**Solar Capstone Project**

From: Raphael Najera, Johnson Liang and Adrian Caprini  
Discipline: Computer Engineering Technology  
Date: 4/23/2018

# Declaration of Joint Authorship

Adrian Caprini, Raphael Najera and Johnson Liang the group members of the project Solar Capstone, confirm that this report submitted for assessment is the joint work of ourselves, and research which is expressed in our own words. Any uses made within our own works of any other author, in any form (ideas, figures, previous technologies, tables, programs, texts) are properly acknowledged at the point of use. A list of the references used is included. For our group members we have evenly divided the work as follows: Adrian Caprini worked on the Database, Raphael Najera worked on the mobile application, and Johnson Liang worked on the web application.

# Approved Proposal

## Executive Summary

As a student in the Computer Engineering Technology program, we will be integrating the knowledge and skills we have learned from our program into this Internet of Things themed capstone project. This proposal requests the approval to do the software portion that will connect to a database as well as to a mobile device application and web application. The internet connected hardware will include a custom PCB with the following sensors and actuators Solar Panels, PV3. The database will store the data retrieved from the four solar panels PV1, PV2, PV3, and PV4. The mobile device functionality will include information retrieved from the database from the four solar panels. This will give the audience a visual aspect of how much solar energy has been collected and depleted. and will be further detailed in the mobile application proposal. We will be collaborating with the following company/department Kerry Johnston, Humber College Institute of Technology & Advanced Learning North Campus, Prototype Lab, and Humber College Sustainable Energy and Building Technology. In the winter semester we plan to form a group with the following students, who are also building similar hardware this term and working on the mobile application with Raphael, Johnson Liang, and Adrian Caprini. The hardware will be completed in CENG 317 Hardware Production Techniques independently and the application will be completed in CENG 319 Software Project. These will be integrated together in the subsequent term in CENG 355 Computer Systems Project as a member of a 3-student group.

## Background

The problem solved by this project is Solar power is clean renewable energy collected from the sun. As a result, by using solar energy it helps reduce greenhouse gas emissions and relying on fossil fuels. Fossil fuels is a heavily relied on source to produce energy however, it will deplete one day. Thus, solar energy should be invested into which has an unlimited supply. A bit of background about this topic is the sun produces renewable energy where it is clean and does not generate harmful environmental emissions. If the properties are harnessed then that source of energy can be manipulated to produce electricity, heat, and other valuable energy properties. A solar panel is made of many cells which consist of a positive and negative layer. When the photons collide with the semiconductors on the panel it creates an electric field which are harnessed by the positive and negative layers. The produced energy is multiplied by the number of cells within a panel and the number of panels in a solar array.

Existing products on the market include [1]. I have searched for prior art via Humber’s IEEE subscription selecting “My Subscribed Content” [4] and have found and read [5] which provides insight into similar efforts.

In the Computer Engineering Technology program, we have learned about the following topics from the respective relevant courses:

• Java Docs from CENG 212 Programming Techniques in Java,

• Construction of circuits from CENG 215 Digital and Interfacing Systems,

• Rapid application development and Gantt charts from CENG 216 Intro to Software Engineering,

• Micro computing from CENG 252 Embedded Systems,

• SQL from CENG 254 Database with Java,

• Web access of databases from CENG 256 Internet Scripting; and,

• Wireless protocols such as 802.11 from TECH152 Telecom Networks.

This knowledge and skill set will enable me to build the subsystems and integrate them together as my capstone project.

## Methodology

This proposal is assigned in the first week of class and is due at the beginning of class in the second week of the fall semester. My coursework will focus on the first two of the 3 phases of this project:

Phase 1 Hardware build.

Phase 2 System integration.

Phase 3 Demonstration to future employers.

*Phase 1 Hardware build*

The hardware build will be completed in the fall term. It will fit within the CENG Project maximum dimensions of 12 13/16" x 6" x 2 7/8" (32.5cm x 15.25cm x 7.25cm) which represents the space below the tray in the parts kit. The highest AC voltage that will be used is 16Vrms from a wall adaptor from which +/- 15V or as high as 45 VDC can be obtained. Maximum power consumption will be 20 Watts.

*Phase 2 System integration*

The system integration will be completed in the fall term.

*Phase 3 Demonstration to future employers*

This project will showcase the knowledge and skills that I have learned to potential employers.

The brief description below provides rough effort and non-labour estimates respectively for each phase. A Gantt chart will be added by week 3 to provide more project schedule details and a more complete budget will be added by week 4. It is important to start tasks as soon as possible to be able to meet deadlines.

