**Boston University**
**Electrical & Computer Engineering**
**EC464 Capstone Senior Design Project**

User's Manual

# Smart Watering with IoT

Submitted to

Mathworks

by

Team 29

Team Members
Kaede Kawata kkawata@bu.edu
Karin Luna kaluna@bu.edu
Nyah Madison nyahbm@bu.edu
Ines Saavedra Villafruela inessv@bu.edu
Jordan Remar jremar@bu.edu

Submitted: April 17, 2023

# Smart Watering with IoT

## Table of Contents



## Executive Summary


Our group's project aims to optimize water usage in crops for farmers by developing an IoT device. The device collects real-time data on soil moisture levels, temperature, humidity, and other environmental parameters to optimize irrigation schedules. The data is then analyzed by machine learning algorithms to determine the optimal irrigation schedule for each crop. The device, charged by solar panels, communicates with a cloud-based platform that provides farmers with insights into their crop's water usage and enables them to adjust irrigation schedules remotely. The device is designed to be low-cost, durable, sustainable, and easy to install, making it accessible to small-scale farmers. The project is expected to result in significant water savings, increased crop yields, and improved farmer livelihoods, while also contributing to sustainable agriculture and water conservation.

# 1 Introduction

Water is an essential resource for life and agriculture is one of the main sectors that depends on this resource. However, due to climate change, growing population and other factors, fresh water in the world is rapidly depleting. In many countries, agriculture is responsible for a large share of freshwater consumption and therefore efficient water management is crucial for the future of our planet.

Lack of fresh water has become an increasingly serious global problem. It is estimated that more than 2 billion people worldwide do not have access to safe drinking water and lack of access to water is affecting food security, health and economic development. In addition, climate change has further exacerbated the problem by increasing the frequency and intensity of droughts and floods, making efficient water management even more important.

Agriculture is one of the sectors most affected by water scarcity and, therefore, it is critical to develop solutions that enable efficient water use. Fortunately, IoT (Internet of Things) technology has advanced enough to allow the creation of smart devices that can collect and analyze data in real time.

Smart Watering with IoT is a project that uses IoT technology to collect data from soil pressure, humidity, temperature, soil moisture and soil ph sensors. This data is sent to an online platform where it is processed and analyzed to determine the optimal timing to irrigate crops in real time. This allows farmers to irrigate crops with the exact amount of water they need, avoiding over-irrigation and water wastage.

This project is a smart and efficient solution for water management in agriculture. The device is easy to install and can be used on any type of soil. It is also scalable and can be used on small and large farms. Farmers can access real-time data through a mobile app or online platform and make informed decisions about irrigating their crops.

The Smart Watering with IoT project is not only beneficial for the environment, but can also be beneficial for farmers' economics by saving on water costs and reducing overall production costs. In addition, by irrigating crops with the exact amount of water they need, better yields and healthier crops can be obtained.

This device can also be a valuable tool for precision agriculture. Precision farming is a farming technique that uses advanced technologies such as GPS, sensors and drones to collect data and make informed crop decisions. The Smart Watering with IoT project can be integrated into precision farming systems to further improve irrigation efficiency and water management.

In addition to efficient water management in agriculture, the Smart Watering with IoT project can also have a positive impact on the conservation of natural ecosystems. Excessive irrigation can have negative effects on soil quality and biodiversity, while efficient water use can help preserve these ecosystems.

The Smart Watering with IoT project can also be a valuable tool for education and awareness of efficient water management in agriculture. By providing real-time information on water consumption and efficient irrigation management, understanding and awareness of the importance of water and the need for its responsible use can be increased.

Moreover, it meets Mathworks' requirements of being an environmentally friendly, low-cost device made for small and medium-sized farms. What makes Smart Watering with IoT a sustainable system is that it is charged using solar energy. Solar panels are connected and feed the ESP32 and sensors. The ESP32 acts both as a micro controller and microprocessor, sending the data to ThingSpeak, which is the database used in this project. Once the data is uploaded to the cloud, there are some visualizations made to analyze the data and to make it user friendly, a React Native application has been developed so that the user can access the data in real time. . This app allows farmers to create users and view the data of their crops. This app also sends an alert when it is optimal to water the crops.

In conclusion, the Smart Watering with IoT project is a smart and efficient solution for water management in agriculture. IoT technology enables real-time data collection, allowing farmers to irrigate crops with the exact amount of water they need, avoiding over-irrigation and water wastage. This solution is not only beneficial for the environment, but can also be beneficial for the economy of farmers and the conservation of natural ecosystems. In addition, the project can be a valuable tool for education and awareness of efficient water management in agriculture. In summary, Smart Watering with IoT is a promising solution to address the problem of water scarcity in agriculture and contribute to the sustainability of our planet.

