Design and Implementation of a Low-cost IoT Node for Data Processing, Case Study: Smart Agriculture

Merihan E. Amr[1], Amr A. Al-Awamry[2], Marwa A. Elmenyawi[2], and Adly S. Tag Eldien[2]
[1] Valley Higher Institute for Engineering & Technology, Cairo, Egypt
[2] Faculty of Engineering, Benha University, Benha, Egypt
Email: merihan.essam1@gmail.com; {amr. awamry, marwa.elmenyawi}@bhit.bu.edu.eg; adlytag@feng.bu.edu.eg

Abstract**—**The majority of IoT nodes work within specific scenarios and can be configured in different ways. This paper seeks to design and implement a low-cost Internet of Things node for research applications to make it suitable for a wider variety of scenarios. The design is divided into hardware board and mobile application. With Bluetooth, the mobile application can connect to the node, and the node can collect data, store this data in the ThingSpeak database, control some connected devices, and check if the connected devices are on or off. The node was designed and tested for research purposes using smart agriculture as the case study. The system detects temperature, humidity, and soil moisture using node sensors, enabling data collection and interpretation by smartphone and web application. There are many challenges associated with the collected data preparation, analysis, visualization, and prediction using the Softmax function for optimal future management. Python was utilized to apply necessary data analysis techniques. The system saves time and makes farming more convenient as it uses few resources in terms of hardware and cost.

Index Terms**—**Internet of Things, system design, ThingSpeak, smart agriculture, data analysis, Softmax

I. INTRODUCTION

On account of the rapid development of the Internet and the Internet of Things, we are all highly integrated on a different scale. Internet of things is often shortened as IoT refers to the interconnection of various devices or any appliances Through any possible situation[1]. Using IoT, we can program devices that we use in daily life or for specific repetitive, and some non-repetitive, tasks automatically, and that is what we call automation which means reduce human interference in the process. That is why we can say that the Internet of Things (IoT) has begun to play a critical role in everyday life. In an IoT system, data are collected from different sensors stored in a server for analysis, simply monitoring, or the data can be used in various other processes according to the system's functionality. IoT systems can be used to automate processes, like collect data for a set amount of time or control devices by turning them on and off automatically when a condition happens or for a set amount of time. This paper offers a simple and low-cost solution besides being less complex as it consists of only two parts as shown in Fig. 1 and Table I one is the hardware node. The other is the gateway or the system interface represented in a simple mobile app from where you can monitor the data and control the devices even if you are thousands of miles away from the experiment or the process you need to monitor. The Internet has many considerations that cause problems as it is not a single network. Therefore, there are many challenges of IoT like security, connectivity, cost, regulation, standard, development, etc.[2].

The paper introduces an IoT node that provides a convenient solution for two IoT challenges: connectivity and cost. Internet of things applications are developed and published in several areas such as transportation and logistics, healthcare, retail and supply chain, industry, and environment [3], [4].One of the most important applications in agriculture.

There are many IoT applications in agriculture, such as data collection of temperature, rain, humidity, and soil content [5] as it helps farmers to obtain temperature and soil moisture information.

TABLE I: SYSTEM PARAMETERS

Parameters	Type	Voltage
Input	Sensors	Depending on application
Output	Actuators	
CPU	Microcontroller	
Wireless Communication	WiFi / Bluetooth	
User Interface	Mobile App	

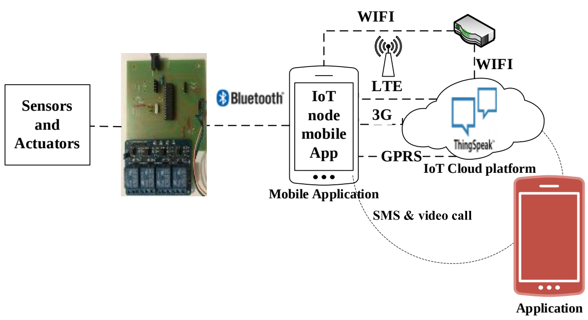


Fig. 1. Node design

A new idea was introduced in a system that provides additional work in the making of exhibits. Promote commercial farming using the Internet of Things to improve automation and reduce farmers' manual labor, energy consumption, and water scarcity. For example, farmers spend the most time in the field to monitor the

Manuscript received July 4, 2021; revised January 15, 2022.
Corresponding author email: merihan.essam1@gmail.com.
doi:10.12720/jcm.17.2.99-109

level of moisture, and they need timely watering. To overcome this problem, we are nowadays in the modern world, there is an advanced technology used in agriculture called Smart Agriculture utilizing the Internet of things. It means the use of sensors to collect data to achieve greater accuracy and progress in agricultural processes, for example, soil, nutrients, energy, and waste management [6]–[8], which can affect the future of agriculture. As a case study, we choose to apply the experiments in the smart agriculture research field. With the help of research focused on smart agriculture systems, we understood the agricultural parameters better and which parameters to enhance or increase agricultural production efficiencies.

The system was found to be very beneficial for agricultural systems as it can collect data related to environmental standards such as humidity, soil moisture, temperature, and pH [9]–[11] over long periods without the need for human interaction. The data uses different sensors to make the farming process more efficient and accurate [12]. Several articles visualize the relationship between Smart Farming and IoT / WSN. Researchers are focusing on how to use this high-tech optimization to develop functional intelligence data transmission systems [13]. Description of IoT technology of the latest revolution in every field in the common person's life by making everything intelligent and Clever. IoT refers to a network of things that make a self-configuration network. Intelligence development IoT smart agriculture-based devices day to day convert the face of agricultural production is not only by strengthening it but also makes it cost-effective and reduces waste. This paper aims to propose a novel smart IoT based on agriculture that helps farmers get live data (Temperature, soil moisture) for effective environmental monitoring. That will enable them to do intelligent cultivation and increase the total yield and the quality of the products [14].

Moreover, the structure of IoT is also detailed in many researchers and organizations. Furthermore, the structure of IoT is also detailed in protected Agriculture, including platforms, modern wireless technologies, cloud computing, machine learning, big data, animal husbandry, hardware, software challenges, etc. [15]. Many researchers and developers are working on microcontroller boards (for example, Arduino or Raspberry Pi) for IoT applications which are relatively expensive and complex. Current IoT nodes often provide the internet connection through one technology and 'it's a cost issue. As the number of sensors and modules integrated increases, the cost of IoT nodes increases.

