Table Of Contents

Acknowledgement

I would like to express my heartfelt gratitude to all those who have contributed to the successful completion of this mini project report on the solar-powered smart irrigation system.

First and foremost, I am immensely grateful to my project guide, **Prof. Shree Gowri S. S.,** for their invaluable guidance, constant support, and expert advice throughout the project. Their vast knowledge and experience in the field of smart agriculture have been instrumental in shaping the direction and scope of this project.

I would also like to extend my sincere thanks to **Karnatak Law Society Vishwanathrao Deshpande Institute of Technology** for providing the necessary resources and infrastructure that facilitated the smooth execution of this project. The access to the library, research materials, and laboratory facilities greatly enhanced the quality of research and analysis conducted.

I would like to acknowledge and thank all the individuals and organizations who have directly or indirectly contributed to the successful completion of this mini project. Your support and contributions have played a pivotal role in the realization of this project.

Abstract

The solar-powered smart irrigation system is an innovative solution that combines renewable energy, advanced sensing technologies, and intelligent algorithms to optimize irrigation processes in agriculture. This project aimed to design and implement a sustainable and efficient system that addresses the challenges of traditional irrigation methods while minimizing energy consumption and environmental impact.

The project focused on harnessing solar power as a renewable energy source to drive the irrigation system. Solar panels were employed to generate electricity, which powered sensors, actuators, and communication devices. This allowed for real-time monitoring of crucial environmental parameters such as soil moisture, temperature, and humidity.

The collected data was then processed and analyzed using data analytics and intelligent algorithms. This provided farmers with actionable insights and decision support, enabling them to make informed and precise irrigation decisions. By automating the irrigation process, water usage was optimized, and crop health was improved.

Introduction

Agriculture plays a vital role in ensuring food security and supporting the growing global population. However, traditional agricultural practices often suffer from inefficiencies, resource wastage, and environmental impact. In recent years, there has been an increasing need to develop sustainable and technologically advanced solutions that address these challenges and optimize agricultural processes. One such solution is the solar-powered smart irrigation system.

The solar-powered smart irrigation system combines renewable energy, advanced sensing technologies, and intelligent algorithms to revolutionize irrigation practices in agriculture. By harnessing the power of solar energy, this system offers a sustainable and environmentally friendly alternative to conventional irrigation methods that rely on non-renewable energy sources.

Irrigation is a critical component of agricultural operations, as it directly affects crop yield, water usage, and resource management. Traditional irrigation practices often involve manual monitoring, which can be time-consuming, inefficient, and prone to human error. Additionally, excessive or inadequate water supply can lead to water wastage, decreased crop productivity, and negative environmental consequences.

The solar-powered smart irrigation system addresses these issues by integrating advanced sensors that continuously monitor environmental parameters such as soil moisture, temperature, and humidity. These sensors provide real-time data, enabling farmers to make informed and precise decisions regarding irrigation. By automating the irrigation process, water usage can be optimized based on the specific needs of the crops, leading to improved water efficiency and enhanced crop health.

Components Description

1. Capacitive Soil Moisture Sensor

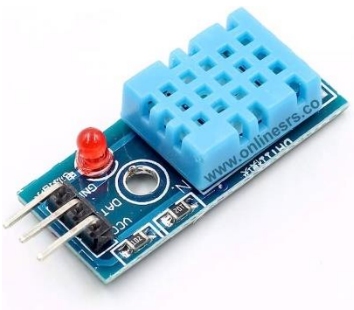
This is an analog capacitive soil moisture sensor which measures soil moisture levels by capacitive sensing. This means the capacitance is varied on the basis of water content present in the soil. You can convert the capacitance into voltage level basically from 1.2V minimum to 3.0V maximum. The advantage of Capacitive Soil Moisture Sensor is that they are made of a corrosion-resistant material giving it a long service life.

Features & Specifications

1. Supports 3-Pin Sensor interface
2. Analog output
3. Operating Voltage: DC 3.3-5.5V
4. Output Voltage: DC 0-3.0V
5. Interface: PH2.0-3P
6. Size: 99x16mm/3.9×0.63″

2. DHT11 Humidity Temperature Sensor

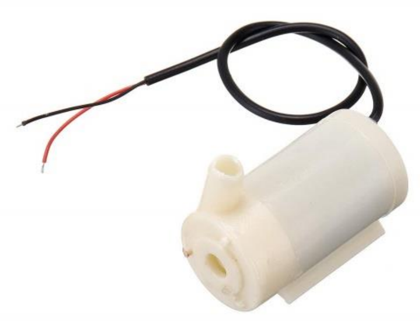
The DHT11 is a basic, ultra low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air. It spits out a digital signal on the data pin



It’s fairly simple to use, but requires careful timing to grab data. The only real downside of this sensor is you can only get new data from it once every 2 seconds. So when using the library, sensor readings can be up to 2 seconds old. In this project, we will use this sensor to measure the air temperature and humidity.

