SMART IRRIGATION USING INTERNET OF THINGS AND MACHINE LEARNING

SUBMITTED TO Dr.MAHENDRA KUMAR SHUKLA

SUBMITTED BY

MARAM VIGNESH 2021BCS-039

WHAT IS SMART IRRIGATION?

Smart irrigation technology uses soil moisture data to determine the irrigation need of the landscape.

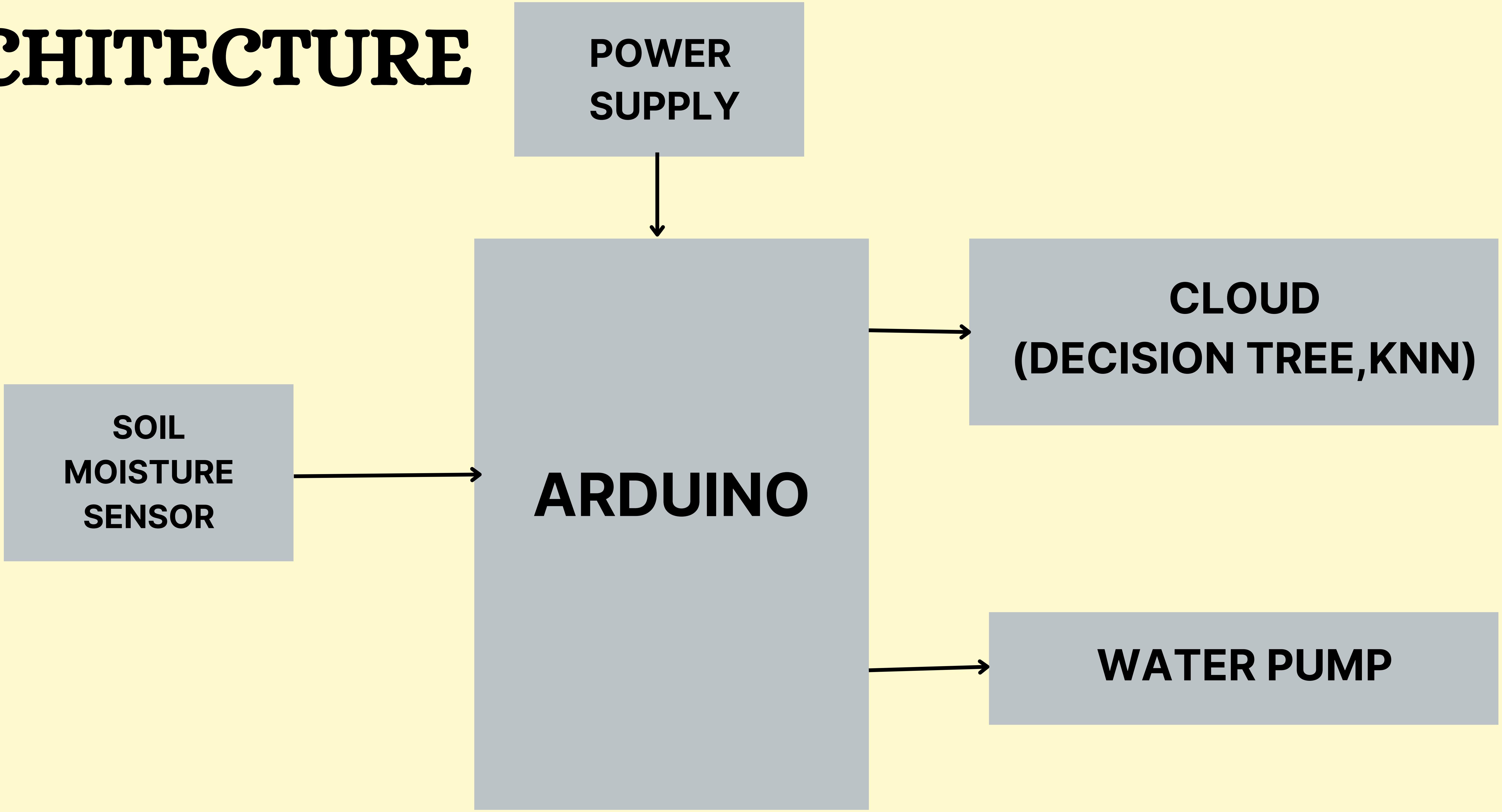
HOW IoT AND ML IS USED FOR SMART IRRIGATION?

