Then when navigating you will see two data displays the readings page (see Figure 3) this will display the current data that was updated in the database, and the history (see figure 3) this will display all the data from the entire day. (this is currently work in progress). You can see an info page (see figure 4) this will have the information describing what lux is and what the readings mean. There is a weather page that will prompt a permission from the user to use the GPS to find the weather in that area. (see figures 5 and 6). The navigation drawer has the following minor pages a blog which explains the app development and testcases, and settings which will change the apps design (see figure 7).

Figure 1-serial number

Figure 2. Serial number.

Status

/1 Hardware present?

/1 Memo by student A + How did you make your Mobile Application? (500 words)

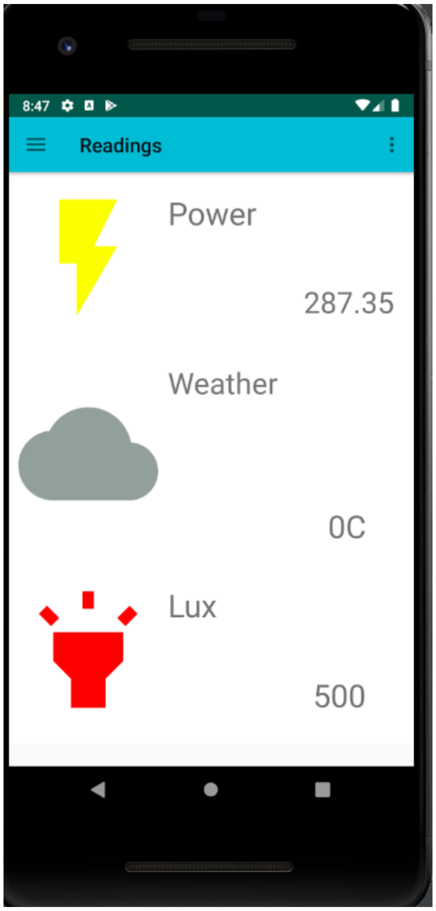
/1 Login activity

/1 Data visualization activity

/1 Action control activity

Include screenshots such as **Error! Reference source not found.**. Testing. Progress.