# 2 System Overview and Installation

## *2.1 Overview block diagram*

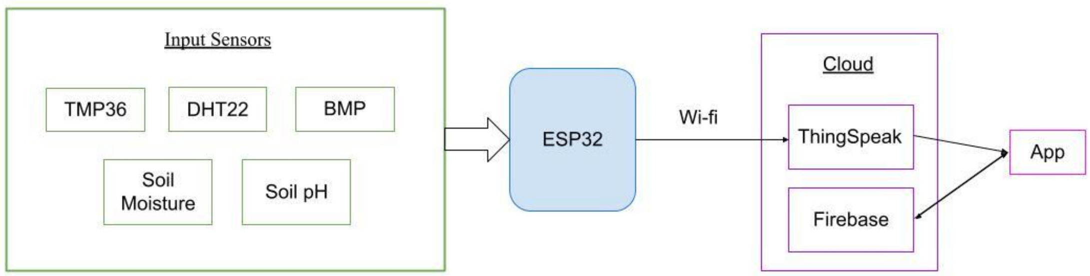
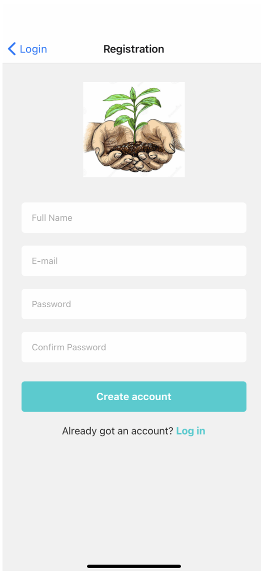


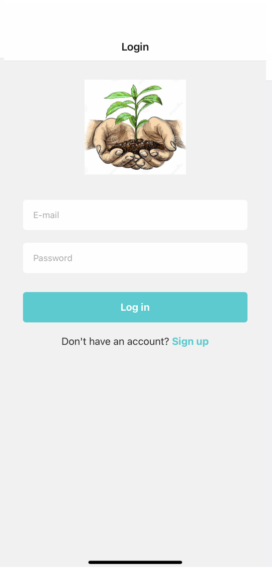

*Figure 2.1 Block diagram showing how data moves from sensors to the app.*

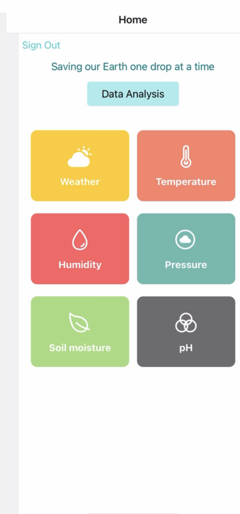
The input sensors measure needed data. BME 280 sensor measures the temperature, humidity, and pressure. The soil moisture sensor and soil pH sensor measure moisture and pH of the soil respectively. The microcontroller uses WiFi to send the collected data to the cloud, ThingSpeak. It will receive data, use it to visualize it and predict when to water the crops. Firebase is used to authenticate users and store their data. The app receives the prediction and offers explanations to the user about why the crops need watering at the time of prediction.

## *2.2 User interface.*

The user interface of the “Smart Watering with IoT” app is simple and easy to navigate. The first screen is the login screen as shown in Image 1 . User should input his/hers credentials and click on the Login button. If the user does not have an account, he/she should press “Sign Up”, enter the information and click on “Create an Account”as shown on Image 2. The home screen is in the form of six boxes (see Image 3). Clicking on the “Weather” box, the user will navigate to a screen where the weather of his/her location will be displayed. Clicking on the “Temperature” box, the user will see the temperature graph fetched from ThingSpeak. Below, a box where the user can select what time period he/she wants to see as shown on Image 4.That will be the same for Humidity, Pressure, Soil Moisture and Soil pH. The alerts will be shown whenever the data gathered does not belong to the optimal interval as we can see on Image 5. Lastly, if the user clicks on “Data Analysis” button, it will display what's shown on Image 6: a table with the optimal values and charts from ThingSpeak.



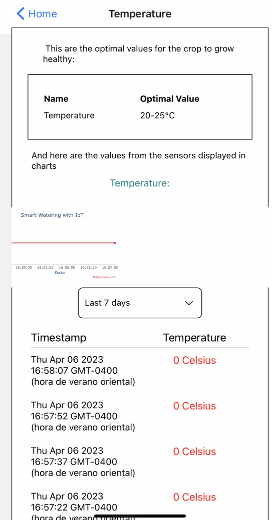
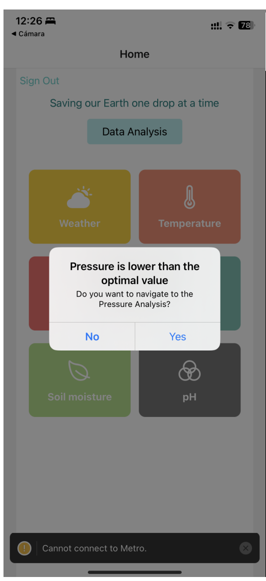




Image 1. Login Screen Image 2. SignUp Screen Image 3. HomeScreen Image 4. Temperature



