International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

Remote Monitoring, Managing & Disease Detection of Plantation using IoT framework

Dr. Aruna Gawade, Dr Kranti Ghag, Rashmil Panchani, Miloni Sangani, Priyal Patel
Department of Computer Engineering
Dwarkadas J. Sanghvi College of Engineering Mumbai
Mumbai, India
aruna.gawade@djsce.ac.in kranti.ghag@djsce.ac.in

Abstract— Living in a country having an Agrarian economy, one of the fundamental objectives of this paper is to cater to the urban class, who because of their busy schedules don't indulge in gardening activities. This IoT Application enables the user to remotely access and irrigate the garden, become cognizant of its current state, monitor the surrounding environment of the garden to steer clear of potential danger and obtain graphical representations of the data collected from various sensors. Further, this system allows the user to obtain a live video feed of the plants, obtain information on the ideal requirements of the plant and detect the possibility of any diseases through leaves. At the application level, a web-app is designed and implemented that can be installed on users phone, thereby bestowing the users with the power to access the system at their fingertips.

Keywords- Internet of Things; IoT; Smart garden; Capacitive Soil Moisture Sensor; Temperature Sensor; Relay module

I. INTRODUCTION

Gardens, as per our understanding of them today, are either food sources or ways to beautify and protect our property. The concept of gardens dates back to the times when rulers engaged in home gardening practices and the gardens in the courtyards of people were popularly known as vatikas. Eventually with the rapid industrial development and commercialization of various food products, the concept of gardening failed to remain a norm and was reduced to a mere pastime. During World War II, umpteen Americans planted victory gardens in their backyards, eventually catering to the nation's hunger with 40 percent of its homegrown fruits and vegetables. Once the war was over, these urban farms withered away, and were supplanted by increasingly efficient and large-scale rural agriculture practices. Similarly, even in India, the government has been making numerous attempts over the years to encourage citizens to indulge in urban farming as it is rightly observed that a family that practices gardening tends to be slightly more inclined towards adopting a healthy lifestyle and have more vegetables and fruits. It further promotes values of science, environmental stewardship and healthy eating among children. This practice teaches us to appreciate the value of food better and thereby provide an insight into the pain that farmers take for producing food. Further, the materialistic mindset of the urban class will take a liking to the natural environment.

The eternal need of every human being in this world is oxygen. Plants play a vital role in maintaining the carbon dioxide and oxygen content in the air. Number of plants are being destroyed each and every day for the urbanization process. The number of plantings made is also reduced. Apart from these things more plants die due to lack of maintenance. The main aim of this paper is to maintain the nature of the plants by continuously monitoring the parameters leading to the increased life of both plants and human beings. At this epoch of time, people are covetous of automating various mundane tasks in order to enable them to multi-task. Thus, by making a smart gardening system named 'Garduino', we plan to do our bit towards contributing to a greener Earth.

II. RELATED WORK

One of the cardinal issues that need to be tackled while building this smart system includes building a robust system that can be remotely accessed and further handle network failures or other impediments. By using the principles of Internet of Things and the multifarious opportunities provided by the usage of Cloud Technologies, many people have worked on similar projects in the past decade. To gain ample knowledge of this ideology, we studied various research papers [1], [2], [3], [4], [5] that throws light on the methods implemented in the past.

On scrutinizing these papers, we were furnished with the different ways in which similar projects were implemented by individuals around the world. Most of the projects employed the use of the following sensors and devices:

A. Raspberry Pi

Raspberry Pi was the name given to a series of single-board computers developed by the Raspberry Pi foundation and it aimed to provide cheap computing facilities to the users. Its enticing price, operation on Linux and the provision of various GPIO pins have appealed to people all over the world to use it for learning programming skills, build hardware projects involving automation and solving many industrial problems. To

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

remotely access the Raspberry Pi over the internet methods such as Port forwarding and remote SSH are widely used [6], [7]. The system will be connected to the internet through WiFi and a cloud interface will be created to control the appliances from a remote location. One disadvantage subjective to different applications is its incapability to read analog signals and its exorbitant price as compared to Arduino.

Figure 1. Raspberry Pi and Arduino to show the sensor values on an LCD Display monitor [14].

B. Arduino

Arduino, an open-source system, which is widely used in many projects involving the Internet of Things (IoT) consists of a physically programmable circuit board (which is often referred to as a microcontroller) and the software part (Integrated Development Environment) which can be conveniently used to write and upload the code onto the physical board. With the help of some code, arduino can effectively interact with light, temperature, pressure, proximity and moisture sensors which can be used in projects similar to our proposed smart gardening system. While an advantage of Arduino is it's easy to understand structure and libraries, a major disadvantage in such projects is its lack of internet connectivity, small RAM capacity and its design of not being able to multitask [8], [9]. Thus, it can be useful as a dedicated device but not as an integrated one.

C. Relay Module

A key component of any smart gardening system is its irrigation module since giving an opposite amount of water to plants is an imperative aspect of gardening. Relays basically serve as switches that are used for disconnecting and connecting the circuits electronically as well as electromagnetically. The advantage of relays is that it takes a relatively small amount of power to operate the relay coil, but the relay itself can be used to control motors, heaters, lamps or AC circuits which themselves can draw a lot more electrical power [10]. Thus, this can be effectively used to manage the water supply to the plants with the help of solenoid valves when low moisture content is detected by sensors.