No other purchases are required for this project as we will be using solar panels located on the roof on the L-wing.

Raspberry Pi 3: used to run the python code to retrieve the data from the website that contains the data of the solar panel and then save the data to the firebase

## Concluding remarks

This proposal presents a plan for providing an IoT solution for this concept that could be used for homes and businesses that have installed solar panels on their roofs. This would show the data from the solar panels from the sunlight when they are running. With the information that is retrieved, the data will be stored in the database. The user will be able to retrieve the information by using the app on their smart phone. As a result, users will be able to keep track the amount of energy the solar panels have collected, CO2 avoided, and energy depleted. This is an opportunity to integrate the knowledge and skills developed in our program to create a collaborative IoT capstone project demonstrating my ability to learn how to support projects such as the initiative described by [5]. I request approval of this project.

# Abstract

Solar power is clean renewable energy collected from the sun. As a result, by using solar energy it helps reduce greenhouse gas emissions and relying on fossil fuels. The four solar panels located on the roof of the L building at Humber College (North) will monitor how much solar energy is collected and the total amount of energy collected every 30 minutes. This data gathered from the four solar panels will then be stored in a database. The data will be available for users to access through our mobile and PC application. The mobile application will retrieve the data from the solar panel PVs and display the information. The web application will also display the data retrieved from the solar panel in the form of a PC GUI. The users can then access this data globally from our web or mobile application. This system can be used to help educate the community about the significance of using renewable energy rather than fossil fuels. For example, the purpose of using clean renewable energy and the importance of avoiding greenhouse gas emissions and the purpose for using clean renewable energy. This project we collaborated with Kerry Johnston whose part of Humber's Sustainable Energy and Building Technology. At the start of this project, we visited room L240 to inspect how the solar panels were communicating and connected.

Table of Contents

[Declaration of Joint Authorship 2](#_Toc510448366)

[Approved Proposal 3](#_Toc510448367)

[Executive Summary 3](#_Toc510448368)

[Background 4](#_Toc510448369)

[Methodology 5](#_Toc510448370)

[Concluding remarks 6](#_Toc510448371)

[Abstract 8](#_Toc510448372)

[Illustration Lists 10](#_Toc510448373)

[1. Introduction 11](#_Toc510448374)

[2. Project Description 13](#_Toc510448375)

[2.1 Problem 13](#_Toc510448376)

[2.2 Rationale Behind Project 13](#_Toc510448377)

[2.3 Project Scope 13](#_Toc510448378)

[2.4 Software Requirement Specifications 13](#_Toc510448379)

[2.5 Build Instructions 14](#_Toc510448380)

[2.5.1 Introduction 14](#_Toc510448381)

[2.5.2 System Diagram 15](#_Toc510448382)

[2.5.3 Bill of Materials/Budget 16](#_Toc510448383)

[2.5.4 Time Commitment 16](#_Toc510448384)

[2.5.5 Power up 17](#_Toc510448385)

[2.5.6 Software Assembly 17](#_Toc510448386)

[2.5.7 Database 19](#_Toc510448387)

[2.5.8 Mobile Application 20](#_Toc510448388)

[2.5.9 Web Application 21](#_Toc510448389)

[2.5.10 Unit Testing 22](#_Toc510448390)

[2.5.11 Production Testing 23](#_Toc510448391)

[2.6 Problems Encountered 24](#_Toc510448392)

[2.7 Approaches 24](#_Toc510448393)

[2.8 Walkthrough of System 24](#_Toc510448394)

[3. Progress Reports 25](#_Toc510448395)

[3.1 Report 1 25](#_Toc510448396)

[3.2 Report 2 27](#_Toc510448397)

[3.3 Report 3 29](#_Toc510448398)

[4. Conclusions 32](#_Toc510448399)

[5. Recommendations 33](#_Toc510448400)

[6. Technical References 34](#_Toc510448401)

[7. Appendices 35](#_Toc510448402)

# Illustration Lists

2.5.2a: Solar Capstone Project System Diagram p.14

2.5.8a: Screenshot of PV1 on Solar Light application p.20

2.5.9a: Screenshot of the Web application p.21

2.5.10a: Testing data on the firebase p.22

2.5.11a: Screenshot of the solar capstone code running p.23

# 1. Introduction

Fossil fuel has been the Earth's main source of energy for many centuries, but this source will one day be depleted and extinct. Instead, we have to educate and make ourselves aware of an alternate energy source such as solar energy. Unlike fossil fuels, solar energy is renewable and clean from the sun, and with the current advancements it can help reduce greenhouse gas emission and the cost of money. We will integrate a solution which will help the community be aware of the significance revolving around clean energy.

