DAYANANDA SAGAR UNIVERSITY

KUDLU GATE, BANGALORE – 560068

Bachelor of Technology
in
COMPUTER SCIENCE AND ENGINEERING

Major Project Phase-II Report

SMART AGRICULTURE USING IOT AND MACHINE LEARNING

By

Manoj P	**-ENG18CS0160**
Shejan Shriram R	**-ENG18CS0259**
Shreyas U	**-ENG18CS0269**
Shyamsundar B	**-ENG18CS0273**
Surya S	**-ENG18CS0291**

Under the supervision of

Prof. Vaidehi Verma, Assistant Professor
Department of Computer Science and Engineering

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING,
SCHOOL OF ENGINEERING
DAYANANDA SAGAR UNIVERSITY,

(2021-2022)

School of Engineering
Department of Computer Science& Engineering
Kudlu Gate, Bangalore –560068
Karnataka, India

CERTIFICATE

This is to certify that the Phase-II project **"SMART AGRICULTURE USING IOT AND MACHINE LEARNING"** is carried out by **Manoj P (ENG18CS0160)**, **Shejan Shriram R (ENG18CS0259)**, **Shreyas U (ENG18CS0269)**, **Shyamsundar B (ENG18CS0273), Surya S (ENG18CS0291)** a bonafide students of Bachelor of Technology in Computer Science and Engineering at the School of Engineering, Dayananda Sagar University, Bangalore in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Engineering, during the year **2021-2022**.

Prof. Vaidehi Verma	**Dr. Girisha G S**	**Dr. A Srinivas**
Assistant Professor Dept. of CSE, School of Engineering Dayananda Sagar University Date:	Chairman CSE School of Engineering Dayananda Sagar University Date:	Dean School of Engineering Dayananda Sagar University Date:

Name of the Examiner **Signature of Examiner**

1.

2.

DECLARATION

We, **Manoj P (ENG18CS0160)**, **Shejan Shriram R (ENG18CS0259)**, **Shreyas U (ENG18CS0269)**, **Shyamsundar B (ENG18CS0273)**, **Surya S (ENG18CS0291),** are **students of eight semester B.Tech in Computer Science and Engineering, at School of Engineering, Dayananda Sagar University, hereby declare that the phase-II project titled "SMART AGRICULTURE USING IOT AND MACHINE LEARNING" has been carried out by us and submitted in partial fulfilment for the award of degree in Bachelor of Technology in Computer Science and Engineering during the academic year 2021-2022.**

Student	**Signature**
Manoj P **USN:ENG18CS0160**	
Shejan Shriram R **USN:ENG18CS0259**	
Shreyas U **USN:ENG18CS0269**	
Shyamsundar B **USN:ENG18CS0273**	
Surya S **USN:ENG18CS0291**	

Place : Bangalore

Date :

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of this project work.

First, we take this opportunity to express our sincere gratitude to School of Engineering & Technology, Dayananda Sagar University for providing us with a great opportunity to pursue our Bachelor’s degree in this institution.

We would like to thank **Dr. A Srinivas. Dean**, **School of Engineering & Technology**, **Dayananda Sagar University** or his constant encouragement and expert advice.

It is a matter of immense pleasure to express our sincere thanks to **Dr. Girisha G S, Chairman**, **Department of Computer Science and Engineering**, **Dayananda Sagar University,** for providing right academic guidance that made our task possible.

We would like to thank our guide **Prof. Vaidehi Verma, Assistant Professor**, **Dept. of Computer Science and Engineering**, **Dayananda Sagar University**, for sparing his/her valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project.

We would like to thank our Project Coordinator **Dr. Meenakshi Malhotra and Dr. Bharanidharan N** and all the staffmembers of Computer Science and Engineering for their support. We are also grateful to our family and friends who provided us with every requirement throughout the course. We would like to thank one and all who directly or indirectly helped us in the Project work.

Name	USN	*Signature of Student*
Manoj P	**-ENG18CS0160**	
Shejan Shriram R	**-ENG18CS0259**	
Shreyas U	**-ENG18CS0269**	
Shyamsundar B	**-ENG18CS0273**	
Surya S	**-ENG18CS0291**	

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

IoT	Internet of Things
ML	Machine Learning
GUI	Graphical User Interface

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Agriculture balances both food requirement for mankind and supplies indispensable raw materials for many industries, and it is the most significant and fundamental occupation in India. The advancement in inventive farming techniques is gradually enhancing the crop yield making it more profitable and reduce irrigation wastages. The proposed model is a smart irrigation system which predicts the water requirement for a crop, using machine learning algorithm. Moisture, temperature and humidity are the three most essential parameters to determine the quantity of water required in any agriculture field. This system comprises of temperature, humidity and moisture sensor, deployed in an agricultural field, sends data through a microprocessor, developing an IoT device with cloud. Machine learning algorithms is applied on the data sensed from the field in to predict results efficiently, which helps in decision making regarding water supply in advance.

Keyword – IoT, Machine Learning, Decision Tree, Greenhouse.

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

The development and usage of Smart Agriculture systems based on IoT and Machine Learning is changing the field of agriculture sector by not only improving the crop production but also making it cost effective. The world seems to be making advancements in the field of technology and it is necessary to make reasonable advancements in the field of agriculture as well. According to the World Bank, the food consumption would increase by 50% by 2050 if the global population continues to rise at its current pace. As a matter of fact, the effects of drastic changes in climatic conditions have seen crops yield falling by more than a quarter. The combination of IoT and Machine Learning can certainly help in lowering the cost and also help in increasing the scale of production through the collection of time series data from sensors. Nearly 51% of the crop yield is dependent on the influence of these factors. These factors include precipitation, temperature, humidity, and moisture and pH concentration.

1.1. WHAT WE PLAN TO DO:

At first, we will create a greenhouse and inside the green house we are going to set up the grow tray (or channel) and the reservoir that contains water and nutrients.

Types of plants we are growing: -

- Rose.
- Tomato
- Brinjal

For supplying water for reservoir to the plants tray the water pump connecting with soil moisture sensor and setting the threshold valve so the if the moisture level is less than the threshold valve then the water will be sent from reservoir to the plant trays through drip irrigation.

1.1.1. PLANTS :

The next step is to set up the sensor which are useful for the prediction of the growth of the plants such as soil moisture, temperature and humidity sensor, UV sensor.

- **Sensors**

The soil moisture sensor is used to determine the moisture value in the soil, temperature and humidity sensor is used to measure the temperature and humidity value in the surrounding the greenhouse and UV sensor calculate the u v rays falling in the plants. The next step is to set up the sensor which are useful for the prediction of the growth of the plants such as soil moisture, temperature and humidity sensor, UV sensor. The soil moisture sensor is used to determine the moisture value in the soil, temperature and humidity sensor is used to measure the temperature and humidity value in the surrounding the greenhouse and the UV sensor calculate the UV rays falling in the plants. The data from this sensor are collected in the cloud using Arduino Uno withESP8266 Wi-Fi module to connect to think speak cloud using API key.

Fig1. Smart agriculture using IoT with Greenhouse protocol

PURPOSE:

By introducing this product in the market this will be helpful to the farmers who are growing vegetables and fruits in small and medium scale using greenhouse protocol and they can provide an assurance to the quality and quantity of the yield produced. It is also beneficial for both farmers and middleman to the sellers by producing high yield and the best quality, as we are monitoring the plants to get best quality and quantity that is used to get high profit for all of them.

1.2. SCOPE

In our project we are constructing the green house, plants and water reservoir. We are implementing the sensor to receive the data from the plants and we are using sensors such as soil moisture sensor, temperature and humidity sensor and UV sensor.

We will collect data from the sensors and it will be sent to Arduino. From Arduino the data will be send to cloud.

And for future prediction of growth of plant we are using Machine Learning Algorithms. The current data will be visualized in the mobile application which we have made.

CHAPTER 2

PROBLEM DEFINITION

CHAPTER 2

PROBLEM DEFINITION

In current situation using manual which is traditional method of agriculture, we could not monitor nor measure the necessary values like temperature, humidity, pH, UV Radiation value with respect to various parameters due restricted knowledge, technology and other circumstances. So that we couldn’t provide necessary or required nutrients to plants this leads to lack of growth which is very essential for a proper yield and there is no prediction for the growth of the plants due to change of climatic condition (greenhouse effect by pollution)

Solution: The solution is to create an open source cloud-based control and data collection based smart greenhouse systems. We need to build a hardware system that works with the cloud-platform in order to both demonstrate the capabilities of the system as well as provide the prediction of the future yield of the plants.

CHAPTER 3

LITERATURE REVIEW

CHAPTER 3

LITERATURE REVIEW

SL NO.	TITLE	AUTHORS	TECHNOLOGY USED	DESCRIPTION	YEAR
1.	**AI and IoT Based Monitoring System for Increasing the Yield in Crop Production**	Richa Singh, Sarthak Srivastav a, Rajan Mishra	AI and IoT	The research is performed on a marigold plant to detect the most suitable conditions for plant growth. The effect of physical conditions like humidity, temperature, soil temperature and moisture and light intensity on the plant growth, is monitored using IoT based monitoring system.	ICE3 - February 2020
2.	**IoT based Smart Agriculture using Machine Learning**	Kasara Sai Pratyush Reddy, Y Mohana Roopa, Kovvada Rajeev	IoT and Machine Learning	The system is programmed to be trained from the given dataset using all the sensed data from the soil moisture,temperature and humidity sensors. By applying the decision tree learning algorithm on the real time data its processes and generates an output yes/no and sends the decision to the farmer through an email.Using this decision, a farmer can decide himself to water the crop	IEEE – September 04,2020

		L N, Narra Sai Nandan		only when required, avoiding the wastage of water use.	
3.	**Smart Agriculture using IoT and Machine Learning**	Sameer M, Mittal B , Sarvesh S, Sagar D	IoT and Machine Learning	The real time readings coming from the sensors along with the application of Machine Learning Algorithms will not only help farmers make informed decisions on which crop to grow in a particular region but also recommend fertilizers based on various factors like soil condition, climatic conditions etc. In addition, from the various machine- learning algorithms implemented, XGBoost seems to give the best results with 99.31% accuracy on the recommendations.	IRJET - 04 April 2021
4.	**IoT and Machine Learning Approaches for Automation of Farm Irrigation System**	Anneketh Vij, Singh Vijendra ,Abhishek, Shivam, Aashima Bassi, Arushi Sharma	IoT and Machine Learning	a proper distributed network contributes to the accuracy of the predictions made by SVR, Random Forrest Regression. Sensor node inter-connectivity will help monitor the complete field thoroughly. To implement a system which would be mobile and can help in every stage of farming	ICCIDS - 2019

5.	**Smart Agriculture Using IoT and Machine Learning**	Sumiksha Shetty & A. B. Smitha	IoT and Machine Learning	In this paper, the assistance of IoT and machine learning in farming, which can expand the productivity of yield production.Different climate parameters are taken into thought from which the best reasonable yield is grown is predicted by supervised learning algorithm, decision tree, utilized in regions like sickness recognition, crop discovery,irrigation system, soil conditions and also the product quality and market analysis.	01 January 2022

CHAPTER 4
PROJECT DESCRIPTION

CHAPTER 4
PROJECT DESCRIPTION

4.1. PROPOSED DESIGN

The first step for the implementation of a Smart irrigation system for farm automation is laying the wireless sensor network field in which each node is inter connected by Wi-Fi module and lays data over a common server. The first step towards establishing an automatic irrigation system is collecting data through various sensors attached throughout the field or the garden. Arduino Uno will act as the gate way node responsible for communicating with all other sensor nodes (Arduino UNO), Each Arduino UNO unit consist of advance soil moisture sensor, Wi-Fi module, Temperature, Humidity and UV Sensor. The connection of various sensors to the microcontroller is based on the fundamental concept of receiver, transmitter, and ground and positive, digital and analog pins along with a number of Vcc and Gnd pins. The first phase of the suggested application is completed after establishing the network topology and the collection of the data. The collection of data through various sensors is the preliminary and the first step for data processing.

4.2. WORKFLOW AND DIAGRAMS

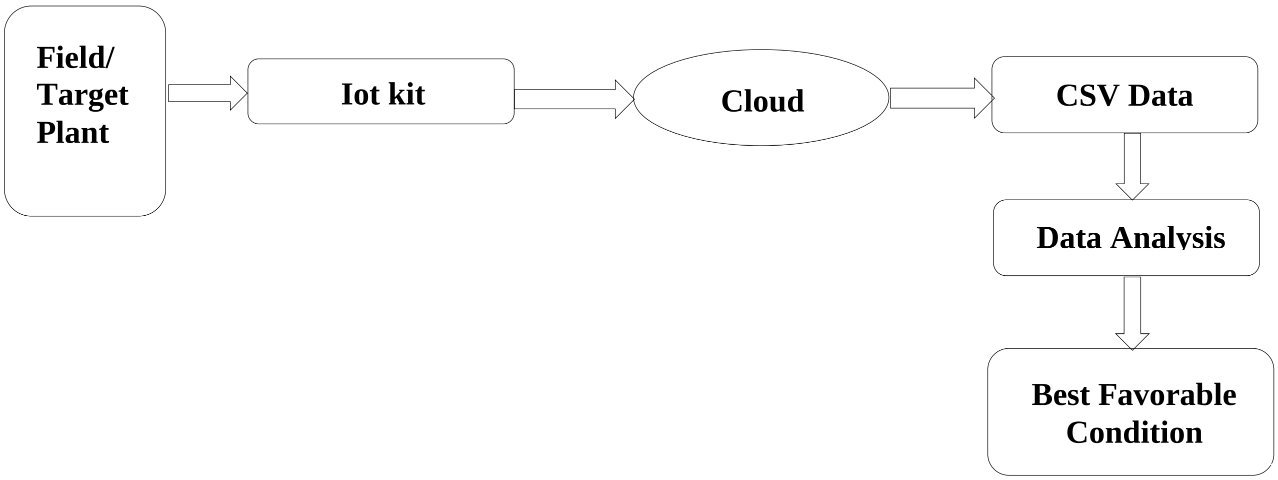


Fig2. Block diagram of ML and IoT based monitoring system.