D. Temperature, Humidity and Moisture Sensors

For obtaining the requisite values pertaining to the environmental conditions around the plants, various sensors are used. The temperature and humidity sensors apprise the user of the prevailing weather conditions around the plants which play a crucial role in determining the amount of water that the plants may need [11]. Further, the moisture sensors available are of capacitive or of resistive types [12]. Across different papers, the advantages of capacitive moisture sensors have been delineated as it is made of a corrosion-resistant material, thereby giving it a long service life.

III. PROPOSED METHODOLOGY

The various approaches enunciated in the numerous research papers aid people to locally monitor the plants and automate the tasks by giving instructions over their local network. Our proposed methodology involves capacitating the users with the ability to give instructions and monitor their plants over a remote network via a web application to provide a lucid interface to users who are not acquainted with the intricacies of programming.

The working architecture diagram of our system as per our final methodology has been shown in the figure 3 below.

A. Explanation of The Proposed Architecture

The sensors along with the relay modules are connected to the Raspberry Pi.

The Relay switch will be further connected to the solenoid valves that will be responsible for controlling the water irrigation in our system.

The Raspberry Pi will act as the central station where the data will come from all the sensors, then they will be processed and then they will be pushed to the Hosted Web Service.

Upon receiving the signal to water the plant, the raspberry pi will instruct the relay switches to activate and water the particular plant.

Thus, upon receiving the command from the user regarding irrigation or the live video feed of the plant, the raspberry pi activates the appropriate modules and helps the user perform the desired activity.

Thus, the sum and substance of the connections between the Raspberry Pi and the web-server are provided in the figure 2 shown below:

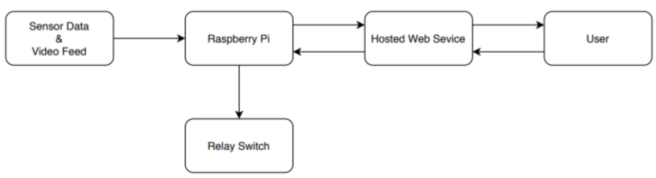

Figure 2. Diagram showing connections between Raspberry Pi and web server.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

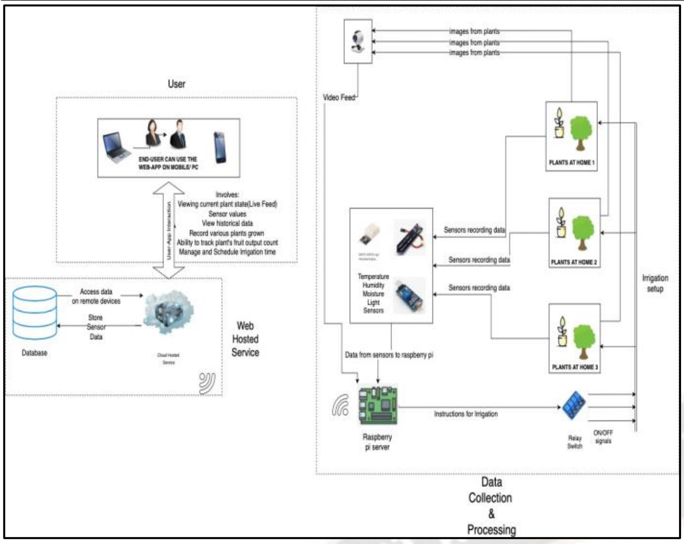

Figure 3. Architecture

B. Proposed Modules

1) Data Collection and Processing:

The Data Collection module forms the basis of the entire paper as the various decisions are taken by a relative comparison of the data obtained by the various sensors with the ideal threshold values mentioned in the information system.

Further, the data collected over a period of time can help provide valuable insights to the user regarding which approach of irrigation was a success and which was a failure. The statistical visualizations can help the user to analyze the data over a period of time and get a summary of the same. It can advise the users who are amateurs at gardening regarding the ideal setup and conditions for different varieties of household plants.

2) Irrigation Module:

The irrigation module of the system is implemented with the help of moisture sensors, relay module, the information system, raspberry pi and the solenoid valves connected to the pipes for regulated water supply.

A Flowchart depicting the working of the irrigation module has been shown in figure 4.

The air temperature, air humidity and soil moisture values are taken from the sensors and are compared with the threshold values in the data management system to decide on the appropriate quantity of water that the plants may require. Further, there are two provisions provided in the web application: Automatic irrigation (where the system takes timely readings and decides on the quantity of water) and Manual irrigation (where the user controls the system and instructs the system to water the plants). A toggle button is provided to switch between the two features.

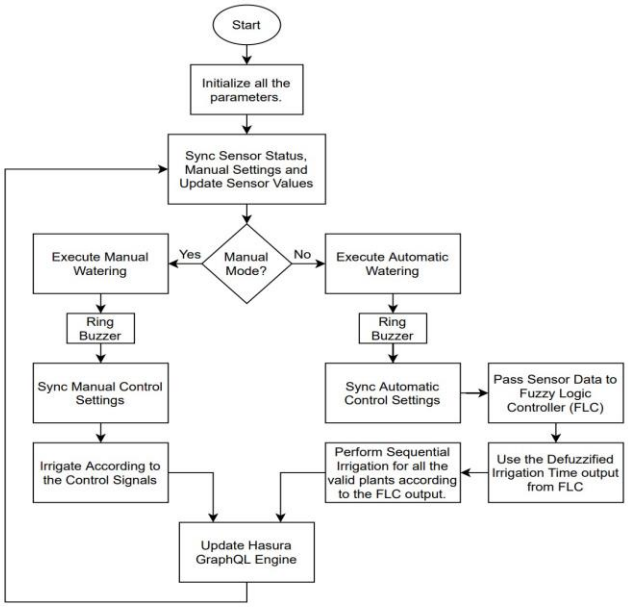

Figure 4. Flowchart depicting working of irrigation module

3) User Module and web hosted service:

The other part of the entire project consists of the Software part where the user gains the ability to tap into the garden system and perform various functions that the person wishes to perform.