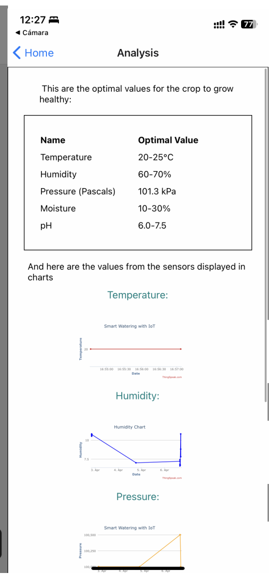
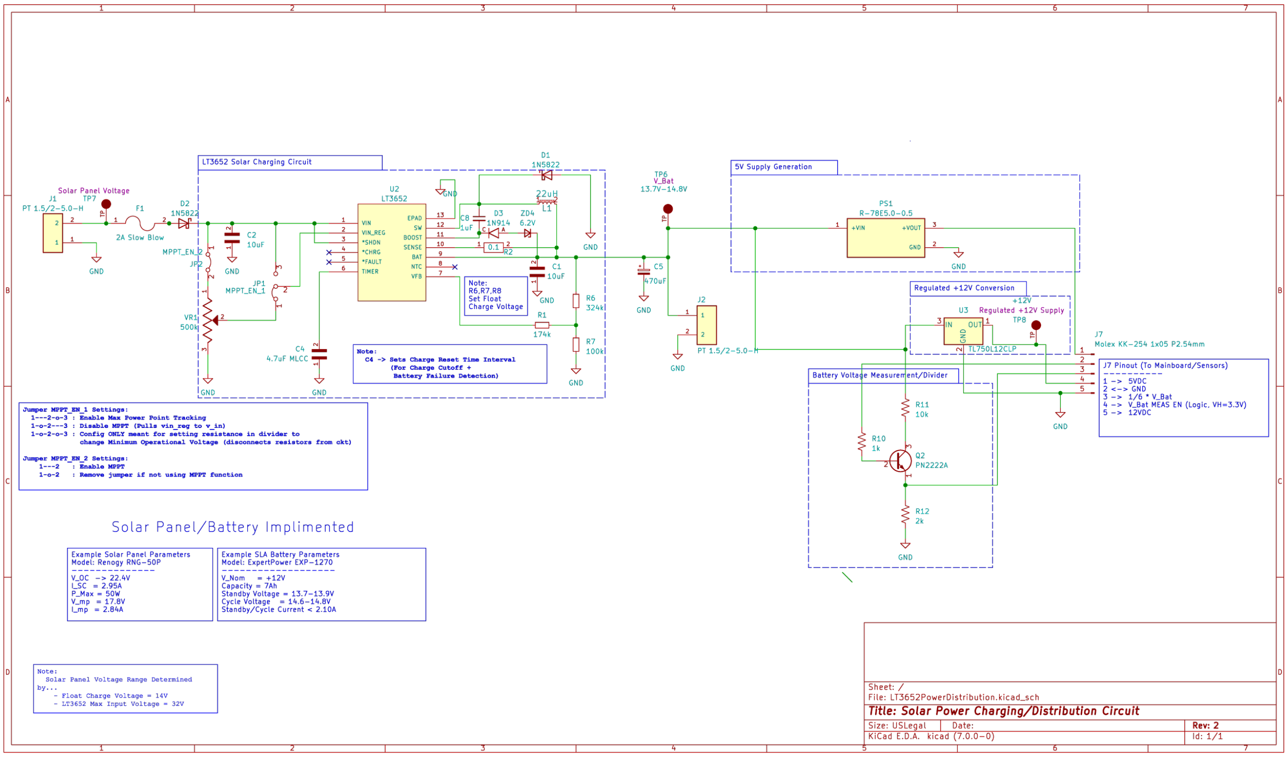
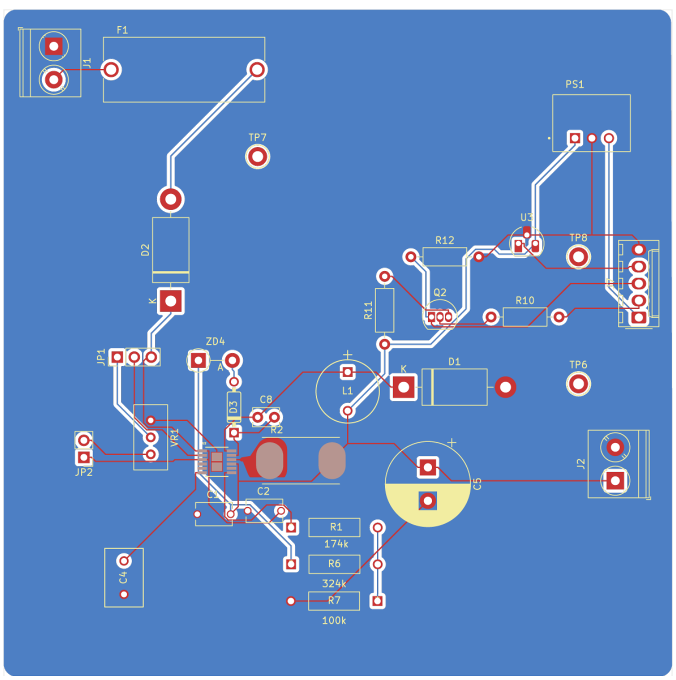