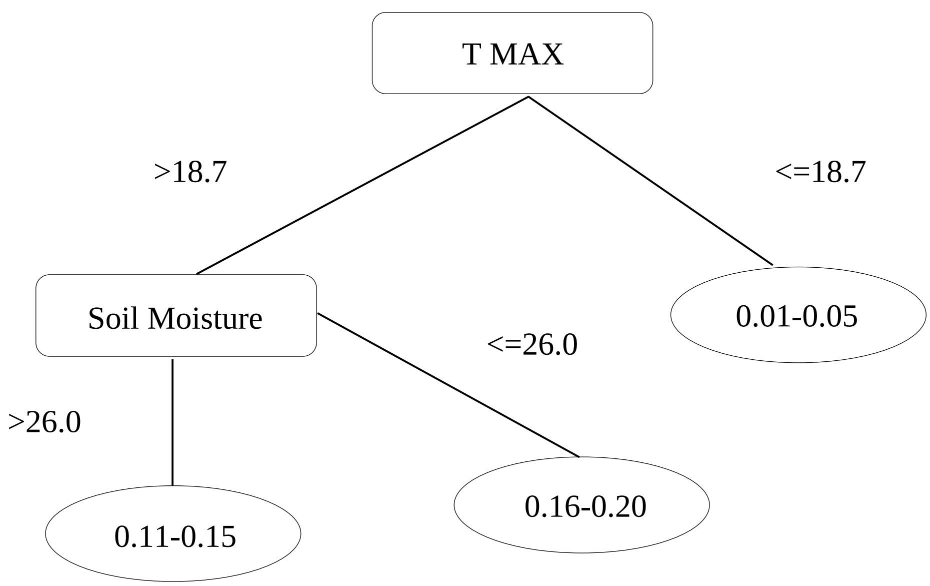


Fig3. Sample Decision Tree.

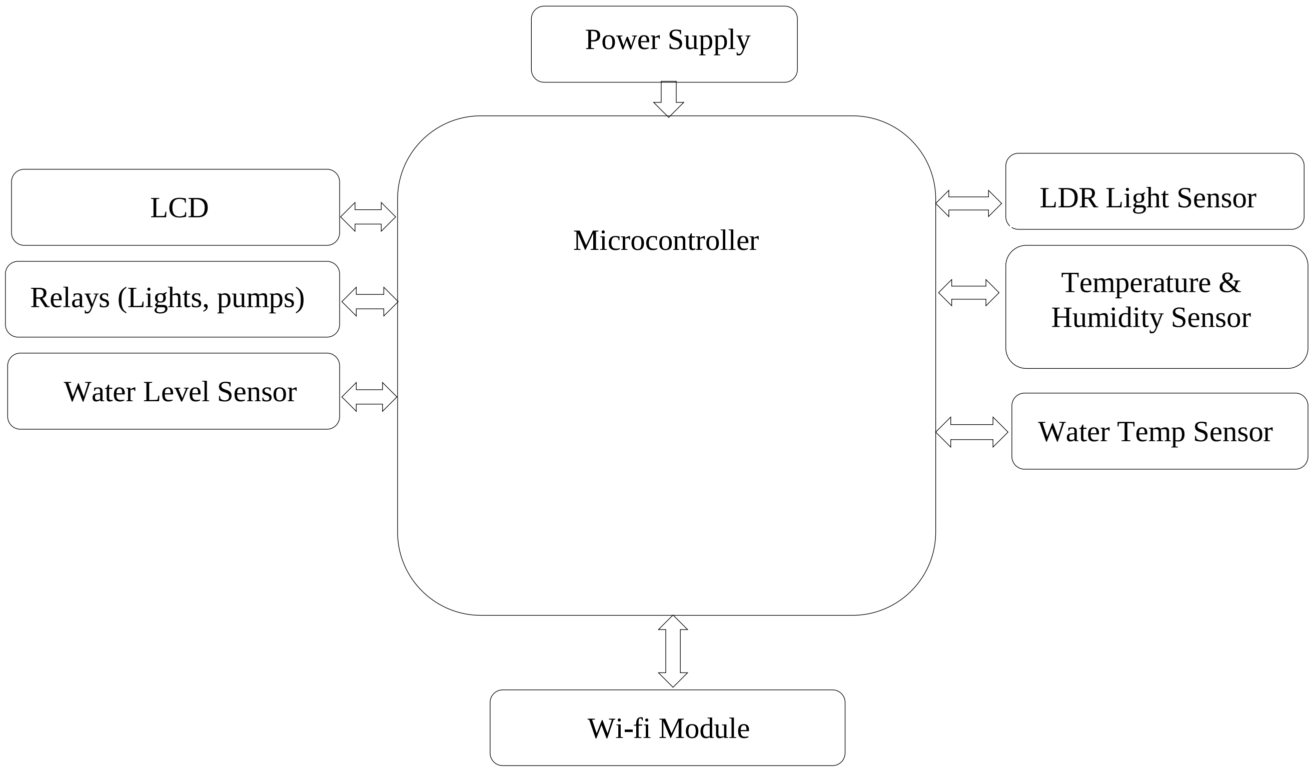


Fig4. Block diagram for IoT Connections

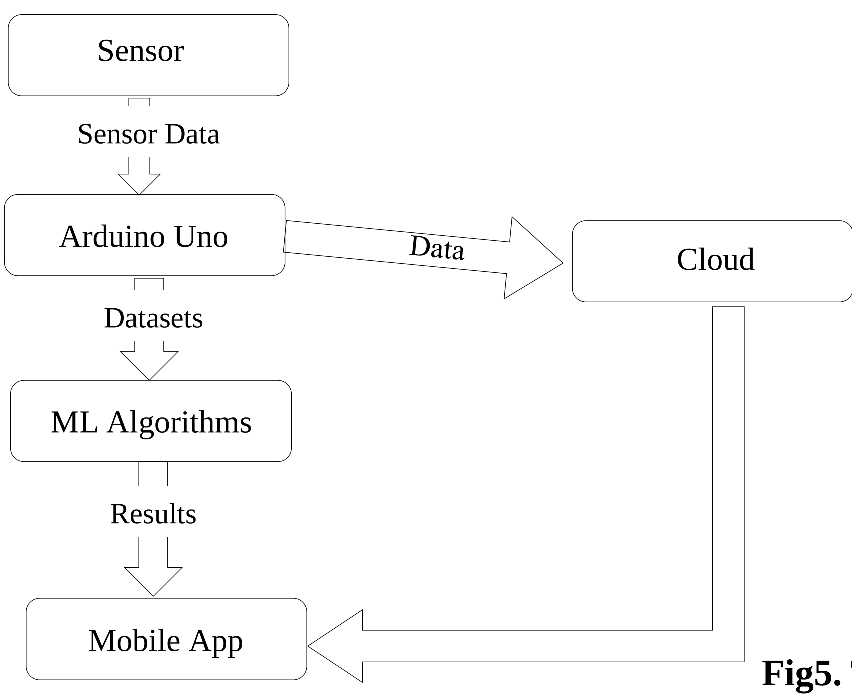


Fig5. Thingspeak Cloud

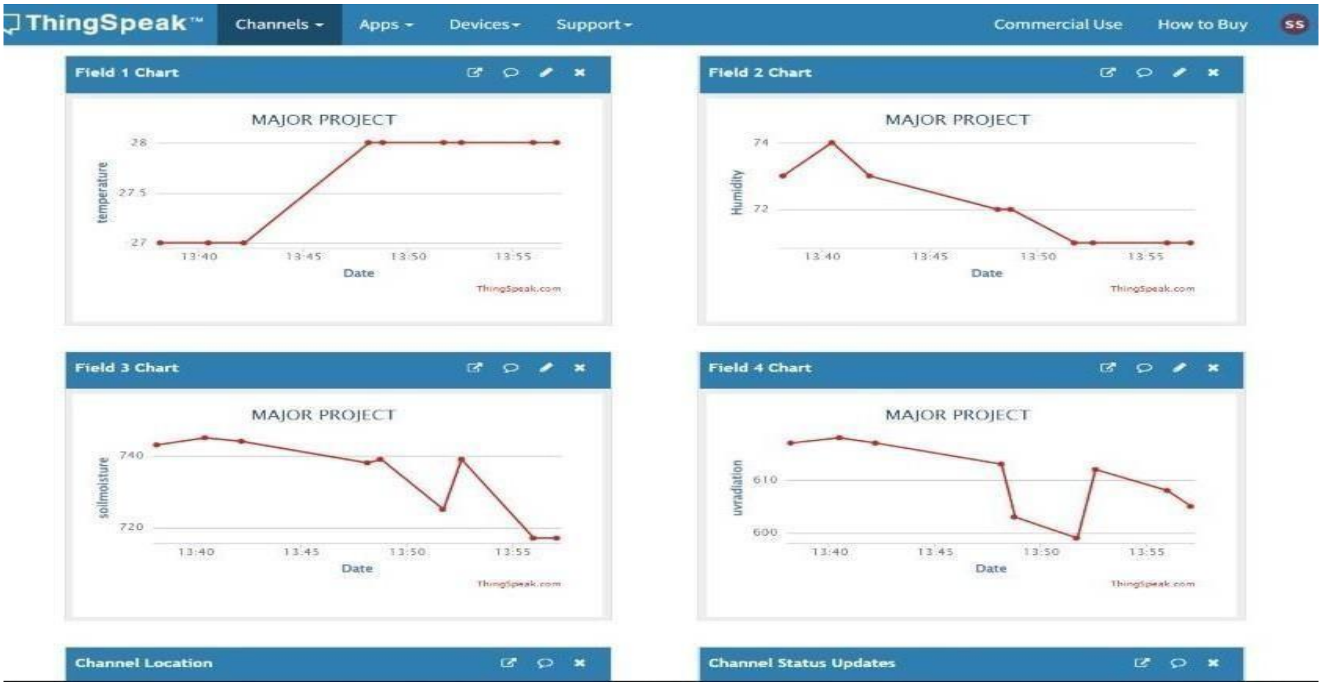


Fig6. Graphical Representation of Sensors Data

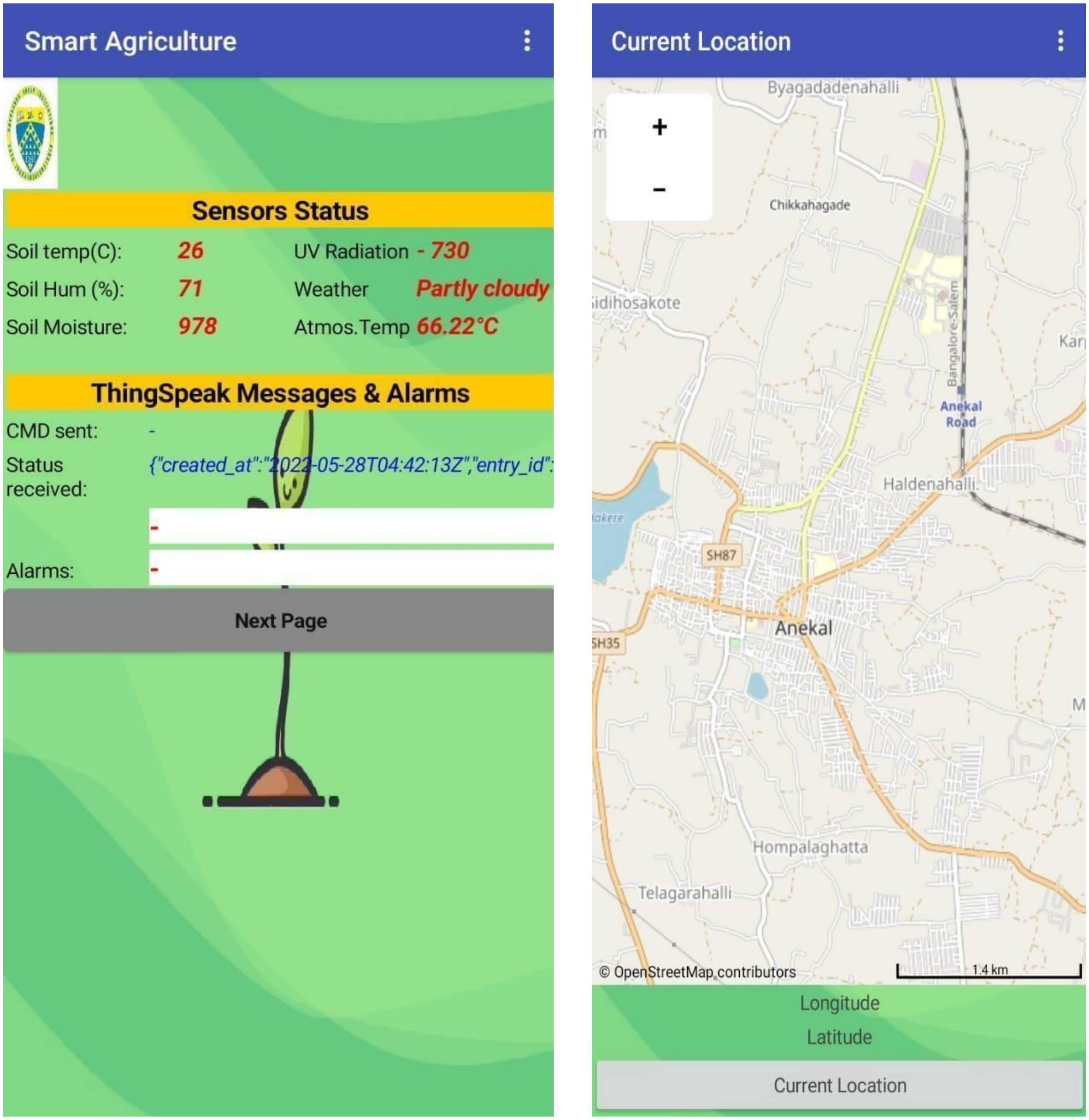


Fig7. Real Time Sensors data in Mobile Application

CHAPTER 5
REQUIREMENTS

CHAPTER 5
REQUIREMENTS

5.1. FUNCTIONAL REQUIREMENTS

Data Processing:

All the data that has been gathered by the sensors needs to be analysed and processed so that the further signal can be send to the actuators as well as alerts can be sent to the end user in case if manual intrusion is necessary. Weather data will be continuously fetched from online open-source API's and if the weather indicates a probability of rain more than 98% then the field will not be irrigated but as a safety measure if the humidity drops below a certain threshold for a specific crop, then the field will be irrigated. The Irrigation would be based on soil type and crop specific, also for automating the systems, the soil type would be determined using Machine Learning Algorithms. The best algorithm for soil type classification as per literature review is Support vector classification.