The user will use the web app to control his/her garden. The person can perform the various activities such as:

- Set watering schedule.
- Visualize Temperature, Humidity, Soil Temperature and Moisture and light data.
- Search information of the various plants.
- See requirements by the plants in the various stages of growth
- To maintain soil temperature water can be regulated automatically.
- Automatic and Manual Irrigation.
- Keep track of the age of the plant.
- Ability to view and store the information of each plant grown.

To visualize how the user will get to communicate to the Hosted Service refer figure 5.

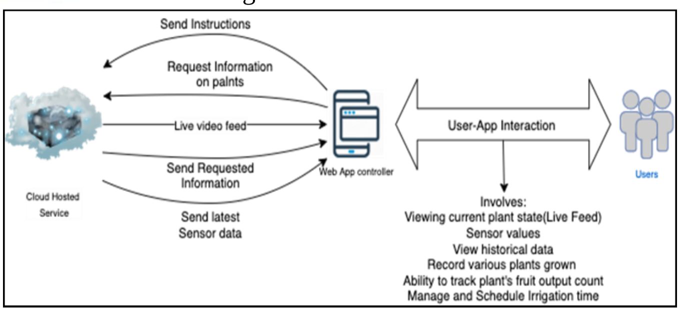

Figure 5. Web Hosted Service and User Implementation

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

As clearly seen from the diagram the Web app will be communicating with the Hosted Web Service which will be our center point. It will be the place where all the data will be stored, and one can access the information stored there anytime and request the latest data also.

This will trigger a request to the Raspberry Pi where the data will be collected from the sensors and will be posted to our hosted runtime.

With the other features, our main goal is to provide the user with a database of various plants with their climate, water and soil requirements that are essential for the growth of the plant. We will try to provide a step-by-step guide on how to take care of different plants and help them detect any deficiencies that the plant might be facing.

4) Fuzzy Logic Controller for smart irrigation:

A fuzzy logic controller is designed to determine the time required to water each plant. A rule base is designed for 2 different sets of input variables; for air temperature and air moisture and for soil temperature and soil moisture. The intermediate variables (s1, s2, s3, a1, a2 and a3) from the air and soil components are further used to compute the final time in seconds according to the output rule base (the output variables being w1, w2, w3, w4 and w5). The different parameters are divided into 5 distinct categories: very low, low, medium, high and very high respectively. Further defuzzification is performed by using Mean of max method.

The mean of the recommended range is taken into account to give a crisp output value for the irrigation time.

The basic rule base of the methodology has been shown in the in the figures 6, 7 and 8.

STemp\SMois	V Low	Low	Med	High	V High
V Low	x	x	x	s1	x
Low	x	x	s1	s1	s1
Med	s2	s2	x	s1	s1
High	s3	s3	s1	x	x
V High	s3	s3	s2	s1	x

Figure 6. Rule base for Soil Parameters

ATemp\AMois	V Low	Low	Med	High	V High
V Low	x	a1	a1	x	x
Low	a1	x	a2	a1	a1
Med	a2	a2	x	a1	a1
High	a3	a3	a2	x	a2
V High	a3	a3	a3	a2	x

Figure 7. Rule base for Air Parameters

A\S	s1	s2	s3
a1	w1	w2	w3
a2	w2	w3	w4
a3	w3	w4	w5

Figure 8. Output Rule Base

5) Leaf Disease Detection:

The economy is highly dependent on agricultural productivity during a year. The farmers have to be vigilant about potential diseases that the plants may be exposed to are already suffering from. Thus, the existing method of naked-eye observation of the plants becomes very inefficient. An automatic detection technique would take less time, effort and would be more accurate. It has been found that the first signs of diseases are conspicuous on the leaves of the plants and thus various researchers have used image processing techniques to detect potential diseases.

The classification of common diseases is done by training the model with a dataset consisting of 14 different plants and testing on the picture of the leaf captured by the User's phone which is to be uploaded on our web app. The basic workflow of the methodology has been shown in the flowchart in the figure 9.

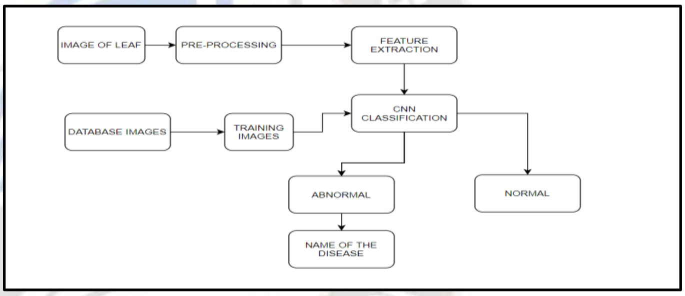

Figure 9. Flowchart for Leaf Disease Detection

6) Live Image stills of the Garden:

This module is implemented to help the user in monitoring the plants by obtaining the live feed of the plant and protecting it from possible intrusions.