We propose a node that uses a mobile phone and a simple embedded system with a Bluetooth connection in between. The main tasks performed by the board are to collect sensor readings, receive commands, and transmit data wirelessly, which is then received by the mobile phone connected to the node. Then the mobile application deals with internet communications and that is how we solve the connectivity challenge by utilizing the built-in mobile phone modules such as WIFI, LTE, CDMA, GPRS, GSM. It also solves the Cost challenge as most people own smartphones and they have various price ranges and do not have to buy a specific device for the system.

In the case study, we used this node in an agriculture app to measure moisture, temperature, and humidity and analyzed the results. Every sensor and module (camera, GPS, Touchscreen, Microphone, Audio output ...) in the mobile phone can be used in the IoT node for other systems and other users. The mobile phone provides them with no need to install these modules. Any other module not included in the mobile phone can be integrated using the piece of hardware (Fingerprint, RFID…).

The paper is organized as follows: The state of the art is discussed in Section II. In section III, we investigate the node. In section IV, you will find a brief overview of the case study. Section V deals with data analysis and prediction. There is a discussion of the results in section VI. In VII, the findings and future work are discussed. And finally, the conclusion.

II. State of the Art

A. *The Hardware*

The hardware board is intended to connect the physical world to the mobile application. Using Bluetooth, the board executes actions and collects readings that are sent to the mobile phone. Fig. 2 shows the system flowchart. The system's initialization process includes the following steps:

- Configuring the analog input pins that take sensors readings.
- Configuring the output pins that transmit the actuator actions.
- Initialize timer, which counts by second to control the sensor reading rate.
- Initialize Bluetooth link using UART serial communication protocol.

With the free ThingSpeak subscription rate [16], the update time is 16 seconds. Therefore, the sensor readings update rate has been calibrated to be one reading every update. When the serial is not sending or receiving data, the system asks if the user is sending something or not. If the user sends an item, it asks if this is a setting frame? Or is this a command frame? the setting frame is the frame that comes from the mobile phone to automate the behavior of the outputs and set it to operate for a specific duration. We can choose to loop it indefinitely or for a certain amount of time and a command frame is a frame that comes from the mobile phone to change the state of one or more actuators. If one of them, it will parse the frame.

The system then searches for the keywords: settings and command. As soon as the frame parses the number, it will go to the function of the board setting and translate the numbers in it corresponding to board settings. In contrast, when the frame parses the number of the command, it will change the actuator state corresponding to the parsed number of the frame. The frame is then saved to an EEPROM (Electrically Erasable Programmable ROM).

If the serial has no data, either the frame was sent incorrectly or the user didn't send any frames. Hence, the processor asks about the serial condition, receives the sensor data, and reads it using ADC.

A timer interrupt will be triggered every second to increment a counter. After 16 seconds; data will be sent over Bluetooth to the mobile application, which will send this data over Wi-Fi to ThingSpeak.

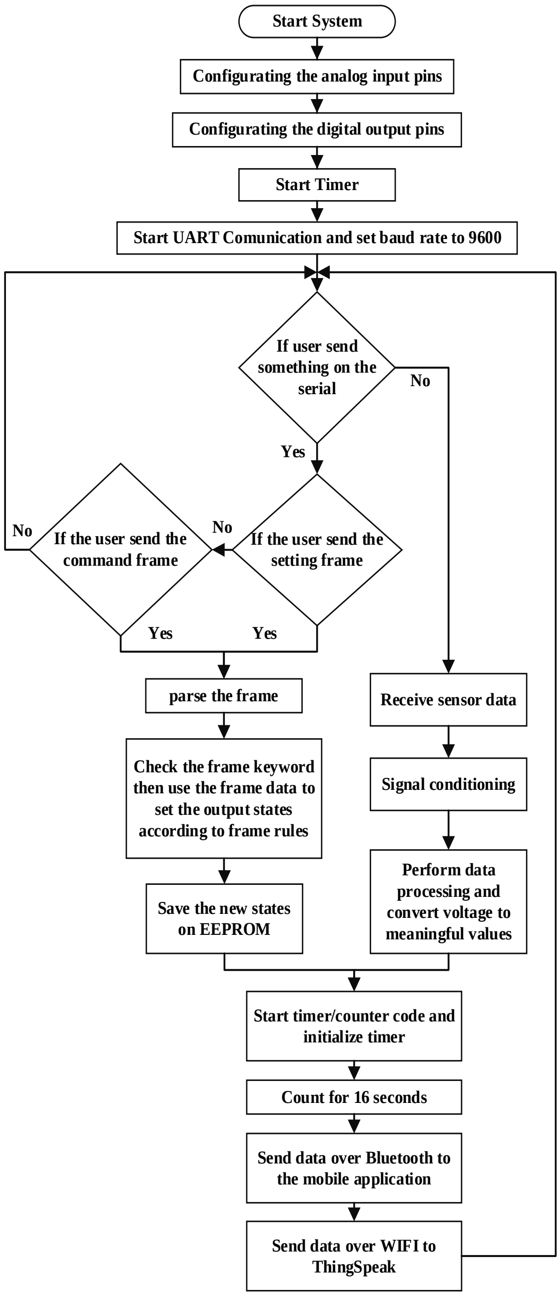


Fig. 2. The system flowchart

B. The Mobile Application

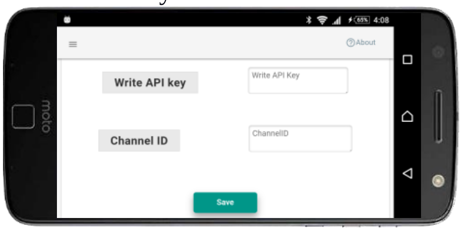
As one of the key elements of the system, the mobile application follows the following guidelines:

- The touch screen serves as a user interface.
- Communicates with the cloud provider for IoT.

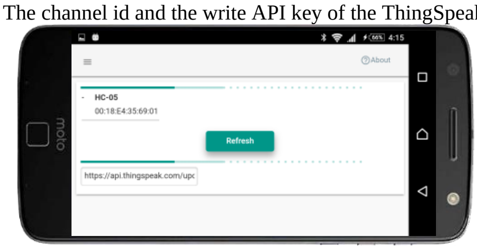
Cloud service providers offer features such as cloud storage, data analysis, and data visualization.