Data Analysis:

For understanding the various crops grown in India and their production quantity and cost aspects an exploratory data analysis is done on "Agriculture Production in India ". The analysis is done in python. This analysis lays the foundation of significance and need for automation in the sector to reduce cost and increase productivity.

Data presentation:

The data needs to be available on a dynamic website, which will represent the real time data analytics and the time stamped irrigation pattern so that in case of any anomaly corrective measures can be taken easily. Visual line graphs tend to provide a temperature and humidity values with the time stamps thus enabling an easy to understand, fast and agile implementation of the system.

Soil Moisture and Weather Data Processing:

The soil humidity is being measured using DHT11 (Digital Humidity and Temperature) sensor, which measures both humidity and temperature. To save and use energy, resources more efficiently machine learning algorithms were applied on the previously collected humidity states, so as to optimize the crop irrigation. For predicting the soil moisture, a comparative study of random forest regression was carried out. The best algorithm for predicting the relative humidity based on the past data and the precipitation probability fetched from open-source weather API providers using R programming and rvest package of R. The crop type, weather conditions and the growth stage of crop plays a significant role in the amount of water required and thus the generic approach towards water needs of crops tends to provide wrong details. Thus, with exact water need data, irrigation is to be carried out in an efficient and sustainable manner.

5.1.1. HARDWARE REQUIREMENTS

The major part of the proposed system will be a hardware device that constitutes of different kinds of sensors such as Soil Moisture Sensor (EK1361), Humidity and Temperature Sensor (DHT22) and UV Radiation Sensor (ML-8511). The device collects the surroundings attributes (soil moisture, temperature, radiation intensity) independently and processes it with the help of its microcontroller ATMega328p. The microcontroller will then send the received data immediately to the AWS S3 database or open source free Thingspeak cloud. 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, and spits out a digital signal on the data pin (no analog input pins needed). It fairly simple to use, but requires careful timing to grab data. The Soil Moisture Sensor measures soil moisture grace to the changes in electrical conductivity of the earth (soil resistance increases with drought). The electrical resistance is measured between the two electrodes of the sensor, A comparator activates a digital output when a adjustable threshold is exceeded The ESP8266 WiFi Module is a self contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all WiFi networking functions from another application processor. The ML8511 sensor breakout is an easy to use ultraviolet light sensor. The MP8511 UV Sensor outputs an analog signal in relation to the amount of UV light it detects. This can be handy in creating devices that warn the user of sunburn or detect the UV index as it relates to weather conditions.

5.2. NON-FUNCTIONAL REQUIREMENTS

The non-Functional requirements include:

SECURITY REQUIREMENTS:

I. **Authentication**: This attribute encompasses a technique to ensure that an entity is genuine during its identification, before providing any access to the system.

II. **Integrity:** It is to ensure that information that is received by the receiver is the same as that sent by the sender, without even minor change.

III. **Confidentiality:** It is to ensure that information that is sensitive to the system is never disclosed to any party, who does not have the authority to its access.

IV. **Availability:** An entity that has the authorization to access a service, should never be denied the access even under the circumstances of the Denial-of-Service (DoS) attacks.

V. **Non-repudiation:** It allows no entity to repudiate the services or actions that were taken up by that entity. This property in turn ensures traceability of a service to an entity.

VI. **Authorization:** This feature ensures that only an entity that is given the rights to provide any particular network service, does so.

VII. **Freshness:** This property ensures that a message received is not generated before a threshold time period, which disallows any message to be replayed by an adversary.

VIII. **Forward secrecy:** After a node is detached from the network, either voluntarily or otherwise, it should not be allowed access to any communication that continues within the network.

IX. **Backward secrecy:** A node that has been recently added to the network should not be privy to a communication that had taken place before its addition to the group

➢ SECURITY THREATS:

An IoT-enabled smart agriculture can be vulnerable to many possible attacks and few of them are:

- **Replay attack:** In a replay attack, after a transmission from an entity, an adversary A, may reuse the content from the previous transmission and attempt to deceive an authorized entity.

- **Man-in-the-middle attack**: During transmission between two entities, A can read the transmitted messages and may then attempt to modify or delete the contents of the messages delivered to the receiver.

- **Stolen-verifier attack:** If an access point (gateway node) stores a list or a table of passwords corresponding to the identities, A may attempt to steal the list from the access point, thereby gaining access to all passwords.

- **Stolen/Lost smart card attack:** Once A has obtained lost/stolen smart card, the techniques such as power analysis attacks and timings attacks can be used to extract the credentials stored in the memory of a smart card or a mobile device. The extracted credentials can be then used to further derive the secret data used in the calculation of the credentials.

- **Password guessing attack:** This attack involves attempts to speculate the correct password using the intercepted messages and illegal access to credentials stored in the smart card or the mobile device.

- **Password change attack:** In this attack, after obtaining access to a stolen smart card or a mobile device, attempts can be made by A to change the passwords of the existing registered users in order to be able to gain unauthorized access.

- **Denial-of-Service (Do's) attack:** This is a specific type of attack in which services are denied to an authorized user on account of overuse of the system's resources by other factors such as a failure in hardware or software bugs or over-allocation of bandwidth to certain users.

- **Privileged-insider attack:** In this kind of attack, an existing user within the system attempts to misuse his/her privileges in order to acquire unauthorized rights.

5.2.1. SOFTWARE REQUIREMENTS:

- **Arduino IDE:** Arduino IDE is a platform which allows to program the Arduino microcontrollers which we discussed in the hardware components. We can code the micro controller according to our application and its syntax is relatively simple. It is an open source and we can use embedded C to program. It's simple to use this IDE as there are readymade code examples. The editor has many features which makes the programming look easy. The programs are known as sketches in this platform. The procedure of using this IDE is selecting the board and writing the appropriate code for it. Once the code is completed, compile and check for errors and then upload code to the microcontroller. Once the code is uploaded the board performs the specified operation.

- **Google Collab:** Collab is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup and the notebooks that you create can be simultaneously edited by your team members - just the way you edit documents in Google Docs. Collab supports many popular machine learning libraries which can be easily loaded in your notebook.

- **Jupyter Notebook:** Jupyter Lab is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extension to expand and enrich functionality.

- **Thingspeak Cloud:** Thingspeak is an IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak. With the ability to execute MATLAB code in ThingSpeak you can perform online analysis and processing of the data as it comes in. ThingSpeak is often used for prototyping and proof of concept IoT systems that require analytics.

- **MIT app inventor:** It is a web application integrated development environment originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT). It allows developers to create application software (apps) for two operating systems (OS): Android, and iOS. It uses a graphical user interface (GUI) very similar to the programming languages Scratch (programming language) and the Star Logo, which allows users to drag and drop visual objects to create an application that can run on Android devices.

- **Visual Studio Code:** Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

Some basic functionality requirements in an IoT-based agriculture environment:

1. **Dynamic New Smart Device Addition:** It is extremely essential that a security scheme should support dynamic node addition facility after initial deployment of the nodes including the smart devices in the network.

2. **High Scalability:** Support of high scalability is a basic functionality requirement in a modern day IoT-enabled agriculture environment. High scalability should assure that even if the number of IoT smart devices is going to increase, the overall network performance should not be affected by this factor.

3. **Mutual Authentication:** Mutual authentication and key management are considered as two important techniques to assure secure communication in an IoT-enabled agriculture environment. Resource limitations of the IoT smart devices and their vulnerability to physical capture make the design of a mutual authentication between two IoT smart devices and also between a remote user.

4. **Availability:** The device communication and control in an IoT-based agriculture environment is performed in real time. For instance, a user might require to remotely controlling an agriculture field by monitoring crops condition and require to access data any time.

5. **Efficiency:** In an IoT-based agriculture environment, various smart devices are resource limited, including the limited battery lifetime. In addition, the IoT smart devices might have also other constraints, such as storage.

6. **UX:** One particular barrier to Block chain development is designing the user experience. It's also important from a user experience perspective. That first impression is very important for a new user.

CHAPTER 6

METHODOLOGY

CHAPTER 6
METHODOLOGY

In the proposed system, the main concept implemented is the Internet of Things (IOT). There will be a low-level hardware device that will measure different variables of the surroundings like temperature, humidity, soil moisture, UV radiation, etc. The measured values will then be transferred to AWS S3 database or Thingspeak cloud database, which will later be pulled back to a mobile application for further processing. On the other hand, there will be an option to connect the device to a phone application with the help of API key in Thingspeak cloud database. The data will then bepushed to the database server using the phone. The major part of the proposed system will be a hardware device that constitutes of different kinds of sensors such as Soil Moisture Sensor (EK1361), Humidity and Temperature Sensor (DHT22), and UV Radiation Sensor (ML-8511). The device collects the surroundings attributes independently and processes it with the help of its microcontrollerATMega328p. The microcontroller will then send the data to the AWS S3 or thing-speak cloud database. A mobile application will be made parallelly to send the data. The received data will be utilized to process necessary requirements that a farmer needs, for maximum yield of crops. The data received from the will be analyzed using different machine learning algorithms and researched agricultural datasets. According to the results from different backend machine learning algorithms, a suggestion template will be developed. The main challenge will be to collect different agricultural datasets that will be used to analyze the data. The mobile application will be fully focused on the data representation. We used different platforms to get crops data with all the above-mentioned attributes and got somedatasets in agricultural college in Bengaluru G.K.V.K and also from verified sources (like Kaggle) we collected data, we have parallelly used our monitored data for our project.

CHAPTER 7
EXPERIMENTATION

CHAPTER 7

EXPERIMENTATION

We used different platforms to get crop data with all the above-mentioned attributes and got some datasets from the agricultural college G.K.V.K as well as from other sources (like Kaggle), we have used our monitored data for this existing project to do experiment.

The training dataset used for the training of crop prediction model contains features like nitrogen, potassium, phosphorus, temperature, humidity, and rainfall whereas the fertilizer recommendation model contains features like nitrogen, phosphorus, potassium, pH and soil moisture. Various Machine learning algorithms were applied against the training dataset - Decision Tree, Naive Bayes, Support Vector Machine, Logistic Regression, Random Forest and XG Boost, and were compared based on the model’s accuracy.

The user, which is a Farmer, will be able to use our platform to get accurate recommendations regarding which crop to grow based on different features like humidity, pH, and rainfall. Besides that, the user will be able to predict yield based on different features like moisture, nitrogen, phosphorus and potassium.