- Sensors and a microcontroller can be used to supply the needed amount of water at the right moment.
- The microcontroller can receive input from the sensor data.
- The irrigation operation can be managed and controlled by microcontroller.
- By considering the amount of soil moisture of land it can be implemented in an automated watering system.
- With various tools and a prediction model, machine learning can be used to schedule irrigation.

EXISTING WORK

- Usage of Soil Moisture Sensor,Arduino and machine learning algorithms like decision tree and KNN.
- Soil Moisture Sensor senses the the amount of moisture in the soil and sends it to the arduino.
- The arduino sends this data to Cloud to analyse the data using machine learning algorithms like Decision Tree,KNN.
- After processing the data,the microcontroller sends a signal to turn on/off the water pump based on the result provided by the algorithms.

ARCHITECTURE



LIMITATIONS OF EXISTING MODEL

- Arduino boards typically have limited resources, such as less RAM and flash memory.
- Arduino boards often require additional modules or shields to add connectivity options like Wi-Fi or Bluetooth.
- Used External Cloud for data processing instead of Edge Computing.

LIMITATIONS(continued)

- Decision trees are prone to overfitting, especially when they are deep and complex.This means they may perform well on the training data but poorly on new, unseen data.

- Decision trees may struggle to capture complex nonlinear relationships between features and the target variable.

- Non usage of Proper Cloud facilities and Application for the Users.

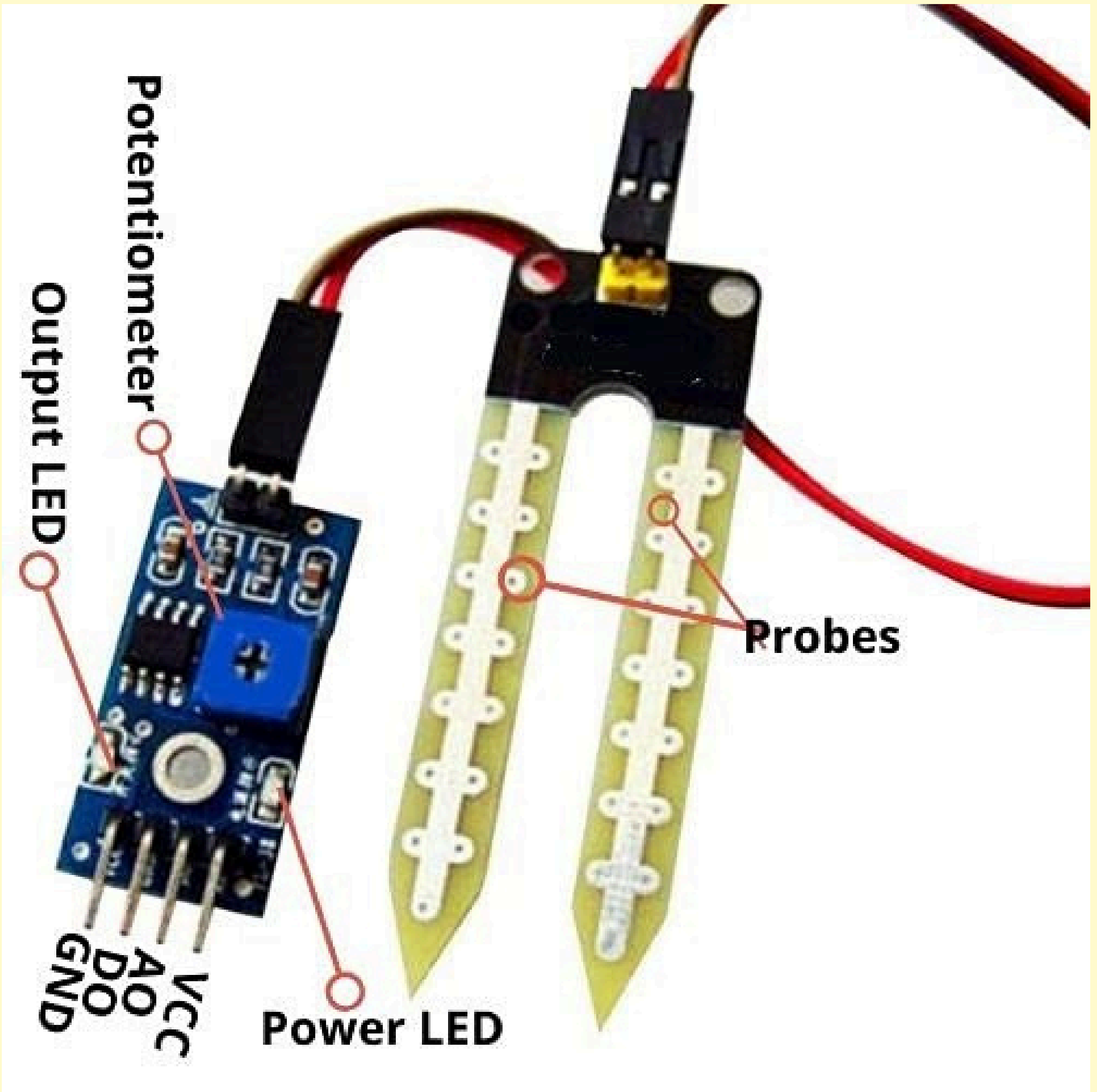
WHAT DID WE DO...