This project revolves around solar panels and the Sunny Boy sensor box to gather data on how much solar energy is collected from each solar panel and the total amount of energy collected every month and year. The information stored on the database will be retrieved so that it can be read on our mobile and web application, so that users can access this information from their cell phones or publicly at Humber College. It is possible to monitor all four solar panels within the application and displaying the information of each one by clicking the different tabs in our mobile application.

The idea of the solar panels is for Humber to use renewable energy and eliminate fossil fuel usage and to show the Humber community how solar energy works. Also, the Humber community would be able to see how much solar energy Humber uses each day, month and year. By having access to this data, individuals interested in this data will have live up-to-date records and observe how much solar energy Humber uses on a daily or monthly basis.

# 2. Project Description

## 2.1 Problem

## 2.2 Rationale Behind Project

## 2.3 Project Scope

## 2.4 Software Requirement Specifications

2.4.1 Database

The project will include four databases, each database contains data from each of the four solar panels. The databases will be used to store the information from the PV's of the four solar panels. The information stored on the databases will be retrieved so that it can be read on our mobile and web application, so that users can access this information from their cell phones or publicly at Humber College. The data being stored in each database will include how much solar energy is collected from each solar panel and the total amount of energy collected every month and year. We will write a python code to get and retrieve the data from the solar panel PV's. (Developed by Adrian Caprini)

2.4.2 Mobile Application

This project will include a mobile application which is currently available for Android platforms. The mobile application will be called Solar Light. This app will take the data from the solar panel PVs, which is stored on the firebase and display them on the app. The user will be able to see how much solar energy has been collected from each of the solar panel PVs each day. In each of the PV's screen will display which PV you are looking at, the current date, power, daily yield, and total yield. It will also show a log of the previous data that it retrieved from the firebase. The app will include a main screen which will display more information about the solar panels and importance of using solar energy. On the action bar, it will display link to the Humber website, Solar Capstone Web application and about information. (Developed by Raphael Najera)

2.4.3 Web Application

The web application will function similarly to the mobile application where both will display data retrieved from the solar panel stored in the database. The difference between the two is one will be accessed from the phone while the other is globally advertised. The web application will be displayed on a monitor in the L building, where the Humber community can observe the school's solar resources. As a result, the community will be able to observe the amount of solar energy collected from the sun, how much energy has been expended, and how much CO2 has been avoided. This graphical user interface is intended to be built using Notepad++. (Developed by Johnson Liang)

## 2.5 Build Instructions

2.5.1 Introduction

The software system project we decided to work on this semester involves four solar panels located on top of the L building of Humber College North Campus. The purpose of this project was to interact and store the gathered data from all four solar panels into a database. Then, with the retrieved information we intend to display it on our mobile and web application. The data we will be retrieving from the solar panels consist of the power, daily and total yield energies. These four solar panels are manufactured from three different companies. For example, the PV1 and PV4 hardware was manufactured by Sunny Webbox, PV2 was manufactured by Envoy Communications Gateway and PV3 manufactured by Outback. The hardware for our project was already provided and installed for us so our main focus was to create the software aspect. At the end of this project we want to be able to retrieve the pushed data on our firebase. This data will be displayed onto our mobile and web applications for the Humber community and users to observe how solar panels work, and how much solar energy has been collected and expended.