Figure 3-Readings



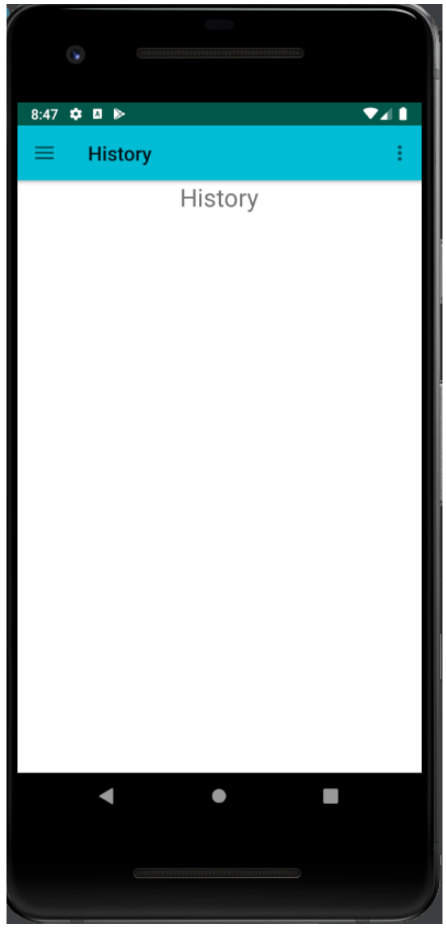


Figure 4-history WIP

Figure 5-info page

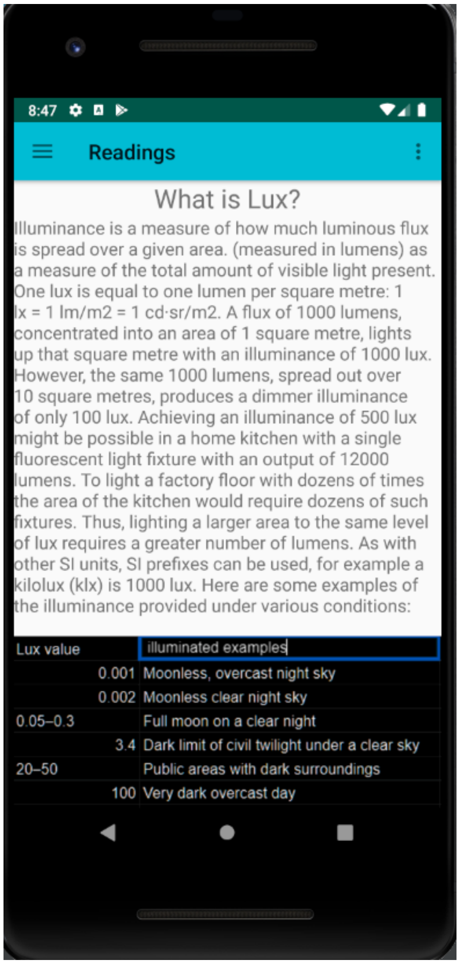


Figure 6-Weather permission

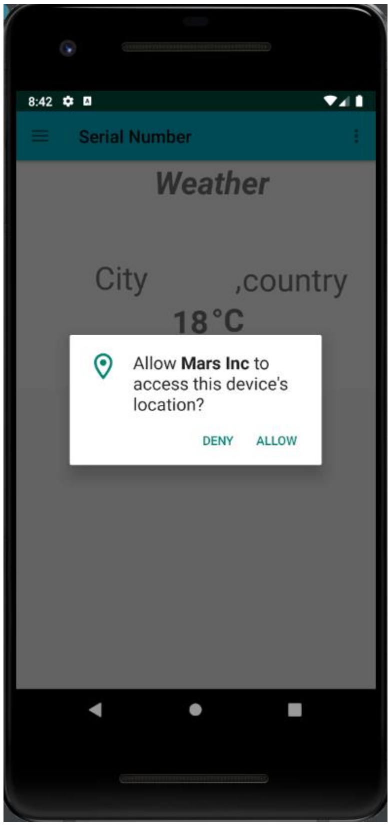
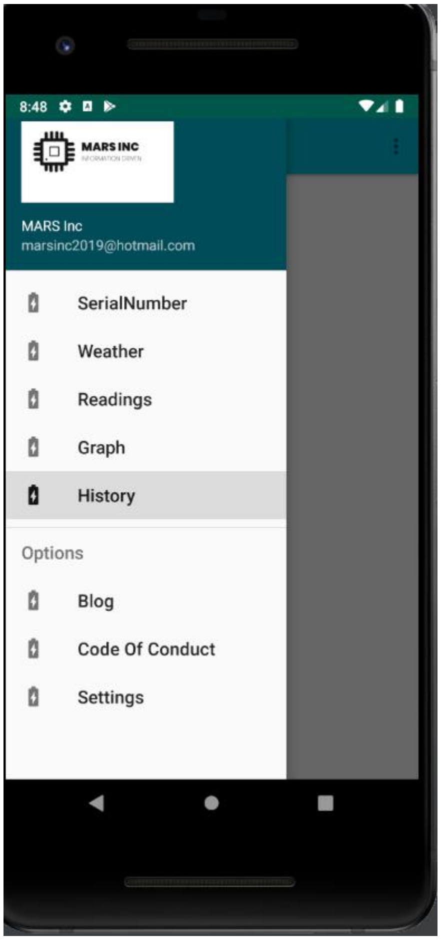


Figure 7- weather

Figure 8-setting and navigation



3.2.2 Image/firmware

The setup for the raspberry pi firmware took numerous steps. Some martials we required to setup the raspberry pi firmware is the raspberry pi, a power cable to power the raspberry pi, a usb computer mouse, a usb keyboard, HDMI cable so we can connect our raspberry pi to a monitor, a micro SD card, and a SD card reader or adapter. The first thing we did was insert our SD card into a SD card adapter and placed it into my laptops SD card slot. After that we opened up the browser, then at the top of the browser we typed in the URL for the raspberry pi website and proceeded to click on NOOBS. To find the link to this website you can go to our GitHub and click on the solar panel gazebo link. After we downloaded the NOOBS zip file and extracted all the files to the micro SD card. Then we inserted our mouse, keyboard, HDMI, and micro SD card into my raspberry pi. After that we booted it up the raspberry pi and clicked on Raspbian to check mark the box. After we clicked on the install button and wait for the installation to be done. After the installation was done and we rebooted the raspberry pi, the firmware and images were up to date.

We used remote desktop to run our code. To setup remote desktop we had to type in a couple commands on the raspberry pi terminal. The first couple of commands we had to type is sudo apt-get update and sudo apt-get upgrade so the raspberry pi will be updated to the latest software. Then we typed in the command sudo apt-get install xrdp and command sudo apt-get install tightvncserver. After we finished typing in these commands, we retrieved the local ip address by typing in the command ifconfig in the terminal. After that we typed in the local ip address into our remote desktop application and clicked on the connect button. Finally, we logged in to raspberry pi and our raspberry pi was connected to the remote desktop application. The raspberry pi 4 has built in WIFI, so we made sure it was enabled and we successfully had wireless connectivity.

