International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075, Volume-9 Issue-12, October 2020

Agrocompanion: A Smart Farming Approach Based on Iot and Machine Learning

Rashi Kaur, Kodali Havish, Thirumala Kaustub Dutt, Gangidi Manohar Reddy

Abstract - Agriculture is one of the cardinal sectors of the Indian Economy. The proposed system offers a methodology to efficiently monitor and control various attributes that affect crop growth and production. The system also uses machine learning along with the Internet of Things (IoT) to predict the crop yield. Various weather conditions such as temperature, humidity, and soil moisture are monitored in real-time using IoT sensors. IoT is also used to regulate the water level in the water tanks, which helps in reducing the wastage of water resources. A machine learning model is developed to predict the yield of the crop based on parameters taken from these sensors. The model uses Random Forest Regressor and gives an accuracy of 87.5%. Such a system provides a simple and efficient way to maintain and monitor the health of the crop.

Keywords: Agriculture, IoT, Sensors, Machine Learning, Random Forest Regressor

I. INTRODUCTION

Agriculture plays a key role in the growth of India's economic status. According to a study in 2020, agriculture is the chief source of income for about 58% of the Indian population. Improved land productivity is the need of the hour. Due to the inconsistent and unpredictable weather conditions, every year many farmers suffer heavy losses. Crop damage also leads to lesser food production in India, which is a major concern. With the increase in population, there is a high demand for the agriculture industry, and they need to meet it regardless of the inconsistent weather conditions. The proposed system aims to provide an efficient system for farming. The system aims to solve problems such as monitoring the crop's health, avoidance of wastage of electricity and water resources by automating and regulating them, and predicting the crop's yield at an early stage to take actions earlier to improve the yield. These issues are solved simply and efficiently, thereby saving time, resources, manpower, and reducing the cost. The system acts as an overall kit for people working in the field of Agriculture by aiding them with automated technology. The system can be used for large conventional farms as well as small ones.

II. LITERATURE SURVEY

Table- I: Literature Survey for the proposed system

S.No	Author and Year	Methodology
[1]	Abhishek L, 2019	Proposed a smart agriculture system that collects data from different sensors and provides the collected data to a machine learning algorithm which suggests the best crop suitable for cultivation in that selected area.
[2]	Agraj Aher,2018	Suggested a system that collects values from sensors that are placed in different locations. A mobile application with cloud services is provided for the farmers to monitor the values gathered from the sensors.
[3]	Nikesh Gondchawar, 2016	This paper aims at the automation of agriculture using IoT. It uses a sophisticated robot that is capable of performing basic agricultural tasks. It also proposed a smart irrigation system.
[4]	Muthunoori Naresh, 2019	Proposed a precision agriculture system which uses wireless IoT sensors. The system updates the farmers with the current conditions of the farm.
[5]	Yemeserach Mekonnen, 2020	Discussed the uniqueness of various machine learning algorithms used for predicting the crop yield. The data used is collected from various IoT sensors.
[6]	P. Kanaga Priya, 2019	Proposed a system that provides real-time crop data to farmers. The system suggests the farmers about the best crop which can be cultivated in a given area using deep neural networks.
[7]	Valavala Nalini Devi Prasanna, 2019	Proposed various machine learning techniques to select the best crop for cultivation during a particular season.
[8]	Manoj Athreya, 2019	Proposed an agriculture-based recommender system that predicts the yield and the price of crops by analyzing the past data.

III. METHODOLOGY

The primary goal of the proposed system is to aid the farmers and others working in the agriculture industry. The proposed system performs various tasks related to the agriculture industry. They are:

- Controlling the devices in hi-tech polyhouses.
- Regulation of water – The system monitors the percentage of water available in the water tank, the flow rate of water, and the amount of liquid transferred.
- Continuous and live monitoring of various weather parameters such as temperature, humidity, and soil moisture.
- Prediction of the crop yield during an early stage of crop growth using data such as longitude, latitude, temperature, dew point, and humidity.

Revised Manuscript Received on September30, 2020.
Ms. Rashi Kaur, Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad.
Mr. HavishKodali,a Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad.
Mr. ThirumalaKaustubDutt, Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad.
Mr. Gangidi Manohar Reddy, Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad.

All the Sensors are connected to a wifi-enabled microcontroller i.eNodeMCU. Firebase is used as the database to store and retrieve all the sensor values. The NodeMCU is connected to the Firebase via the internet. The connection between the Firebase and NodeMCU is established using the Firebase-Arduino module. Whenever the user requests to view any data, the system retrieves that particular data from Firebase and returns the values to the user. When the user changes the state of any device i.e. from ‘ON’ to ‘OFF’ or vice-versa, the change is first updated in the Firebase database. The NodeMCU continuously monitors the values in the database. Once the NodeMCU detects a change in the Firebase value, it updates its value and implements the change in the device’s state. Such a method is easy to implement and keeps the data secure. Since the system is wifi-enabled, the user can access the values of the sensors or control the devices from anywhere at any time. This reduces the overhead of having extra manpower and resources for agriculture.