Image 5. Alert Image 6. Data Analysis

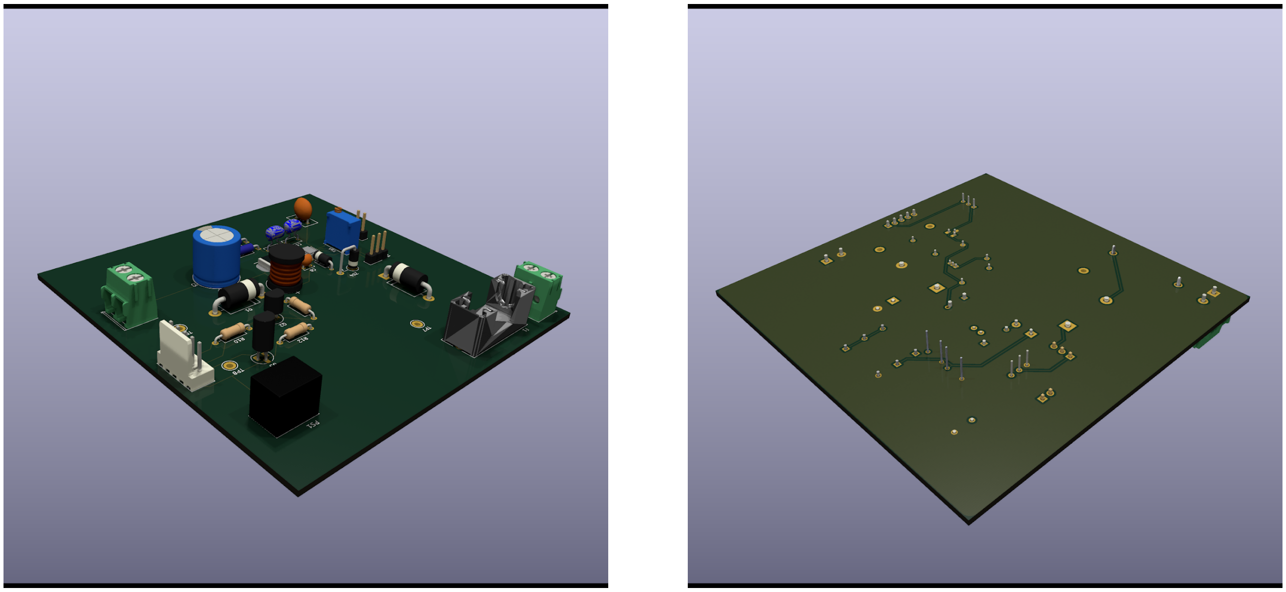
## *2.3 Physical description.*



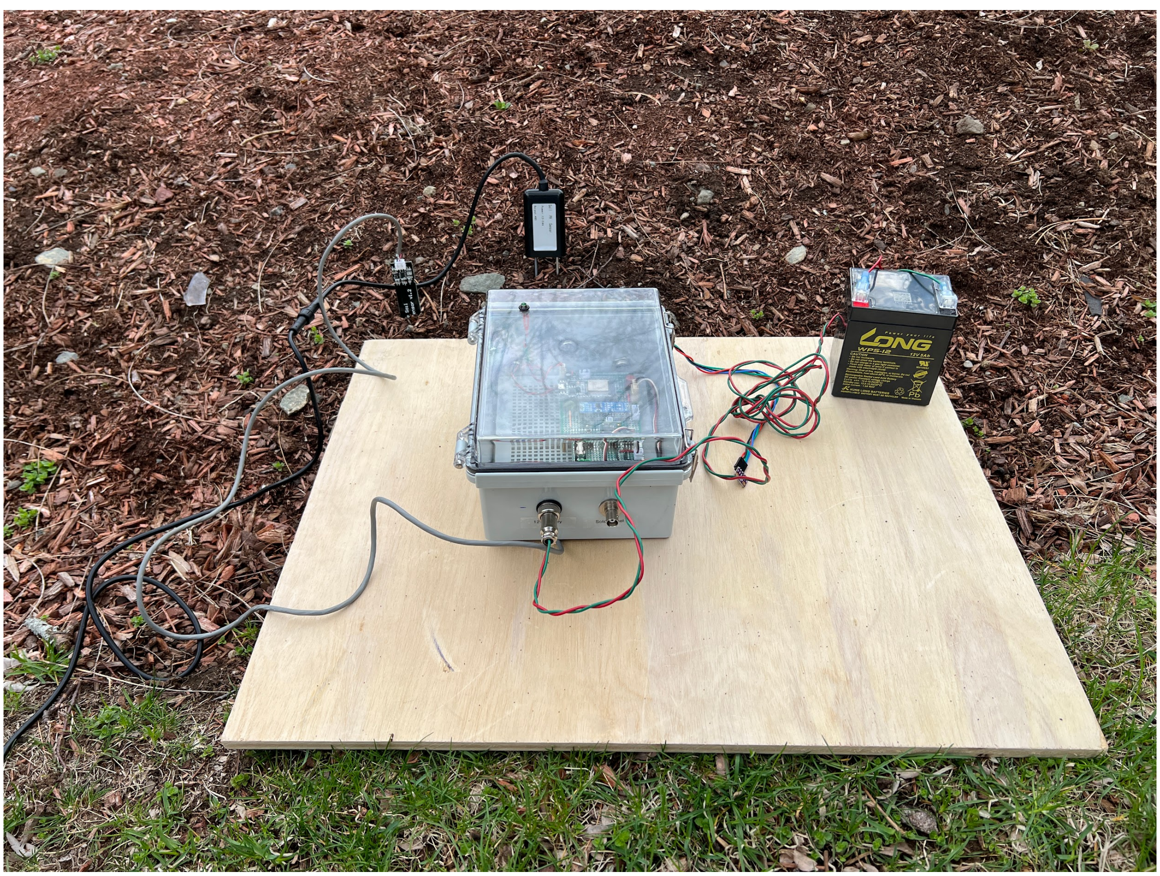
Solar Charging + Power Distribution Circuit Schematic