C. Workflow of The System

The flow of data in the system has been explained in figure 9 with the help of a data flow diagram depicting how a user will be able to interact with the system and what data flows from where to where.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

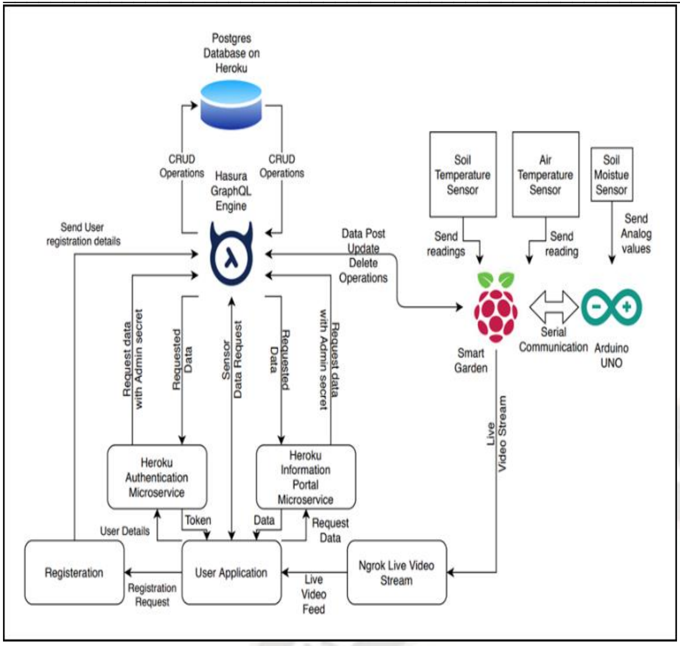

Figure 10. Data Flow

The various features which involve the movement of data as depicted in the diagram, are as follows:

1) Ability to Add a plant:

The user can add a plant to the garden and attach several sensors to it such as solenoid valve, soil moisture/temperature sensor, etc. The information of this plant is added to the Hasura GraphQL engine and the UI of the user is updated.

2) Read Sensor Values:

The user is able to track the sensor values such as the temperature and humidity of the environment around the plants, the soil moisture and the amount of light falling on the plants. The Raspberry Pi collects this data from the sensors and forwards this information to the cloud platform after processing it. Finally, the App can access these readings from the cloud.

3) Irrigation System:

The user sets the irrigation schedule for the plant or uses the recommended settings. The system automatically updates the Raspberry Pi to irrigate the plant. The Irrigation System uses the Information System to decide whether the plant requires water or not. Then the user is notified about the same.

4) Information System:

The amount of water that needs to be supplied to the plants relies on its current conditions. The information system consisting of the ideal growing conditions required for these plants will facilitate the system to determine if the soil needs more water supply based on the current temperature and humidity of the atmosphere around.

5) Fuzzy Logic Controller for Smart Irrigation:

The concept of a fuzzy logic controller is used to decide the opposite time required for irrigating different plants. The atmospheric temperature and moisture values, along with the soil temperature and moisture values are taken into account to decide on the irrigation time for each plant. A rule base is constructed for the input and output variable which aids in this process.

6) Tracking Plant Growth:

The system facilitates the user to update the status of the plants by entering various stats like the height, fruit output by the plant, the number of harvests done etc.

7) Live Image Stills:

If the user wants to have a visual overview of the garden, then the user can request live image stills of the garden via the webcam installed in the garden. This feature then sends an auto-generated email with 10 images from the garden to the users registered email-id.

IV. IMPLEMENTATION

A. Irrigation Module

The irrigation module is implemented using raspberry pi, Arduino UNO, solenoid valves, PVC pipes (lead-free), relay switches and Capacitive Soil Moisture sensor.

TABLE I. IRRIGATION MODULE COMPONENTS

Name	**Quantity**	**Specifications**
Raspberry pi model 4	1	RAM of a minimum of 2 GB is required.
Arduino UNO	1	A standard board for recording analog data from moisture sensors is required.
8 Channel Relay switches	1	A relay switch with 8 channels working on 5V input from raspberry pi is required to at least operate 24V appliances.
Solenoid valves	8	8 x 24V DC solenoid valves are required to operate on 8 different plants.
PVC (lead-free) pipes and connectors	7ft.	length of pipe varies on a use case basis. Required 7ft. long pipes along with connectors to direct the flow of water.
The capacitive soil moisture sensor	8	The version used is V2.0 which is an analog sensor. Operating voltage range of 3.3 5.5V. Output Voltage DC 0-3.0V.

B. Data Collection and Processing

The data collected is used for further processing and for enabling the system to make smart decisions on its own. Various sensors are used to implement this module, the details of which are specified below:

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

TABLE II. DATA COLLECTION AND PROCESSING COMPONENTS

Name	Quantity	Specifications
DHT22 air temperature and humidity	1	Operate Voltage: DC 3 5.5V, TemperatureRange:-40°C 80°C Humidity Range: 0 100% RH, Temperature Measurement Accuracy: ± 0.5 °C HumidityMeasurement Accuracy: ± 2% RH.
DS18B20 soil temperature sensor	1	Temperature Range: -55 to 125°C (-67°F to +257°F), ±0.5°C Accuracy from -10°C to +85°C, Usable with 3.0V to 5.5V power/data.
LDR lightsensor	1	Operating voltage: 3.3V – 5V Output: Analog & Digital
Raspberry pi	1	Covered in TABLE I.
Arduino UNO	1	Covered in TABLE I.