CODE

```
#include <SoftwareSerial.h>

#include <dht11.h>

#define RX 2

#define TX 3

#define dht_apin A2 // Analog Pin sensor is connected to

dht11 dhtObject;

String AP = "UR";       // AP NAME

String PASS = "ravi2001"; // AP PASSWORD

String API = "CK7VI3KOS2KD2RMX";   // Write API KEY
```

```
String HOST = "api.thingspeak.com";
String PORT = "80";
int countTrueCommand;
int countTimeCommand;
int msensor = A1;
int msvalue = 0;
boolean found = false;
int valSensor = 1;
int sensorvalue;

SoftwareSerial esp8266(RX, TX);

void setup() {
 Serial.begin(9600);
  esp8266.begin(115200);
 sendCommand("AT", 5, "OK");
   sendCommand("AT+CWMODE=1", 5, "OK");
  sendCommand("AT+CWJAP=\"" + AP + "\",\"" + PASS + "\"", 20,"OK");
 }

 void loop() {

 String getData = "GET /update?api_key=" + API + "&field1=" + getTemperatureValue() +
 "&field2=" + getHumidityValue()+"&field3=" + getsoilmoisturevalue()+"&field4=" +
 getuvradiationvalue();
 sendCommand("AT+CIPMUX=1", 5, "OK");
```

```
sendCommand("AT+CIPSTART=0,\"TCP\",\"" + HOST + "\"," + PORT, 15, "OK");

sendCommand("AT+CIPSEND=0," + String(getData.length() + 4), 4, ">");

esp8266.println(getData); delay(1500); countTrueCommand++;

 sendCommand("AT+CIPCLOSE=0", 5, "OK");

}



String getTemperatureValue() {


dhtObject.read(dht_apin);

Serial.print(" Temperature(C)= ");

int temp = dhtObject.temperature;

Serial.println(temp);

 delay(50);

 return String(temp);


}



String getHumidityValue() {


dhtObject.read(dht_apin);

  Serial.print(" Humidity in %= ");

int humidity = dhtObject.humidity;

  Serial.println(humidity);
```

```
 delay(50);
return String(humidity);


}
int getsoilmoisturevalue() {
 msvalue = analogRead(msensor);
  Serial.println(msvalue);
  delay(1000);
 return int(msvalue);
}
int getuvradiationvalue(){
sensorvalue=analogRead(36);
  Serial.println(sensorvalue);
  delay(1000);
 return int(sensorvalue);
}


void sendCommand(String command, int maxTime, char readReplay[]) {
Serial.print(countTrueCommand);
Serial.print(". at command => ");
Serial.print(command);
Serial.print(" ");
 while (countTimeCommand < (maxTime * 1))
{
	esp8266.println(command);//at+cipsend
```

```
 if (esp8266.find(readReplay)) //ok
 {
        found = true;
       break;
}

 countTimeCommand++;
}

if (found == true)
 {
       Serial.println("Ok");
       countTrueCommand++;
       countTimeCommand = 0;
 }

 if (found == false)
{
       Serial.println("Fail");
       countTrueCommand = 0;
        countTimeCommand = 0;
 }

found = false;
}
```

Code

```
# Run this program on your local python
# interpreter, provided you have installed
# the required libraries.

# Importing the required packages
import numpy as np
import pandas as pd
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report

# Function importing Dataset
def importdata():
	balance_data = pd.read_csv("feeds.csv")

	# Printing the dataswet shape
	print ("Dataset Length: ", len(balance_data))
	print ("Dataset Shape: ", balance_data.shape)

	# Printing the dataset obseravtions
	print ("Dataset: ",balance_data.head())
	return balance_data
```

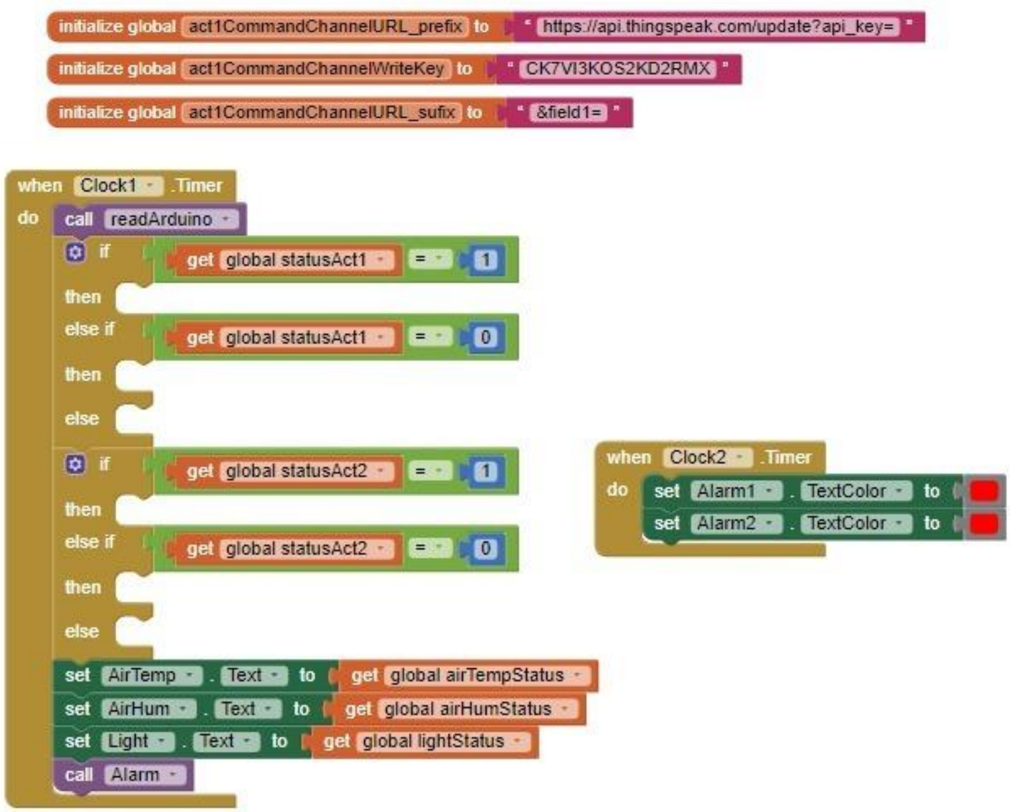
```
# Function to split the dataset
def splitdataset(balance_data):

    # Separating the target variable
    X = balance_data.values[:, 1:5]
    Y = balance_data.values[:, 0]

    # Splitting the dataset into train and test
    X_train, X_test, y_train, y_test = train_test_split(
    X, Y, test_size = 0.3, random_state = 100)

    return X, Y, X_train, X_test, y_train, y_test

# Function to perform training with giniIndex.
def train_using_gini(X_train, X_test, y_train):

    # Creating the classifier object
    clf_gini = DecisionTreeClassifier(criterion = "gini",
    random_state=100,max_depth=3,min_samples_leaf=5)

    # Performing training
    clf_gini.fit(X_train, y_train)
    return clf_gini
```

```
# Function to perform training with entropy.
def tarin_using_entropy(X_train, X_test, y_train):

    # Decision tree with entropy
    clf_entropy = DecisionTreeClassifier(
            criterion = "entropy", random_state = 100,
            max_depth = 3, min_samples_leaf = 5)

    # Performing training
    clf_entropy.fit(X_train, y_train)
    return clf_entropy



# Function to make predictions
def prediction(X_test, clf_object):

    # Predicton on test with giniIndex
    y_pred = clf_object.predict(X_test)
    print("Predicted values:")
    print(y_pred)
    return y_pred

# Function to calculate accuracy
def cal_accuracy(y_test, y_pred):
```

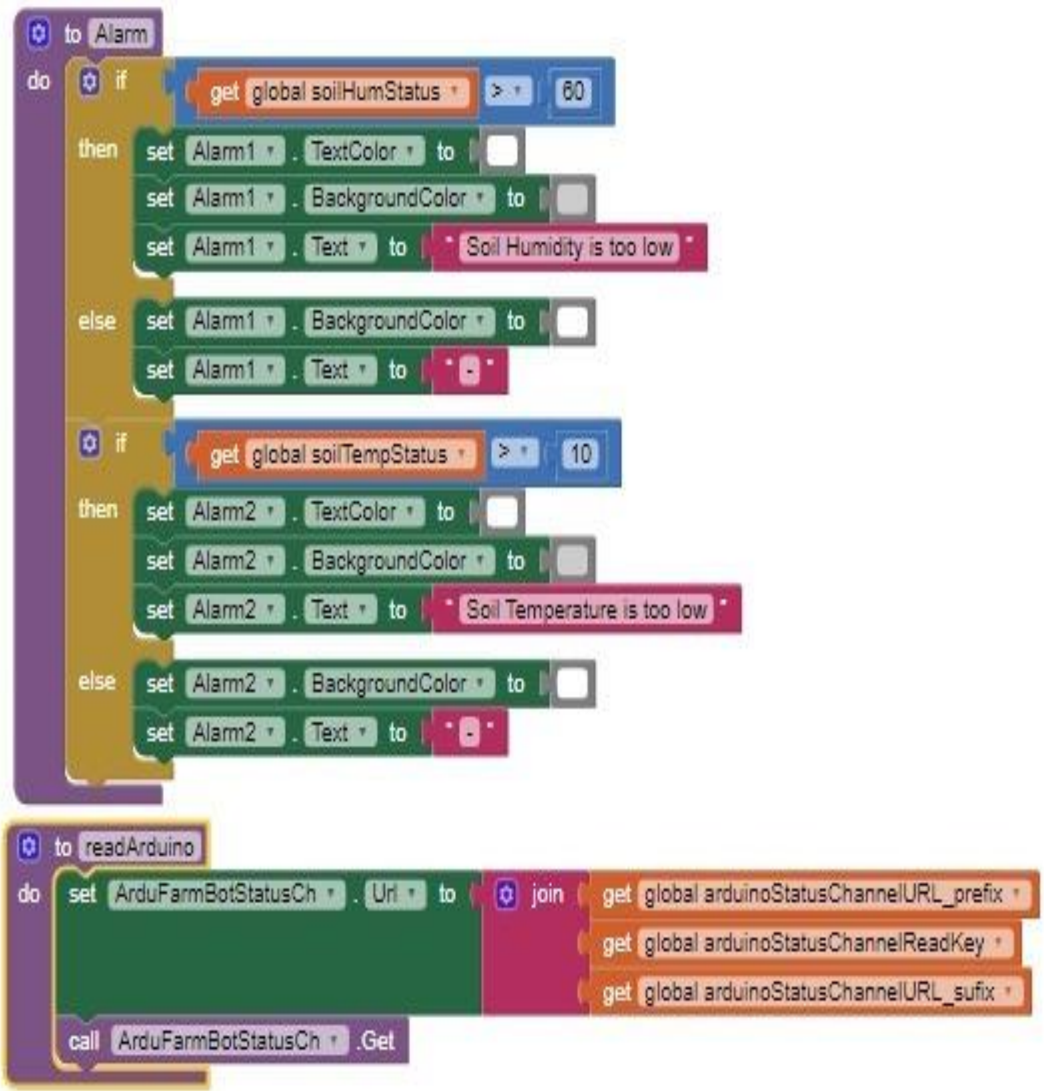
```
    print("Confusion Matrix: ",

        confusion_matrix(y_test, y_pred))


    print ("Accuracy : ",

    accuracy_score(y_test,y_pred)*100)


    print("Report : ",

    classification_report(y_test, y_pred))


# Driver code

def main():


    # Building Phase

    data = importdata()

    X, Y, X_train, X_test, y_train, y_test = splitdataset(data)

    clf_gini = train_using_gini(X_train, X_test, y_train)

    clf_entropy = tarin_using_entropy(X_train, X_test, y_train)


    # Operational Phase

    print("Results Using Gini Index:")


    # Prediction using gini

    y_pred_gini = prediction(X_test, clf_gini)

    cal_accuracy(y_test, y_pred_gini)
```

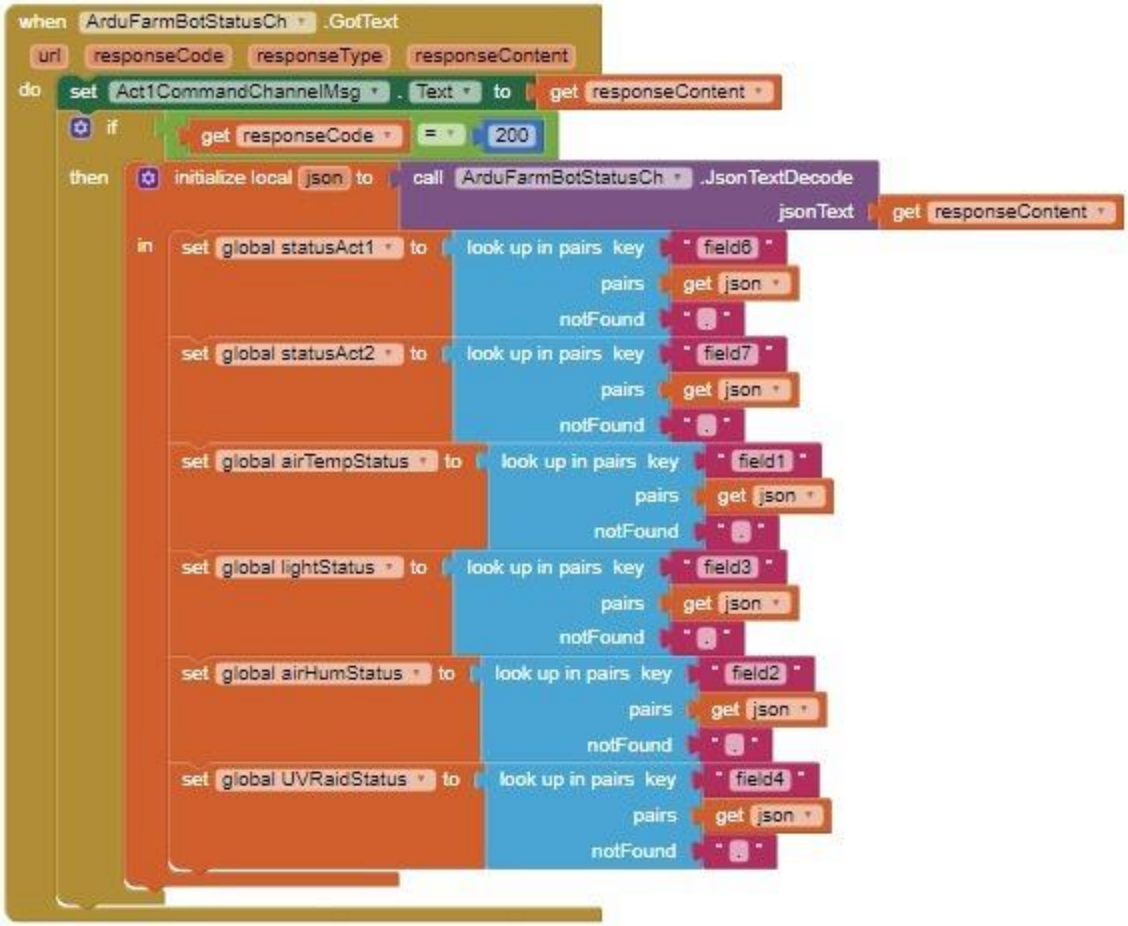
```
    print("Results Using Entropy:")

    # Prediction using entropy

    y_pred_entropy = prediction(X_test, clf_entropy)

    cal_accuracy(y_test, y_pred_entropy)



# Calling main function

if __name__=="__main__":

    main()
```