IV. SYSTEM ARCHITECTURE

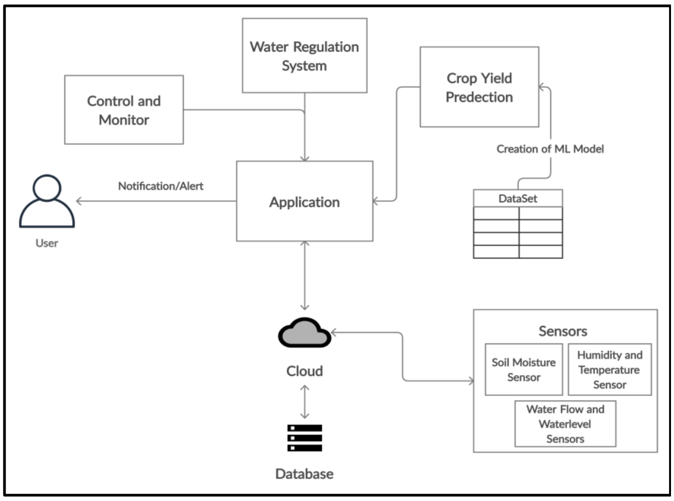

Fig. 1. The system architecture of the proposed system

The system architecture mainly contains four components. These are:

- Firebase - The cloud-based database
- Control and Monitor
- Water Regulation
- Crop Yield Prediction

The user communicates to the system using a web application. The system uses Firebase to store and retrieve the data obtained from the IoT sensors. These sensors include the temperature and humidity sensor, soil moisture sensor, water flow sensor, and ultrasonic sensor. This data is used by all the other components of the system. Based on the component the user is using, the sensor values are retrieved from the Firebase and transferred to the required parts or modules.

A. Control and Monitor Component

The Control and Monitor component has two functions:

a. To control the switches of the various devices in the polyhouse or farm.
b. To monitor the various weather parameters that affect the crop’s health.

This component focuses on controlling the states i.e. on and off of various devices that are being used on the farm. It also monitors the parameters that play a key role in affecting the crop’s health and yield.

This component can be divided into two sub-components i.e. ‘Control’ and ‘Monitor’.

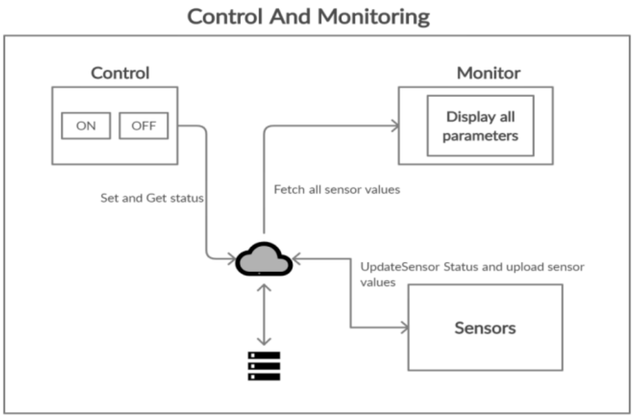

Fig. 2. The architecture of the Control and Monitor Component

Control Component

Under the ‘Control’ component, various devices are connected to the NodeMCU through relay modules. The NodeMCU is connected to the cloud database i.e. Firebase via the Internet. Initially, the user needs to register a set of devices that they want to control using the system. Then, the user selects a specific device from the registered devices. The control component displays the current status of the device chosen i.e. on or off. The user can now change the status of that device from ‘ON’ to ‘OFF’ or vice-versa. When the user changes the status of the device, the change is first updated in the Firebase. Then the NodeMCU detects the change in the database and changes the physical status of the device.

Monitor Component

Under the ‘Monitor’ Component, various parameters that are crucial for the crop’s health and better yield are displayed to the user. The parameters that are monitored are temperature, humidity, and soil moisture. These parameters are monitored in real-time using IoT sensors like DHT11(Temperature and Humidity sensor), HC-SR04 (Ultrasonic sensor), FC-28 (Soil Moisture Sensor), and YF-S201 (Hall-effect water flow sensor). These sensors are placed near the crop that needs to be monitored and are connected to the NodeMCU. The NodeMCU continuously takes the sensor values and updates the Firebase database. When a user wants to view the current value of the parameters, this component retrieves the current updated values from Firebase and displays it to the user. Also, the user can set a preferred range of values for the sensors. If any parameter crosses the set range, then a notification/alert is sent to the user’s email id.

B. Water Regulation Component

The water regulation system provides the following functionalities:

a. It monitors the current level of water available in the water tank.
b. It allows the user to view the data related to water regulation of previous days.

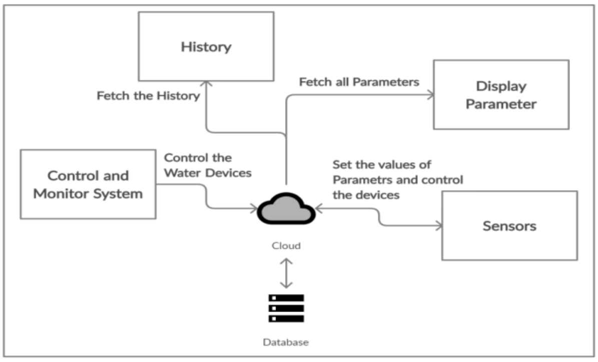

Fig. 3. The architecture of the Water Regulation component

The water regulation component uses the ultrasonic sensor (HC-SR04) and the flow rate sensor (YF-S201). These sensors are connected to the NodeMCU. The ultrasonic sensor is placed on the top inside edge of the water tank. It continuously measures the distance between the water present in the tank and the top of the tank. The current percentage of water available in the tank is monitored using this sensor. The flowrate sensor measures the flow rate of water from the water tank in litres/minute and also calculates the amount of water transferred in a day. NodeMCU stores and retrieves these values in the Firebase. These values are stored in the database along with the date and time. When the user wants to view the statistics of water flow for any given day, the values for that particular day are retrieved and displayed to the user. This lets the user calculate the resources spent on water and helps avoid water wastage, if any, in the future. The user can also view the live parameters of the water tank.

C. Crop Yield Prediction Component

This component is used to predict the yield of the crop using various parameters whose values are taken from the sensors. Machine learning is used to implement this functionality. When the user wants to view the predicted crop yield, this component retrieves the latest values of temperature, humidity, and soil moisture from the Firebase. It takes the values of the longitude and latitude from the user. Once all the values are available, it gives a predicted yield value in bushels/acre.

The architecture of the crop yield prediction component:

This component follows a three-tier architecture as shown in Fig. 4. Such architecture increases the scalability and flexibility of the machine learning model.