Matthew Phillip's code that was store in the repository is for the temperature sensor. This code first creates a function called read_temp_raw that gets the raw readings from the temperature sensor and stores it in a file. After another function called read_temp gets the raw read values from the previous functions and converts into string. The first string makes sure the raw readings is converts to Celsius and the second-string converts makes sure the raw readings is converted to Fahrenheit. Final there is a while loop that prints out the first and second string every one second as long as the temperature sensor produces raw readings. This code was subjected by copyright by a company called Circuit Basics and to find the link to the source code go to our GitHub page.

Ramin Kurkeice's code that's on the repository is for the voltage sensor. The code comes from the learn adafruit website, this is an open source website which provided us code to edit and according to their license on the GitHub they state it is free to distribute the code on our github we will also provide a link to both their webpage to learn how to install the libraries necessary and the original GitHub. The code uses the same i2c conversion code below see (Ahmad's below). The python code is voltage.py and calls the function from the library that is in a loop in the main and will output the readings of the voltage sensor.

Ahmad El-Hajj's code that's on the repository is for the current sensor. The code comes from the learn adafruit website which is an open source website. Used the i2c code that is a function put into the main to read the analog input and convert it to a digital value. The code i used was the ACS712.c and it takes from it's header file and uses the function to read from the 12c readings and prints the values in a loop of the current readings.

3.2.3 Breadboard/Independent PCBs

The capstone project will test all sensors functionality all at once. To do so we will need to test and wire up the sensors to the breadboard. This helps us narrow the result of error before printing the PCB. Based off the distributor of the ACS712 to connect the sensor to the raspberry pi we will need a few a wire, an ads1115 and resistors. Step one was connecting the ads1115. It requires a minimum of 3.3 volts to its power pin. Ground connection should be made to the raspberries pi ground pin. The GPIO2 pin will connect to the SDA port of the ads115. GPIO pin 3 will connect to the SCL pin of the ads115. This helps power and read the reading of the ads1115. To connect the ACS712 we will also need to use the raspberry pi's 3.3 volts to power pin and the ground connection to the ground pin. The Vout of the ACS712 will need 470 ohms resistor to the A1 pin of the ads1115. The A1 pint is an analog to digital converter pin, which will need a 1k ohm resistor to ground. To connect the ZMPT101B to the raspberry pi 3 we need to connect the voltage pin to the raspberry pi's 3.3 volts and the ground pin to the raspberry's ground pin. The Vout pin of the zmpt101b connects to A0 pin of the ads1115. The output of the sensor needs to be converted since it is in analog. The last thing that needs to be connect to pi would be the DS temperature sensor. In order to connect we will require some wires and one 4.7k ohm resistor. The first pin is the voltage pin of the DS1 temperature sensor. We can connect it to the raspberry pi's 3.3-volt source. The second pin is the output of the sensor and will need a connection to the raspberry pi. The GPIO4 pin will connect to the DS1's second pin and will also require a 4.7k ohm resistor to power. The power is the 3.3-volt pin connection to the raspberry pi. The third pin which is the ground pin will be connected to the raspberry's pi ground pin.

The final thing that needs to be added would be the solar panel and a 100-ohm resistor. The positive and negative wires of the solar panel need to be connected to the back side of acs712 current sensor and zmpt101b voltage sensor. The resistor should be connected to the positive side of the solar panel. See Figure 9,as it shows how it all connects to each other on a breadboard. Once the testing phase has been completed, we will now focus on creating a custom PCB board for our sensors to stay secure on the raspberry pi. Software used to create the custom PCB board is fritzing. The initial schematic design, 3, shows where the pins of these design are connected to. The pcb also let to on how the final PCB board design would look. Sockets are made in place of the sensors pin to help avoid damaging the sensors in case we have a faulty board. The sockets require 0.15-inch pin spacing to be able to fit our sockets in the board. The raspberry pi pins will be connected at the bottom of the board. The sensors, resistors and the solar panel will be connected at the top of PCB board.