2.5.2 System Diagram

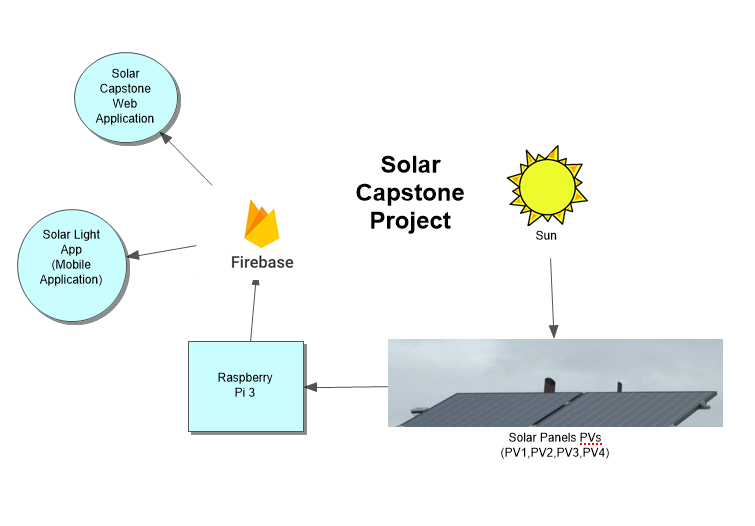


Image 2.5.2a: Solar Capstone Project System Diagram

2.5.3 Bill of Materials/Budget

The materials needed to replicate the Solar Capstone Project are solar panels and a Raspberry Pi. The solar panels were provided by the school and we bought the Raspberry Pi from Amazon. The solar panels were manufactured by the companies Envoy Communications Gateway, Sunny Webbox and Outback Power.

1. Raspberry Pi 3 (CanaKit Starter Kit): CAD $99.99

Link: https://www.amazon.ca/CanaKit-Raspberry-Complete-Starter-Kit/dp/B01CCF6V3A/

1. Solar Panel PVs (Provided):

PV1 and PV4: Envoy Communications

PV2: Envoy Communications Gateway

PV3: Outback Power

2.5.4 Time Commitment

This project can be completed with a group of three people or individually. However, if completed in a group the tasks can be divided and completed quickly compared to individually. This project can be completed in the span of 13 weeks where each task is divided weekly. For example, one week we will write python code to retrieve specific data and another week we can write code to push the retrieved data into a database. At the end of the project we will present our completed application which will be able to successfully retrieve pushed data.

2.5.5 Power up

If you are booting up the Raspberry Pi for the first time, insert the micro SD card to the SD card reader from the Raspberry Pi package and make sure that the correct operating system is pre-installed. If it is not, the OS can be downloaded from this link: https://www.raspberrypi.org/downloads/noobs/. Once the OS has been installed on the Raspberry Pi, insert the micro SD card into the Raspberry Pi and power it. After doing this, the Raspberry Pi should then boot up. Before going further, users will be required to set up the operating system and configure their Raspberry Pi settings and Wi-Fi.

2.5.6 Software Assembly

The data came from four different sources and so, we needed to pick a database which would store the data under one platform. As a result, we decided to use Firebase as it's free to use and can be easily accessed. Additionally, we have experience with Firebase from CENG319: Software Projects. We also had to decide the platform we wanted to use to create the mobile and web application. We choose Android Studio to create our mobile application and Notepad++ to create the web application. The web application is created using HTML and JavaScript which will be locally hosted on the PC.

Furthermore, a Python script is required to retrieve and push data into our Firebase.

The python code we created will retrieve the solar panel data from the solar panel websites hosted on Humber's network and store the data on Firebase. The code can be downloaded here: https://minhaskamal.github.io/DownGit/#/home?url=https:%2F%2Fgithub.com%2FRaphaelNajera%2FSunlight\_Sensor%2Ftree%2Fmaster%2Fdocumentation%2FCENG355%20Solar%20Capstone%2Ffirmware

The steps to run the Solar Capstone python code on the raspberry pi:

1. Power up the raspberry pi and set up VNC server on the raspberry pi
2. On the PC, download and run VNC viewer. With VNC viewer you can remotely connect to the raspberry pi
3. On the raspberry pi before running the code, you have to set the default python to python 3 and install the following modules which is required to run the code.

On the terminal type the following code to set the default python version to python 3.5.3.

nano ~/.bashrc

When you open ~/.bashrc, add new alias to change the default python executable.

alias python='/usr/bin/python3'

After you added the new alias, save the file, and enter this command.

. ~/.bashrc

Next enter the following command to install the modules.

pip install --upgrade setuptools

sudo apt-get install libxml2-dev libxslt-dev python-dev

sudo apt-get install python3-lxml python-lxml

pip install beautifulsoup4

pip install pyrebase

pip install apscheduler or pip3 install apscheduler

The following module are required because with beaultifulsoup 4, it has the function to allow the code to read the html and retrieve the data from the PV’s website. The pyrebase allow us to push the data that the code retrieved from the website to the firebase. The apscheduler allow you to schedule the functions so it will run a certain amount of time

1. Once you got the module installed, you can now run the code. The code will loop every 30 minutes.

python Solar\_Capstone\_PV\_v6.py

2.5.7 Database

Create a Firebase account using Google Mail to access exclusively to your database. Then we had to create the tables for each solar panel PV and test the code we wrote with placeholder values. Next, we wrote the python code that is going to be used to get and retrieve the data from firebase. This python code will execute every 30 minutes to fetch the latest results and push it into firebase. We also need to retrieve the epoch time to be able to delete this data after 30 days. Since we have a limited amount of storage this is the current solution we have come up with. The epoch time will not be displayed to the user it is for the programmer’s use.