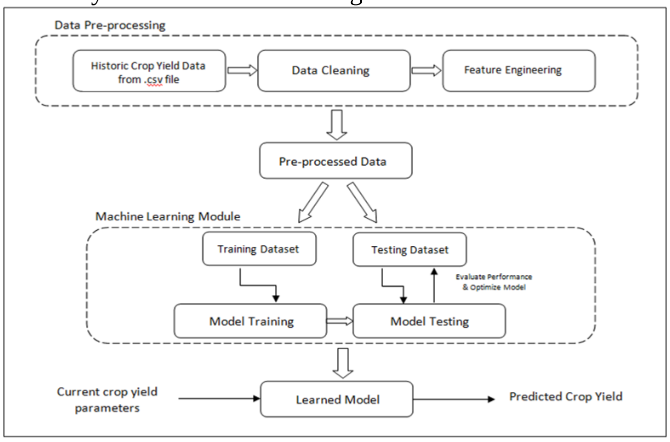

Fig. 4. The architecture of the Crop Yield Prediction component

Tier 1: This tier contains the datasets. The dataset consists of 24 attributes like latitude, longitude, dew point, maximum temperature, minimum temperature, pressure, etc. This dataset is cleaned by filling the missing values and removing a few attributes. Feature Engineering is performed on the cleaned data i.e. new features are created in the data set to make the machine learning model work better. Then, feature scaling is performed on the data to normalize the features of the data.

Tier 2: The middle tier consists of the machine learning modules, which provide the business processes logic and data access. The normalized data from tier 1 is then split into a training dataset and testing dataset. Various machine learning models are tested and evaluated.

Tier 3: In the last tier, the Learned Prediction Model i.e. Random Forest Regressor which is produced from tier-2, takes the current climatic conditions from the sensors like temperature, humidity, soil moisture, longitude and latitude as input and predicts the crop yield as output.

Data collection

The datasets are taken from a data science challenge of the Aerial Intel challenge. There are two datasets available. These datasets are of the year 2013 and 2014. The datasets contain 24 attributes. These are: CountyName, State, Latitude, Longitude, Date, apparentTemperatureMax, apparentTemperatureMin, cloudCover, dewPoint, humidity, precipIntensity, precipIntensityMax, precipProbability, precipAccumulation, precipTypeIsRain, precipTypeIsSnow, precipTypeIsOther, pressure, temperatureMax, temperatureMin, visibility, windBearing, windSpeed, NDVI, DayInSeason, and Yield. Fig. 5. depicts a snippet of the dataset used.

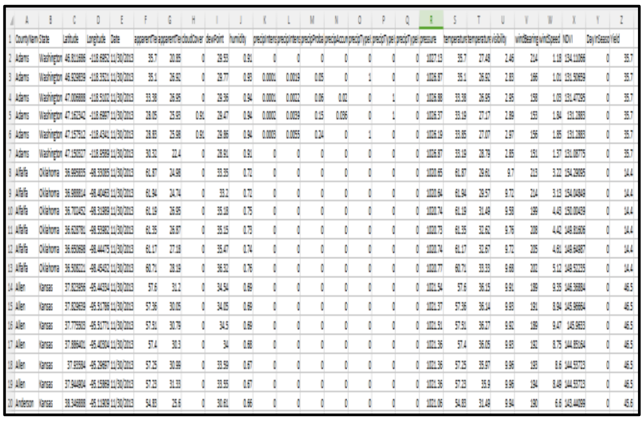

Fig. 5. Screenshot of the dataset used for machine learning

Data preprocessing

The datasets used contain many missing values and null values. To improve the accuracy and efficiency of the machine learning model, we need to clean the data.

a) Cleaning the dataset

Many attributes like cloudCover, precipIntensity, precipIntensityMax, precipProbability, precipAccumulation, precipTypeIsRain, precipTypeIsSnow, precipTypeIsOther and DayInSeason contain zeros for all rows. These attributes are removed from the dataset. The attributes pressure and visibility consist of many null values in both the datasets. The empty fields are filled using the mean of the values for that particular attribute.

b) Feature Selection

A high correlation between the features/attributes can lead to misleading machine learning results. Hence, a correlation matrix is obtained for both the datasets, and one of the highly correlated attributes are removed from the dataset. Attributes such as appararentTemperatureMax, appararentTemperatureMin are removed from the dataset since they are highly correlated to temperatureMax and temperatureMin respectively. The attributes ‘CountyName’ and ‘State’ are highly correlated to the attributes ‘Longitude’ and ‘Latitude’, and so they are removed. Fig. 6. depicts the correlation matrices for the dataset.

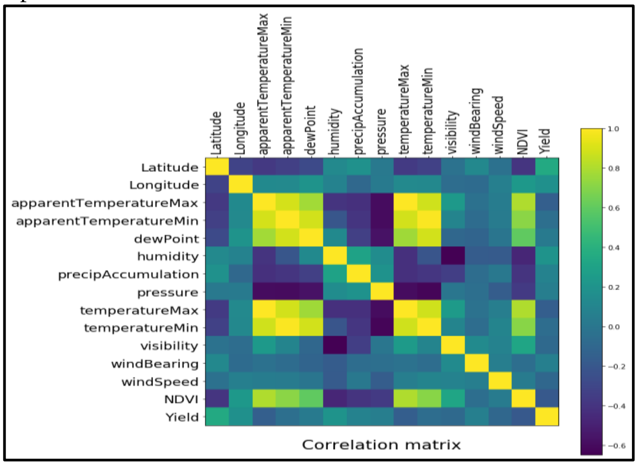

Fig. 6. The correlation matrix of the features in the dataset

c) Feature Engineering

The dataset contains multiple observations for the same area and same day. Hence, we perform feature engineering on the dataset. Here, the average of the observations recorded on the same date and the same area is taken and the multiple observations are converted into a single observation. The average of attributes such as ‘temperatureMax’ and ‘temperatureMin’ is taken and placed as ‘temp’ in the dataset. Since the value of ‘pressure’ and ‘NDVI’ can’t be calculated using the IoT sensors, these attributes are removed.