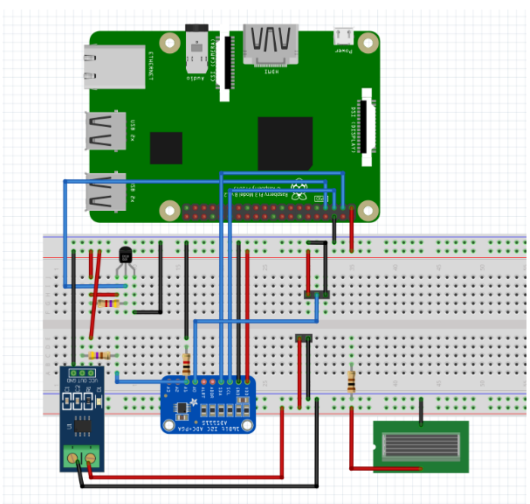


Figure 9 - breadboard diagram

The initial schematic design, Figure 10, based on datasheets (Bosch Sensortec, 2019) led to a breadboard layout Figure 11 that was realized Figure 12.

How did you build your Prototype: Breadboard?

Then a PCB was designed, **Error! Reference source not found.**, and populated (**Error! Reference source not found.**). Bill of Materials, Case, Time commitment. Testing. Progress.

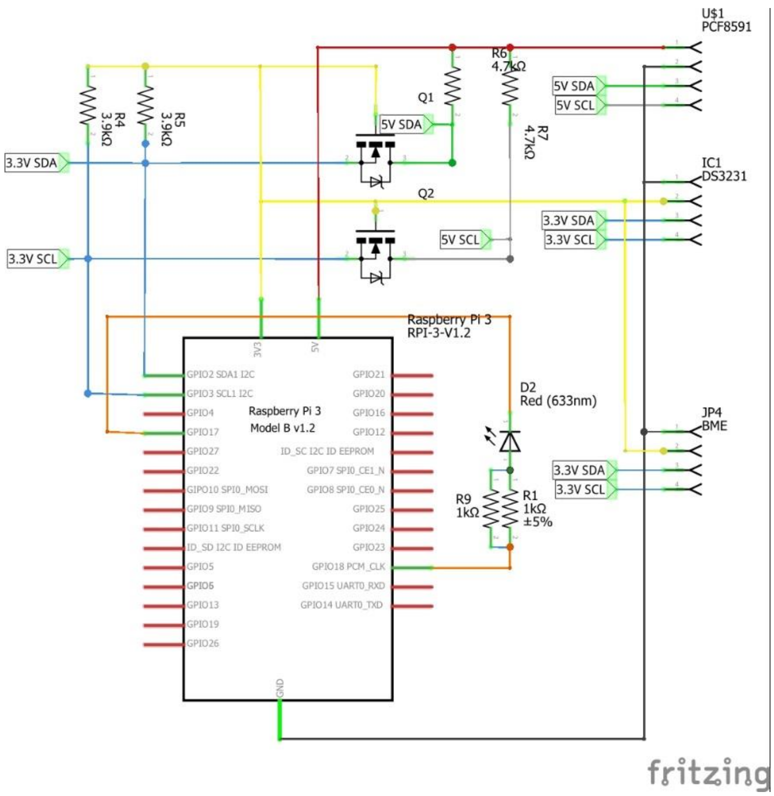


Figure 10. Initial schematic. This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0.

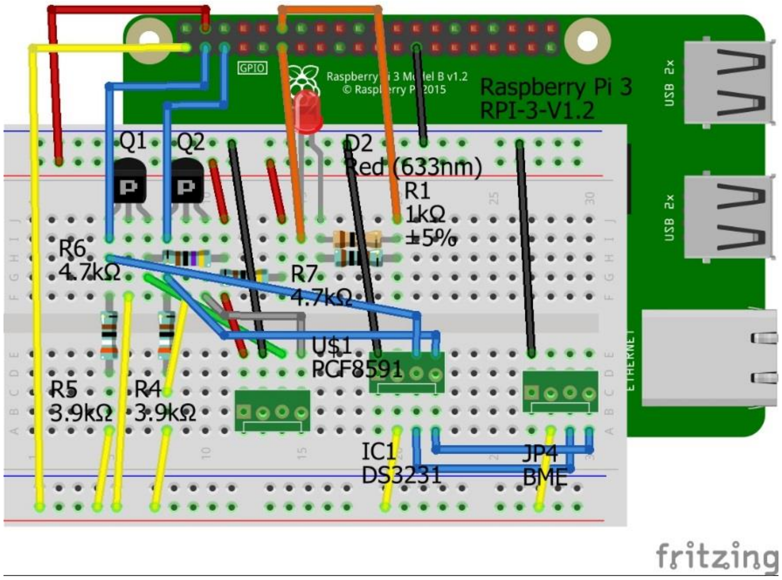


Figure 11. This work is a derivative of "http://fritzing.org/parts/" by Fritzing, used under CC:BY-SA 3.0.