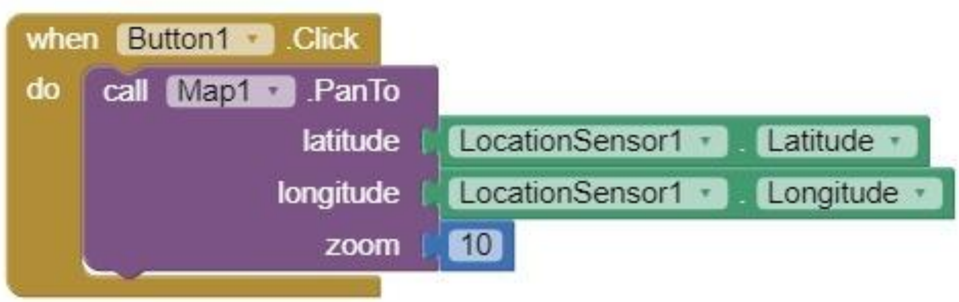
Mobile App Code:

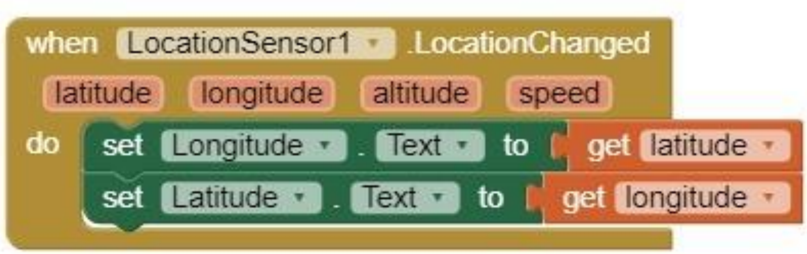


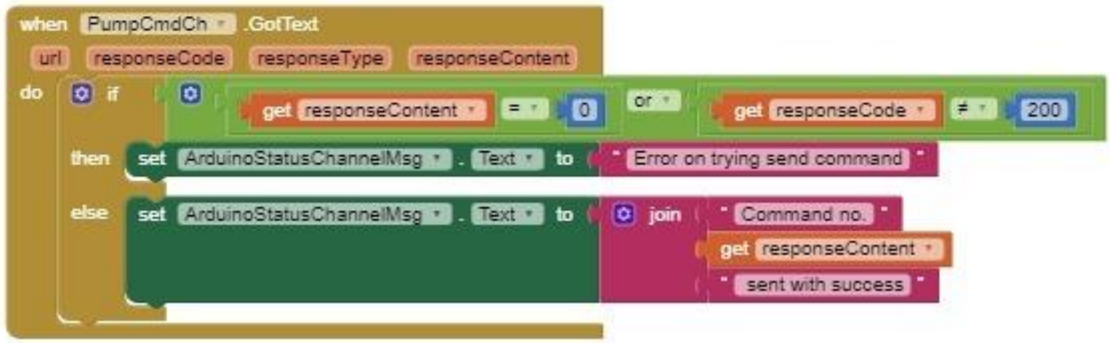

initialize global act1CommandChannelURL_prefix to " https://api.thingspeak.com/update?api_key= "
initialize global act1CommandChannelWriteKey to " CK7VI3KOS2KD2RMX "
initialize global act1CommandChannelURL_sufix to " &field1= "


to Alarm
do
if get global soilHumStatus > 60
then set Alarm1 . TextColor to
set Alarm1 . BackgroundColor to
set Alarm1 . Text to " Soil Humidity is too low "
else set Alarm1 . BackgroundColor to
set Alarm1 . Text to " - "
if get global soilTempStatus > 10
then set Alarm2 . TextColor to
set Alarm2 . BackgroundColor to
set Alarm2 . Text to " Soil Temperature is too low "
else set Alarm2 . BackgroundColor to
set Alarm2 . Text to " - "
to readArduino
do set ArduFarmBotStatusCh . Url to join
get global arduinoStatusChannelURL_prefix
get global arduinoStatusChannelReadKey
get global arduinoStatusChannelURL_sufix
call ArduFarmBotStatusCh .Get

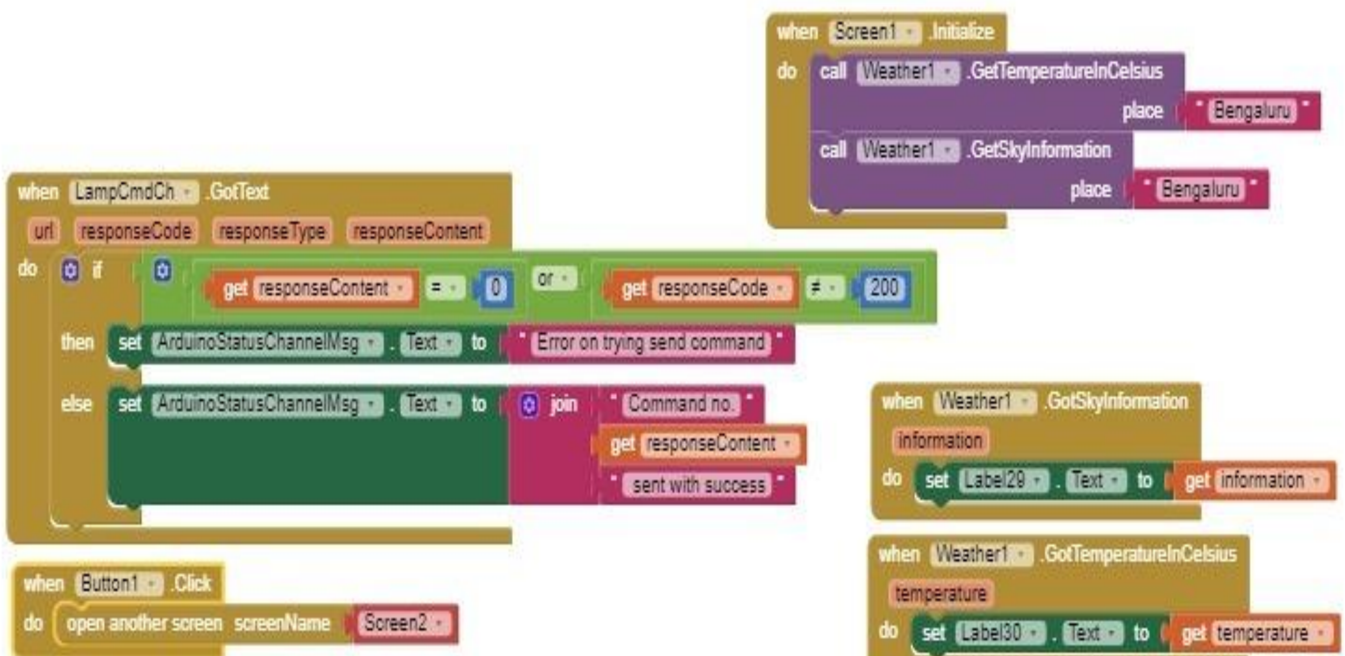

when ArduFarmBotStatusCh .GotText
url responseCode responseType responseContent
do set Act1CommandChannelMsg . Text to get responseContent
if get responseCode = 200
then initialize local json to call ArduFarmBotStatusCh .JsonTextDecode
jsonText get responseContent
in set global statusAct1 to look up in pairs key " field6 "
pairs get json
notFound " "
set global statusAct2 to look up in pairs key " field7 "
pairs get json
notFound " "
set global airTempStatus to look up in pairs key " field1 "
pairs get json
notFound " "
set global lightStatus to look up in pairs key " field3 "
pairs get json
notFound " "
set global airHumStatus to look up in pairs key " field2 "
pairs get json
notFound " "
set global UVRaidStatus to look up in pairs key " field4 "
pairs get json
notFound " "


when Button1 .Click
do call Map1 .PanTo
latitude LocationSensor1 . Latitude
longitude LocationSensor1 . Longitude
zoom 10


when LocationSensor1 .LocationChanged
latitude longitude altitude speed
do set Longitude . Text to get latitude
set Latitude . Text to get longitude


when PumpCmdCh .GotText
url responseCode responseType responseContent
do if
get responseContent = 0 or get responseCode ≠ 200
then set ArduinoStatusChannelMsg . Text to "Error on trying send command"
else set ArduinoStatusChannelMsg . Text to join
"Command no."
get responseContent
"sent with success"

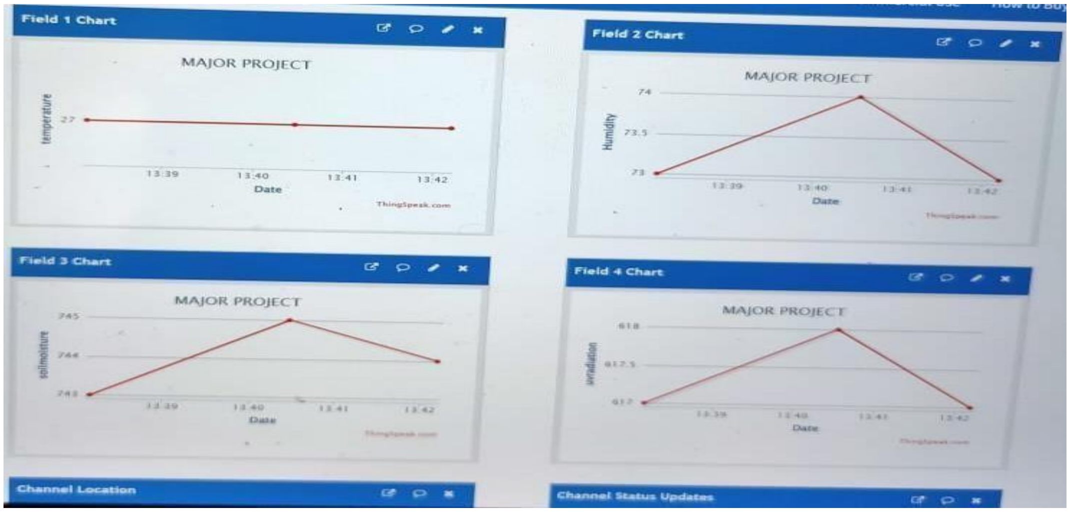

initialize global act2CommandChannelURL_prefix to "https://api.thingspeak.com/update?api_key="
initialize global act2CommandChannelWriteKey to "CK7VI3KOS2KD2RMX"
initialize global act2CommandChannelURL_sufix to "&field1="


when Screen1 .Initialize
do call Weather1 .GetTemperatureInCelsius
place "Bengaluru"
call Weather1 .GetSkyInformation
place "Bengaluru"
when LampCmdCh .GotText
url responseCode responseType responseContent
do if
get responseContent = 0 or get responseCode ≠ 200
then set ArduinoStatusChannelMsg . Text to "Error on trying send command"
else set ArduinoStatusChannelMsg . Text to join
"Command no."
get responseContent
"sent with success"
when Weather1 .GotSkyInformation
information
do set Label29 . Text to get information
when Button1 .Click
do open another screen screenName Screen2
when Weather1 .GotTemperatureInCelsius
temperature
do set Label30 . Text to get temperature