C. *User Module and Web Hosted Service*

The remote access to the smart garden is a crucial aspect of this paper. The details of the technologies used for implementing the user module are specified below:

1) Hasura GraphQL Engine:

This provides us with a hosted- cloud platform through which we can query and mutate the database using the GraphQL language. It also pro- vides us with the functionality of subscriptions through which we can keep track of any updates in the database.

2) Heroku Microservices:

Various NodeJS Servers incorporating business logic are hosted using Heroku microservices which ensure smooth interaction between users and the Hasura GraphQL Engine. Information, disease detection and Authentication systems are implemented by hosting them on Heroku.

3) Web Application:

The user web application is built in React JS which communicates with the database and microservices to provide a user, who is a novice at gardening, with an easy interface of automating procedures.

Figure 11. Fig. 10. The setup of our plant with the valves and sensors

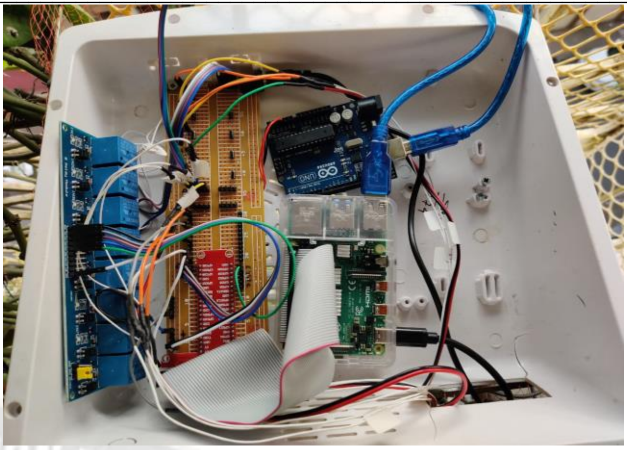

Figure 12. Fig. 11. The connection of various sensors for data collection

D. *Fuzzy Logic Controller*

The graphs for the fuzzy logic controller are shown in figures 12 & 13. They depict the range selected for both input and output variables.

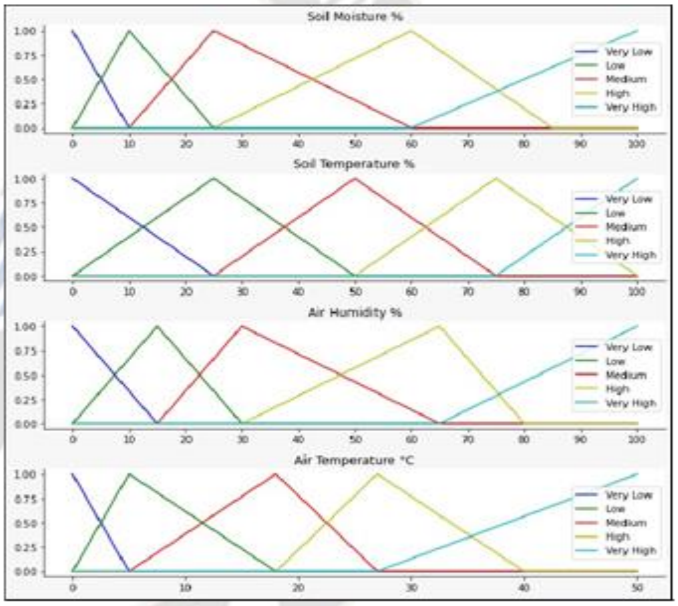

Figure 13. Fig. 12. Plot of Input variables to the fuzzy logic controller

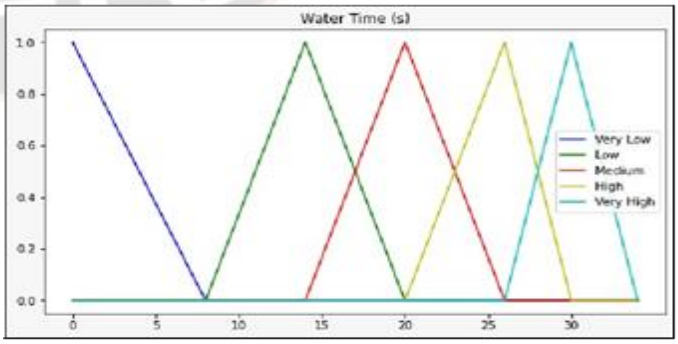

Figure 14. Fig. 13. Plot of Output variable from the fuzzy logic controller

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

E. Leaf Disease Detection

To implement leaf disease detection module, a convolutional neural network architecture is built that consists of various Convolution layers, Max Pooling layers and Dense layers with relu activation functions. The model summary can be seen in figure 14. The data set used for the training consisted of 15 classes and over 1000 images in each class. The dataset is picked from Kaggle named PlantVillage Dataset [13]. The model is trained in Google Colab using Keras, Tensorflow and Sklearn for Model building, training, data preprocessing and evaluation of the model.