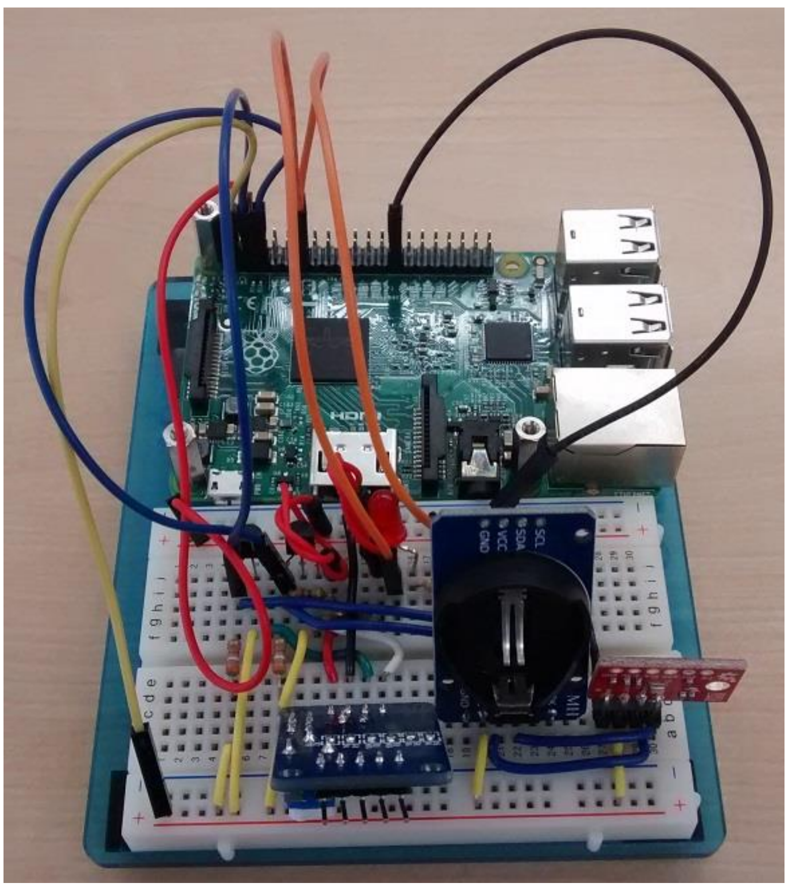


Figure 12. Breadboard prototype.

3.2.4 Printed Circuit Board

The PCB was designed through the software fritzing, we designed the breadboard from our test to the virtual wiring this help with the schematic design which gave us a plan to start designing the PCB and with two versions we came with this design, we soldered the PCB with lead free sold and clipped all wires. The PCB has 7 sockets – pi(1 socket), voltage sensor (2 sockets), current sensor (2 sockets), ADC (1 socket), and temp sensor (1 socket). We also had 4 resistors and a inputs for the solar panel. We tested the PCB with a multimeter and checked each pin for any shorts in the soldering or mistakes with the design. Until we felt comfortable with the test results we then tested on the raspberry pi and ran the code.