- Developed a better architecture using a better microcontroller NodeMCU.
- Usage of Temperature Sensor.
- Usage of RFID tags to identify the plant.
- Utilization of Edge Computing.
- Used Logistic Regression to train the model which provides better accuracy.

COMPONENTS
SENSORS

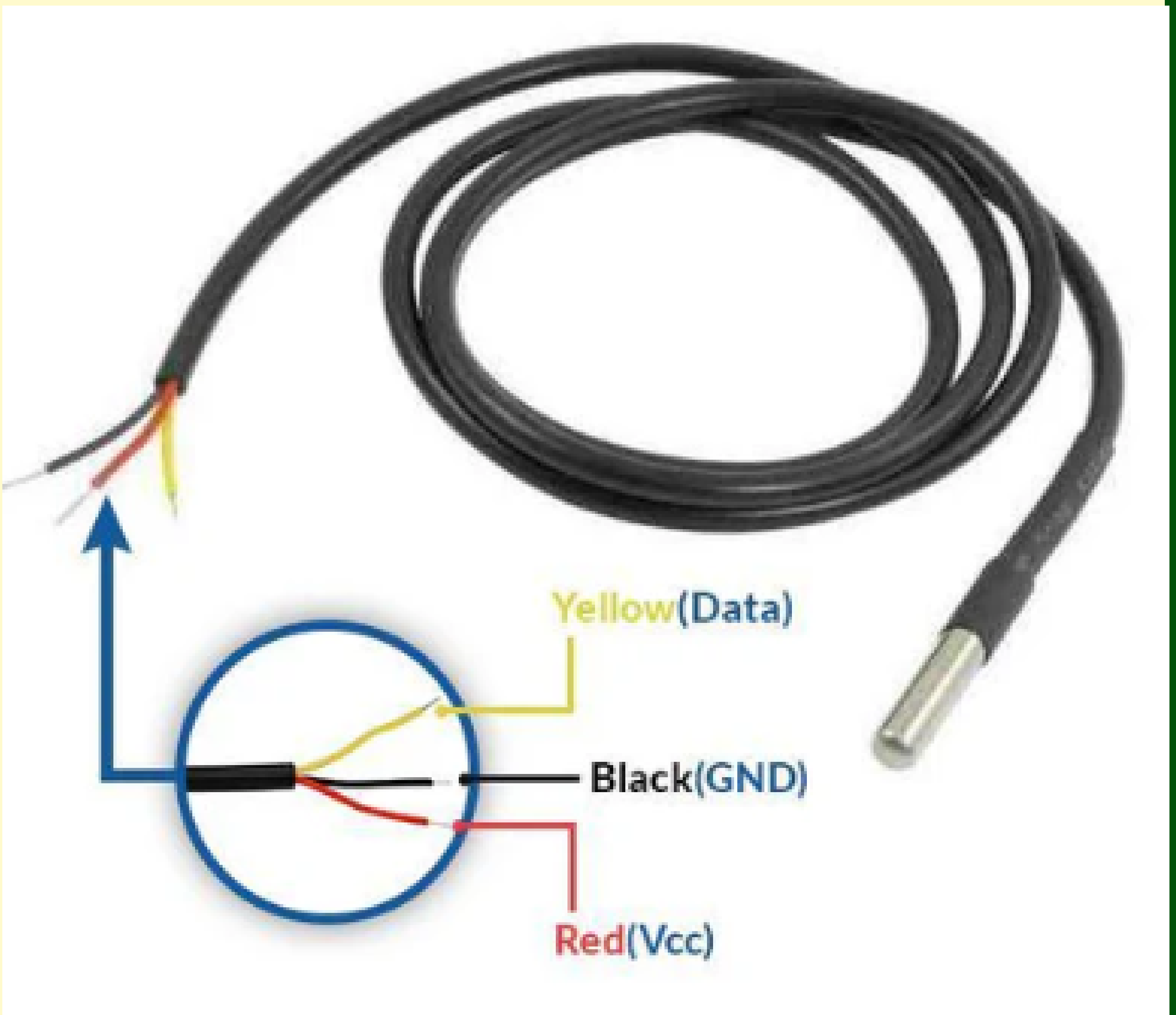
SOIL MOISTURE SENSOR

- The water content is measured by a Soil Moisture sensor.
- This sensor measures the permittivity of water content in the soil by using the capacitance.
- The moisture content from the soil is sent to the micro controller unit which analyzes this data to issue necessary signals.



TEMPERATURE SENSOR

- Temperature sensor is used to measure the temperature of the environment of the field.
- A Popular temperature sensor that can be used in a smart irrigation system with a NodeMCU is the DS18B20 digital temperature sensor.
- This sensor is widely used in IoT applications due to its accuracy, digital output, and ease of integration with microcontrollers like the NodeMCU.
- The DS18B20 sensor communicates over a one-wire interface, which means it requires only one pin on the NodeMCU for both power and data communication.
- This makes it convenient for connecting multiple sensors to the NodeMCU without using too many pins.