2.5.8 Mobile Application

Android Studio was the software we decided to use to make the counterpart of the mobile application. Our app connects to Google’s Firebase where it will retrieve existing live data ran 24/7 on an external device running a python script. Our application is called “Solar Light” because it our project revolves around collecting solar energy produced from the sun. Once the app is created we then setup the app to connect our firebase by clicking Tools, and then Firebase. On the firebase assistant we clicked on analytics and then log an analytics event. We followed the steps such as login to the firebase, so the app can connect to the firebase and then adding the code to read from the firebase. Then we create four different screens for each of the four PV's by creating a navigation drawer for the solar PV's. Each of the PV's will display which PV you are looking at, the current date, power, daily yield, and total yield. It will also show a log of the pervious data that it retrieved from the firebase. The app will include a main screen which will display more information about the solar panels and the importance of using solar energy. On the action bar, it will display link to the Humber website, solar capstone web application and about information.

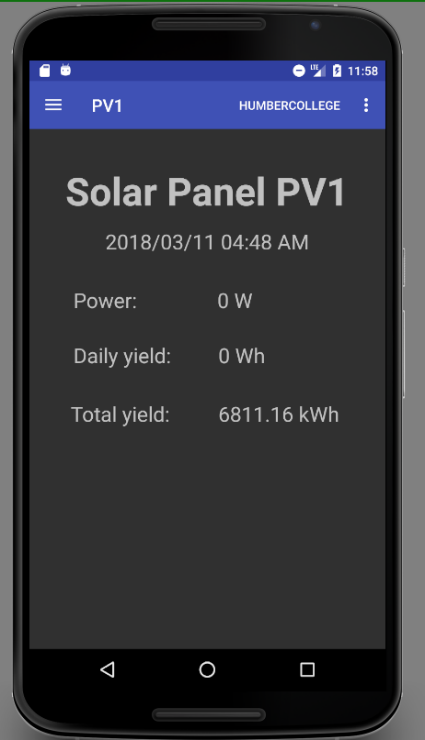


Image 2.5.8a: Screenshot of PV1 on Solar Light app

Link to the GitHub Solar Light application:

https://github.com/RaphaelNajera/Solar-Capstone-App

2.5.9 Web Application

The web interface was simply created using Notepad++ where one HTML document linked to four separate HTML forms. The HTML linked to four different frames and consisted of different panes that displayed four different sections. It is chronologically ordered so that users have a good idea of which solar panel is which. In each of these frames it will display the current PV panel and latest data retrieved from the last query. In other words, it would be the latest data retrieved from the database and placed under the <table> tag. Additionally, it will retrieve an array of data which is simply the history of all the fields. As such this allows the user to observe how much data has been collected every 30 minutes or so. This HTML can be hosted locally or through a domain service such as GitHub and will be able to be open on any browsing platform.

Link to the GitHub Solar Capstone Web application:

https://github.com/j-liang/solarcapstone\_web

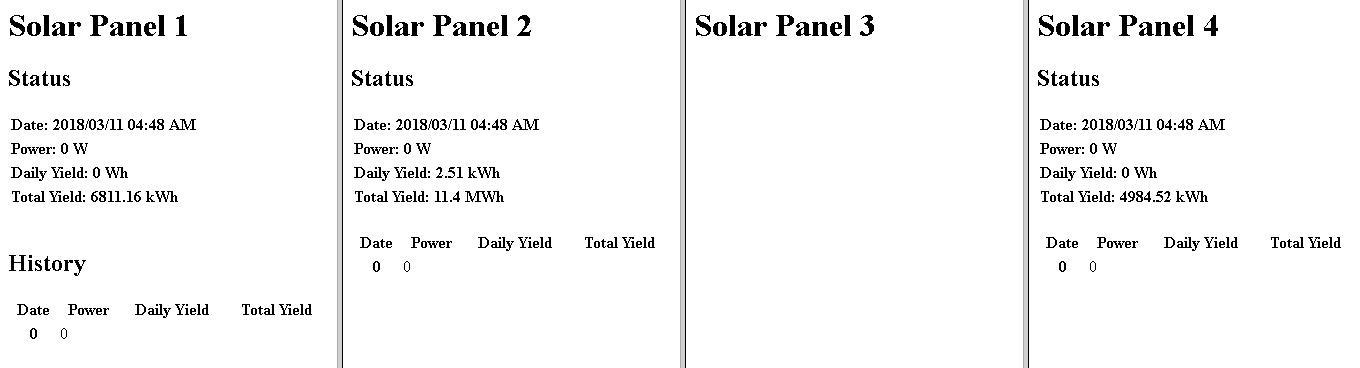


Image 2.5.9a: Screenshot of the Web application

2.5.10 Unit Testing

We did a lot of debugging and testing in order to check if specific tags were found with certain data. In order to achieve this, we wrote a python script to complete this task then we defined functions to push data into a pre-made firebase. However, prior to running the script we have to ensure that the raspberry pi is set up with installed python functions in the terminal which is explained under software assembly. Once the data has been stored it can be retrieved and be listed into our mobile and web applications. The web application can be ran on any Internet browser and it will retrieve the required information.

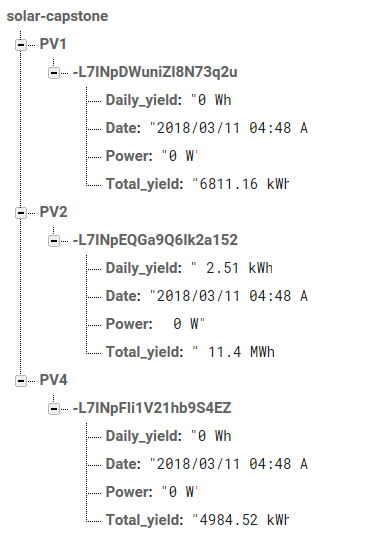


Image 2.5.10a: Testing data on the firebase

2.5.11 Production Testing

Once we successfully ran the python code on the raspberry pi, it will display the name of the solar panel PV's, epoch time, date, power, daily yield and total yield information from each of the solar panel. Our python code will be running 24/7 which the code will keep looping so that as it progresses the status will change. This will allow our mobile application and web application to display date, power, daily yield, and total yield from each of the solar panel with the current and updated status.

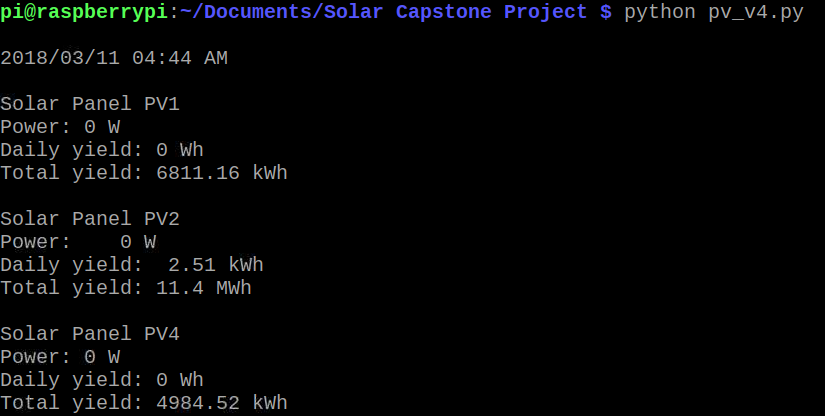


Image 2.5.11a: Screenshot of the solar capstone code running

## 2.6 Problems Encountered

## 2.7 Approaches

## 2.8 Walkthrough of System

# 3. Progress Reports

## 3.1 Report 1

To: Austin Tian <Austin.tian@humber.ca>

Cc: Johnson Liang <johnny.son@live.ca>, Adrian Caprini<adrianc\_34@hotmail.com>

Subject: Solar Capstone Project Status Update A

Hi Austin,

This is our status A update on Solar Capstone. I will be student A, Johnson will be student B and Adrian will be student C. Up to this week, we have created templates for the mobile and PC applications, Database, Software Requirements Specification, Declaration of authorship, Abstract and Introduction. We have recently submitted the Declaration of authorship, Abstract, and Introduction. We are now working on the status update A. As of right now, we are on track.

For the database, we are using firebase to store the data we retrieve from the Solar Panel PVs. Adrian has created a table on firebase where the data is stored in four different sections. Johnson and Raphael has written python code which filters the HTML <table> in order to retrieve specific information. In this case, we are retrieving the power, daily and total yield of energy. Initially, we tested code to see if the data would push to our database using placeholder data. Now we are combining the code where it filters the data and the code to push the data into our database. So far, we have successfully push data from PV1 and PV4.