3. DC 3-6V Micro Submersible Mini Water Pump

The DC 3-6 V Mini Micro Submersible Water Pump is a low cost, small size Submersible Pump Motor. It operates from a 2.5 ~ 6V power supply. It can take up to 120 liters per hour with a very low current consumption of 220mA. Just connect the tube pipe to the motor outlet, submerge it in water, and power it.



Features & Specifications

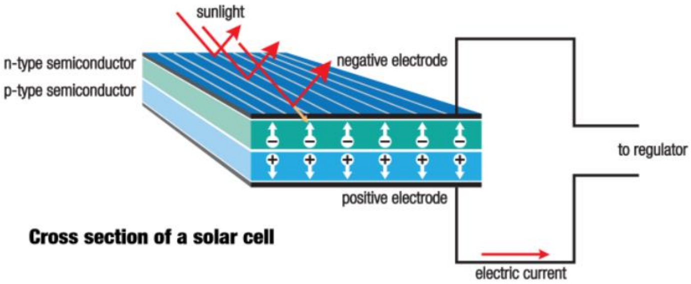
1. Operating Voltage : 2.5 ~ 6V
2. Operating Current : 130 ~ 220mA
3. Flow Rate : 80 ~ 120 L/H
4. Maximum Lift : 40 ~ 110 mm
5. Outlet Outside Diameter: 7.5 mm
6. Outlet Inside Diameter: 5 mm

4. Node-MCU ESP8266

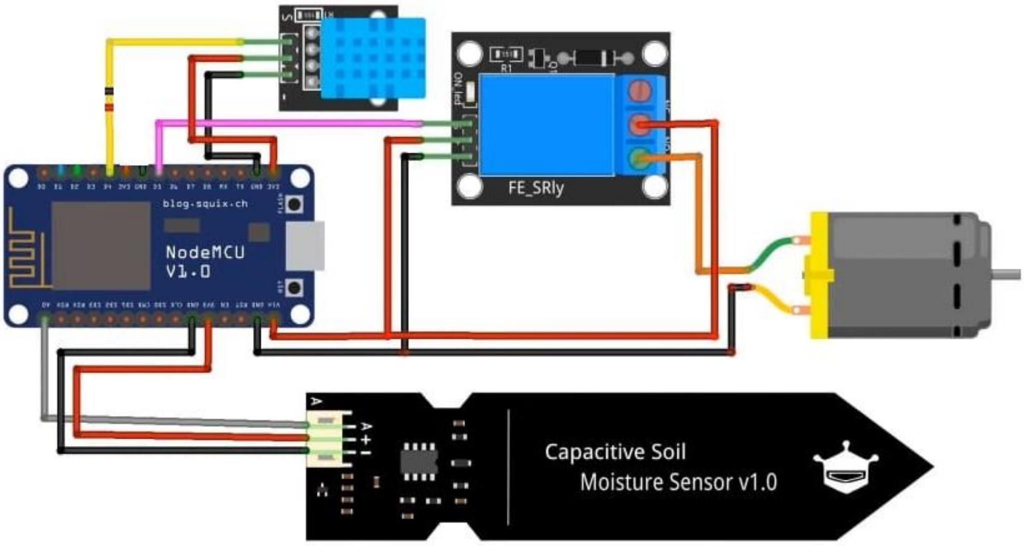
ESP8266 is a cost effective and optimal power consumption Wi-Fi microchip with full TCP/IP stack and 32-bit microcontroller capability which allows microcontroller to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. It operated on a power supply of +- 3.3 V and is programmed using Lua Script. With chip clock rate acceleration and an ADC (Analog to Digital Converter) to improve the sensitivity, integrating with soil humidity sensor is a common application.

5. Solar Panel

Whenever the sun light hits on the solar panel with photons (particles of sunlight), the panel converts those photons into electrons of direct current (DC) electricity. Naturally, the sunnier it is, the more energy is produced by the panels. Solar energy is a resource that is not only sustainable for energy consumption, it is indefinitely renewable. Solar power can be used to generate electricity, it is also used in relatively simple technology to heat water.