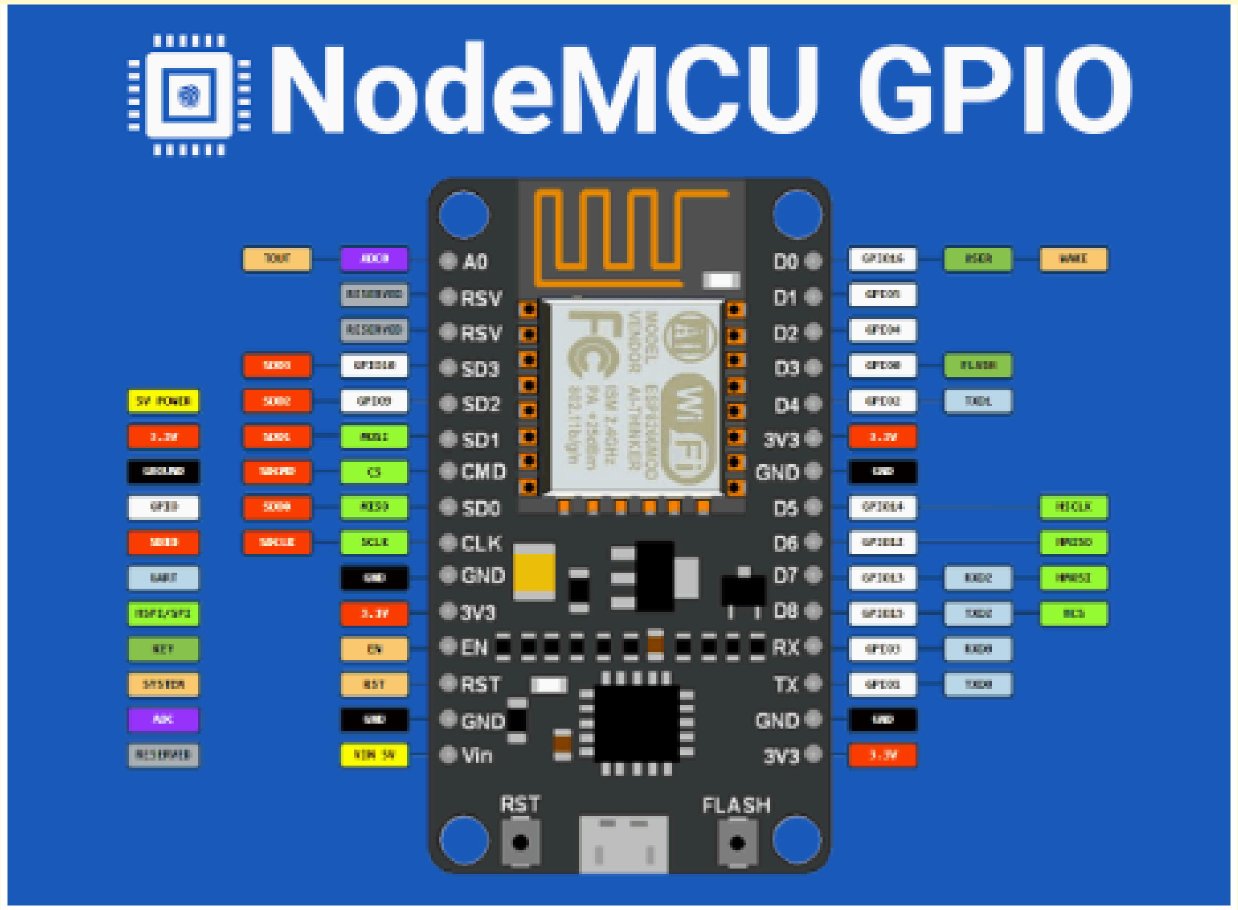


RFID(Radio-Frequency Identification) Tags

- RFID tags or chips can be attached to individual plants.
- Each RFID tag contains a unique identifier that corresponds to specific information about the plant.
- RFID readers installed in the irrigation system can read the RFID tags as plants move within range.
- The readers capture the unique identifiers from the tags and transmit this data to a central control system or database.

NODEMCU

- NodeMCU is a microcontroller which is often used for IoT (Internet of Things) projects due to its built-in Wi-Fi connectivity and relatively low cost.
- NodeMCU can be used in a smart irrigation system to automate and optimize the process of watering plants based on various factors such as soil moisture levels, temperature, and plant types.
- NodeMCU can control actuators such as pumps that regulate the flow of water to the plants.
- When NodeMCU determines that irrigation is required based on the sensor data, it activates the appropriate pumps to start watering.
- Machine Learning Algorithms are used here to analyse the incoming data and send the appropriate signals to the microcontroller.



RELAY

- In a smart irrigation system, a relay is a device that controls the flow of electricity to activate or deactivate irrigation components such as pumps, valves, or sprinklers.

- It acts as a switch that can be remotely controlled by the smart system's software or sensors.

- This automation helps in efficient water management and conservation in agricultural or garden settings.