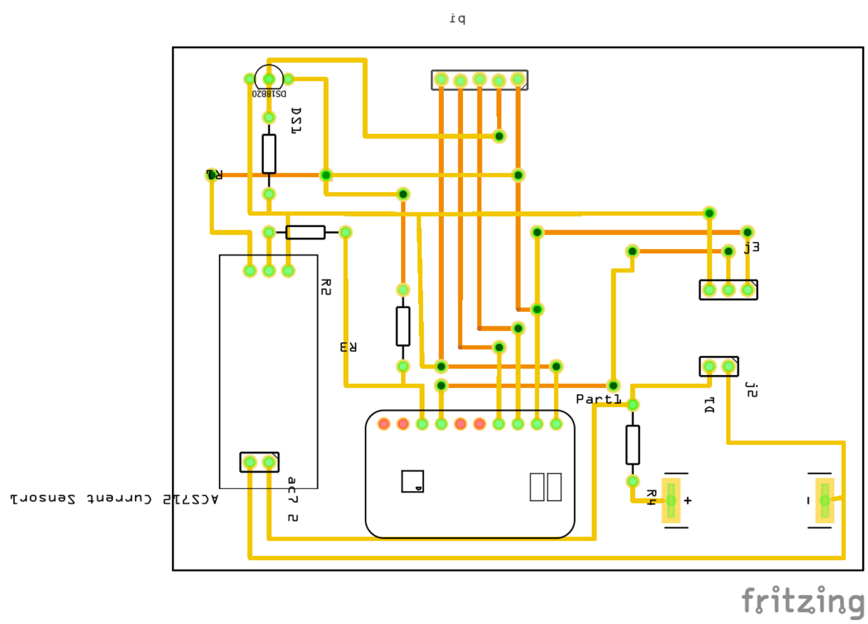


Figure 13 - Pcb prototype

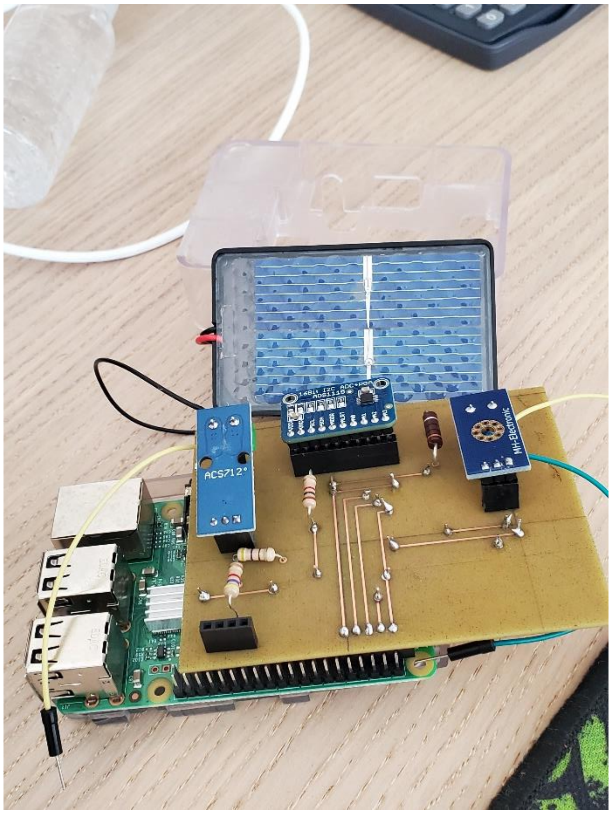


Figure 14- raspberry pi with the PCB

3.2.5 Enclosure

I used Inkscape as development platform for multiple reasons. Inkscape can save files has a svg, which will be really useful when using the laser cut. The second reason I used inscape is how efficient and easy it is to use the development platform. Inkscape is also really affordable because their platform is free for all users and this is the last reason why I used this development platform to build my case. These are some the reasons why inscape was the perfect platform to build my case for this project.

I first started by creating the top cover for the case. I made the top of the case a square with a length of 132mm and a width of 85mm. Then I added a big window with a length of 76mm and a width of 80mm in the middle of the cover so it could hold the solar panel in place at the top of the case. After I added 4 handles on each side of the cover so the top cover would be secured to the case. Finally, I added an etching of the company name at the bottom of the cover.

Then I did the bottom piece of the case by making it a square with a length of 132mm and a width of 85mm. After I added four holes around the middle of the bottom piece so I could lock in the raspberry pi to the case. Finally, I added joints on the sides of the bottom piece to connect the sides of the case to bottom piece.

After I started with the left side piece of the case by making it a length of 132mm and a width of 52mm. I also added joints at the of end of the side piece so it would clip the front and the back of the case. At the top left and right side of the side piece I added small circle cut outs so I can fit the handles from the top cover into them. On the middle left of the side piece I added a small window so users would have access to the power and HDMI plugs of the raspberry pi. Then I added a circle cut out on the middle right side of the side piece to have access to the audio port of the raspberry pi. Finally, at the bottom left and right I added square cut outs so the joints from the bottom piece would be able to connect the left side piece to the bottom piece.

After I started on the right-side piece by making it a length of 132mm and a width 52mm. On top left and right of the side piece I added circle cut outs to connect the handles from the top cover to the right-side piece. In the middle I created a small crevice to help with ventilation. Finally, at the I added two square cut outs on the bottom of the right-side piece so the bottom piece joints would have something to connect to.

Next, I did the front side by making it a square with a length of 82mm and a width of 50mm. in the middle I added a big window so users would have access to the usb and ethernet ports. Then square cut outs were inserted at each corner so the side pieces would be able to connect to the front piece.

Finally, the back piece was made a length of 82mm and a width of 50mm. This piece only has 4 square cut outs on each corner of it so the side pieces can connect to it.