Circuit Diagram



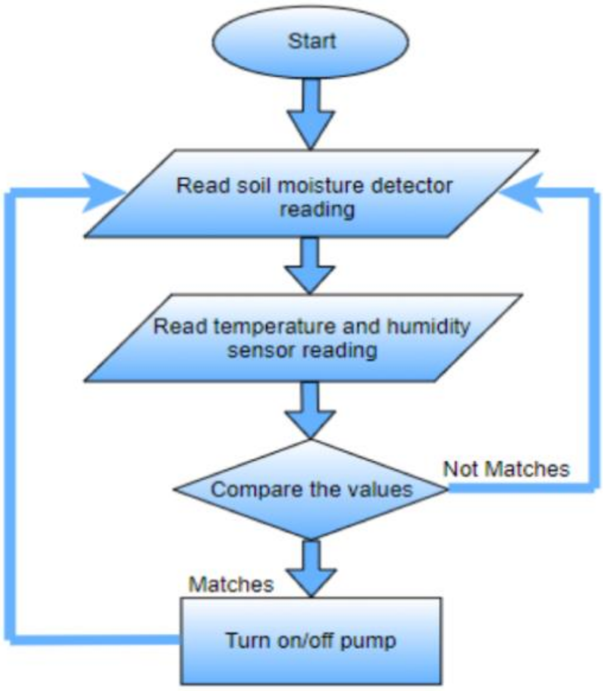
Working Process

1. **Solar Energy Generation:** The system utilizes solar panels to harness solar energy. These panels convert sunlight into electricity, providing a renewable and sustainable power source for the entire system.
2. **Sensor Deployment:** Advanced sensors are strategically deployed in the agricultural field to monitor key environmental parameters such as soil moisture, temperature, humidity, and light intensity. These sensors continuously collect real-time data, providing insights into the crop's water requirements and overall health.
3. **Data Collection and Transmission:** The sensors transmit the collected data wirelessly to a central control unit or a cloud-based server for further analysis. The data transmission can be facilitated through wireless communication technologies such as Wi-Fi, Zigbee, or LoRaWAN.
4. **Data Analysis and Decision Support:** The collected data is processed using data analytics techniques and intelligent algorithms. These algorithms analyze the data, considering various factors such as crop type, growth stage, and weather conditions. Based on the analysis, the system generates actionable insights and recommendations for optimized irrigation schedules and water usage.
5. **Automation of Irrigation:** The system is equipped with actuators and control mechanisms that automate the irrigation process. Based on the recommendations from the decision support system, the actuators regulate the

flow of water to specific areas of the field, ensuring that crops receive the right amount of water at the right time.

6. **Monitoring and Feedback:** Throughout the irrigation process, the system continues to monitor the environmental parameters using the deployed sensors. This real-time monitoring allows for continuous adjustment and optimization of irrigation schedules based on changing conditions.
7. **User Interface and Control:** The system provides a user-friendly interface, typically accessible through a web or mobile application. This interface allows farmers or users to monitor the system's performance, view data analytics, and manually control irrigation operations if necessary.
8. **Maintenance and Troubleshooting:** Regular maintenance of the system, including cleaning and inspection of solar panels, sensors, and actuators, ensures its proper functioning. In case of any issues or anomalies, the system should have mechanisms for troubleshooting and alerting the user about potential problems.

Flow Chart



Algorithm

1. Here we have taken random variable sensor which will store the value of the moisture content of the soil.
2. When the soil moisture sensor receives the value of 1023 that means that the soil is 0% moist but since the real conditions are not ideal, we will take the value of 800. That is soil is dry if the value received by the sensor is more than 800 and if the value received is less than 800 that means the soil has moisture content and the pump will stop.
3. A loop function is then applied such that the values received from the sensor is than stored in a new variable Analog Read.
4. Furthermore, if the value we receive from the sensor is high, i.e., The display will show ‘Pump ON’ if value is higher than 800 and the water pump will start. And if the value we receive is less than 800 then display will show ‘Pump OFF’ and no water will be supplied.
5. The above presented code can be modified for different crops according to their water requirements