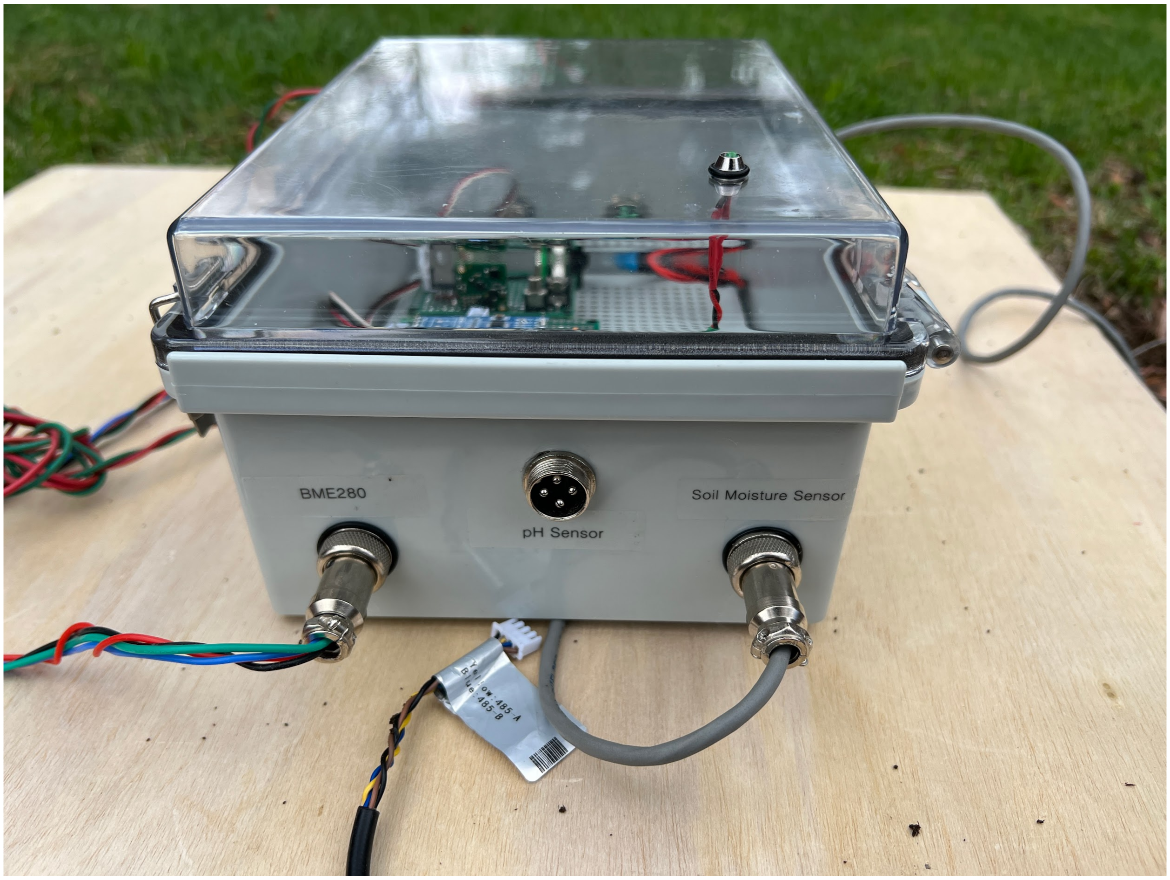


Board design of the Solar Charging + Power Distribution Circuit (Created using KiCAD 7.0.0)



3D-Rendering of PCB design for the Solar Charging + Power Distribution Circuit (generated using KiCAD 7.0.0)






Project in its weather-proof enclosure. Sensors are mounted outside of the enclosure and interface with the ESP32/Solar Charging + Power Distribution Circuit using GX16 style connectors. All connectors and indicator LEDs are sealed using O-Rings to prevent water ingress under normal operating conditions.