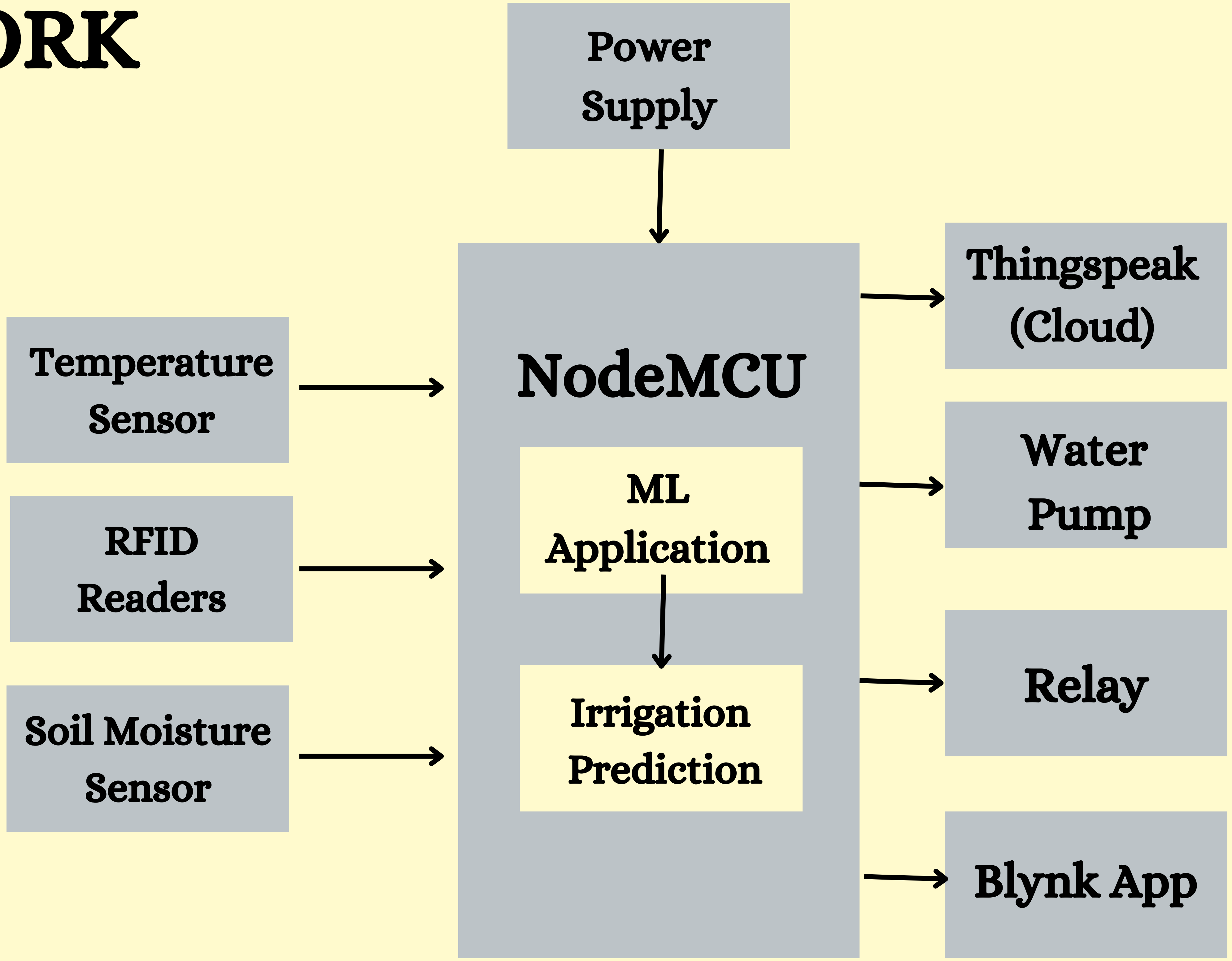
THINGSPEAK CLOUD

- Thingspeak is a platform that enables IoT (Internet of Things) devices to collect, analyze, and visualize data in real-time.
- It can be used in a smart irrigation system to monitor and control various parameters related to irrigation.
- IoT devices such as NodeMCU equipped with sensors (e.g., soil moisture sensors, temperature sensors) collect data related to environmental conditions relevant to irrigation.
- Thingspeak stores the incoming sensor data in its cloud-based database.