Code

```
#include <DHT.h>
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ThingSpeak.h>
#define MOISTURE_PIN A0    // Analog pin for moisture sensor
#define PUMP_PIN 14        // Digital pin for water pump
const int moistureThreshold = 500;  // Moisture threshold value
#define DHTPIN 2   // Pin where the DHT11 is connected
#define DHTTYPE DHT11   // DHT sensor type
DHT dht(DHTPIN, DHTTYPE);
const char* ssid = "Redmi K50i";              // Replace with your network credentials
const char* password = "kartik99";
const char* server = "api.thingspeak.com";
const unsigned int httpPort = 80;
const String apiKey = "W21CX2771FGEEC8N"; // Replace with your ThingSpeak
API Key
WiFiClient client;
const int AirValue = 790;   // You need to replace this value with Value_1
const int WaterValue = 390; // You need to replace this value with Value_2
const int SensorPin = A0;
int soilMoistureValue = 0;
int soilMoisturePercent = 0;
void setup() {
  Serial.begin(9600);
  pinMode(PUMP_PIN, OUTPUT);
  digitalWrite(PUMP_PIN, LOW);  // Initialize the pump as off
  dht.begin();
  connectWiFi();
}
void loop() {
  int moistureValue = analogRead(MOISTURE_PIN);
  Serial.print("Moisture level: ");
```

```
  Serial.println(moistureValue);  // dry soil  = maximum moistureValue
  if (moistureValue > moistureThreshold) {
    digitalWrite(PUMP_PIN, LOW);  // Turn on the water pump
    Serial.println("Water Pump ON");
  } else {
    digitalWrite(PUMP_PIN, HIGH);   // Turn off the water pump
    Serial.println("Water Pump OFF");
  }
  delay(1000);
  float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();
  Serial.print("Humidity: ");
  Serial.print(humidity);
  Serial.print(" %\t");
  Serial.print("Temperature: ");
  Serial.print(temperature);
  Serial.println(" °C");
  soilMoistureValue = analogRead(SensorPin);
  soilMoisturePercent = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
  Serial.print("Soil Moisture: ");
  Serial.println(soilMoistureValue);
  sendToThingSpeak(temperature, humidity, soilMoistureValue, soilMoisturePercent);
}
void connectWiFi() {
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();
  Serial.println("Connected to WiFi!");
```

```
  Serial.print("IP Address: ");
  Serial.println(WiFi.localIP());
}
void sendToThingSpeak(float temperature, float humidity, int soilMoistureValue, int soilMoisturePercent) {
  Serial.println("Sending data to ThingSpeak...");
  if (client.connect(server, httpPort)) {
    String url = "/update?api_key=" + apiKey +
                 "&field1=" + String(temperature) +
                 "&field2=" + String(humidity) +
                 "&field3=" + String(soilMoistureValue) +
                 "&field4=" + String(soilMoisturePercent);
client.print(String("GET ") + url + " HTTP/1.1\r\n" +
                 "Host: " + server + "\r\n" +
                 "Connection: close\r\n\r\n");

    Serial.println("Data sent!");
  } else {
    Serial.println("Failed to connect to ThingSpeak");
  }
  delay(1000);
  client.stop();
}
```

IOT Interface (Thingspeak)

ThingSpeak is an open-source Internet of Things (IoT) platform that allows users to collect, analyze, and visualize data from various connected devices. It provides an easy-to-use interface and a set of APIs for data collection, storage, and retrieval, making it ideal for building IoT applications and projects.

Here's an overview of the key features and components of ThingSpeak:

Data Collection: ThingSpeak enables users to collect data from sensors, devices, and other sources. It supports a wide range of IoT protocols such as MQTT, HTTP, and TCP/IP, allowing seamless integration with diverse devices and systems.

Channel Structure: The core concept in ThingSpeak is a "channel." A channel represents a collection of data streams, where each stream corresponds to a specific data field. Users can define and customize the fields to suit their specific project requirements.

Data Storage: ThingSpeak provides cloud-based storage for the collected data. It stores the data in a time-series format, allowing users to access and analyze historical data easily. The data storage capacity depends on the user's subscription plan, ranging from free accounts to commercial plans with expanded storage capabilities.

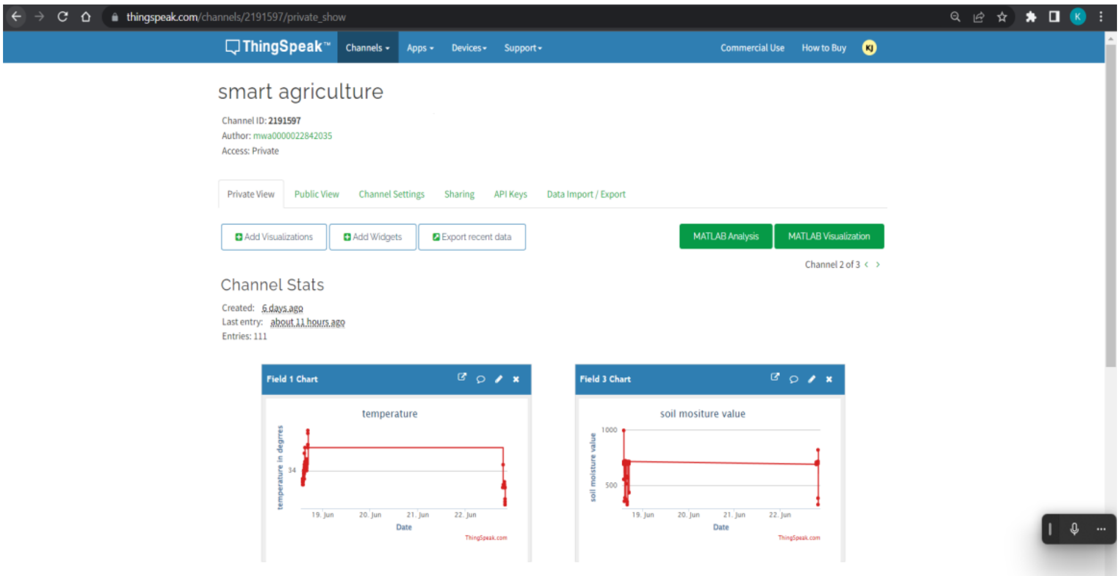
Data Visualization: ThingSpeak offers built-in visualization tools that allow users to create customizable charts, graphs, and gauges to visualize their data. These visualizations can be embedded in webpages or shared with others for monitoring and analysis.