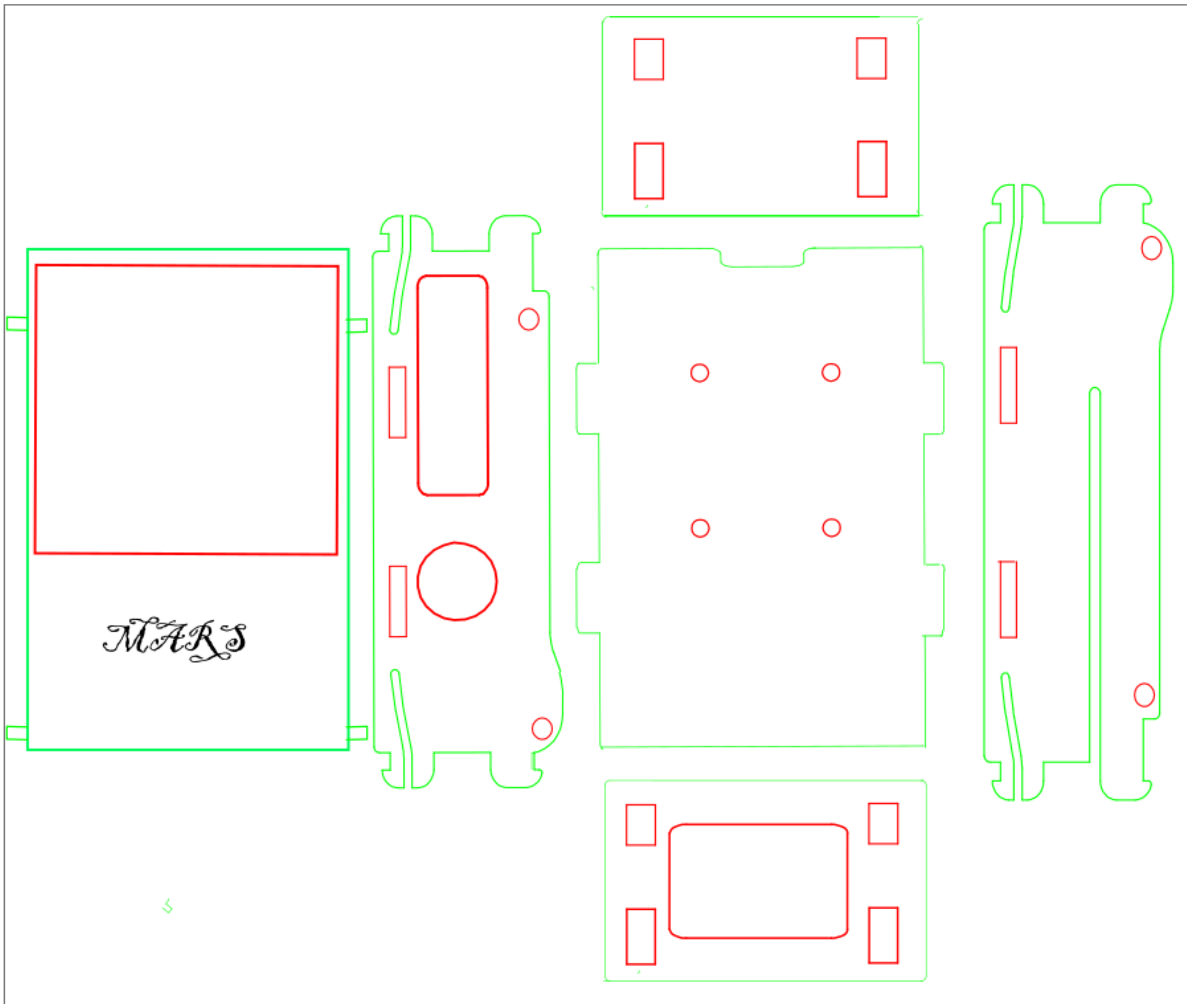

MARS

Figure 15. Example enclosure.

3.3 Integration

We chose to integrate our database with Google's firebase, as it was made for mobile applications and has a secure and reliable system.

3.3.2 Database Configuration

The database was created through firebase and connected through the android studio with the database function that tests the serial number from the input and when clicking the update button, the code will authenticate the serial number and then pull from the records to an array list to post onto a list view that displays the current, voltage, and temperature from the database that is given from the raspberry pi readings that is pushed to it.

3.3.1 Enterprise Wireless Connectivity

The data from the database was connected from firebase onto the mobile application. Unfortunately, due to conditions of the COVID-19 pandemic we were unable to connect the mobile application to the prototype as all equipment facilities dire to the prototypes manufacturing and completion were postponed due to quarantine. Although the enterprise connecting through the Humber encrypted connection is not done.

3.3.3 Security

The security is not encrypted instead it is hashed from the user and the app itself is hiding the input of the serial number from the user and the security of the app is at the hands of the firebase connection and security of the database. The rules on firebase allows people to read only and no write permission. This is due to the scope of our project as each device has a unique serial number and those devices have access to write to it.