For the mobile application, as of right now, Raphael has created the template for each of the Solar Panel PVs by using drawerlayout. It will display the data it retrieves from the firebase. The next step is once the database has been fully implemented we will be able to retrieve the stored data on our firebase and display that information on the app.

For the web application, Johnson has created the template. Our current template divides the screen into four screens and each screen is assigned to each solar panel. Each screen will retrieve information from its corresponding table on the firebase. The next step is once the firebase is ready and filled with solar panel data we can continue working on the web application.

As of right now, we can connect and use python code to retrieve data from PV1 and PV4. Once the data has been retrieved it is immediately pushed into firebase for storage. We're currently figuring out how to retrieve the data from PV2 and PV3.

The problems we encountered was communicating to the solar panels as all the IP addresses were not active for connection. The second problem was we were limited on when and where we can work on the project. Since the solar panels are tied to Humber's network, we were not able to test code unless we're at Humber. However, this issue will be resolved once we set up a remote connection with a raspberry pi at Humber. The third problem we encountered is retrieving PV3's data as the data is not available in the HTML. When we checked PV3's HTML, the HTML in inspect mode is different from the HTML shown on the view page source.

Our financial status didn't increase and is on budget because everything for this project was provided for us. We didn't have to spend additional money to make this project possible. We will be using a raspberry pi purchased last semester to remotely connect to the solar panel which can only be accessed on Humber's network using VNC.

Solar Capstone Github - https://raphaelnajera.github.io/Sunlight\_Sensor/

Thanks,

Raphael Najera - Solar Capstone

## 3.2 Report 2

To: Austin Tian <Austin.tian@humber.ca>

Cc: Adrian Caprini<adrianc\_34@hotmail.com>, Raphael Carlo Najera <rcnajera@outlook.com>

Subject: Solar Capstone Project Status Update B

Hi Austin,

The following email will discuss the status of the Solar Capstone project since status A. In status B, the team has made significant progress in terms of retrieving and displaying data into our applications. Additionally, the team has been frequently updating the OACETT documentation based on the feedback received from our advisor.

There weren't any major changes to the database as it was already retrieving real-time data from the solar panels. However, one feature Adrian added to the python script was a field called "epoch" into the tables. The purpose of this additional field was so the database can flush out the tables every 30 days. Firebase has a limited storage to 1GB per account and if the python script runs every 30 minutes for 30 days, the database will have an estimation of 5760 entries across four PVs. As a result, if the data is old the team believes it should be removed to limit the amount of entries. Johnson and Raphael wrote the python sections to retrieve PV2 meanwhile Adrian retrieved PV3. However, the filtered data from PV3 is still unclear to the team.

The mobile application transitioned from a basic template to being able to retrieve the latest entry from the Firebase. As a result, Raphael was able to display the latest power, daily yield, total yield from the PV’s. Raphael is currently working on a feature which displays the history of the accumulated data. The layout for each PV's is divided into two layouts, one being the latest data and the other being a history of data. The history of data will use a ListView to display previous entries from the Firebase. Both layouts will be accessed by using Tab layout. https://github.com/RaphaelNajera/Solar-Capstone-App

The web application was originally planned to be made with NetBeans however, the team decided to use HTML instead. Johnson was able to retrieve and display the latest date, power, daily yield, and total yield from all solar panels. The HTML contains four different frames and each frame corresponds to its solar panel. As of right now, the HTML is able to retrieve a history of 16 entries for the power section with the help of Johnson and Raphael working together. The array is still a working progress and we are attempting to retrieve the history of all dates, power, and daily yield. Additionally, this HTML file will be hosted on GitHub so that it can be publicly viewed. https://github.com/j-liang/solarcapstone\_web

The team is still having issues with PV3 although the python script is capable of filtering out the data. The team is unsure at what data should be pushed into the database. We also switched the web from java app to HTML because there's no SDK support for Firebase.

There are no financial updates to this project from status A. As of current, the project only requires a Raspberry Pi to run the python script. The device must be present on Humber's network otherwise the solar panel's IP addresses are not retrievable. As a result, the project's financial status remains the same. Solar Capstone Project Github: https://github.com/RaphaelNajera/Sunlight\_Sensor

Thanks,

Johnson Liang - Solar Capstone

3.3 Report 3

To: Austin Tian <Austin.tian@humber.ca>

Cc: Raphael Carlo Najera <rcnajera@outlook.com>, Johnson Liang <johnny.son@live.ca>

Subject: Solar Capstone Project Status Update C

Hi Austin,

The following email will discuss the status of the Solar Capstone project since status B. In status C, the team has made progress retrieving and displaying data from PV3 into our applications. Additionally, the team has been updating the technical report handed in including all the documentation handed in to this point based on the feedback received from our advisor. The team believes the mobile and web application should be completed by the end of this semester as we are on track.