After performing feature engineering, the dataset consists of the following attributes:

- Longitude
- Latitude
- Temperature
- Dew Point
- Humidity
- Yield

d) Feature Scaling

The dataset contains numeric features which are based or measured on different scales. We perform feature scaling to normalize the values of the dataset to a particular scale. A scaler is set up to bring all the features to zero mean and unity standard deviation.

Developing the machine learning model

While developing the machine learning model, both the datasets i.e. 2013 and 2014 datasets are combined. Various machine learning models such as Random Forest, Gradient Boosting algorithm, Nearest neighbour algorithm, Support Vector Machine(RBF Kernel), Support Vector Machine(Linear Kernel), Support Vector Machine(Polynomial Kernel) are tested and evaluated. Since we have a limited data sample, we use a 5-fold cross-validation scoring to evaluate the models. Fig. 7. depicts the accuracies of the models.

```
In [35]: # Random forest regression
est = RandomForestRegressor(n_estimators=100, n_jobs=-1, max_features= 'sqrt')
scores = cross_val_score(est, X_train, y_train, cv=kf, scoring=cv_scoring)
print("Accuracy: %0.4f (+/- %0.4f)" % (scores.mean(), scores.std() * 2))
rf = scores.mean()*100

Accuracy: 0.8750 (+/- 0.0342)

In [36]: # Gradient boosted regression
est = GradientBoostingRegressor(n_estimators=100, learning_rate=0.1,
                                random_state=0,loss='ls')
scores = cross_val_score(est, X_train, y_train, cv=kf, scoring=cv_scoring)
print("Accuracy: %0.4f (+/- %0.4f)" % (scores.mean(), scores.std() * 2))
gb=scores.mean()*100

Accuracy: 0.8011 (+/- 0.0353)

In [37]: # Nearest neighbor regression
est = neighbors.KNeighborsRegressor(5, weights='uniform', n_jobs=-1)
scores = cross_val_score(est, X_train, y_train, cv=kf, scoring=cv_scoring)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std() * 2))
nn= scores.mean()*100

Accuracy: 0.78 (+/- 0.08)

In [38]: # Support vector regression - RBF kernel
est = SVR(kernel='rbf', C=1, gamma=0.1)
scores = cross_val_score(est, X_train, y_train, cv=kf, scoring=cv_scoring)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std() * 2))
svr1 = scores.mean()*100

Accuracy: 0.40 (+/- 0.13)

In [39]: # Support vector regression - linear kernel
est = SVR(kernel='linear', C=1)
scores = cross_val_score(est, X_train, y_train, cv=kf, scoring=cv_scoring)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std() * 2))
svr2 = scores.mean()*100

Accuracy: 0.26 (+/- 0.09)

In [40]: # Support vector regression - polynomial kernel
est = SVR(kernel='poly', C=1e3, degree=2)
scores = cross_val_score(est, X_train, y_train, cv=kf, scoring=cv_scoring)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std() * 2))
svr3 = scores.mean()*100

Accuracy: 0.34 (+/- 0.10)
```

Fig. 7. Accuracies of the machine learning models

Fig.8. consists of a graph depicting the accuracies of the various machine learning models evaluated.

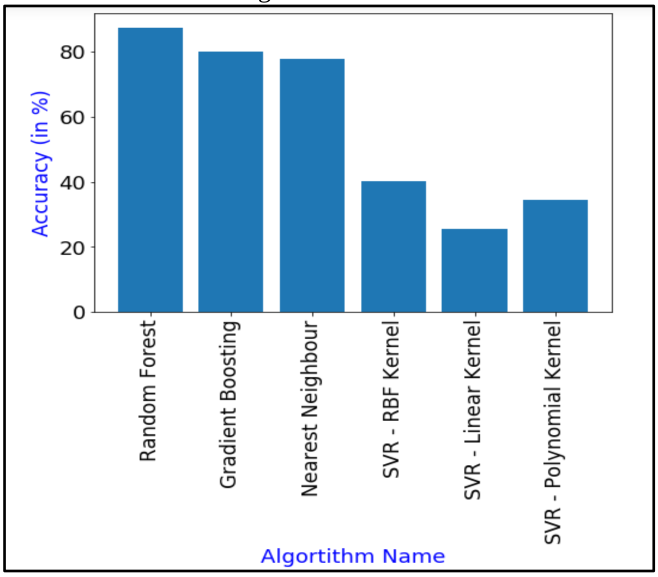

Fig.8. Graph depicting the accuracies of the machine learning models

The Random Forest model gives the best accuracy i.e. 87.5%. Hence it is selected and is deployed in the proposed system. Fig.9. depicts the model’s performance.

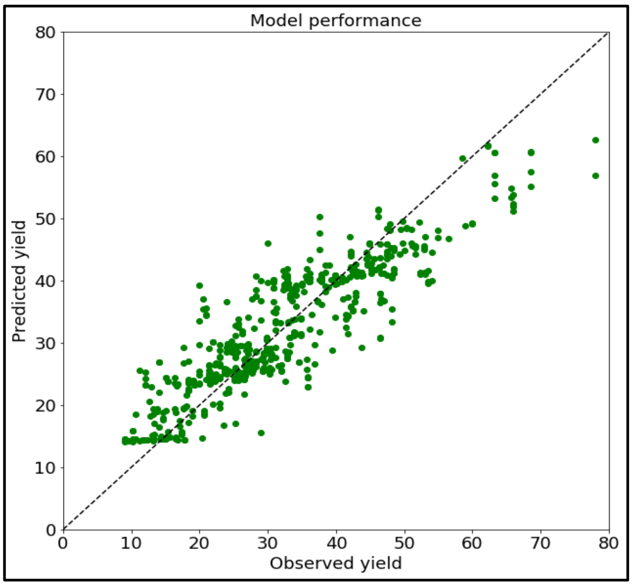

Fig.9. Model performance graph of the Random Forest Algorithm

V. PIN DIAGRAM

The proposed system uses NodeMCU, which is a WiFi built-in microcontroller. It consists of nine digital pins (D1-D9) and one analog pin (A0). The digital and analog pins are used when one needs to take input from a sensor or give output to a sensor or device. Each digital pin has two modes i.e. ‘0’ which means ‘no voltage’ and ‘1’ which denotes high voltage. The analog pins takes or gives a value between the range 0 to 1023. The proposed system uses a breadboard to connect more than one sensor to the NodeMCU. It aids in creating complex circuits having a single external power source.