One key feature to be noted here is that the model is only trained to differentiate between 15 classes of various diseases in some plants. A much deeper and much wider range of dataset has to be collected to build a model that is able to generalize well over any inputs provided to it.

```
Model: "sequential"
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 256, 256, 32)      896
activation (Activation)      (None, 256, 256, 32)      0
batch_normalization (BatchNo (None, 256, 256, 32)      128
max_pooling2d (MaxPooling2D) (None, 85, 85, 32)        0
dropout (Dropout)            (None, 85, 85, 32)        0
conv2d_1 (Conv2D)            (None, 85, 85, 64)        18496
activation_1 (Activation)    (None, 85, 85, 64)        0
batch_normalization_1 (Batch (None, 85, 85, 64)        256
conv2d_2 (Conv2D)            (None, 85, 85, 64)        36928
activation_2 (Activation)    (None, 85, 85, 64)        0
batch_normalization_2 (Batch (None, 85, 85, 64)        256
max_pooling2d_1 (MaxPooling2 (None, 42, 42, 64)        0
dropout_1 (Dropout)          (None, 42, 42, 64)        0
conv2d_3 (Conv2D)            (None, 42, 42, 128)       73856
activation_3 (Activation)    (None, 42, 42, 128)       0
batch_normalization_3 (Batch (None, 42, 42, 128)       512
conv2d_4 (Conv2D)            (None, 42, 42, 128)       147584
activation_4 (Activation)    (None, 42, 42, 128)       0
batch_normalization_4 (Batch (None, 42, 42, 128)       512
max_pooling2d_2 (MaxPooling2 (None, 21, 21, 128)       0
dropout_2 (Dropout)          (None, 21, 21, 128)       0
flatten (Flatten)            (None, 56448)             0
dense (Dense)                (None, 1024)              57803776
activation_5 (Activation)    (None, 1024)              0
batch_normalization_5 (Batch (None, 1024)              4096
dropout_3 (Dropout)          (None, 1024)              0
dense_1 (Dense)              (None, 15)                15375
activation_6 (Activation)    (None, 15)                0
=================================================================
Total params: 58,102,671
Trainable params: 58,099,791
Non-trainable params: 2,880
```

Figure 15. Fig 14. Model Summary for leaves disease detection

F. Image Stills of the garden

Live image stills of the garden can be remotely obtained even when the user is away from his local network. This can help in apprising the user about the environment around the plants. Further, the user can be notified of possible intrusions around the plant. The image stills are sent to the user on his registered email id.

TABLE III. LIVE IMAGE STILLS COMPONENTS

Name	Quantity	Specifications
Web Cam	1	at least 720p resolution.