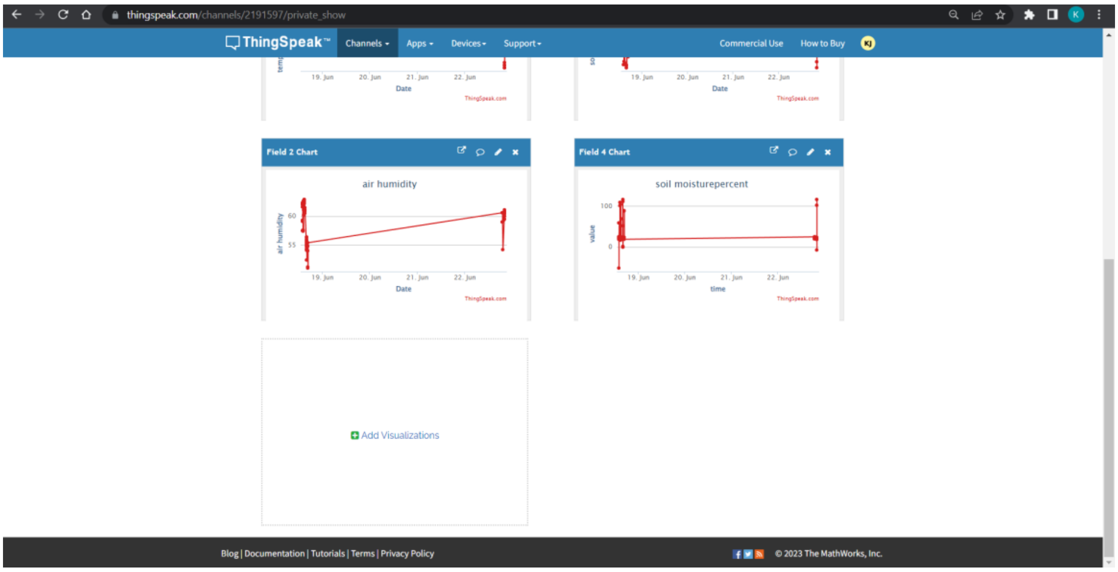
IoT Integrations: ThingSpeak provides integration with various IoT platforms and services, allowing users to connect and share data with other systems easily. This includes integration with popular IoT platforms like Arduino, Raspberry Pi, and Particle, enabling seamless communication and data exchange.

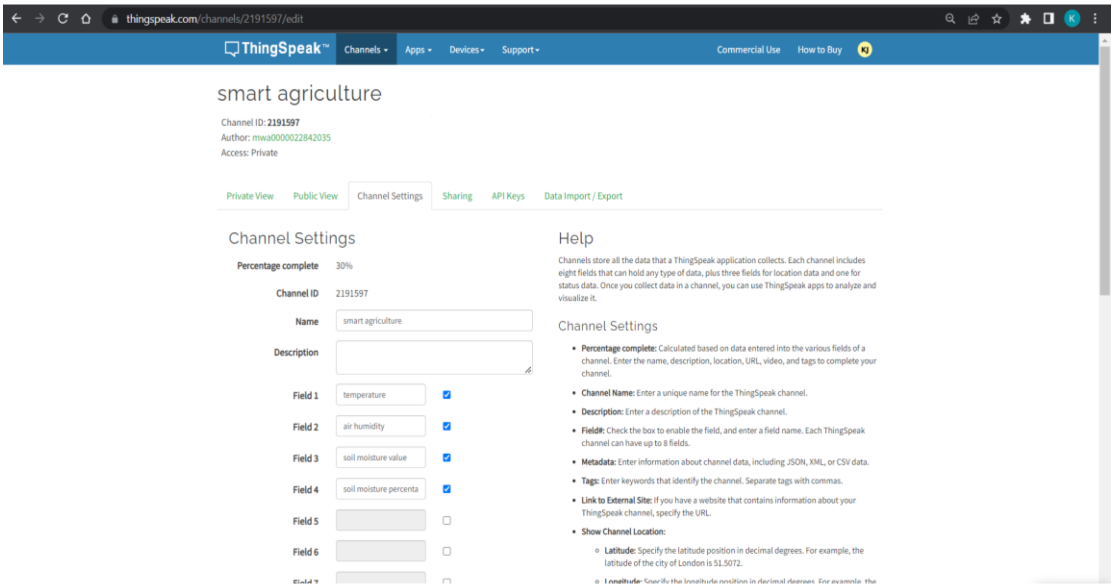
User Access Control: ThingSpeak offers user management features, allowing project owners to control access and permissions for different users. This ensures data security and privacy by restricting access to authorized individuals or teams.