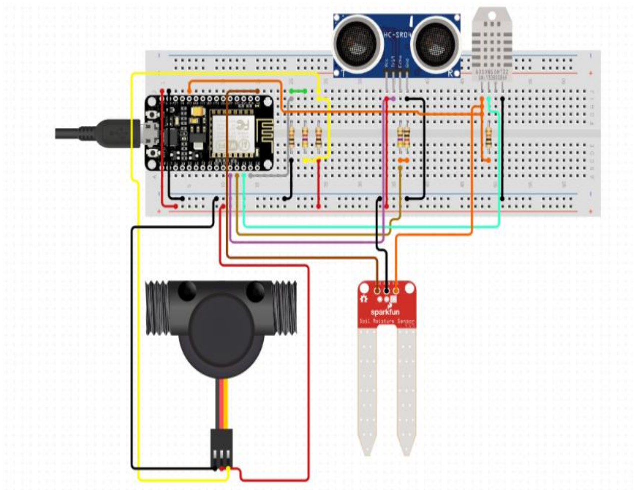

Fig. 10. Pin Diagram of the system

In fig. 10., the NodeMCU is connected to the hall-effect water flow sensor (YF-S201) at pin D1(GPIO 5) with a 4.7k Ω and a 10k Ω resistor connected in parallel. Another 10k Ω resistor is connected in series with the circuit, and this closes the circuit. An ultrasonic sensor (HC-SR04) is connected at pin D4 (GPIO 2) as a trigger and pin D3 (GPIO 0) as echo pins. Two resistors i.e. 470 k Ω and 1 k Ω are connected in series with the ultrasonic sensor. The digital humidity and temperature sensor (DHT11) is connected to the NodeMCU at pin D2 (GPIO4) with 10 k Ω resistor in parallel. The soil moisture sensor (FC-28) is connected at one analog pin (A0). All these sensors are powered through an external source of power generated by NodeMCU into the breadboard.

VI. TECHNOLOGY REQUIREMENTS

A. Hardware Requirements

ESP 8266 NodeMCU

The ESP 8266 NodeMCU is a wifi-enabled microcontroller that fetches values from sensors and sends them to the firebase cloud database. It supports both analog and digital signals. It is powered by a USB 2.0 cable with an input voltage of 7V -12V. It has 30 pins.

Temperature and humidity sensor (DHT11)

The DHT11 sensor is used to measure temperature and humidity. It has one digital signal. It has 4 pins i.e. VCC (3.3V), Data, NC, and GND. It can measure temperatures between 0-50 ℃ and Humidity between 20-90%. It gives a temperature accuracy of ± 2 ℃ and a Humidity accuracy of ± 5 % RH. It has a resolution of 1.

6.1.3. Ultrasonic sensor (HC-SR04)

The HC-SR04 sensor is used to measure the distance between two objects. This can be used to measure the percentage of water. It has two digital signals. It uses 4 pins i.e. VCC (+5V), Trigger (an input pin that triggers the ultrasonic wave for initial measurement), Echo (receives the ultrasonic wave sent by the trigger and gives the time taken which is converted into distance), GND (ground). The proposed system uses the distance given by echo and converts it into a percentage.

Soil moisture sensor (FC-28)

The FC-28 sensor is used to measure soil moisture. Its output signal is in analog as well as digital. It has 4 pins i.e VCC (3.3V - 5V), A0 (Analog output), D0 (Digital output), and GND (ground). We use the analog output(A0) of the sensor. Analog pin (A0) has a range of values between 0-1023 which is mapped to values between 0-100 % of soil moisture.

Hall-effect water flow sensor (YF-S201)

The YF-S201 sensor is used to measure the flow rate of water in L/min. It has one digital pin. It has 3 pins i.e. VCC (+5V), PWMoutput (gives rpm which is converted into water flow rate), and GND (ground).

B. Software Requirements

Firebase

Firebase is a mobile platform developed by Google for the development of web applications. This platform consists of a great set of development tools. The proposed system uses the real-time database development tool of Firebase. It acts as a cloud database that is used by the NodeMCU to store and retrieve the sensor readings. Firebase is selected as the database since it makes it easy to maintain the data in the cloud, without having to run it on the local server.

Flask

Flask is a web framework that provides tools, support for libraries, and technologies for developing a web application.

It is categorized to be a micro-framework with a few or no dependencies on external libraries and is light-weight in nature. The proposed system uses Flask to manage the requests and responses from the user via the web-application. It accomplishes this by routing the web pages.

VII. RESULTS

The above mentioned functionalities are deployed into a web-app using Flask. This makes the system user-friendly and easy-to-use.