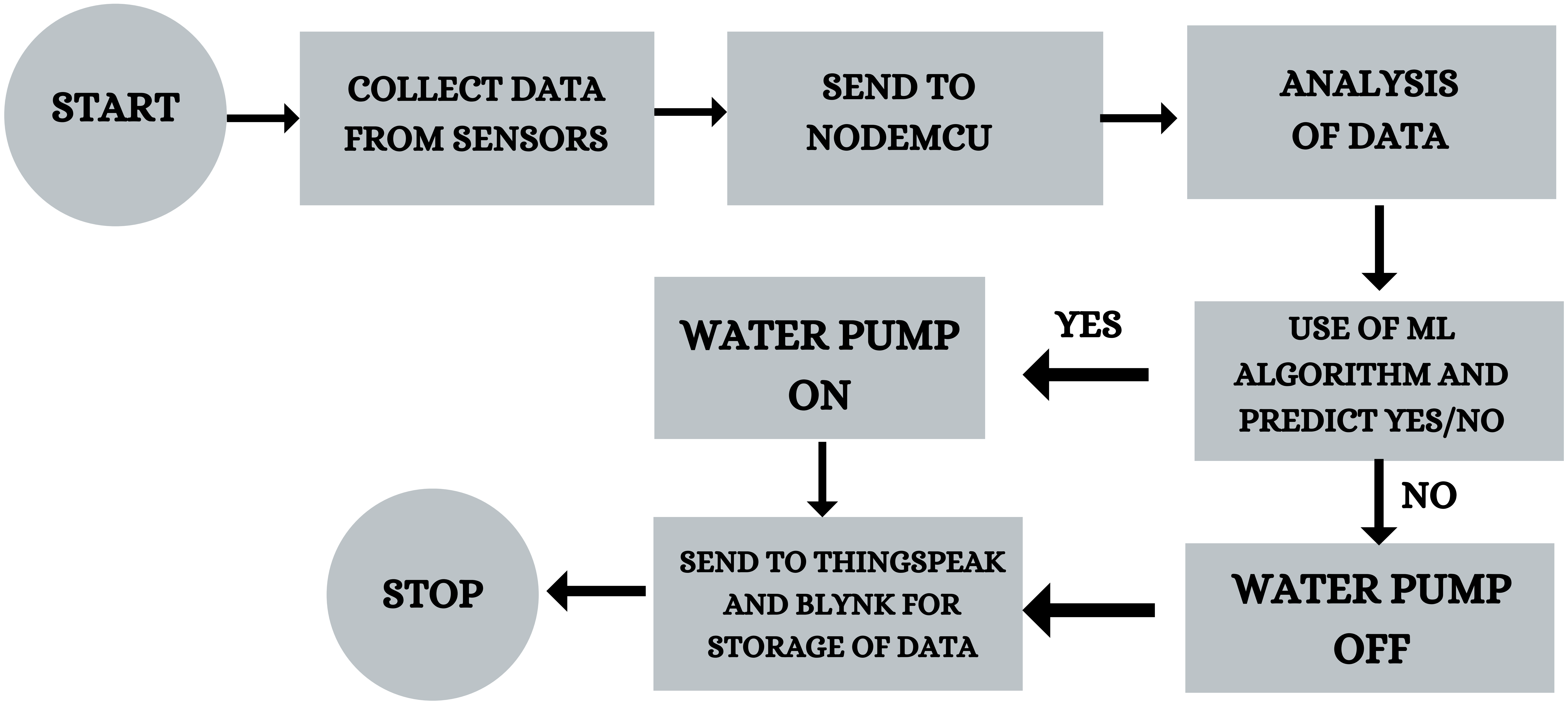
BLYNK APP

- Blynk is a popular IoT platform that allows users to create custom mobile applications for controlling and monitoring IoT devices and projects.
- It provides an easy-to-use interface and a range of features that make it suitable for integrating with smart irrigation systems.
- The app can display real-time data such as soil moisture levels, temperature, plant name and water flow status.
- With Blynk, users can remotely monitor and control their smart irrigation system from anywhere with an internet connection.
- They can check the status of sensors, turn on/off irrigation pumps or valves, and adjust settings on the go using the mobile app.
- Blynk supports push notifications and email alerts, enabling users to receive instant notifications about critical events such as low soil moisture levels,excess water.
- This helps in timely intervention and maintenance of the smart irrigation system.

ARCHITECTURE OF OUR WORK



FLOWCHART


START
COLLECT DATA FROM SENSORS
SEND TO NODEMCU
ANALYSIS OF DATA
USE OF ML ALGORITHM AND PREDICT YES/NO
YES
WATER PUMP ON
NO
WATER PUMP OFF
SEND TO THINGSPEAK AND BLYNK FOR STORAGE OF DATA
STOP

WORKING

- The data is collected using Temperature Sensor,Soil Moisture Sensor and RFID Tags.
- This Data is sent to NodeMCU and data is analysed using machine learning algorithms.
- This gives a result whether to water or not and sends signal to the pump.
- The data is sent to ThingSpeak Cloud and Blynk App from NodeMCU using its inbuilt WiFi Module where the data is stored.
- Connected sensors collect data for every 1000 miliseconds.

USE OF MACHINE LEARNING ALGORITHMS