Aside from the features above, ThingSpeak is the cloud provider of choice because MATLAB code can carry out data analysis and manipulation. Thus, academic researchers favor ThingSpeak.

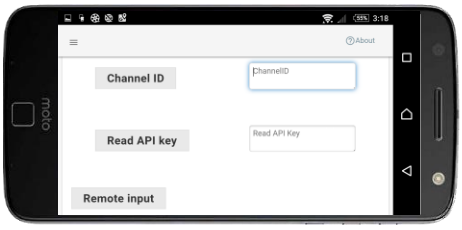
ThingSpeak communicates in two ways, sending and receiving data. The mobile application in Fig. 3, controls the application both ways:



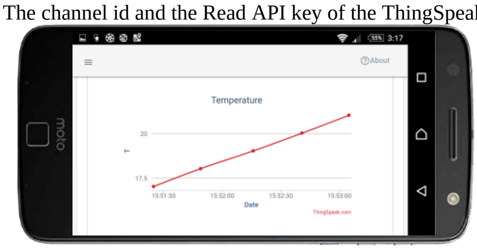
a. The channel id and the write API key of the ThingSpeak



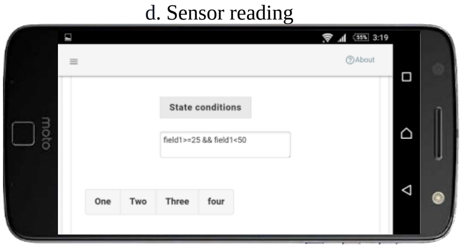
b. Pair and connect with the hardware board



c. The channel id and the Read API key of the ThingSpeak



d. Sensor reading



e. State conditions

Fig. 3. The mobile application

- In the case of Sending data to ThingSpeak, after creating a ThingSpeak account and creating a channel to receive readings from IoT sensors, as shown in Fig. 3(a), the user will need to enter the channel ID and the API key that is generated by ThingSpeak. Following that, we connect the mobile phone to the hardware board, as shown in Fig. 3(b). Finally, the user can view the readings gathered through the channel readings page, reflected in Fig. 3(d).

- In the case of receiving data from ThingSpeak, to select the remote output control page, the user can use the navigation icon to open the side menu. Fig. 3(c) shows the channel ID and the Read API key entered by the user.

The user is then required to set states and see the corresponding action on the relay on the state conditioning page as shown in Fig. 3(e).

To develop the mobile application, we used the Apache Cordova [17] cross-platform development framework. Since Cordova-platform development can target multiple platforms using almost the same codebase, we chose it for cost savings. This mobile application was designed by Nitobi using HTML5, CSS3, and JavaScript.

C. *IoT Architecture for Big Data Analytics*

The Internet of Things architecture is based on the abstraction and identification of the IoT domain.

In addition, it identifies relationships between various IoT sectors, such as smart transportation, smart homes, and smart health. Data can be extracted with the help of the big data analytics architecture.

Various IoT architectures have been described in the literature [18], [19]. The IoT architecture of our proposed node is shown in Fig. 4. The sensor layer (the perception layer) contains all sensors on a wireless network.

Our system uses Wireless Network connections to interact with sensors (peripheral network) like Bluetooth and telecommunications networks (host network) as LTE, WI-FI, 4G, 3G … etc. The IoT gateway enables Internet and Web communication.

The application layer relates to big data analytics, where a large amount of data from sensors is stored in the cloud and accessed through big data analytics applications. With the mobile application used, API management and a dashboard assist in interacting with the processing engine.

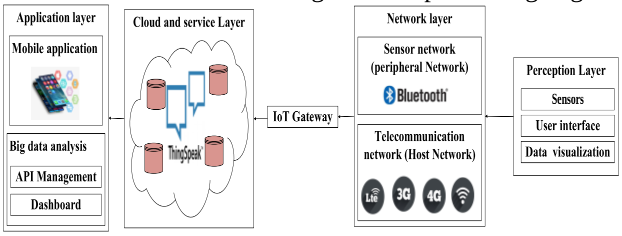


Fig. 4. IoT architecture for big data analytics

IV. Investigation of the Proposed Node

To investigate the node, we have two scenarios: uplink and downlink as shown in Fig. 5.

The uplink scenario is verified by using four rotary potentiometers to act as sensors. These potentiometers change their resistance values by rotating them and each one is connected to a pin in the microcontroller that is configured as input ADC. In Each potentiometer has a field on the ThingSpeak channel that node updates voltage values. The downlink scenario is verified by using a four relays module. Each relay module is connected to a pin in the microcontroller that is configured as an output. The node downloads the relay states set by the user then it updates the relays connected to the node.