There weren't any major changes made to the database since the last status update except having a python script to push PV3 values into Firebase. Additionally, the python script now includes code to delete tables after 30 days. This was achieved by using the epoch time we retrieved in status B. Adrian helped Raphael with adding an image to the main screen of the app, creating a splash screen and about page for the mobile application.

There have been a few changes made to the mobile application, but it was already able to retrieve the latest entry and history from the Firebase. Since the last update of the mobile application, Raphael has been able to display the history of the accumulated data up to 24 entries and added the labels for the logs. Raphael also added the link to the web app on the action bar, on the current data it now shows the date and time of last update and updated to display the current data of PV3. Raphael is currently working on displaying the PV3 history and adding icon and image to the app. <https://github.com/RaphaelNajera/Solar-Capstone-App>

There have been a few changes made to the web application since the last status update, but it was already able to retrieve the latest entry from the Firebase. Since the last update of the web application, Johnson has modified the HTML with CSS to enhance the appearance of the webpage. As a result, minor details such as the background color, font family, and Humber logo were added. Additionally, Johnson was able to push the values of PV3 to Firebase since the last update. Johnson was also able to retrieve and display the latest entries from the Firebase to the HTML frame for PV3. As of current, the web application is almost complete but we have minor implications to enhance its capabilities. <https://github.com/j-liang/solarcapstone_web>

The problem we had was displaying the data from PV3 to our Firebase, mobile and web application. We were also unclear on what we were going to display until last week when it became clearer to us. The reason we have not shown this project to the collaborator is because we are currently working on the mobile and web app on displaying the data from the 4 solar panel PV's. We are planning on showing the mobile and web app to our collaborator in week 11.

There are no financial updates to this project from status B. The project only required a Raspberry Pi to run the python script. The device must be present on Humber's network otherwise the solar panel's IP addresses are not retrievable. As a result, the project's financial status remains the same. Solar Capstone Project Github: <https://github.com/RaphaelNajera/Sunlight_Sensor>

Sincerely,

Adrian Caprini - Solar Capstone

# 4. Conclusions

The purpose of this project is to develop a monitoring system which keeps track of the solar energy collected from the solar panels. The panels are located on top of the roof of the L building of Humber College North Campus. This system will monitor the current power status, daily and total energy yield. The data is then sent to the database where it will be stored for 30 days. On the contrary, the status of each solar panels will be updated every 30 minutes and display a history of all previous entries up to 10. As a result, this gives an opportunity to allow the Humber community to recognize the importance of solar energy. The mobile application will include these features but also give an additional educational purpose that teaches users the significance. This will allow the Humber community to monitor the status of the solar panels on the mobile and web application. Additional features will include the current weather temperature on the web application which will make it appealing. Also, it is useful as it provides the current weather status of how hot, warm, or cold it is outside.

# 5. Recommendations

# 6. Technical References

[1] Hudson, G., Noble, G., Lea, T., & Galloway, M. (n.d.). Solar PV. Retrieved February 04, 2018, from https://guide.openenergymonitor.org/applications/solar-pv/

[2] Energy, O. (n.d.). The ultimate beginner's guide to solar panels. Retrieved February 04, 2018, from https://www.ovoenergy.com/blog/green/the-ultimate-beginner-s-guide-to-solar-panels.html

[3] AlsoEnergy. (n.d.). Retrieved February 4, 2018, from http://www.alsoenergy.com/PowerTrack/PowerLobby.aspx?sid=33838

[4] Institute of Electrical and Electronics Engineers. (2015, August 28). IEEE Xplore Digital Library [Online]. Available: https://ieeexplore.ieee.org/search/advsearch.jsp

[4] Jain, C., & Singh, B. (2016, December 28). Solar Energy Used for Grid Connection: A Detailed Assessment Including Frequency Response and Algorithm Comparisons for an Energy Conversion System. Retrieved February 04, 2018, from http://ieeexplore.ieee.org/document/7801016/

[5] Liu, S. M. (2009, December 1). Design and Implementation of RGB LED Drivers for LCD Backlight Modules. Retrieved February 01, 2018, from http://ieeexplore.ieee.org/document/5166500/citations

# 7. Appendices