- Machine Learning(ML) is a subset of Artificial Intelligence(AI) that are classified into supervised learning, unsupervised learning.
- KNN, Random Forest, Logistic Regression and Decision Tree belong to the supervised category, which makes prediction based on labeled data set.
- Here we Use Machine Learning Algorithms to analyse the sensor data and predict whether the plant needs to be watered or not.
- Many Algorithms like Decision Tree,Random Forest,KNN were used for the analysis of the data but the results were better when we used Logistic Regression to predict.
- As we get the data from the sensors to the cloud,the data is analysed by the Algorithm.
- We have Used Logistic Regression for the analysis of data.

LOGISTIC REGRESSION

- Logistic regression is a statistical method used for binary classification tasks, where the goal is to predict the probability that an instance belongs to a particular class (e.g., yes/no, true/false, 1/0).
- It's widely used in various fields such as machine learning, statistics, and data analysis.
- Logistic regression uses the logistic (or sigmoid) function to map the linear combination of input features to a probability score.

WHY LOGISTIC REGRESSION?

- Logistic regression offers several advantages over decision trees, random forests, and k-nearest neighbors (k-NN).
- Logistic regression can handle large datasets efficiently, especially compared to k-NN, which requires storing the entire dataset for prediction.
- Logistic regression's computational complexity is generally lower, making it more scalable for large datasets.
- Logistic regression is less prone to overfitting.