### *2.4 Installation, setup, and support*

For the hardware, customers will receive a single device. The device will be a weatherproof box containing all the necessary sensors. There is no need to attach any of the sensors onto the device as they are already attached. The device consists of five sensors. Three of the five sensors are ready for use. Two of the sensors, the soil moisture sensor and the pH sensor have leads that the user needs to insert into the ground near the plant.

The system is powered by a lead acid battery that comes with the system. No battery replacement is required as the battery is charged by a solar panel that also comes with the device. The solar panel can be placed on a stand near the rest of the device.Once powered, the device will start collecting data. To make sure data communication is working properly, the user will have to connect the device to wi-fi.

The device comes with an app that users must download. To use the app, the user will first create an account. This is done by clicking the “Create an Account” button on the landing page after opening the app. Once an account is created, the user can log in to see information about their crops.

# 3 Operation of the Project

## *3.1 Operating Mode 1: Normal Operation*

Under normal circumstances, the app works as follows:

1. Upon opening the app, the user will receive an alert if any environmental factors measured are sub-optimal.
2. The user should see different color buttons with labels corresponding to the five different types of information being collected. Click any button to open analytics about the type of data.
3. After clicking the button, a page will open showing the user a graph and a table with data collected over time.
4. Every so often, the app will notify the user when to water their crops.

All proper user interaction with the device is through the app.

## *3.2 Operating Mode 2: Abnormal Operations*

Data collection errors may occur if soil measurement sensors are not properly positioned in the soil. Make sure the soil moisture sensor and soil pH sensor are inserted into the soil at all times.

Wi-fi connection is needed to see information about crops on the app. If data fails to load, check to make sure the device is connected to the internet.

## *3.3 Safety Issues*

The battery used to power the Smart Watering System is a mix-chemistry battery. Use caution by making sure to always follow proper handling as noted on the battery. In the case of battery leaks, carefully remove the battery while wearing gloves and protective eyewear. Make sure to not touch any liquid that comes out of the battery. Clean the battery compartment. Store the battery in a plastic bag and dispose. Replace using the same battery type.

For program running on the microcontroller, because USB micro port is available, it is possible to overwrite the program. In this case, the original program will be lost and the system will not work properly. It might be possible to analyze the binary data on the microcontroller and reverse engineer the original program. However, the original program does not have any access to database that has authentication information, so the risk of getting cyberattacks through hardware is minimized.

# 4 Technical Background

## 4.1 *Sensors and Control System*

The system will be powered off of a +12V Sealed Lead Acid battery to supply the required +12V, +5V, and +3.3V. This circuit will also be capable of using various solar panels to recharge the battery, making use of the LT3652 IC. This IC allows for implementation of Mean Power Point Tracking (MPPT) to regulate the charging current such that the solar panel operates in its maximum power efficiency range. Since the peak power voltage, Vp, will vary between solar panels, this setting is adjustable through a 500kΩ trim potentiometer. If desired, this function can be disabled by using two jumpers. The LT3652 is configured to have a peak charging current of 1A, set by using a 0.1Ω shunt resistor connected to the SENSE pin of the LT3652. This current will allow for greater flexibility of the +12V battery being used.

The battery voltage needs to be regulated down to +12V, as the actual battery voltage will be ~13.2V to 14.2V depending on charge. This is accomplished using a 12V low dropout linear regulator (Using the TL750L12 in a TO-92 package). A linear voltage regulator would be unsuitable for the +5V generation, as the voltage difference would be dissipated as heat. Instead, a +5V step-down DC-DC converter IC (Recom R-78E5.0-0.5 SIP) is used to efficiently drop the voltage to +5V. Since the 5V goes on to power everything except the pH sensor, overcurrent protection is implemented using a socketed 1A radial fuse. The +5V rail will supply the MAX485 RS-485 transceiver with the needed +5V. Furthermore, the +5V supply will also power the ESP32 through the Vin pin. This way, the +3.3V supply is generated using the 3.3V LDO included on the ESP32 Development Board.

The hardware part of the Smart Watering System includes sensors connected to a microcontroller to collect environmental data. The microcontroller is programmed using C++ and PlatformIO with VS Code as an integrated development environment. Another essential component that microcontroller manages is pushing data to ThingSpeak. The microcontroller is connected to WiFi, and it follows HTTP and uses the Write API key for the ThingSpeak channel to make environmental data available in real time.