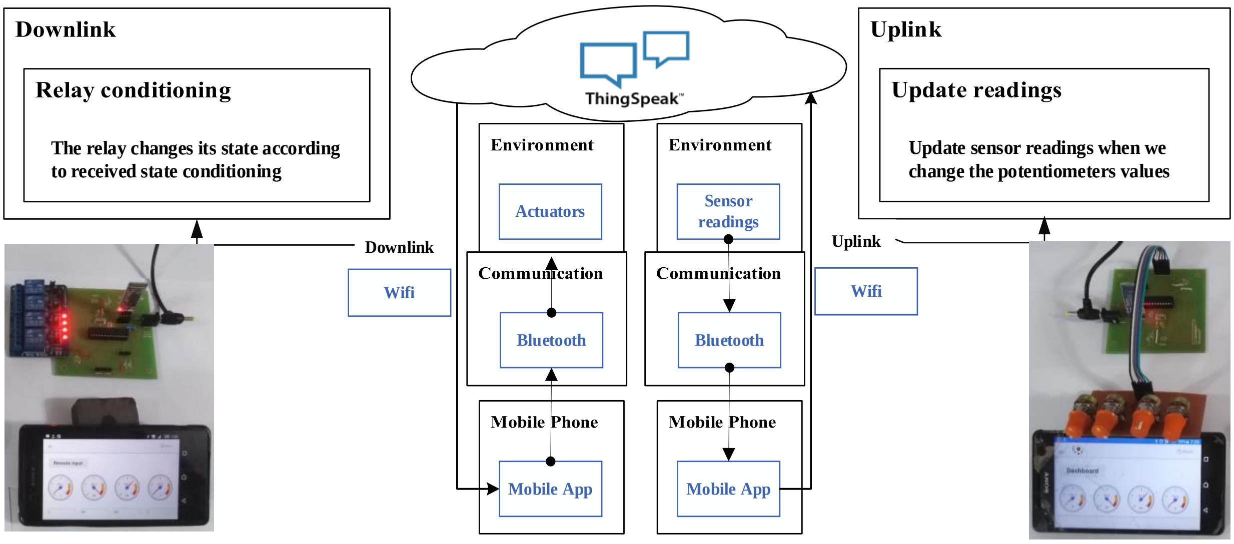


Fig. 5. Node readings: uplink and downlink scenarios

III. Case Study: System Overview

Fig. 6 and Table II demonstrate our IoT node in agriculture application for monitoring soil moisture, temperature, and humidity over a long period of time for four types of soil. On the Thing Speak server, where the data is stored for monitoring and archiving, two weeks of data are displayed. Data is transferred to the cloud using a Wi-Fi modern system, and the time difference between each instance is 16 seconds. To prepare for analysis, the collected data can be arranged and converted into an Excel sheet.

The hardware board responsible for data collection is measured by the sensors. The used controller is the node

Microcontroller unit Atmega328, which is well suited for simple tasks like collecting data and sending it via Bluetooth. In addition, the HC-05 Bluetooth module is used because it is more than sufficient to serve as the communication link between the hardware and the mobile phone.

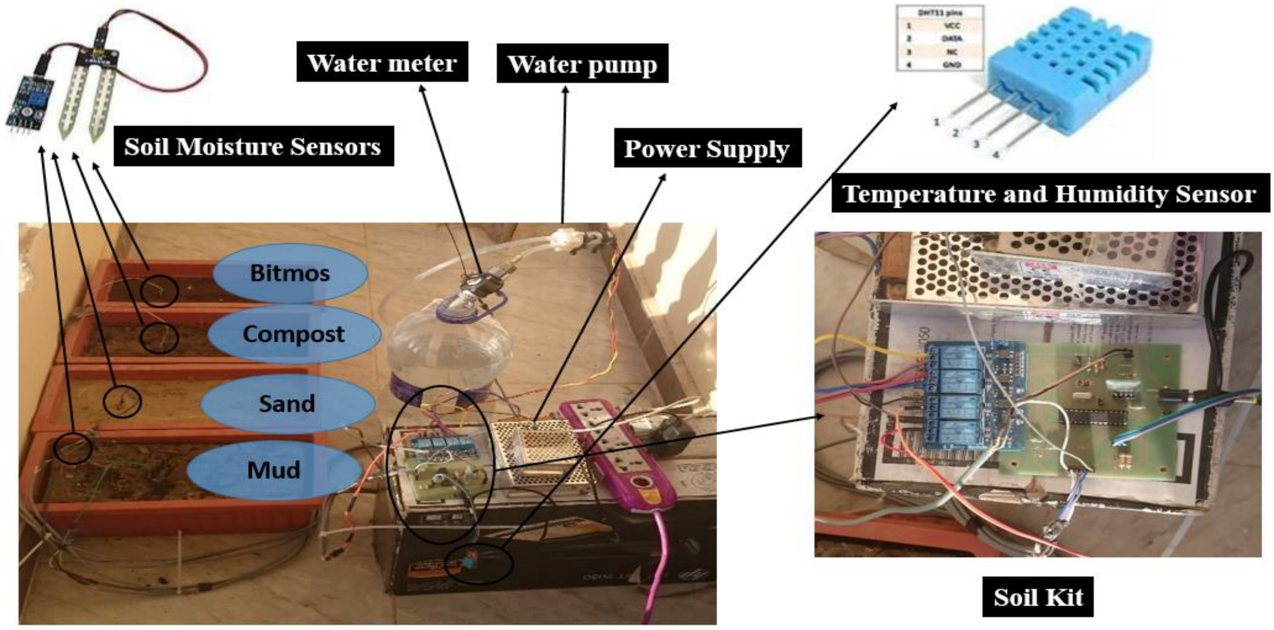


Fig. 6. The soil analysis system

TABLE II: SYSTEM PARAMETERS VALUES

Parameters	type	Power
Sensors	Soil moisture, Temperature, Humidity sensor , and water meter.	5 volts
Actuators	Water pump, relay module	5 / 12 volts
Microcontroller	AVR 328	5 volts
Wireless Communication	Wi-Fi/Bluetooth	5 volts
Mobile App	Soil Analysis	5 volts

There are four soil moisture sensors used to measure moisture in different soil types (mud, bitmos, compost, and sand), one for each soil type, and each sensor has its field on the ThingSpeak channel.

Temperature and humidity sensor DHT11 is used since it is a combined temperature and humidity sensor that provides a wider temperature range and better accuracy. Specifically, its low consumption makes it more appropriate for most smart farming systems. The water meter sensor is used to monitor water consumption.

Additionally, coupled with the relay module, the output from the device can be used to start irrigating the soil remotely with the amount of water required. The data collected by sensors attached to analog and digital board pins (ADC) can be stored locally or sent to the server.

The HTTP (Hypertext Transfer Protocol) protocol is used to transfer data to the server. During data collection from the sensors, the IoT node functions as the client and sends an HTTP request to the web server over the Internet. Another role for the IoT node is that the user can also be the recipient. By definition, the webserver runs on the client device and serves requests from the network. A user can access and control the node, as well as relays attached to it. The mobile application is used as the user interface to communicate with the cloud through Wi-Fi.

V. DATA ANALYSIS AND PREDICTION

We cannot apply analysis directly to raw data in most experiments. We have to transform it first into a suitable format for analysis. This process is referred to as Data Preparation.

Afterward, we will discuss predictions with an introduction to the Softmax Machine Learning model that we used. In each part, you will see data visualization to analyze how efficient that part is.