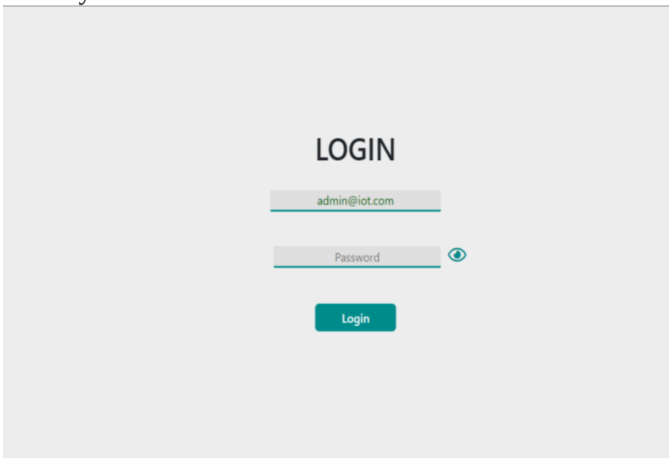

Fig. 11. Login Page – Screen 1

Fig.11. displays the first screen of the web-app. The user needs to enter their email id and password to log in.

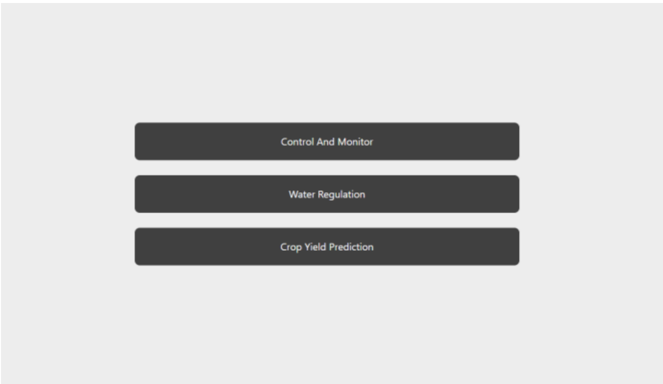

Fig. 12. Screen 2 – First menu displayed

Once the user logs into the web-app, the page in Fig. 12. appears. It displays three options to the user i.e. 'Control and Monitor', 'Water Regulation', and 'Crop Yield Prediction'. The user can click and select any one of the options as required.

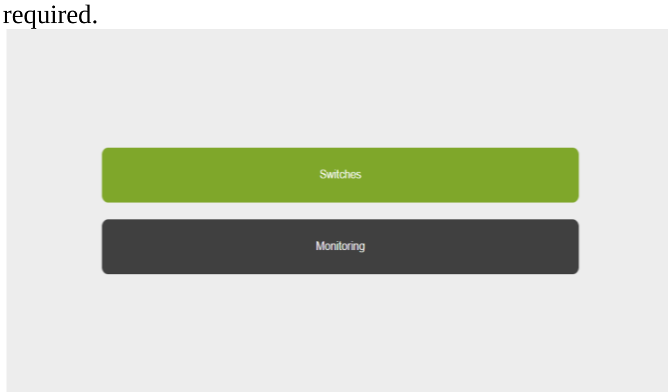

Fig. 13.The menu that appears when the user selects 'Control and Monitor'

If the user selects the 'Control and Monitor' option, the page in fig. 13. appears. Now the user can select between 'Switches' and 'Monitoring'. If the user wants to change the state of the switches, he/she must select 'Switches'. If the user wants to see the real-time values of weather parameters, he/she must select 'Monitoring'.

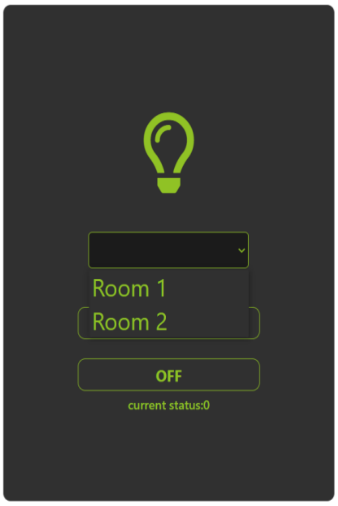

Fig. 14. The page that appears when the user selects 'Control'

If the user selects the 'Switches' option, the page displayed in fig.14. appears. The drop-down list contains the various registered devices which the user can control.

Fig. 15. Control options that are displayed to the user

Once the user selects a room/device from the drop-down list, the left screen in fig. 15. appears. Here, the current status of the switch is shown. Eg: In fig. 15. the left screen shows the current status of the switch in room1 as 'off' or 0. The user can change the state of the switch by clicking on either 'ON' or 'OFF'. The right screen in fig. 15. appears when the user changes the state of the device. Eg: If the user clicks on 'ON' for the devices in 'Room 1', the change is made in the Firebase database, and then it is reflected on the screen. So, the new state of the device is 'ON' or 1.

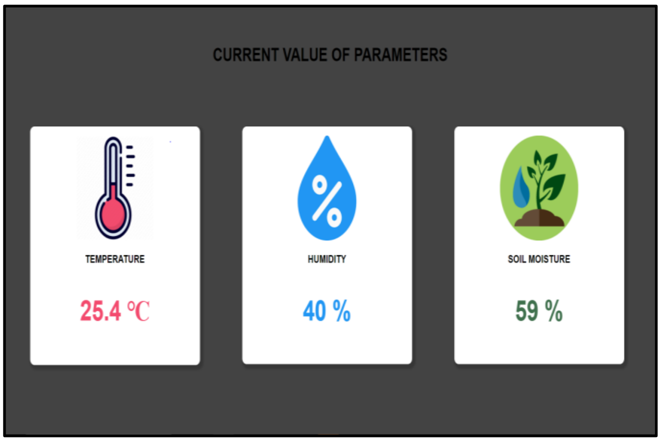

Fig. 16.The screen that appears when the user selects the 'Monitoring' tab.

In fig.13. , if the user selects 'Monitoring', the screen in fig.16. appears. The user can view the real-time values of parameters such as temperature, humidity, and soil moisture. These values are taken from the sensors placed near the crops.

Fig. 17. Options that are displayed to the user when the user selects the 'Water Regulation' Tab

In fig. 12. , if the user selects the 'Water Regulation' Tab, the screen in fig. 17. is displayed. Here, the user can select 'History' to see the previously stored parameters regarding water regulations. To view the current values of parameters of the water tank, the user can select the 'Water Tank' tab.