V. RESULTS AND DISCUSSION

Figure 16. Fig. 15. Plant Information Page

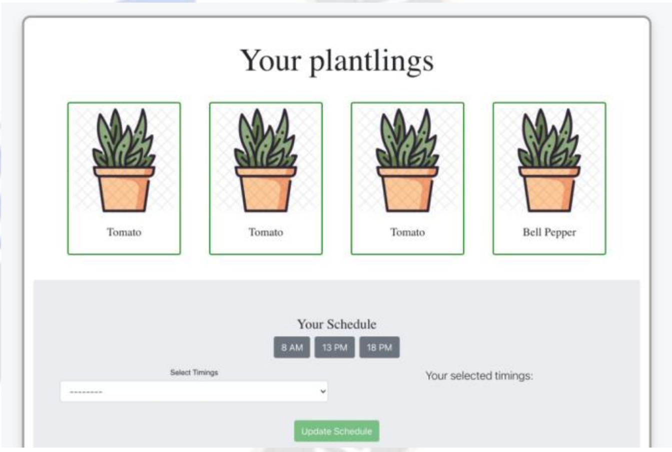

Figure 17. Fig. 16. Dashboard of Plants in User's Garden

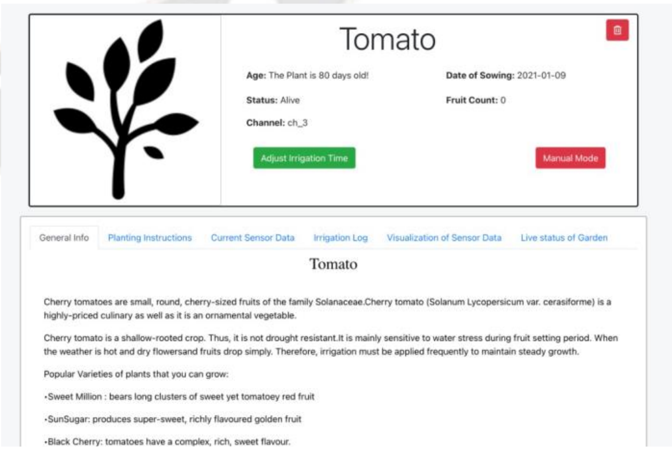

Figure 18. Fig. 17. Plant Detail Page

Figure 19. Fig. 18. Visualization of Sensor Data

Figure 20. Fig. 19. Plant Sensor Data Page

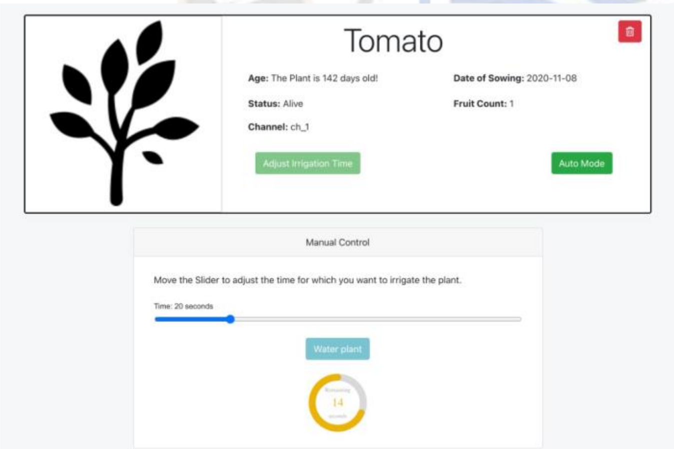

Figure 21. Fig. 20. Manual mode Irrigation of a particular plant

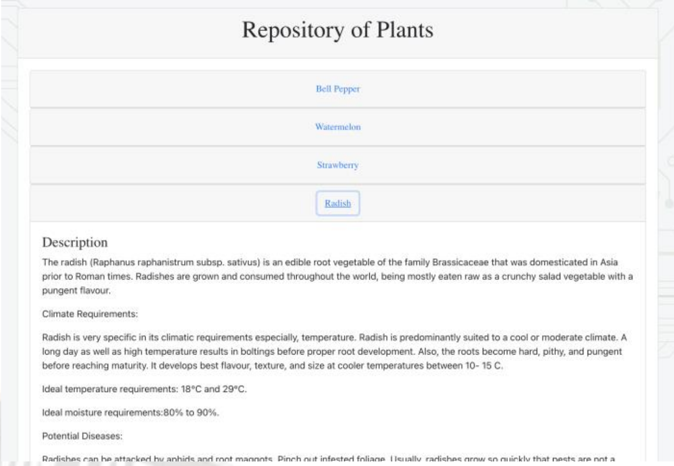

Figure 22. Fig. 21. Information System for 15 commonly home-grown plants

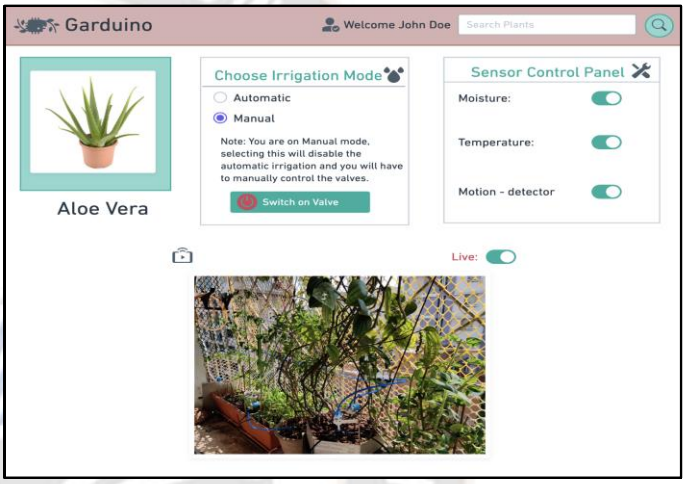

Figure 23. Fig. 22. Plant Monitor Page with video feed from the camera installed in the garden.

Apart from the web app, the model we have trained earlier for our Leaf Disease detection, has tested to be 89.09% accurate on the Training data and 68% accurate on the Test data. The training and the validation loss and accuracy plot provides enough information that there is scope of improvement in a better model approach. We simply lack the resources to build the network and train it. Figure 23 and 24 depicts the graphs of Training and Validation Loss and Accuracies.

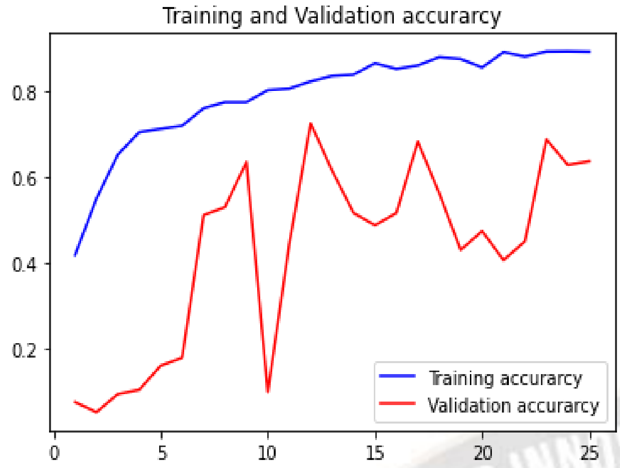

Figure 24. Fig. 23. Plot of Training and Validation Accuracy.

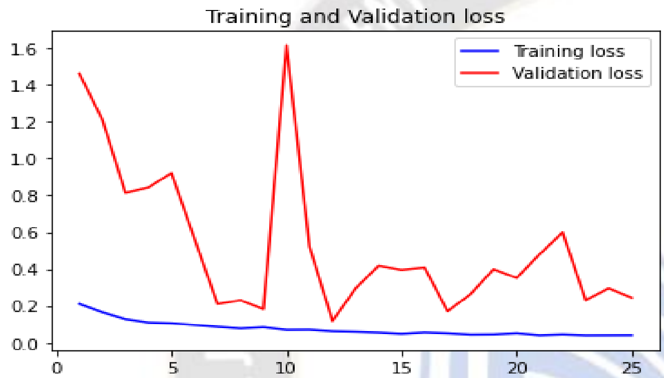

Figure 25. Fig. 24. Plot of Training and Validation Loss

VI. FUTURE SCOPE

The paper can be extended to include various other features as well:

Early detection of diseases is a major challenge in the field of agriculture science. In the early days the pest identification was done manually. It is very time consuming and it also requires continuous monitoring by experts. The farmers try to organize regular spray programs without considering the presence of pests or the density of pests. This process inversely affects the environment.