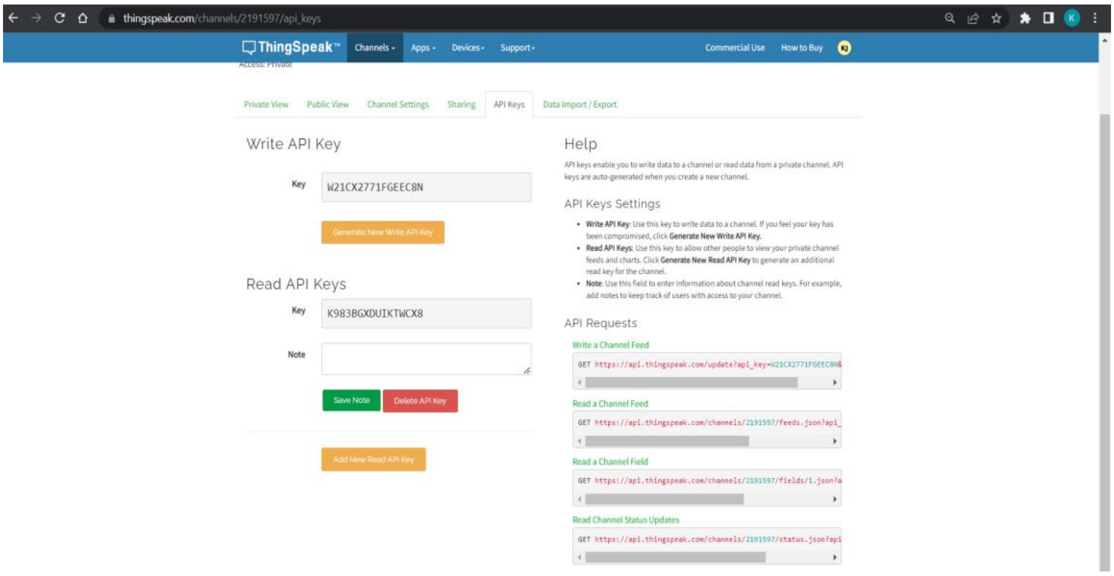
APIs and Webhooks: ThingSpeak provides APIs and webhooks that allow users to interact with their data programmatically. These APIs enable seamless integration with external systems, automation of processes, and real-time data retrieval.

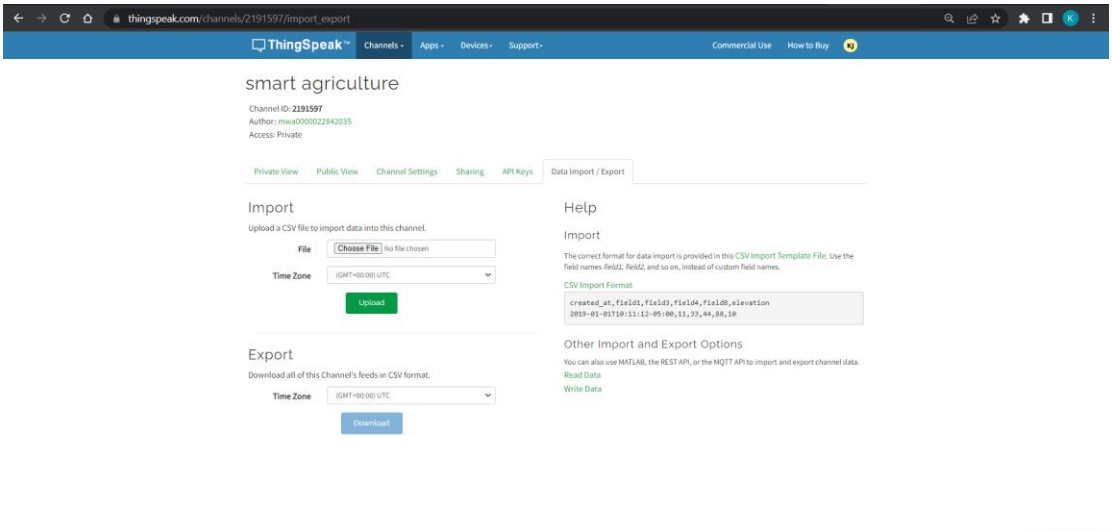
ThingSpeak is widely used by IoT enthusiasts, researchers, and developers for various applications such as environmental monitoring, smart agriculture, home automation, and industrial IoT. Its user-friendly interface, extensive documentation, and community support make it a popular choice for IoT projects of all scales.