CHAPTER 8
TESTING AND RESULTS

CHAPTER 8

TESTING AND RESULTS



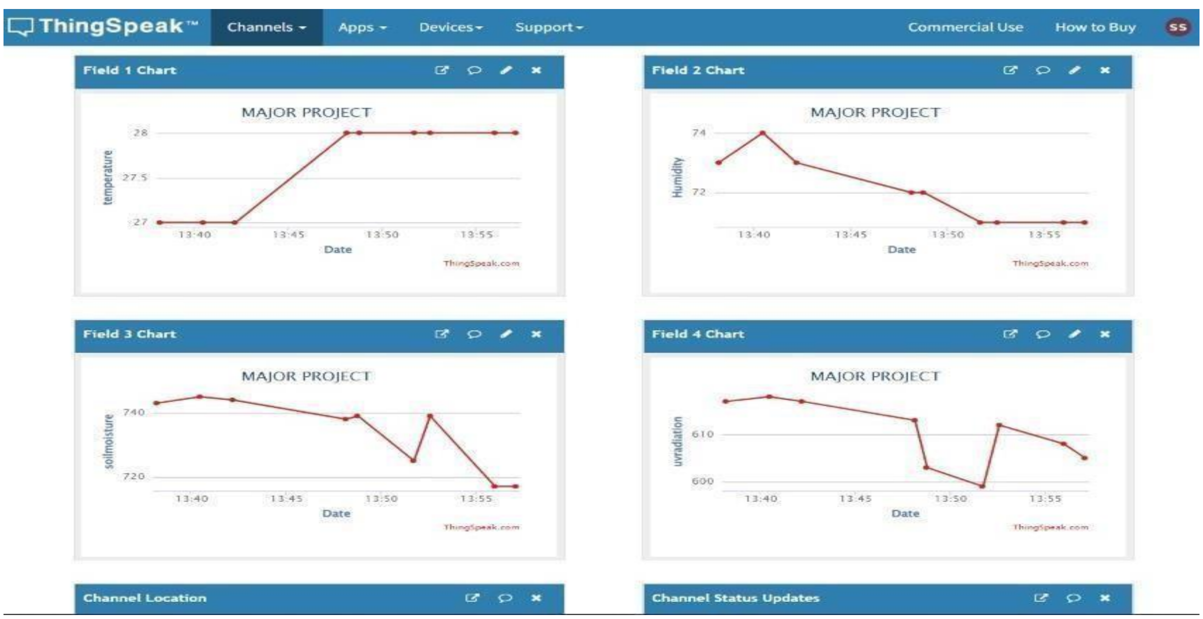


Fig8. Displays the Real-Time Sensors value from cloud

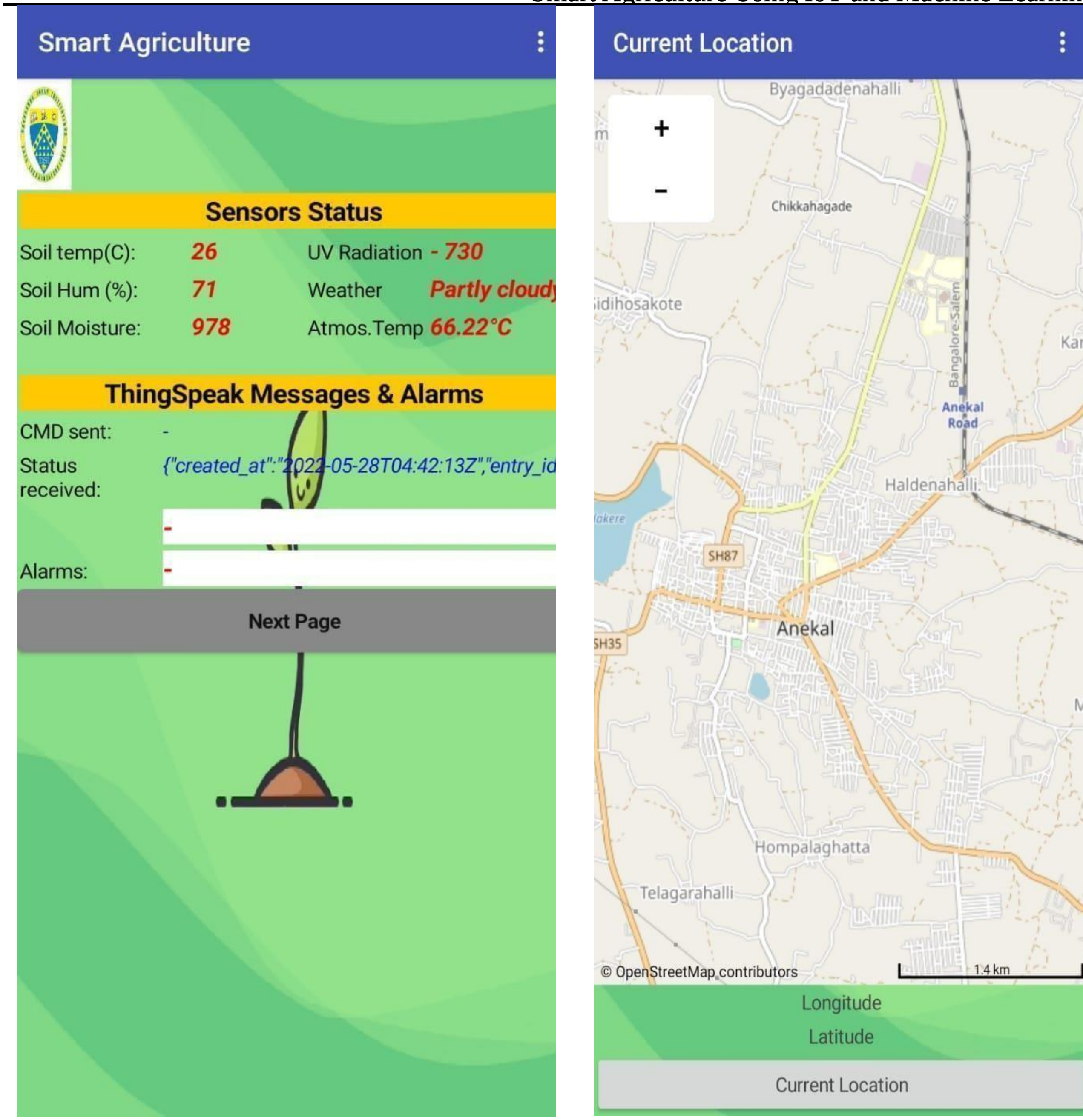


Fig9. Cloud data and system location

```
PS F:\Major Project> python app.py
                         date  entry_id  temperature  humindity  soilmoisture  uvradiation
0    2022-05-26T08:38:47+00:00         1           28         71           974          728
1    2022-05-26T08:39:20+00:00         2           28         71           968          724
2    2022-05-26T08:39:57+00:00         3           28         70           973          728
3    2022-05-26T08:41:40+00:00         4           27         71           967          724
4    2022-05-26T08:42:29+00:00         5           27         71           983          724
..                         ...       ...          ...        ...           ...          ...
98   2022-05-26T11:30:35+00:00        99           27         67           968          725
99   2022-05-26T11:36:06+00:00       100           27         66           985          743
100  2022-05-26T11:36:39+00:00       101           27         66           976          724
101  2022-05-26T11:38:36+00:00       102           27         66           976          730
102  2022-05-26T11:38:54+00:00       103           27         66           977          730

[103 rows x 6 columns]
PS F:\Major Project>
```

```
PS F:\Major Project> python app1.py
Dataset Length:  103
Dataset Shape:  (103, 6)
Dataset:                         date  entry_id  temperature  humindity  soilmoisture  uvradiation
0  2022-05-26T08:38:47+00:00         1           28         71           974          728
1  2022-05-26T08:39:20+00:00         2           28         71           968          724
2  2022-05-26T08:39:57+00:00         3           28         70           973          728
3  2022-05-26T08:41:40+00:00         4           27         71           967          724
4  2022-05-26T08:42:29+00:00         5           27         71           983          724
Results Using Gini Index:
Predicted values:
['2022-05-26T08:38:47+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T09:59:45+00:00'
 '2022-05-26T09:59:45+00:00' '2022-05-26T10:33:55+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T10:32:55+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T09:59:45+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T10:32:55+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T10:32:55+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T10:33:55+00:00'
 '2022-05-26T09:59:45+00:00' '2022-05-26T09:59:45+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T10:52:25+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T08:38:47+00:00']
Confusion Matrix:  [[0 0 0 ... 0 0 0]
 [1 0 0 ... 0 0 0]
```

```
0  2022-05-26T08:38:47+00:00        1          28         71          974         728
1  2022-05-26T08:39:20+00:00        2          28         71          968         724
2  2022-05-26T08:39:57+00:00        3          28         70          973         728
3  2022-05-26T08:41:40+00:00        4          27         71          967         724
4  2022-05-26T08:42:29+00:00        5          27         71          983         724
Results Using Gini Index:
Predicted values:
['2022-05-26T08:38:47+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T09:59:45+00:00'
 '2022-05-26T09:59:45+00:00' '2022-05-26T10:33:55+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T10:32:55+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T09:59:45+00:00' '2022-05-26T09:41:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T10:32:55+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T10:32:55+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T10:33:55+00:00'
 '2022-05-26T09:59:45+00:00' '2022-05-26T09:59:45+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T10:52:25+00:00' '2022-05-26T08:38:47+00:00'
 '2022-05-26T08:38:47+00:00']
Confusion Matrix:  [[0 0 0 ... 0 0 0]
 [1 0 0 ... 0 0 0]
 [1 0 0 ... 0 0 0]
 ...
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
Accuracy :  0.0
```

```
2022-05-26T09:30:46+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:33:09+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:35:46+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:38:07+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:41:01+00:00       0.00      0.00      0.00       0.0
2022-05-26T09:45:15+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:58:44+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:59:45+00:00       0.00      0.00      0.00       0.0
2022-05-26T10:15:10+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:28:26+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:32:55+00:00       0.00      0.00      0.00       0.0
2022-05-26T10:33:55+00:00       0.00      0.00      0.00       0.0
2022-05-26T10:36:25+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:37:44+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:46:30+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:52:25+00:00       0.00      0.00      0.00       0.0
2022-05-26T10:53:17+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:54:27+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:02:44+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:07:50+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:09:30+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:11:54+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:14:24+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:24:05+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:30:35+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:36:06+00:00       0.00      0.00      0.00       1.0

                 accuracy                           0.00      31.0
                macro avg       0.00      0.00      0.00      31.0
             weighted avg       0.00      0.00      0.00      31.0

Results Using Entropy:
Predicted values:
```

```
Predicted values:
['2022-05-26T08:38:47+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T10:56:25+00:00' '2022-05-26T11:18:01+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:17:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T11:18:01+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:17:01+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T08:39:20+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T10:56:25+00:00' '2022-05-26T10:56:25+00:00'
 '2022-05-26T10:56:25+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T11:18:01+00:00' '2022-05-26T10:56:25+00:00'
 '2022-05-26T08:38:47+00:00']
Confusion Matrix:  [[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
Accuracy :  0.0
```

2022-05-26T09:28:10+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:28:52+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:30:46+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:33:09+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:35:46+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:38:07+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:45:15+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:58:44+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:02:13+00:00	0.00	0.00	0.00	0.0
2022-05-26T10:15:10+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:17:01+00:00	0.00	0.00	0.00	0.0
2022-05-26T10:28:26+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:36:25+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:37:44+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:46:30+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:53:17+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:54:27+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:56:25+00:00	0.00	0.00	0.00	0.0
2022-05-26T11:02:44+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:07:50+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:09:30+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:11:54+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:14:24+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:18:01+00:00	0.00	0.00	0.00	0.0
2022-05-26T11:24:05+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:30:35+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:36:06+00:00	0.00	0.00	0.00	1.0
accuracy			0.00	31.0
macro avg	0.00	0.00	0.00	31.0
weighted avg	0.00	0.00	0.00	31.0

Fig10. Showing the efficiency of the machine learning algorithm

REFERENCES

1.1. G. Eason, B. Noble, and I.N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. A. Lieto, M. Bhatt, A. Oltramari, D. Vernon (2018). The role of cognitive architectures in general artificial intelligence, Cognitive Systems Research, 2018; 48:1-3. doi: https://doi.org/10.1016/j.cogsys.2017.08.003.

1.2. J. Jin, C. Song, H. Li, K. Gai, J. Wang and W. Zhang (2018). Real-Time Bidding with Multi-Agent Reinforcement Learningin Display Advertising. CIKM '18: Proceedings of the 27th ACM International Conference.

1.3. Sameer. M. Patel1, Mittal. B. Jain2, Sarvesh. S. Pai3, Sagar. D. Korde4," SmartAgriculture using IoT and Machine Learning", International Research Journal of Engineering and Technology, April, 2021.

1.4. Kasara Sai Pratyush Reddy, Y Mohana Roopa, Narra Sai Nandan, Kovvada Rajeev L N," IoT based Smart Agriculture using Machine Learning", IEEE, ICIRCA 2020.

1.5. Shridhar M, Chinmay P, Piyush W, Aniket P, Vaishali D, "A Survey on Predicting Suitable Crops for Cultivation Using IoT," International Journal of Innovative Research in Computer and Communication Engineering, January, 2017.

1.6. T Raghav, Bhagavatula A, Aashish S, Drishti J, Natesh B, Varshini S, “Smart Management of Crop Cultivation using IOT and Machine Learning,” International Research Journal of Engineering and Technology, November, 2018.

1.7. R. Pallavi R, B. Vinitha, K. Rishita, K. Pranavi, “Crop Monitoring and Recommendation System using Machine Learning and IoT”, International Journal of Innovative Technology and Exploring Engineering, July, 2020 Vol-9 Issue-9.

1.8. Anuja Chandgude, Nikita Harpale, Diksha Jadhav, Punam Pawar and Suhas M. Patil, “A Review on Machine Learning Algorithm Used for Crop Monitoring System in Agriculture”, International Research journal of Engineering and Technology (IRJET), vol. 05, Issue 4, Apr.2018.

1.9. M. K. Gayatri, J. Jayasakthi andG.S.Anandha Mala, “Providing Smart Agriculture Solutions to Farmers for better yielding using IoT”, IEEE Technological Innovation in ICT for Agriculture and Rural Development (TIAR), 10-12 Jul. 2015.

1.10. Kiran Shinde, Jerrin Andrei, AmeyOke, “Web Based Recommendation System for Farmers”, International Journal on Recent and Innovation Trends in Computing and Communication 2, Volume: 3, Mar. 2014.

1.11. Rushika G, Juilee K, Pooja M, Sachee N, Priya R, “Prediction of Crop Yield using Machine Learning,”International Research Journal of Engineering and Technology, February, 2018.

APPENDIX

Arduino IDE

Arduino IDE is a platform which allows to program the arduino microcontrollers which we discussed in the hardware components. We can code the micro controller according to our application and its syntax is relatively simple. It is an open source and we can use embedded C to program. It's simple to use this IDE as there are readymade code examples. The editor has many features which makes the programming look easy. The programs are known as sketches in this platform. The procedure of using this IDE is selecting the board and writing the appropriate code for it. Once the code is completed, compile and check for errors and then upload code to the microcontroller. Once the code is uploaded the board performs the specified operation.

MIT app inventor

It is a web application integrated development environment originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT). It allows developers to create application software (apps) for two operating systems (OS): Android, and iOS. It uses a graphical user interface (GUI) very similar to the programming languages Scratch (programming language) and the Star Logo, which allows users to drag and drop visual objects to create an application that can run on Android devices.

Decision tree

Decision trees use multiple algorithms to decide to split a node into two or more sub-nodes. The creation of sub-nodes increases the homogeneity of resultant sub-nodes. In other words, we can say that the purity of the node increases with respect to the target variable.