Thus, the scope can further be extended to automating the task of detecting pests. An IR sensor and ultrasonic sensor can be used to detect the number of insects and their subsequent growth. Further through OpenCV and Neural networks we classify and recognize the type of pest.

Further, pH sensor can be used to determine the quality of water being supplied to the plants since much of the water today is highly polluted. This can prevent the plants from getting affected.

By using motion sensors and appropriate image processing techniques, an alarm system can be set up to alert the user if a bird or an animal makes an attempt to harm the plant.

Further, a disease detection module can be implemented to detect potential diseases from the leaf by image processing techniques.

This system can also incorporate Hydroponic/Aquaponic method of growing plants in order to obtain higher efficiency and yields by controlling Air Temperature and moisture, Water Temperature and its nutrients.

VII. BENEFITS AND SCOCIAL IMPACT

Promotes a healthy lifestyle by imparting the values of appreciating nature, cultivating home-made, organic vegetables and fruits and ensuring that greenery is sustained amidst the rising levels of pollution.

Encourages people, who tend to avoid gardening due to their busy schedules and frequent travelling, to continue taking care of their plants by automating tasks and educating them with the ideal conditions for plants.

Prevents the untimely death of plants due to diseases.

Aids in the conservation of water as the system decides on the appropriate quantity based on multifarious factors and requirements.

It will help in water conservation which in turn will have a positive impact on the environment. The system will help in increasing the green cover in cities.

VIII. CONCLUSION

The authors were successful in implementing the proposed design and the prototype of the smart garden for agriculture. There are various features that can be added as a part of the future scope which are mentioned above. This prototype can further be used as a part of a bigger paper catering to huge agricultural fields and greenhouses.

REFERENCES

[1] Smart Irrigation System using Raspberry Pi: Jainish Kumar, Anghan, Parveen Sultana H, School of Computing Science and Engineering (SCOPE), VIT University, Vellore. International Journal of Scientific Engineering Research Volume 9, Issue 6, June-2018.

[2] Sahu Tanu, and Ashok Verma. "Automated smart irrigation system using Raspberry Pi." International Journal of computer applications 172, no. 6 (2017): 9-14.

[3] Pawar Swapnali B., Priti Rajput, and Asif Shaikh. "Smart irrigation system using IOT and raspberry pi." International Research Journal of Engineering and Technology 5, no. 8 (2018): 1163-1166.

[4] Sahoo Khirod Chandra, and Umesh Chandra Pati. "IoT based intrusion detection system using PIR sensor." In 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communica- tion Technology (RTEICT), pp. 1641-1645. IEEE, 2017.

[5] Yadhav S. Yegneshwar, T. Senthilkumar, S. Jayanthy, and J. Judeson Antony Kovilpillai. "Plant Disease Detection and Classification using CNN Model with Optimized Activation Function." In 2020 International Conference on Electronics and

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

Sustainable Communication Systems (ICESC), pp. 564-569. IEEE, 2020.

[6] Sukhdev, Nikhil, Naman Nahata, Sunil Sridhara, and Gnana Swamy. "IoT Enabled Smart Gardening." In 2018 Fourteenth International Con- ference on Information Processing (ICINPRO), pp. 1-2. IEEE, 2018.

[7] Al-Omary Alauddin, Haider M. AlSabbagh, and Hussain Al-Rizzo. "Cloud based IoT for smart garden watering system using Arduino Uno." (2018): 33-6.

[8] Islam Asif, Prem Saha, Masud Rana, Md Mim Adnan, and Bishwajit Banik Pathik. "Smart Gardening Assistance System with the Capability of Detecting Leaf Disease in MATLAB." In 2019 IEEE Pune Section International Conference (PuneCon), pp. 1-6. IEEE, 2019.

[9] Ishak S. N., N. N. N. Abd Malik, NM Abdul Latiff, N. Effiyana Ghazali, and M. A. Baharudin. "Smart home garden irrigation system using Raspberry Pi." In 2017 IEEE 13th Malaysia International Conference on Communications (MICC), pp. 101-106. IEEE, 2017.

[10] Sharma Sachin, Abhinav Sharma, Tanya Goel, Rohan Deoli, and Seshadri Mohan. "Smart Home Gardening Management System: A Cloud- Based Internet-of-Things (IoT) Application in VANET." In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1-5. IEEE, 2020.

[11] Al-Ali, A. R., Murad Qasaimeh, Mamoun Al-Mardini, Suresh Radder, and Imran A. Zualkernan. "ZigBee-based irrigation system for home gardens." In 2015 International Conference on Communications, Signal Processing, and their Applications (ICCSPA'15), pp. 1-5. IEEE, 2015.

[12] Hadi, Mokh Sholihul, Pradipta Adi Nugraha, I. Made Wirawan, Ilham Ari Elbaith Zaeni, Muhammad Alfian Mizar, and Mhd Irvan. "IoT Based Smart Garden Irrigation System." In 2020 4th International Conference on Vocational Education and Training (ICOVET), pp. 361-365. IEEE, 2020.

[13] Tairu Oluwafemi Emmanuel, Plant Village Dataset: Kaggle, version1, Available: https://www.kaggle.com/emmarex/plantdisease [Accessed: November 1, 2020]

[14] Xiao Fan Mok, Smart Garden project, Available: https://www.hackster.io/mokxf16/smart-garden-raspberry-pi-arduino-65c7b7, Feb 2019.