GLIMPSE OF OUR DATASET

crop	moisture	temp	pump
cotton	638	16	1
cotton	522	18	1
wheat	741	22	1
paddy	798	32	1
cotton	690	28	1
paddy	558	23	1
wheat	578	12	1
paddy	673	35	1
wheat	642	45	1
cotton	723	11	1
wheat	671	23	1
cotton	758	34	1
wheat	507	45	1
cotton	586	33	1
wheat	703	45	1
paddy	716	25	1
cotton	76	35	0
wheat	434	19	0
wheat	487	16	0
paddy	493	18	0
wheat	421	39	0
paddy	379	23	0
cotton	177	12	0
wheat	482	27	0
paddy	168	10	0
cotton	208	39	0
wheat	162	29	0
cotton	306	25	0
paddy	394	27	0
wheat	184	29	0
cotton	429	17	0

CODE SNIPPETS

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
```

```
In [11]:    from sklearn.model_selection import train_test_split
```

```
In [12]:    X = soildata.drop(['Arrosage','crop'], axis=1)
            y = soildata['Arrosage']
```

```
In [13]:    X_train, X_test, y_train, y_test = train_test_split( X, y,test_size=0.2,stratify=y, random_state=42)
```

In [14]:

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix
model = LogisticRegression(solver='liblinear', random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
y_pred_prob=model.predict_proba(X_test)
```

In [15]:

```
from sklearn.metrics import accuracy_score
print(accuracy_score(y_test, y_pred), ": is the accuracy score")
from sklearn.metrics import precision_score
print(precision_score(y_test, y_pred), ": is the precision score")
from sklearn.metrics import recall_score
print(recall_score(y_test, y_pred), ": is the recall score")
from sklearn.metrics import f1_score
print(f1_score(y_test, y_pred), ": is the f1 score")
```

```python
print(y_pred)
```

```
[1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 0 0 1 1 0 1 1 1 1 1 1 0 0 1 1 0 1 0 1 1 1 1
 1 1 1]
```

- Our Model Predicts whether the water pump has to be ON/OFF
- 1 indicates for Water Pump to be ON
- 0 indicates for the Water Pump to be OFF

RESULTS

We have achieved 95% Accuracy using our model with 93.75% Precision and 100% Recall

```
0.95 : is the accuracy score
0.9375 : is the precision score
1.0 : is the recall score
0.967741935483871 : is the f1 score
```

COMPARISION

ARCHITECTURE	ACCURACY(%)	PRECISION(%)	RECALL(%)
DECISION TREE	85.71	89.21	86.71
RANDOM FOREST	91.02	90.57	86.02
K-NN	93	91.36	97.8
LOGISTIC REGRESSION	95	93.75	100

- Set of these 4 algorithms are applied by considering three parameters i.e type of plant,temperature and soil moisture.

- From the application of Logistic Regression algorithm, highest accuracy is obtained.

- In comparison,the accuracy of prediction is less by using Decision Tree and Random Forest model.K-NN predicts more accurately with 91% in comparison to Decision Tree and Random Forest whereas Logistic Regression model gives the highest accuracy with 95% in comparison with other three modes on our work.

- So, Logistic Regression can be used for best prediction.

CONCLUSION

- A Smart irrigation system is presented in which NodeMCU is used as microcontroller and sensors(temperature,soil moisture) to record the required data.
- The Usage of NodeMCU helped us to send the data to the cloud due to the availability of inbuilt WiFi module.
- The Gathered data is analysed by ML algorithm in NodeMCU itself.
- Because of EdgeComputing,the results are faster.
- We have got the best results when using Logistic Regression.The water will be pumped based on the prediction of the Algorithm.
- As per the requirement of water for crops, the pump became on or off.
- The database will be stored in ThingSpeak and the user can access data using blynk app.

FUTURE SCOPE

- This system can be upgraded using a higher microcontroller like RaspberryPi.
- A higher level data set can be used using different clouds like Microsoft Azure, AWS etc.
- The model can be trained according to the updated data.
- This smart irrigation can be implemented in drip irrigation as well as in Floriculture in the future.
- Consider integrating additional sensors such as humidity sensors, rain sensors, or light sensors to gather more comprehensive data for irrigation management.
- Implement energy-efficient techniques such as sleep modes, low-power sensors, or solar-powered solutions to optimize energy consumption in the system.

THANK YOU