Research Paper Details

- Paper Title: **SMART AGRICULTURE USING IOT AND MACHINE LEARNING**
- Status :
- Conference date :
- Conference location :
- Conference publication :

GitHub Link

MAJOR PROJECT LINK - https://github.com/ENG18CS0160MANOJP/SMART-AGRICULTURE-USING-IOT-AND-MACHINE-LEARNING

SMART AGRICULTURE USING IOT AND MACHINE LEARNING

Asst. Prof. Vaidehi Verma, Manoj.P, Shejan Shriram.R, Shreyas.U, Shyamsundar.B, Surya.S
Department of Computer Science Engineering,
Dayananda Sagar University
Bangalore-560068

Abstract – **Agriculture plays a very important role in both fields, such as food necessity for human beings and providing necessary stocks for many food industries, and it is one of the most effective and the backbone of India. The future of innovation in creating farming methods is moderately reinforcing the crop yield to make it more commercial and reduce irrigation debris. In this research paper, we are pleased to introduce our prototype for Smart Agriculture using IoT and Machine Learning. Firstly, we will construct a greenhouse and then test different kinds of crops grown inside. By using IoT devices, we will collect various datasets consisting of moisture, temperature, and humidity, which are the three most vital parameters that are required in any agriculture field. This system comprises temperature, humidity, and moisture sensors, installed in the greenhouse, and sends data through an Arduino Board, developing an IoT device with the cloud. Machine learning algorithms are applied to the dataset which is collected from the greenhouse field to predict results proficiently.**

Keywords*:* IoT, Machine learning, plants.

I. INTRODUCTION

Agriculture plays a very important role; it is one of the main sources of food and a majority of people in India are counting on agriculture as their major source of income. The development and utilization of the Smart Agriculture structure are based on IoT and Machine Learning is advanced in the field of agriculture division by not only boosting the crop production but also making it cost effectual. There is no dubiety that the government needs to invest in the agriculture sector for it to bloom. The effects of desperate changes in climatic infirmity have seen crop yield drop by more than a sector. There needs to be a focus on the accomplishment of smart automation in the field of agriculture to yield quality and mass production of crops. The collaboration of IoT and Machine Learning can surely, help in decrement of cost and also help grow the scale of manufacturing through the collecting of time sequence data or the information from sensors. There are certain elements, which play a major role in the production of crops. Such as humidity, temperature and soil moisture.

II. LITERATURE SURVEY

• AI and IoT Based Monitoring System for Increasing the Yield in Crop Production Richa Singh, Sarthak Srivastava, Rajan Mishra AI and (0) IoT. The research is performed on a marigold plant to detect the most suitable conditions for the plant. The effect of physical conditions like humidity, temperature, soil temperature and moisture and light intensity on the plant growth, is monitored using IoT based monitoring system (ICE3 - February 2020)

• IoT based Smart Agriculture using Machine Learning Kasara Sai Pratyush Reddy, Y Mohana Roopa, Kovvada Rajeev L N, Narra Sai Nandan IoT and Machine Learning The system is programmed to be trained from the given dataset using all the sensed data from the soil moisture, temperature and humidity. By applying the decision tree learning algorithm to the real-time data its processes and generates an output yes/no and sends the decision to (IEEE – 04 September 2020) the farmer through an email. Using this decision, a farmer can decide to water the crop only when required.

• Smart Agriculture Using IoT and Machine Learning Sameer M, Mittal B, Sarvesh S, Sagar D IoT and Machine Learning The real-time readings coming from the sensors along with the application of Machine Learning Algorithms will not only help farmers make informed decisions on which crop to grow in a particular region but also recommend fertilizers based on various factors like soil condition, climatic conditions (1) etc. In addition, from the various machine-learning algorithms implemented, XGBoost seems to give the best results with 31% accuracy on the recommendations (IRJET - 04 April 2021)

• IoT and Machine Learning Approach for Automation of Anneketh Vij, Singh Vijendra, IoT and Machine Learning

It provides a sustainable and computationally efficient approach based on IoT based establishing (ICCIDS – 2019) Farm Irrigation System Abhishek, Shivam, Aashima Bassi, Arushi Sharma a proper distributed network contributes to the accuracy of the predictions made by SVR, Random Forrest Sensor node inter-connectivity will help monitor the complete field thoroughly. To implement a system which would be mobile and can help in every stage of farming. (2)
• Smart Agriculture Using IoT and Machine Learning.
• Sumiksha Shetty & A. B. Smitha IoT and Machine Learning In this paper, the assistance of IoT and machine learning in farming, can expand the productivity (3)
• Smart Agriculture Using IoT and Machine Learning Sumiksha Shetty & A. B. of yield Different climate parameters are taken into thought from which the best reasonable yield is grown is predicted by a supervised (4) learning algorithm, decision tree, utilized in regions like sickness recognition, crop discovery, irrigation system, soil conditions and also the product quality and market analysis.

III. METHODOLOGY

In this Project, the most important and implemented concept is the Internet of Things (IoT). There will be different types of devices that will measure different variables of the surroundings like temperature, humidity, soil moisture, UV radiation, etc

Then the data received from these devices or sensors will be then be transferred to the Thinkspeak. com cloud Database, which displays the data given from the sensors in the form of a graph. After this, the data collected from the sensors will be stored in a CVS Configuration File and will be used to implement Machine Learning Algorithms which will predict the growth of the plant-based on the surroundings. On the other hand, there will be an option to connect the device to a phone application with the help of a Bluetooth module. The data will then be pushed to the database server using the phone. (6) The major part of the project will be a hardware device that constitutes of different kinds of sensors such as Soil Moisture Sensor (EK1361), Humidity and Temperature Sensor (DHT22), and UV Radiation Sensor (ML-8511).

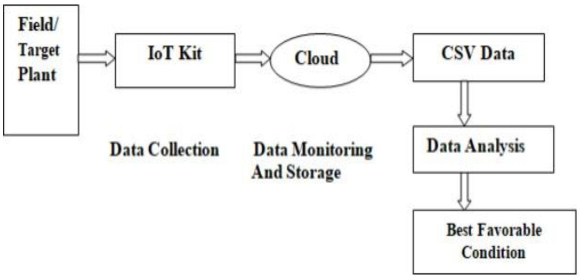


Fig1. Workflow Diagram of Smart Agriculture using IoT And Machine Learning

The device collects the attributes of the surrounding (soil moisture, temperature, radiation intensity) independently and processes. The WIFI Module ESP8266 will then send the received data immediately to the Think speak database. And after Implementing Different Machine Learning Algorithms we can predict the Yield or growth of the plant. And also, from the mobile application, we can view the data received from the sensors. The received data will be utilized to process requirements that a farmer needs to take care of, for maximum yield of the crops.

IV. PROPOSED METHODOLOGY

The major part of the proposed methodology will be a hardware device that constitutes of different kinds of sensors such as Soil Moisture Sensor (EK1361), Humidity and Temperature Sensor (DHT22), pH Sensor (SKU: SEN0161) and UV Radiation Sensor (ML-8511). (8) The device collects the attributes of the surrounding (soil moisture, temperature, pH, radiation intensity) independently and processes them with the help of its microcontroller ATMega328p.2 The microcontroller will then send the received data immediately to the AWS S3 database and open source free think speak cloud. In the proposed methodology, the main concept implemented is the Internet of Things (IoT). There will be a low-level hardware device that will measure different variables of the surroundings like temperature, humidity, soil moisture, UV radiation, etc. (9) The measured values will then be transferred to the AWS S3 database or Thinkspeak cloud database, which will later be pulled back to a mobile application for further processing. On the other hand, there will be an option to connect the device to a phone application with the help of an API key in the Thinkspeak cloud database. The data will then be pushed to the database server using the phone. (6) The major part of the proposed methodology will be a hardware device that constitutes of different kinds of sensors such as Soil Moisture Sensor (EK1361), Humidity and Temperature Sensor (DHT22), and UV Radiation Sensor (ML-8511). The device collects the attributes of the surrounding (soil moisture, temperature, radiation intensity) independently and processes them with the help of its microcontrollerATMega328p.

The microcontroller will then send the received data immediately to the AWS S3 database or think to speak cloud database. On the other hand, a Bluetooth module will be connected to the microcontroller so that the (6) user gets to connect to the device locally, and send the data manually through a Mobile application.

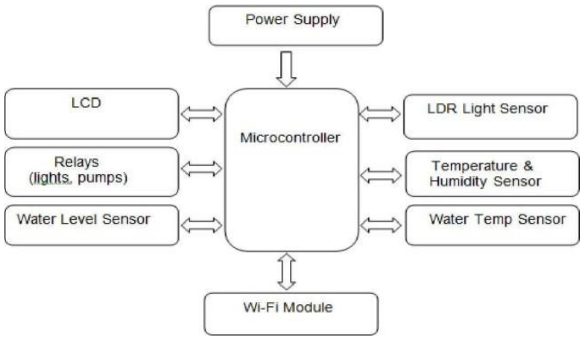


Fig2. Block diagram for IoT Connections

The received data will be utilized to process requirements that a farmer needs to take care of, for maximum yield of the crops. To interface the user with the device's data. The data received from the hardware device will be analyzed using different machine learning algorithms and researched agricultural datasets in the past. According to the results from different backend machine learning algorithms, a suggestion template will be developed. so that the farmer knows what to do next. The main challenge will be to collect different agricultural datasets that will be used to analyse the data received from the hardware device. (12) The mobile application will be fully focused on the data visualization and helpful suggestions of each user.

V. FIGURES AND TABLES

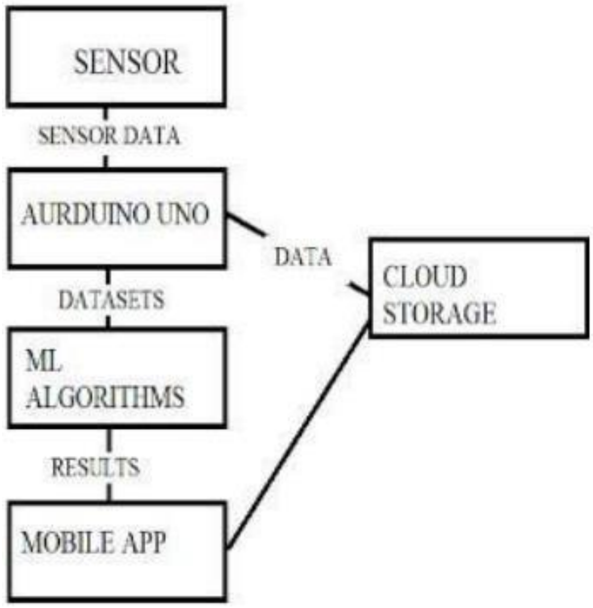


Fig4. Interconnection of Systems and Dataflow.

Fig3. Smart agriculture using IoT with Greenhouse protocol

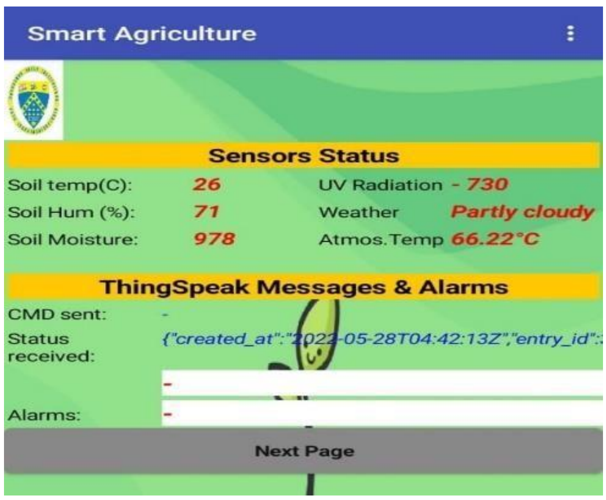


Fig6. Real Time Sensors data in Mobile Application

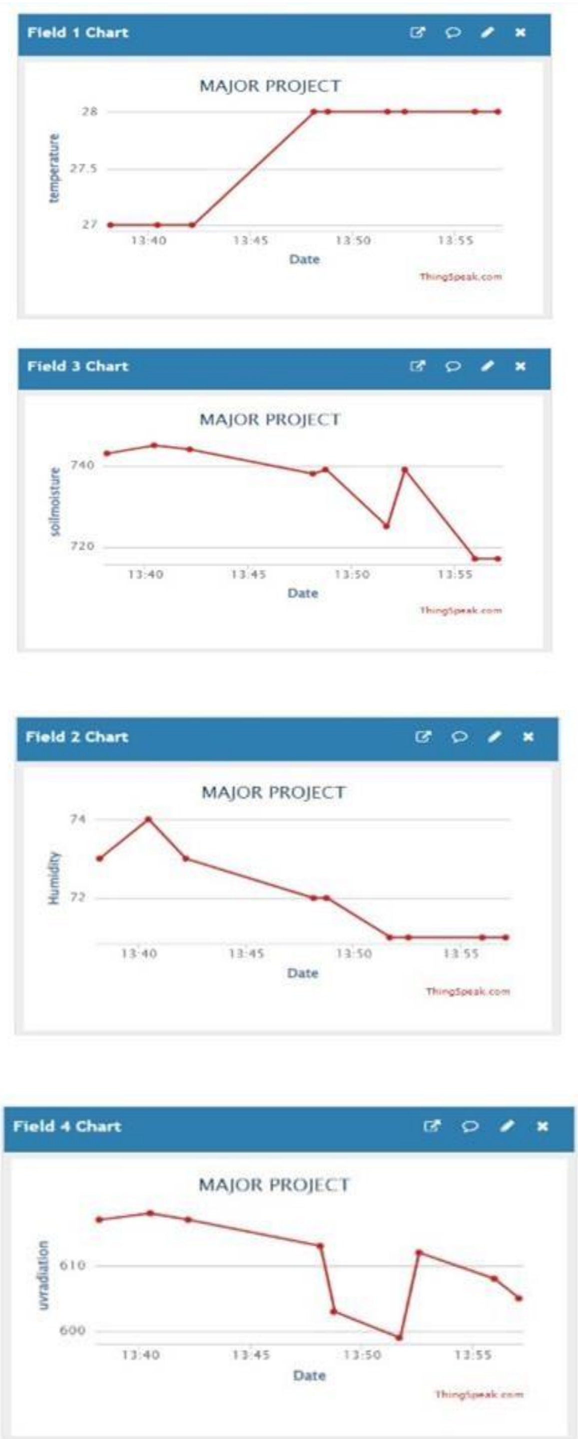
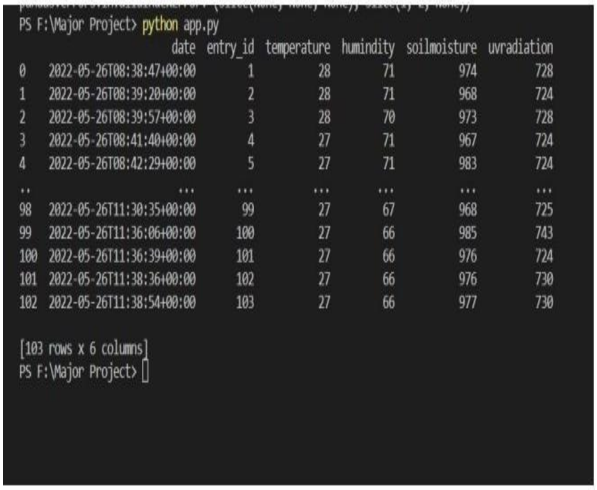