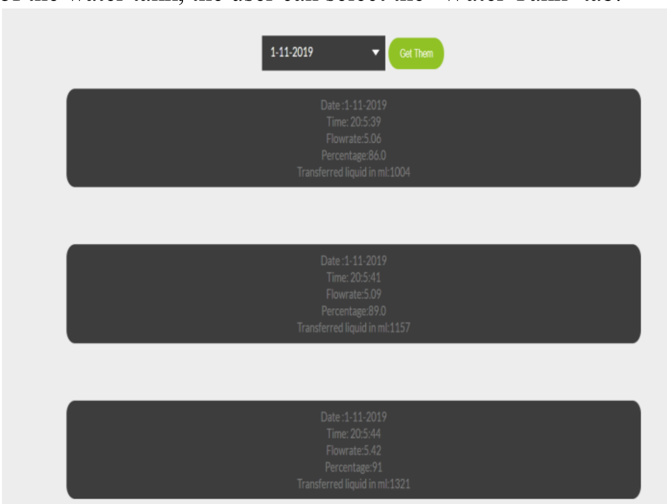

Fig. 18. The 'History' page under the 'Water Regulation' Tab

If the user selects the 'History' Tab, the user needs to select a date from the given drop-down menu. Once the user selects the date and clicks on 'Get Them', the screen in fig. 18. is displayed. Here the user can view the parameters such as 'Flowrate', 'Percentage', and 'Transferred liquid in ml' for the selected date and time. This is useful for the user to know the amount of water resources used for the farm in a specific period.

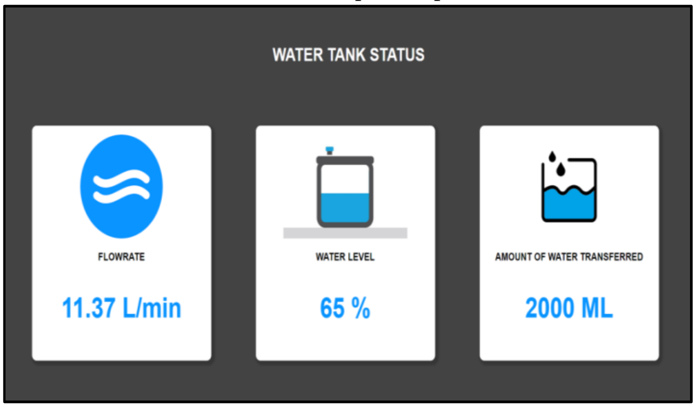

Fig. 19. Control options that are displayed to the user

If the user selects the 'Water Tank' tab in fig. 17., the real-time values of parameters such as the 'Flowrate', 'Water Level' and 'Amount of water transferred' are displayed as shown in fig. 19.

Fig. 20. The menu that appears when the user selects the 'Crop Yield Prediction' Tab

If the user selects the 'Crop Yield Prediction' Tab in fig. 12., a menu appears as shown in fig. 20. If the user wants to predict the yield of the crop based on the real-time values from the sensor, the user must select the 'Sensor Prediction' tab. The user should select the 'User Prediction' tab if he/she wants to predict the crop yield for some specific user-entered values.

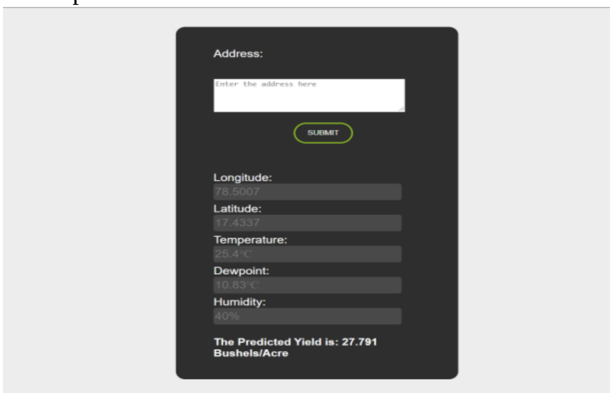

Fig. 21.The page that appears once the user has entered and submitted the values required to predict the crop yield.

When the user selects the ‘Sensor Prediction’ Tab, the user is prompted to enter the address of the farm. This is done to obtain the longitude and latitude values of the area. When the user clicks on ‘Submit’, a screen is displayed as shown in fig. 21. The values of ‘Temperature’ and ‘Humidity’ are obtained from the sensors. The value of ‘Dewpoint’ is calculated using the temperature and humidity values. The predicted yield is displayed in Bushels/Acre. Eg: In the above fig.21., the value entered for address is ‘Secunderabad Women’s Hospital, Hyderabad’.

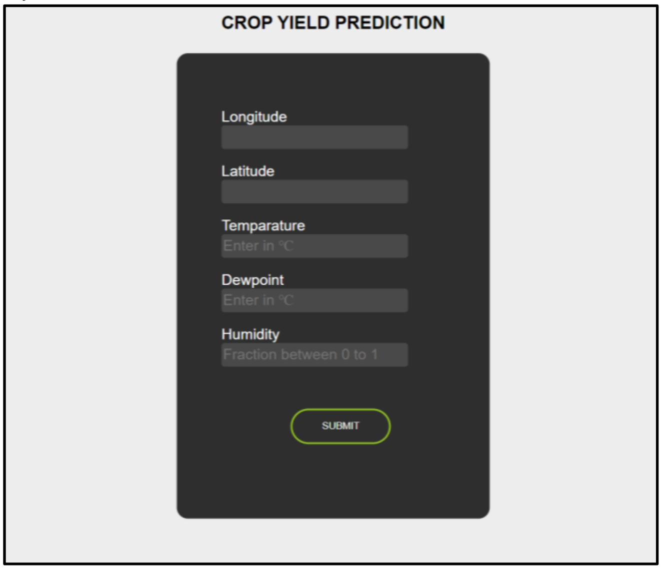

Fig. 22. Control options that are displayed to the user

If the user selects the ‘User Prediction’ Tab in fig.20, the screen displayed in fig. 22 appears. Here the user needs to manually enter the values for ‘Longitude’, ‘Latitude’, ‘Temperature’, ‘Dewpoint’, and ‘Humidity’.

