**EMERGENCY STATIONS**

**PROJECT EXPLAINED**

Version 1

Date: 11/6/2019

Document’s Purpose: To explain the innerworkings of the project to achieve the project’s vision. This is an extensive document elaborating the software and hardware of the project.

**MECHANICAL HARDWARE**

\*\*\*

This section needs some work and the CAD files for the mechanical hardware is missing!

\*\*\*

**ELECTRICAL HARDWARE**

A picture containing object

Description automatically generated

Figure 2.1: Electrical Hardware Diagram

This section deals with the electrical hardware of the system, consisting of the solar panels, solar charger controller, rechargeable 12V Lithium batteries, and the control system. The control system is composed of digital clock switch for the pole’s light, a microcontroller (Arduino) and the communication module (RockBLOCK).

The power obtained from the solar panel is processed and handle by the solar charger controller to properly charge the 12V batteries. Then the solar charger controller has a USB port that is used to run the Arduino and the RockBLOCK. Whenever the solar panel is not outputting enough voltage to power the solar charger, the voltage of the batteries ensure that the solar charger controller is powered and supplying power to the control systems.

**SOFTWARE**

A screenshot of a cell phone

Description automatically generated

Figure 3.1: Software Subsystem Diagram

The software subsystem of the project is composed of two main components: the emergency station software within the Arduino, and the Raspberry Pi Server. The weight within a barrel is recorded by three weight sensors. The weight values are summed and converted to a value in pounds (lbs.). Weight values are recorded every 90 minutes for a total of eight times. One cycle of data collection is 12 hours. Afterwards, these values are transmitted by the Communication Module (RockBLOCK) that is connected to the Arduino.

When the Communication Module transmits data, it is sent as an email where the subject contains the GPS location of the module and the weight data recorded by the Arduino. The Raspberry Pi Server is routinely checking for new emails from the ESs. If a new email is detected, the weight data within the new email is read, recorded, and saved to a master excel file within the gmail’s google drive. Additionally, the Raspberry Pi generates other excel files that update the GPS locations within the Google Maps plugin and the most up to date ESs’ inventory values within a Table plugin. These plugins are supplied by WordPress, which is the platform used to operate the website.

The latest 100 points from the data points of the ES is used to plot the changing weight and sensor values. This helps diagnose and troubleshoot possible problems for individual ES.