A. Data Cleaning

The system was built for experiments, so let's imagine an expected behavior happened like a power surge, sudden environmental phenomenon, sensors go rusty which happened a lot in our system, etc. Due to these conditions and more, there will be wrong inaccurate readings, or it can take no readings at all. In this case, it will affect our analysis badly causing incorrect results and speculations.

Using Python and Pandas, we simulated unclean data in Fig. 7(a). You can see the temperature data trend throughout the whole testing period with simulated null values, no readings, before cleaning, and how it differs after cleaning in Fig. 7(b).

To fill the missing data, many algorithms [20] can be applied:

- We can fill with the data mean but it is unsuitable for many experiments with high data diversity.
- We can remove the rows with missing data, but it faces some challenges because if more than 30% of the data rows had missing data, it would result in wrong data representation
- Another way is to fill the missing data with the average of both the last and next data to keep my data visualization consistent and clean. But it 'wasn't very accurate in this case as there was some consecutive null data as in (1).

$$missing\ data \approx \frac{last\ data + next\ data}{2} \quad (1)$$

- The most suitable way for this case is to fill the missing data with the mode of specified intervals. The day was divided into 6 periods, each one with 4 hours. The mode of each period was used to fill the missing data in those 4 hours as in (2).

$$missing\ data\ per\ peiod \approx \text{mode of the period} \quad (2)$$

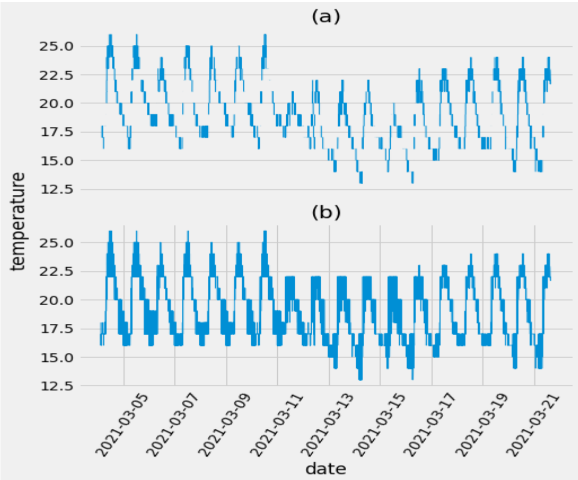


Fig. 7. Temperature data before and after cleaning

B. *Data Wrangling*

As soon as our data is cleaned, we wrangle it [21], which means removing useless data or data that is not relevant to our analysis. It is up to us to decide whether the data is useless or not based on how it can be fully visualized. In this study, the original readings of the soil moisture sensors before being divided by 1023 are irrelevant to the analysis. Not only useless data affect the data visualization, as some data cause the wrong presentation to the other data in the study, or data with different value ranges affect the analysis badly, causing incorrect results and speculations. The data trend can be presented using Python and Pandas but not all of the data is included in the presentation as seen in Fig. 8.

Temperature and humidity are the only properties that can be visualized, while the other properties have values near 0. For the sake of presenting the other, Fig. 9, shows that they are not only near 0, but they can have an effective trend. Thus, in this system, for any experiment, data will be categorized according to the value ranges. Each range will be visualized in a unique graph to the grantee that we see the data trends of all experiment properties.

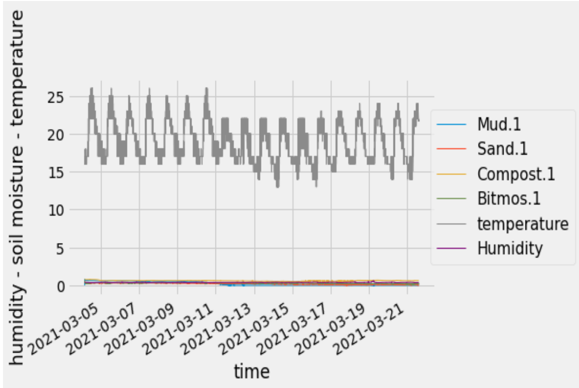


Fig. 8. Data before wrangling

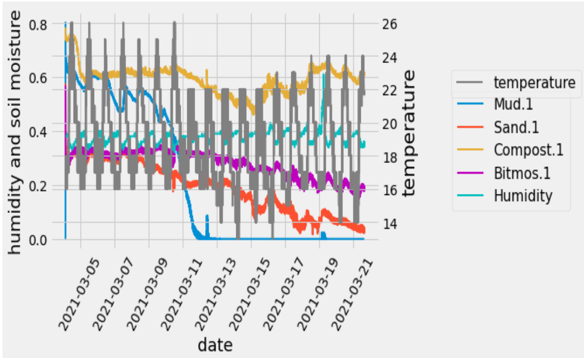


Fig. 9. Data after wrangling

C. *Data Visualization*

Using custom queries .xlsx files were extracted from the ThingSpeak database and generate automated graphs using the Pandas graphing extension. Fig. 9 presents the data collected from March 5, 2021 to March 21, 2021. Throughout the testing period, it shows the data trend of the major properties. Python and Pandas were used to show the temperature graph in Fig. 10, the humidity graph in Fig. 11, and the soil moisture graph in Fig. 12. We found that analyzing the data like this will provide crucial information for the agriculture application and verifies the system's efficiency.

Graphs like these are used to help visualize the environmental properties. However, extra work needs to be done to determine the daily and weekly patterns of the best soil to use for agriculture. From the extensive data collected and with graph generation automated with the help of Python and Pandas, a graph per day of the week was easily created for each property for different times of the day. In Fig. 10, you will find four graphs describing how temperature was at the critical periods of the day noon, sunrise, sunset, and midnight. Compared to the humidity distribution in Fig. 11, you will find that every two curves for each period, for example, the noon graphs, are opposite to each other, which validates our system efficiency. In Fig. 12, you can see the soil moisture distribution over the same four periods. The four graphs are nearly identical, proving that the moisture is not dependent on the temperature, humidity, or natural light (different periods of the day). One possible recommendation is that we can add another sensor to collect data of another property related to soil moisture to compare them together and do a more effective analysis.