thingspeak.com/channels/2191597/edit
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
smart agriculture
Channel ID: 2191597
Author: mwa0000022842035
Access: Private
Private View
Public View
Channel Settings
Sharing
API Keys
Data Import / Export
Channel Settings
Percentage complete 30%
Channel ID 2191597
Name smart agriculture
Description
Field 1 temperature
Field 2 air humidity
Field 3 soil moisture value
Field 4 soil moisture percenta
Field 5
Field 6
Help
Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.
Channel Settings
Percentage complete: Calculated based on data entered into the various fields of a channel. Enter the name, description, location, URL, video, and tags to complete your channel.
Channel Name: Enter a unique name for the ThingSpeak channel.
Description: Enter a description of the ThingSpeak channel.
Field#: Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
Metadata: Enter information about channel data, including JSON, XML, or CSV data.
Tags: Enter keywords that identify the channel. Separate tags with commas.
Link to External Site: If you have a website that contains information about your ThingSpeak channel, specify the URL.
Show Channel Location:
Latitude: Specify the latitude position in decimal degrees. For example, the latitude of the city of London is 51.5072.


thingspeak.com/channels/2191597/api_keys
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
Private View
Public View
Channel Settings
Sharing
API Keys
Data Import / Export
Write API Key
Key W21CX2771FGEEC8N
Generate New Write API Key
Read API Keys
Key K983BGXDUIKTWCX8
Note
Save Note
Delete API Key
Add New Read API Key
Help
API keys enable you to write data to a channel or read data from a private channel. API keys are auto-generated when you create a new channel.
API Keys Settings
Write API Key: Use this key to write data to a channel. If you feel your key has been compromised, click Generate New Write API Key.
Read API Keys: Use this key to allow other people to view your private channel feeds and charts. Click Generate New Read API Key to generate an additional read key for the channel.
Note: Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.
API Requests
Write a Channel Feed
GET https://api.thingspeak.com/update?api_key=W21CX2771FGEEC8N
Read a Channel Feed
GET https://api.thingspeak.com/channels/2191597/feeds.json?api_
Read a Channel Field
GET https://api.thingspeak.com/channels/2191597/fields/1.json?a
Read Channel Status Updates
GET https://api.thingspeak.com/channels/2191597/status.json?api


thingspeak.com/channels/2191597/import_export
ThingSpeak™
Channels
Apps
Devices
Support
Commercial Use
How to Buy
smart agriculture
Channel ID: 2191597
Author: mwa0000022842035
Access: Private
Private View
Public View
Channel Settings
Sharing
API Keys
Data Import / Export
Import
Upload a CSV file to import data into this channel.
File Choose File No file chosen
Time Zone (GMT+00:00) UTC
Upload
Export
Download all of this Channel's feeds in CSV format.
Time Zone (GMT+00:00) UTC
Download
Help
Import
The correct format for data import is provided in this CSV Import Template File. Use the field names field1, field2, and so on, instead of custom field names.
CSV Import Format
created_at,field1,field3,field4,field8,elevation
2019-01-01T10:11:12-05:00,11,33,44,88,10
Other Import and Export Options
You can also use MATLAB, the REST API, or the MQTT API to import and export channel data.
Read Data
Write Data

Objective of the Project:

The objective of the solar-powered smart irrigation system project is to design, develop, and implement an innovative and sustainable solution for optimizing irrigation practices in agriculture. The project aims to achieve the following objectives:

1. Integration of Renewable Energy: The project seeks to harness solar energy as a renewable and sustainable power source for the irrigation system. By utilizing solar panels, the objective is to reduce dependency on non-renewable energy sources and promote environmentally friendly agricultural practices.

2. Advanced Sensing and Data Collection: The project aims to deploy advanced sensors to monitor crucial environmental parameters such as soil moisture, temperature, humidity, and light intensity. The objective is to collect real-time data to accurately assess crop water requirements and overall health.