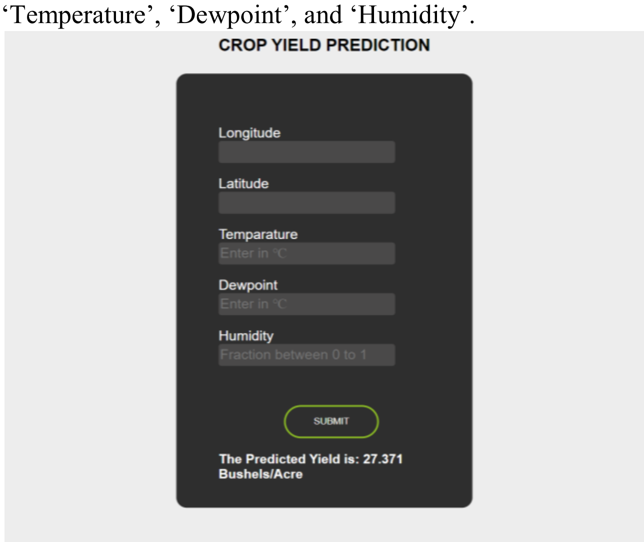

Fig. 23. Control options that are displayed to the user

Once the user enters the values of the parameters and clicks on ‘Submit’, the predicted crop yield is displayed in bushels/acre as shown in fig. 23. Eg: In fig. 23., the values entered are longitude= 35.4, latitude = 40.8, temperature = 26, dewpoint = 20.7 and humidity = 0.5. The machine learning algorithm used to predict the crop yield gives an accuracy of 87.5%. So, the result of the predicted yield might differ slightly for the same values of the parameters.

VIII. CONCLUSION

The proposed system aims to provide the farmers and agriculturalists with an efficient and easy to use mechanism to monitor and control their crop’s health, which in turn helps in producing a better yield. It uses IoT sensors and machine learning to perform these tasks. The system also saves the water resources in the agriculture sector by regulating the water system.

The water regulation component of the system always keeps a check on the water levels in the water tank. In addition, the control and monitoring system alerts the users about the environmental conditions if necessary. The Yield prediction component of the system predicts the crop yield using the parameters from the sensors. It is developed using the Random Forest regressor which gives an accuracy of 87.5%. This helps the user estimate the yield of their crop at a very early stage of crop growth. Hence, the system provides a holistic approach to improve the crop’s health for small-scale as well as large-scale farmers. It also helps in reducing the wastage of crops due to uncertain and ever-changing weather conditions. Moreover, it is user-friendly and affordable. Since agriculture is one of the major sources of India’s national income, such a system will help in increasing the country’s economy.

REFERENCES

1. Abhishek, L.; Rishi Barath ,B. (2019): Automation in Agriculture Using IOT and Machine Learning : (IJITEE) ISSN: 2278-3075, Volume-8 Issue-8 June, 2019
2. Agraj, Aher.; et al(2018): Smart Agriculture using Clustering and IOT :(IRJET) e-ISSN: 2395-0056 p-ISSN: 2395-0072,Volume: 05 Issue: 03 | Mar-2018
3. KanagaPriya,P.; Dr N. Yuvaraj. (2019): An IoT Based Gradient Descent Approach for Precision Crop Suggestion using MLP : 2019 J. Phys.: Conf. Ser. 1362 012038
4. Manoj Athreya,A.; Hrithik Gowda, S.; Madhu, S.; Ravikumar, V. (2019): Agriculture Based Recommender System using IoT (IJRTE) ISSN: 2277-3878, Volume-8 Issue-2S8, August 2019
5. Yemeserach, Mekonnen. (2019): Machine Learning Techniques in Wireless Sensor Network Based Precision Agriculture : 2019 J. Phys.: Conf. Ser. 1362 012038
6. Naresh, Muthunoori.; Munaswamy, P. (2019): Smart Agriculture System using IoT Technology:(IJRTE) ISSN: 2277-3878, Volume-7 Issue-5, January 2019
7. Nikesh, Gondchawar.; Prof. Dr. R. S. Kawitkar. (2016): IoT based Smart Agriculture :ISSN (Online) 2278-1021 ISSN (Print) 2319 5940 Vol. 5, Issue 6, June 2016
8. Prasanna, Valavala Nalini Devi.; Dr.B.Kezia Rani. (2019): A Novel IOT Based Solution for Agriculture Field Monitoring and Crop Prediction Using Machine Learning (IJIRSET) ISSN(Online): 2319-8753 ISSN (Print): 2347-6710

AUTHORS PROFILE

Ms. Rashi Kaur, a student of Bachelor of Technology in the field of Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad. She has worked as an intern at Quikr India and ParallelDots Pvt. Ltd.The research areas she is interested in include Data Mining, Data Analytics, Machine Learning and IOT.

Mr. HavishKodali,a student of Bachelor of Technology in the field of Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad. He has worked as an intern at Bharat Electronics Ltd, Hyderabad.

Mr. ThirumalaKaustubDutt,a student of Bachelor of Technology in the field of Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad. He has worked as an intern at Cognizant and Bharat Electronics Ltd, Hyderabad.

Mr. Gangidi Manohar Reddy,a student of Bachelor of Technology in the field of Computer Science & Engineering at Mahatma Gandhi Institute of Technology, Hyderabad. He has worked as an intern at Zemoso, Hyderabad and Bharat Electronics Ltd, Hyderabad.