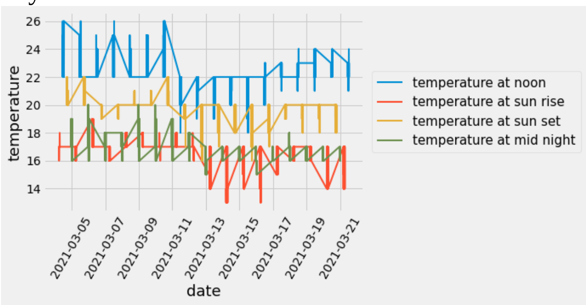


Fig. 10. Temperature data at different times of the day

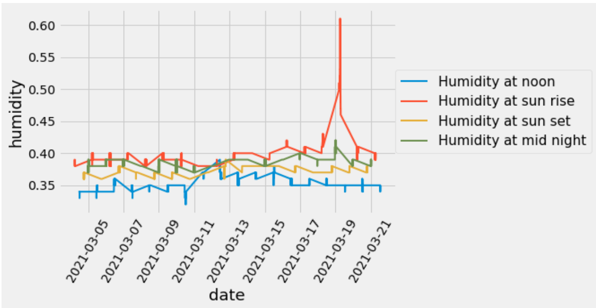


Fig. 11. Humidity data at different times of the day

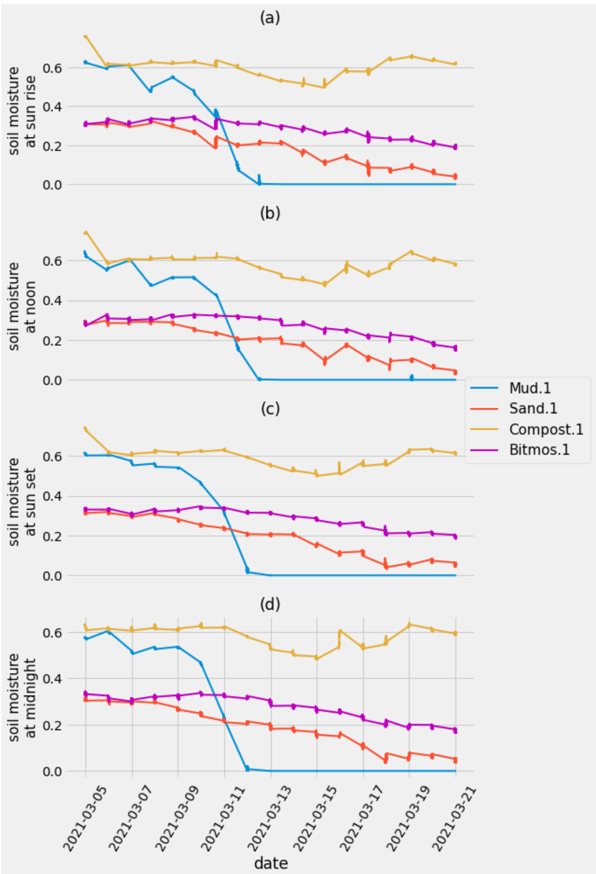


Fig. 12: Soil moisture data at different times of the day

D. Data Prediction

Many researchers need to forecast the future readings or the environment sensors and to do that accurately. We need to use an AI model that will increase the dependency on the generated data.

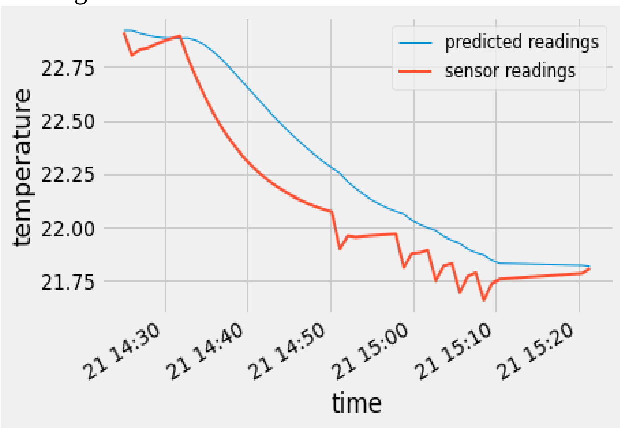


Fig. 13. Data prediction

Fig. 13 illustrates the AI rule in our system and you can see how we used the Softmax function [22] to make predictions. Softmax is very helpful because it converts scores into a normal probability distribution, which can be viewed by the user or used as an entry point for other systems.

Therefore, it is normal to add the Softmax function as the final layer to the neural network. Since many multi-layered neural networks terminate in a penultimate layer that produces results that are not properly scaled and that can be not easy to handle. The function is defined mathematically as in (3).

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_i}} \qquad (3)$$

$\vec{Z}$ is the input vector (z0,z1,... zK) to the Softmax function. e^{z_i} an exponential function is applied to each element of the input vector. K is the number of categories in the multi-category workbook. Here, all the vector elements of Z_i input vector elements and can take any real value, positive, zero, or negative. The normalization term $\sum_{j=1}^{K} e^{z_i}$ at the bottom of the formula ensures that all job output values will add up to one, which constitutes a valid probability distribution.

VI. Results

A. Analysis Results

Using Python and Pandas we can determine the maximum and minimum of each parameter as shown in Table III. We have to keep in our minds that it will differ from an environment to another.

From the first look, we can falsely claim that mud is the best soil because it has the highest moisture. However, looking back at the graphs in Fig. 9, we can say that compost is the best soil after that Bitmos then sand and surprisingly mud is the worst of the four types of soil. Although compost has lower maximum moisture than mud, it keeps the moisture for longer periods same as Bitmos when compared to sand. On the other hand, the mud is the worst when it comes to keeping the moisture.

For the temperature and humidity, we mentioned before that the humidity is low relative to the environment when the temperature is high and vice versa.

TABLE III: VALUE RANGE ANALYSIS

Parameters	Value range
Temperature	18° – 26°
Humidity	32% – 61%
Soil Moisture - Mud	0.0% – 80.0%
Soil Moisture - Sand	1.661% – 57.67%
Soil Moisture - Compost	46.13% – 78.40%
Soil Moisture - Bitmos	14.37% – 56.89%

B. Soil Classification

Figs. 10-11-12 demonstrate that soil moisture is not affected by temperature or humidity. Therefore, watering the soil at any time of day will have the same effect for the same season with close temperature values. The recommended watering intervals can be determined using (4).

$$watering\ interval = d_{Xi} - d_{X1} \quad \forall\ [(X_1 - X_i) * 100]_i^n > 30 \text{ and } i = 1,2,..,n \tag{4}$$

where the watering interval is the period of time, in days, that lasts between each time the soil needs to be watered. d_{X1} is the day of the first observation while $d_{Xi}$is the day of the i^th observation. $X_1$is the moisture value of the first observation, and X_i is the moisture value of the i^th observation. The number n represents the number of observations.