The microcontroller receives 5V from a battery, which is charged using solar cells. Sensors receive power from the microcontroller except for the soil pH sensor. This sensor receives 12V from its own power line and is connected to a converter before it connects to the pins of the microcontroller. The converter also receives 5V from the microcontroller. BME 280 sensor uses I2C protocol to send temperature, humidity, and pressure data to the microcontroller. The soil moisture sensor uses the analog reading and calculates the relative moisture percentage based on the analog reading when the sensor is in the dry air and in water.

## *4.2 ThingSpeak*

ThingSpeak is an open-source, web-based platform for the Internet of Things (IoT) that allows users to collect, store, and analyze sensor data from IoT devices.

One of the key features of ThingSpeak is its ability to integrate with a wide range of IoT devices and sensors. ThingSpeak supports a range of communication protocols, including MQTT, HTTP, and HTTPS, and can be used with devices that use Arduino, Raspberry Pi, and other microcontrollers.

To collect data from agricultural IoT sensors using ThingSpeak, the sensors are typically configured to send data to ThingSpeak's cloud-based servers using HTTP requests. This data can include sensor readings, such as temperature, humidity, and soil moisture levels, as well as other relevant information such as the date and time the data was collected.

Once the data has been collected by ThingSpeak, it can be stored in the platform's cloud-based database. ThingSpeak provides a range of tools and APIs for analyzing this data, including built-in visualizations and custom MATLAB scripts that can be used to perform complex data analysis.

ThingSpeak also provides a range of features for sharing and collaborating on data. Users can create public channels, which allow them to share their data with others and collaborate on data analysis projects. ThingSpeak also supports real-time data streaming, which allows users to monitor sensor data in real-time and receive alerts when certain conditions are met.

In conclusion, ThingSpeak is a powerful platform that is collecting, storing, and analyzing the data from our agricultural IoT sensors. Its ability to integrate with a wide range of devices and sensors, as well as its built-in data analysis tools, make it an ideal choice for farmers and agricultural workers looking to optimize crop growth and reduce water usage. With ThingSpeak, farmers can gain valuable insights into their crops, allowing them to make data-driven decisions and improve the overall health and yield of their crops.

## *4.3 React-Native App*

Smart Watering with IoT is a mobile application built using React Native framework. It is designed to collect data from sensors installed in an agricultural environment and send alerts to the user on when it is optimal to irrigate crops. The app is built with the aim of automating and improving the irrigation process in agriculture, thereby reducing water wastage and increasing crop yields. In this technical background it will be discussed the architecture of the app, the technologies used, and the implementation of the Smart Watering with IoT.

The architecture of the Smart Watering with IoT app is divided into two parts: the client-side and the server-side. The client-side is the mobile application that runs on the user's smartphone, while the server-side is the cloud-based backend that stores and processes the data collected from the sensors.The client-side of the app is built using the React Native framework, which allows the app to run on both iOS and Android platforms. The app uses a variety of libraries, including Axios for making HTTP requests, and React Navigation for handling navigation between screens.

The server-side of the app is built using two cloud-based services: ThingSpeak and Firebase. ThingSpeak is a platform for IoT data collection and analytics, while Firebase is a cloud-based platform for building mobile and web applications. ThingSpeak is used to collect data from the sensors and Firebase is used to create user accounts.

The Smart Watering with IoT app uses a variety of technologies to collect data from sensors and send alerts to the user. The IoT sensors collect data on environmental factors such as temperature, humidity, and soil moisture, and send this data to ThingSpeak using HTTP requests. The app uses ThingSpeak's REST API to read and write data to and from the ThingSpeak platform.

To read data from ThingSpeak, the app sends an HTTP GET request to ThingSpeak's API endpoint, including the API read key in the request URL. ThingSpeak then returns the requested data in a JSON format, which the app parses and displays to the user.

To write data to ThingSpeak, the app sends an HTTP POST request to ThingSpeak's API endpoint, including the API write key in the request URL and the data to be written in the request body. ThingSpeak then stores the data in its cloud-based database.

The app uses Firebase for user authentication, as previously described. The app would use Firebase's Authentication API to create and manage user accounts.

# 5 Relevant Engineering Standards

Our Senior Design Project is focused on developing a smart device for watering plants that collects agricultural data using IoT sensors and uploads it to ThingSpeak. The data is then displayed in a React Native application that provides real-time updates and recommendations for when it is optimal to water the plants.

As our project involves the development of a device that will be used in the agricultural industry, it is critical that we adhere to relevant engineering standards to ensure its safety, reliability and functionality. Failure to follow these standards could result in negative impact on plant growth, loss of crops, and damage to the environment.

1. IEEE 802.15.4 - This standard ensures that our IoT sensors and devices use a low-power wireless network that is suitable for agricultural environments.

2. ISO 25119 - This standard outlines the functional safety requirements for agricultural machinery, which will be used to guide the development of our device.

3. General Data Protection Regulation (GDPR) - Since the project is going to be available in Europe or will collect data from European citizens, we adhere to GDPR requirements. This regulation provides guidelines on how to collect, store, process, and protect user data, including the right to access, rectify, or erase personal data.