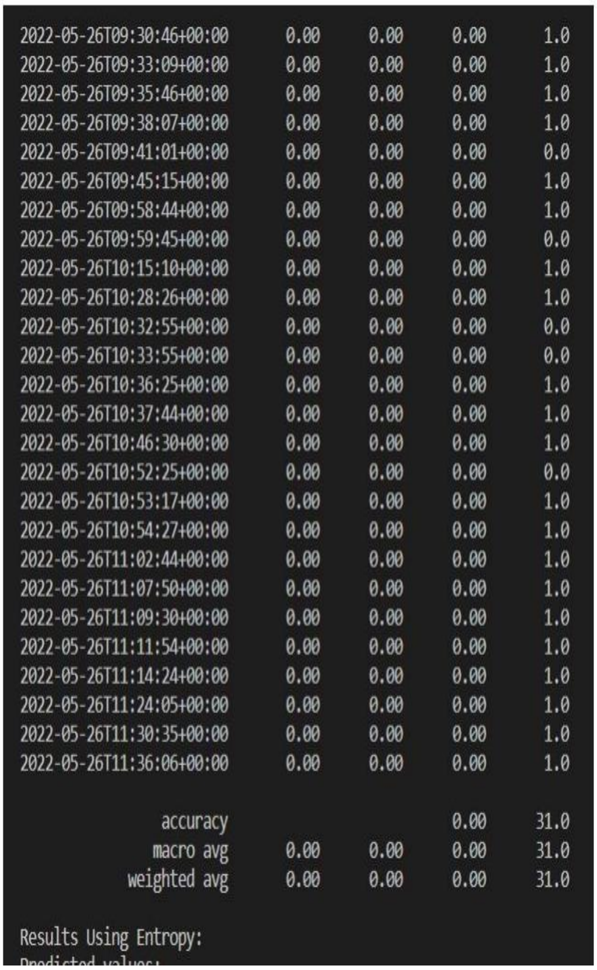
Fig5. Graphical Representation of Sensors Data.



PS F:\Major Project> python app.py

	date	entry_id	temperature	humindity	soilmoisture	uvradiation
0	2022-05-26T08:38:47+00:00	1	28	71	974	728
1	2022-05-26T08:39:20+00:00	2	28	71	968	724
2	2022-05-26T08:39:57+00:00	3	28	70	973	728
3	2022-05-26T08:41:40+00:00	4	27	71	967	724
4	2022-05-26T08:42:29+00:00	5	27	71	983	724
..	...	...	...	...	...	...
98	2022-05-26T11:30:35+00:00	99	27	67	968	725
99	2022-05-26T11:36:06+00:00	100	27	66	985	743
100	2022-05-26T11:36:39+00:00	101	27	66	976	724
101	2022-05-26T11:38:36+00:00	102	27	66	976	730
102	2022-05-26T11:38:54+00:00	103	27	66	977	730

[103 rows x 6 columns]
PS F:\Major Project>



2022-05-26T09:30:46+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:33:09+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:35:46+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:38:07+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:41:01+00:00	0.00	0.00	0.00	0.0
2022-05-26T09:45:15+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:58:44+00:00	0.00	0.00	0.00	1.0
2022-05-26T09:59:45+00:00	0.00	0.00	0.00	0.0
2022-05-26T10:15:10+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:28:26+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:32:55+00:00	0.00	0.00	0.00	0.0
2022-05-26T10:33:55+00:00	0.00	0.00	0.00	0.0
2022-05-26T10:36:25+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:37:44+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:46:30+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:52:25+00:00	0.00	0.00	0.00	0.0
2022-05-26T10:53:17+00:00	0.00	0.00	0.00	1.0
2022-05-26T10:54:27+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:02:44+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:07:50+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:09:30+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:11:54+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:14:24+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:24:05+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:30:35+00:00	0.00	0.00	0.00	1.0
2022-05-26T11:36:06+00:00	0.00	0.00	0.00	1.0
accuracy			0.00	31.0
macro avg	0.00	0.00	0.00	31.0
weighted avg	0.00	0.00	0.00	31.0

Results Using Entropy:

```
Predicted values:
['2022-05-26T08:38:47+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T10:56:25+00:00' '2022-05-26T11:18:01+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:17:01+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T11:18:01+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:17:01+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T08:39:20+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T08:39:20+00:00' '2022-05-26T10:02:13+00:00'
 '2022-05-26T10:56:25+00:00' '2022-05-26T10:56:25+00:00'
 '2022-05-26T10:56:25+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T08:38:47+00:00' '2022-05-26T08:42:29+00:00'
 '2022-05-26T11:18:01+00:00' '2022-05-26T10:56:25+00:00'
 '2022-05-26T08:38:47+00:00']
Confusion Matrix:  [[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
Accuracy :  0.0
```

```
2022-05-26T09:28:10+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:28:52+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:30:46+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:33:09+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:35:46+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:38:07+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:45:15+00:00       0.00      0.00      0.00       1.0
2022-05-26T09:58:44+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:02:13+00:00       0.00      0.00      0.00       0.0
2022-05-26T10:15:10+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:17:01+00:00       0.00      0.00      0.00       0.0
2022-05-26T10:28:26+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:36:25+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:37:44+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:46:30+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:53:17+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:54:27+00:00       0.00      0.00      0.00       1.0
2022-05-26T10:56:25+00:00       0.00      0.00      0.00       0.0
2022-05-26T11:02:44+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:07:50+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:09:30+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:11:54+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:14:24+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:18:01+00:00       0.00      0.00      0.00       0.0
2022-05-26T11:24:05+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:30:35+00:00       0.00      0.00      0.00       1.0
2022-05-26T11:36:06+00:00       0.00      0.00      0.00       1.0

                 accuracy                           0.00      31.0
                macro avg       0.00      0.00      0.00      31.0
             weighted avg       0.00      0.00      0.00      31.0
```

Fig7. Showing the efficiency of the machine learning algorithm

V. EXPERIMENTAL RESULTS

We used different platforms to get crop data with all the above-mentioned attributes and got some datasets from the agricultural college G.K.V.K as well as from other sources (like Kaggle), we have used our monitored data for this existing project to do experiment. The training dataset used for the training of crop prediction model contains features like nitrogen, potassium, phosphorus, temperature, humidity, and rainfall whereas the fertilizer recommendation model contains features like nitrogen, phosphorus, potassium, pH and soil moisture. Various Machine learning algorithms were applied against the training dataset - Decision Tree, Naive Bayes, Support Vector Machine, Logistic Regression, Random Forest and XGBoost, and were compared based on the model's accuracy. The user, which is a Farmer, will be able to use our platform to get accurate recommendations regarding which crop to grow based on different features like humidity, pH, and rainfall. Besides that, the user will be able to predict yield based on different features like moisture, nitrogen, phosphorus and potassium.

VII. CONCLUSION

We conclude that the sensors (such as temperature and humidity sensor, Soil moisture sensor, and UV radiation sensor) which will be fixed in the greenhouse are taken as input, through the microcontroller and esp8266 Wi-Fi module and the data received from the sensors will be sent to Thinkspeak cloud. Then this data collected from sensors will be analyzed by a machine learning algorithm (decision tree) by this algorithm using known parameters we can predict the unknown parameter (growth rate of the plant) and the current sensors value will be displayed through application which is created using MIT mobile application software.

IX. APPENDIX

- **Arduino IDE**

Arduino IDE is a platform which allows to program the Arduino microcontrollers which we discussed in the hardware components. We can code the micro controller according to our application and its syntax is relatively simple. It is an open source and we can use embedded C to program. It's simple to use this IDE as there are readymade code examples. The editor has many features which makes the programming look easy. The programs are known as sketches in this platform. The procedure of using this IDE is selecting the board and writing the appropriate code for it. Once the code is completed, compile and check for errors and then upload code to the microcontroller. Once the code is uploaded the board performs the specified operation.

- **MIT app inventor**

It is a web application integrated development environment originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT). It allows developers to create application software (apps) for two operating systems (OS): Android, and iOS. It uses a graphical user interface (GUI) very similar to the programming languages Scratch (programming language) and the Star Logo, which allows users to drag and drop visual objects to create an application that can run on Android devices.

- **Decision tree**

Decision trees use multiple algorithms to decide to split a node into two or more sub-nodes. The creation of sub-nodes increases the homogeneity of resultant sub-nodes. In other words, we can say that the purity of the node increases with respect to the target variable.

X. ACKNOWLEDGMENT

It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of this project work. First, we take this opportunity to express our sincere gratitude to the School of Engineering & Technology, Dayananda Sagar University for providing us with a great opportunity to pursue our Bachelor's degree in this institution. We would like to thank Dr. A Srinivas. Dean, School of Engineering & Technology, Dayananda Sagar University for his constant encouragement and expert advice. It is a matter of immense pleasure to express our sincere thanks to Dr. Girisha G S, Department Chairman, Computer Science and Engineering, Dayananda Sagar University, for providing the right academic guidance that made our task possible. We would like to thank our guide Professor Vaidehi Verma, Associate Professor, Dept. of Computer Science and Engineering, Dayananda Sagar University, for sparing her valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project. We would like to thank our Project Coordinator Dr. Meenakshi Malhotra and Dr. Bharanidharan N, and all the staff members of Computer Science and Engineering for their support. We are also grateful to our family and friends who provided us with every requirement throughout the course. We would like to thank one and all who directly or indirectly helped us in the Project work.

REFERENCES

[1] Kasara Sai Pratyush Reddy, Y Mohana Roopa, Narra Sai Nandan, Kovvada Rajeev L N," IoT based Smart Agriculture using Machine Learning", IEEE, CIRCA

[2] Shridhar M, Chinmay P, Piyush W, Aniket P, Vaishali D, (13)

[3] Kasara Sai Pratyush Reddy, Y Mohana Roopa, Narra Sai Nandan, Kovvada Rajeev L N," IoT based "A Survey on Predicting Suitable Crops for Cultivation Using IoT," International Journal of Innovative Research in Computer and (14)

[4] Kasara Sai Pratyush Reddy, Y Mohana Roopa, Narra Sai Nandan, Kovvada Rajeev L N," IoT based Communication Engineering, January 2017. (15)

[5] Kasara Sai Pratyush Reddy, Y Mohana Roopa, Narra Sai Nandan, Kovvada Rajeev L N," IoT based

[6] T Raghav, Bhagavatula A, Aashish S, Drishti J, Natesh B, Varshini S, "Smart Management of Crop Cultivation (15) using IoT and Machine Learning," International Research Journal of Engineering and Technology, November 2018.

[7] Pallavi R, B. Vinitha, K. Rishita, K. (17) using IoT and Machine Learning," International Research Journal of Engineering and Technology, November 2018.

[8] Pallavi R, B. Vinitha, Pranavi, "Crop Monitoring and Recommendation System using Machine Learning and IoT", International Journal of Innovative Technology and Exploring Engineering, July 2020 (1) using IoT and Machine Learning," International Research Journal of Engineering and Technology, November 2018.

[9] Pallavi R, B. Vinitha, Vol-9 Issue-9. (18) using IoT and Machine Learning," International Research Journal of Engineering and Technology, November 2018.

[10] Anuja Chandgude, Nikita Harpale, Diksha Jadhav, Punam Pawar and Suhas Patil, "A Review on Machine Learning (18) Algorithm Used for Crop Monitoring System in Agriculture", International Research Journal of Engineering and Technology (IRJET), vol. 05, Issue 4, Apr. 2018.

[11] Kiran Shinde, Jerrin Andrei, AmeyOke, "Web-Based Recommendation System for Farmers", International Journal on Recent and Innovation Trends in Computing and Communication (18) Issue 4, Apr. 2018. 2, Volume: 3, Mar. 2014 (18)