Equation (4) represents the difference in days between the first watering day and the day when the soil moisture decreased by more than 30% of the original moisture.

In Compost soil, the watering intervals are large because the soil retains water well, requiring watering every eleven days. On the other hand, Mud soil has a high moisture content, but it cannot retain water, so it should be watered every three days. It is expected that watering the Bitmos soil every ten days will be sufficient, but it is not suitable for all corps. Furthermore, sand is not good for all corps since it is low in moisture and loses water more quickly than Bitmos, so watering the sand every six days will keep it hydrated**.**

C. Prediction Results

Comparing the predicted data with the measured data reveals high accuracy. As you can see in Fig. 13, we have compared the predicted temperature values against the actual readings from our sensors. To evaluate the overall prediction process, equation (5) calculates the percentage of predicted readings, p_d, that are within a specified range of deviations from the measured readings.

$$p_d = \frac{\sum_{i=1}^{n} f(PV_i, MV_i, d)}{n}$$
$$f(PV_i, MV_i, d) = \begin{cases} 1 & |PV_i - MV_i| \le d \\ 0 & |PV_i - MV_i| > d \end{cases} \tag{5}$$

where MV is the measured value, PV is the predicted value, n is the number of observations and d is the defined deviation. The numerator of (5) sums the values of $f(PV_i, MV_i, d)$ for different i values. It equals 1 when $(|PV_i - MV_i| <= d)$ and 0 when $(|PV_i - MV_i| > d)$. The sum is then divided by n to determine the p_d.

The function was calculated using Python to calculate the p_d for different d values then we found that the p_d is 100% when the $d \ge 0.36$ and drastically decreases when $d < 0.36$, as presented in Table IV.

TABLE IV: PERCENTAGE OF PREDICTED READINGS FOR DIFFERENT DEFINED DEVIATIONS

d	p_d
0.36 or more	100%
0.35	98%
0.3	80%
0.2	52%
0.1	24%

D. Comparison

Comparing our system with an another system in the same field with the same functionality, you will realize that customizing our system, programming a data frame for each case scenario, and using only open source developing tools in developing the website, the mobile app and data analysis were really good ideas and very suitable for this task.

In addition to using a smartphone to be a part of the main system is cost-efficient as it has many features and default modules that can be utilized in future advancements.

A comparison between this system and the system proposed by Muangprathub *et al*. [13] was made and summarized in Table V.

TABLE V: SYSTEM COMPARISON

Parameters	Our system	Muangprathub
Sensors	Humidity Temperature Soil moisture Water Meter	Humidity Temperature Soil moisture Water Meter
Actuators	4 channels Relay Module	5 channels Relay Module

	Water Pump	Water Pump
Hardware (IoT node)	Customized	Readymade Node MCU
Node reconfigurability	Reconfigurable	Can't be reconfigured, only replaced
Microcontroller	AVR	ESP8266
Processing Speed	16 MHz (customizable)	80 MHZ (not customizable)
Connectivity	Bluetooth WIFI	WIFI
e App	Cross-platform (CORDOVA)	line Application
Compatibility with different devices	Existed	Not existed
Data analysis	Automatic data analysis	Manual data analysis
Data prediction	Existed	Not existed

The system, we offer a better deal in many aspects such as:

- In the case of design, we designed our hardware to achieve minimization to keep it as simple as possible with a lower cost.
- In the case of the application, we developed our application using Cordova open-source cross-platform development app, and it can be easily modified according to the experiment.
- In the case of connectivity, we found that Bluetooth is sufficient for most experiments and it can be simply upgraded to WIFI if needed.
- In the case of cloud storage, we used ThingSpeak to collect the data and access it anytime from different devices and it can be integrated with MATLAB and perform analysis.
- In the case of data frames, we have four data transmission frames that support different use cases.
- In the case of data analysis, we used Python, Numby, and Pandas to perform data analysis on the data retrieved from ThingSpeak. Then we performed data prediction using Python and Pandas utilizing the Softmax function.

VII. Discussion And Future Work

The forecast accuracy [23] was determined by calculating mean absolute percent error first, MAPE, as in (6) which gives us the mean percent error.

$$\text{MAPE} = \frac{\sum_{i=1}^{n} \left| \frac{MV_i - PV_i}{MV_i} \right|}{n} \times 100 \qquad (6)$$

Then the forecast accuracy can be calculated by subtracting MAPE from 100 as in (7).

$$Forecast\ Accuracy = 100 - MAPE \qquad (7)$$

The mean absolute deviation, MAD, can be determined as in (8), and it is helpful to estimate the amounts of deviation of the predicted values from the measured values.

$$MAD = \frac{\sum_{i=1}^{n} |MV_i - PV_i|}{n} \qquad (8)$$

The function was calculated using Python, the MAPE is 0.825%, the forecast accuracy was determined to be 99.175% and the mean absolute deviation is 0.18.

For future work, we intend to upgrade our board to use WIFI with an ARM microcontroller and check the changes in the simple and complicated tasks. We designed a smaller kit with neater copper routs, and intend to use it in the next experiments. We developed a way to update the board wirelessly which is a considerable advantage against most customized IoT nodes. We will study deeper analysis and research a way to decrease prediction time and try to develop our own custom AI model. We can move our analysis to MATLAB and integrate it with Python code to achieve better automatic collaboration.