3.3.4 Testing

Testing was done through applying data in the database and checking the display from the list view on the graph page from the app. See figure 16, and figure 17.

Figure 16 - add serial

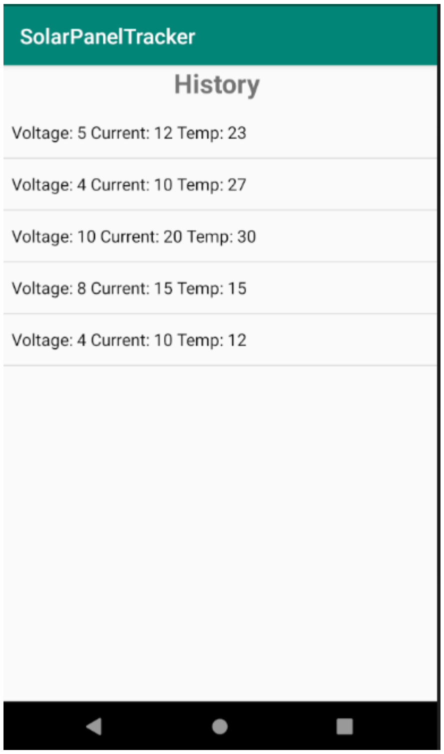


Figure 17 - database

4.0 Results and Discussions

As of today our prototype isn’t perfect we still have some bugs in our code and due to the recent events we are unable to access the needed facilities for further testing although the prototype is complete and runs. During this project we learned a lot of skills like teamwork, communication, problem solving, planning, and we used software like fritzing, GitHub, Android studio, firebase, Inkscape, and Microsoft Word/Excel.

5.0 Conclusions

If we were making a thousand of these devices we would have to test and run every 5 devices for faulty sensors or wiring. This is due to the amount of sensor one device holds and sometimes sensors would have problems in manufacturing. We would conduct testing with PCB first a simple short testing with a multimeter would be fine and then running the PCB on a pi. If there is a problem, we would have to test the sensors individually and find the issue. We hope this device inspire companies to buy green energy devices with the reliable data tracking prototype for solar panels.

6.0 References

Bosch Sensortec. (2019, July). *BME680 - Datasheet.* Retrieved from Robert Bosch GmbH: https://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BME680-DS001.pdf

Ontario Solar Installers. (2019, march 25). *Why do you need a Solar Monitoring*. Retrieved from Ontario solar installers.ca: https://ontario-solar-installers.ca/solar-panel-installers/why-do-you-need-a-solar-monitoring-system/

RecSolar. (2017). *solar for business*. Retrieved from Recsolar.com.

Yerema, R., & Kristina , L. (2019, april 14). Humber college recoginzed as one of canada's greenest employers. *canadastop100*, 1.

7.0 Appendix

7.1 Firmware code

The project was having issues with the wiring of the circuit on the breadboard and the readings of the zmpt101b was unreliable so we bought a smaller scale voltage sensor and had a display before week 6 but an issue of the current sensor wiring was to be addressed due to it not going parallel with the voltage sensor and in series with the current sensor the temp sensor was not much of an issue as it did not obstruct other sensors wiring to the analog to digital converter. The code was written in python to help match with the existing libraries for the sensors we wrote the code in a while loop with a 2 second delay with functions from the header files calling for the readings of the current and voltage sensors, with an exception of the temperature sensor which reads from another header file that is not from the python libraries. Financially the project is doing well expenses are low the last addition to the budget was the new voltage sensor which coat $10.99 (tax included) and took 2 days to ship with amazon prime. Although spending is low, we are expecting that the cost of the 3d printing and laser cutting to be higher as mistakes with scaling could prove an issue. Progress status is going well as we have the wired breadboard providing readings and the PCB v1 to be printed. We plan on testing and soldering before the week due and are a head on the enclosure designs, the app is close to being in a desired state and the database and Its security will be completed 3 weeks earlier than expected. The issue we face is mostly the PCB testing, and the database display on the mobile app without crashing other than those we are on track with completion of the tasks. The code is posted on GitHub.

7.2 Application code

The data from the database was connected from firebase onto the mobile application. Unfortunately, due to conditions of the COVID-19 pandemic we were unable to connect the mobile application to the prototype as all equipment facilities dire to the prototypes manufacturing and completion were postponed due to quarantine. Although the enterprise connecting through the Humber encrypted connection is not done. We have completed the databases code with the app and login activity with the standard emulator to firebase the code is posted on GitHub