3. Data Analytics and Decision Support: By leveraging data analytics techniques and intelligent algorithms, the project intends to process the collected data and generate actionable insights. The objective is to provide farmers with decision support, enabling them to make informed and precise irrigation decisions based on data-driven recommendations.

4. Optimization of Irrigation Processes: The project seeks to automate the irrigation process based on the analyzed data and recommendations. The objective is to optimize water usage, prevent water stress, and enhance crop productivity by delivering the right amount of water at the right time and in the right areas of the field.

5. Cost-Effectiveness and Feasibility: The project aims to assess the economic feasibility of the solar-powered smart irrigation system. The objective is to evaluate the installation costs, maintenance expenses, potential savings in water and energy usage, and overall cost-effectiveness of the system.

6. Scalability and Adaptability: The project intends to design the solar-powered smart irrigation system in a scalable and adaptable manner. The objective is to develop a system that can be easily deployed in different agricultural settings, accommodate various crop types, and be scalable to meet the requirements of different field sizes.

7. Environmental Sustainability: The objective is to contribute to environmental sustainability by reducing water wastage, optimizing energy usage, and minimizing the ecological footprint associated with traditional irrigation practices.

By achieving these objectives, the solar-powered smart irrigation system aims to revolutionize irrigation practices in agriculture, promoting water efficiency, crop health, and overall sustainability in the farming sector.

Conclusion

In conclusion, the solar-powered smart irrigation system project has successfully designed, developed, and implemented an innovative solution for optimizing irrigation practices in agriculture. By integrating renewable energy, advanced sensing technologies, data analytics, and intelligent algorithms, the project has achieved its objectives and demonstrated significant benefits for sustainable agriculture.

Through the utilization of solar energy, the project has reduced reliance on non-renewable energy sources, promoting environmentally friendly agricultural practices. The incorporation of data analytics and intelligent algorithms has provided valuable insights and decision support, enabling optimized irrigation schedules and water usage. By automating the irrigation process based on data-driven recommendations, the project has improved water efficiency, prevented water stress, and enhanced overall crop productivity.

The economic feasibility assessment of the solar-powered smart irrigation system has shown its potential as a cost-effective solution in the long run. With savings in water and energy usage, the system offers economic benefits to farmers while promoting sustainable practices.

Smart Irrigation systems offer a variety of advantages over traditional irrigation systems. One of which is the attention needed to the crop's during irrigation is reduced which is achieved with our project whilst keeping the project environmentally clean as we used solar energy as power source along with saving water as we would suggest using Drip irrigation as the method of irrigation if the crop does support it, as we found out that it is the most efficient method of irrigation with minimum water wastage.

Hence, we conclude that if the farmer can afford the initial one-time investment of setting up the smart irrigation system over traditionally fuelled manually supervised irrigation, then he should opt for the solar powered smart irrigation system as it also provides him with the subsidy from the government. This will further help in reducing the overall cost of the setup and the system will also require less maintenance cost. So, in the long run it does help him financially as well as in saving the water and protecting the environment

References

[1] Alok Gora and M. S. Dulawat, "Solar powered smart irrigation system-an innovative concept", in Research Journal of Agriculture and Forestry Sciences, vol. 5, no. 6, pp. 15-19, June 2017.

[2] Anjana Sengupta, Sougata Santra, Shamik Chattaraj, "Solar Powered Smart Irrigation System Using Humidity Sensor and Float Sensor", in IOSR Journal of Engineering, vol. 9, no. 6, pp. 34-36, June 2019.

[3] Rohan Sable, Sagar Kolekar, Adesh Gawde1, Sagar Takle and Adesh Pednekar, "A Review on Different Irrigation Methods," in International Journal of Applied Agricultural Research, vol. 14, no. 1 pp. 49-60, 2019.

[4] P. Kalaivani, N. Sushma, and S. Ragavi, "Solar Powered Smart Irrigation System Using IoT", in International Journal of Scientific Development and Research, vol. 5, no. 4, pp. 140-146, April 2020

[5] Dursun M and Ozden S. Application of solar powered automatic water pumping in Turkey. Int J Comput Electr Eng 2012; 4: 161–164.

[6] Viani F, Bertolli M, Salucci M, et al. Low-cost wireless monitoring and decision support for water saving in agriculture. IEEE Sensors J 2017; 17: 4299–4309.

[7] Masaba K, Ntakirutimana A and Ustun TS. Design and implementation of a smart irrigation system for improved water-energy efficiency. In: Fourth IET Clean Energy and Technology Conference (CEAT 2016), Proceedings of the CEAT conference, Kuala Lumpur, 2016, pp.1–5.