4. Internet Engineering Task Force (IETF) standards - These standards define protocols for smooth data collection, transfer, and processing over the internet, including HTTP, REST, and WebSockets. Following these standards can ensure that the app can handle different types of data and devices efficiently.

5. User-Centered Design (UCD) guidelines - These guidelines focus on designing apps that are easy to use, accessible, and meet user needs. Following UCD principles can help ensure that the app provides a smooth and intuitive user experience for farmers, making it easier for them to access and interpret their agricultural data.

6. Wi-Fi Alliance standards - Wi-Fi Alliance is an organization that defines and certifies wireless networking standards, including Wi-Fi Protected Access (WPA) and Wi-Fi Direct. These standards ensure secure and reliable wireless communication between devices and access points.

# 6 Cost Breakdown

| Project Costs for Production of Beta Version (Next Unit after Prototype) | | | | |
|---|---|---|---|---|
| Item | Quantity | Description | Unit Cost | Extended Cost |
| LT3652EMSE | 1 | IC BATT CHG MULTI-CHEM 12MSOP | $9.23 | $9.23 |
| 20W 20V Solar Panel | 1 | 20W, 20V Photovoltaic Panel | $39.39 | $39.39 |
| ESP32-Devkit V1 | 1 | ESP32 30-pin Development Board | $6.00 | $6.00 |
| LoRa Module | 2 | Required for communication between microcontrollers | $19.5 | $19.5 |
| Soil Moisture Sensor (SEN-13322) | 1 | Measures soil moisture | $9.99 | $9.99 |
| Soil pH Sensor | 1 | Sensor for the system | $62 | $62 |
| Router | 1 | Required to create local network | $33.25 | $33.25 |
| ESP32 camera | 1 | Microcontroller for the system | $8.99 | $8.99 |
| PCB board kit | 2 | Required to create PCB board | $13.49 | $13.49 |
| MAX485 TTL to RS485 Converter Module | 1 | Converts RS-485 Protocol to TTL for communication with the ESP32 | $9.79 | $9.79 |
| BME 280 Breakout Board | 1 | Measures pressure, temperature and humidity | $9.99 | $9.99 |
| R-78E5.0-0.5 | 1 | 5V DC-DC Step-Down converter for converting +12V to +5V | $3.25 | $3.25 |
| TL750L12CLP | 1 | +12V Low Drop-Out Linear Regulator used for regulating the battery voltage | $0.82 | $0.82 |
| PT 1.5/2-5.0-H | 2 | 2 Position Wire to Board Terminal Block | $0.54 | $1.08 |
| Weather-Proof Junction Box | 1 | Weather-Proof enclosure to house main electronics (excluding sensors) | $27.99 | $27.99 |

# 7 Appendices

## *7.1 Appendix A - Specifications*

| **Requirements** | **Value, range, tolerance, units** |
|---|---|
| Case dimensions (LxWxH) | 220mm x 170mm x 110mm |
| Power supply | +12V DC, +12V SLA Battery (~13.2V to 14.2V) |
| Energy use | 155mA +/- 20mA @ 12VDC (1.62W - 2.1W) |
| Microcontroller | ESP32 |
| User Interaction | App receives data in real time. User has access to all data gathered. |
| Practical | The overall cost of the device should be <100$ |

## *7.2 Appendix B – Team Information*

Team 29, or Smart Watering with IoT, consists of Kaede Kawata, Nyah Madison, Karin Aimee Luna, Jordan Remar and Inés Saaavedra.

Kaede Kawata is a senior Computer Engineering student. She has a passion for many different subjects including IoT devices, decentralized networks, and decentralized machine learning. She hopes to use her skills as an engineer to contribute to creating a safe and secure smart city. She will pursue embedded engineering at a manufacturer company of consumer electronics products.

Nyah Madison is a senior Computer Engineering student. She enjoys learning about digital signals processing and is open to learning more about the topic in different applications, including image processing, audio processing, and telecommunications. She is excited to spend this summer learning more about signals processing through an internship at a wireless security company.

Karin Aimee Luna is a senior Computer Engineering student. She is passionate about learning and being exposed to new fields, from cloud computing to AI. She hopes to use her skills as a societal engineer to innovate sustainable technology and address global issues. She will pursue software engineering at a nonprofit for people-powered change in politics.

Jordan Remar is a Senior Electrical Engineer. He is particularly passionate about analog electronics, and loves to try and solve problems using a hardware-based approach. In his free time he loves to repair broken test equipment to see how various functions have been implemented in the circuit, and how different approaches compare to one another. He hopes to find a job where he can design circuits that will either further research/exploration, or will help with nature preservation.

Inés Saavedra is a Senior Exchange Student from Spain that will be graduating Computer Engineering and Business Analytics. She has a passion for many different subjects including data analytics, IoT devices, smart electric power and much more. She hopes to use his skills as an engineer to work on solving more engineering problems related to data analysis and network communication.