VIII. Conclusion

The Internet of Things was introduced to connect devices over the Internet and facilitate access to information for users. The IoT-based surveillance and control system model presented in this paper can be used in farms. Sensors play a key role in the data collection of the monitoring system, as they collect data then send them to the hardware board, IoT node, which then sends it to the ThingSpeak cloud using the mobile application or web application. The system then can take action automatically or leave it to the user to take action using the same mobile application or web application that controls the data flow. In addition to sensors, we used actuators to control the devices connected to the system. The board is not only about data collection and monitoring, but it can also perform data analysis using MATLAB or Python. In this paper, we verified our idea using Python and Pandas. We analyzed an excel sheet with more than 25000 rows of data and compared the different properties in different relevant conditions. We used the Softmax function to perform data prediction and verified the accuracy of the predicted data using the data collected by sensors. In the end, we can say that the proposed idea was an acceptable, efficient, and affordable solution for research applications.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Through regular discussions, all authors contributed to the paper. Merihan Essam developed the system and collected the data with Amr El-Awamry. Marwa Elmenyawi performed the data analysis and worked on the prediction part and Adly Tag Eldin reviewed the content of the paper. All authors had approved the final version.

ACKNOWLEDGMENT

We would like to thank the InnoEgypt program for supporting, funding this research, and incubating this idea at Ebni Incubator to be launched on the market soon, by the commercial name Tamra. Furthermore, we are obliged to give our thanks and appreciation to Professor Mahmoud Fathy, professor at the basic science department at Benha faculty of engineering, for his academic support during the research.

REFERENCES

[1] D. D. Koo, J. J. Lee, A. Sebastiani, and J. Kim, "An Internet-of-Things (IoT) system development and implementation for bathroom safety enhancement," in *Procedia Engineering*, 2016, vol. 145.

[2] M. G. Samaila, M. Neto, D. A. B. Fernandes, M. M. Freire, and P. R. M. Inácio, "Challenges of securing Internet of Things devices: A survey," *Secur. Priv.*, vol. 1, no. 2, 2018.

[3] K. K. Patel, S. M. Patel, and P. G. Scholar, "Internet of Things-IoT: Definition, characteristics, architecture, enabling technologies, application & Future challenges," *Int. J. Eng. Sci. Comput.*, vol. 6, no. 5, 2016.

[4] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of things: Vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7. 2012.

[5] D. Glaroudis, A. Iossifides, and P. Chatzimisios, "Survey, comparison and research challenges of IoT application protocols for smart farming," *Comput. Networks*, vol. 168, 2020.

[6] R. Dolci, "IoT solutions for precision farming and food manufacturing: Artificial intelligence applications in digital food," in *Proc International Computer Software and Applications Conference*, 2017, vol. 2.

[7] M. Bhagat, D. Kumar, and D. Kumar, "Role of Internet of Things (IoT) in smart farming: A brief survey," 2019.

[8] A. Moon, J. Kim, J. Zhang, and S. W. Son, "Evaluating fidelity of lossy compression on spatiotemporal data from an IoT enabled smart farm," *Comput. Electron. Agric.*, vol. 154, 2018.

[9] P. Sureephong, P. Wiangnak, and S. Wicha, "The comparison of soil sensors for integrated creation of IoT-based Wetting front detector (WFD) with an efficient irrigation system to support precision farming," 2017.

[10] S. Rajeswari, K. Suthendran, and K. Rajakumar, "A smart agricultural model by integrating IoT, mobile and cloud-based big data analytics," in *Proc. International Conference on Intelligent Computing and Control, I2C2 2017*, 2018, vol. January 2018.

[11] P. Patil and V. Sachapara, "Providing smart agricultural solutions/techniques by using IoT based toolkit," in *Proc. International Conference on Trends in Electronics and Informatics, ICEI 2017*, January 2018.

[12] S. Pooja, D. V. Uday, U. B. Nagesh, and S. G. Talekar, "Application of MQTT protocol for real time weather monitoring and precision farming," in *Proc. International Conference on Electrical, Electronics, Communication Computer Technologies and Optimization Techniques, ICEECCOT 2017*, 2018.

[13] J. Muangprathub, N. Boonnam, S. Kajornkasirat, N. Lekbangpong, A. Wanichsombat, and P. Nillaor, "IoT and agriculture data analysis for smart farm," *Comput. Electron. Agric.*, vol. 156, 2019.

[14] A. Nayyar and V. Puri, "Smart farming: IoT based smart sensors agriculture stick for live temperature and moisture monitoring using Arduino, cloud computing & solar technology," 2017.

[15] X. Shi, *et al.*, "State-of-the-art internet of things in protected agriculture," *Sensors (Switzerland)*, vol. 19, no. 8. 2019.

[16] A. I. Abdul-Rahman and C. A. Graves, "Internet of things application using tethered MSP430 to thingspeak cloud," 2016.

[17] M. A. Brovelli, C. E. Kilsedar, and F. Frassinelli, "Mobile tools for community scientists," *Abstr. ICA*, vol. 1, 2019.

[18] R. Duan, X. Chen, and T. Xing, "A QoS architecture for IOT," 2011.

[19] Y. Zhang, *et al.*, "ICN based Architecture for IoT - requirements and challenges," *ICN Res. Gr.*, 2016.

[20] M. S. Ramanathan. (2020). Data cleansing - remove NULL values with automated python code. [Online]. Available: https://www.theanalyticmind.com/post/data-cleansing-remove-null-values-with-automated-python-code

[21] S. Ghosh, K. Neha, and Y. Praveen Kumar, "Data wrangling using python," *Int. J. Recent Technol. Eng.*, vol. 8, no. 2, 2019.

[22] Q. Zhu, Z. He, T. Zhang, and W. Cui, "Improving classification performance of softmax loss function based on scalable batch-normalization," *Appl. Sci.*, vol. 10, no. 8, 2020.

[23] How do I measure forecast accuracy? (2020). [Online]. Available: https://www.forecastpro.com/2020/08/24/how-do-i-measure-forecast-accuracy/.

Merihan Essam Mohamed Teaching assistant at Science Valley Academy of Engineering and Technology in Obour City. B.Sc., Communications Engineering "June 2014" Grade Excellent with Honor graduation project Design, analysis, and implementation of a planner antenna for wireless applications. Master student at Benha faculty of engineering.

Amr A. Al-Awamery Lecturer in Benha Faculty of Engineering. B.S degree from Benha High Institute of technology (BHIT) 1999 communication department, Msc in Microwave Engineering 2005 from BHIT, finally Ph.D. 2012 in Antenna and Microwave devices from Saint Petersburg Electro technical University

Marwa Abdelrazik Elmenyawi received her Ph.D. degree in electronics and communications engineering from Ain Shams University, Cairo, Egypt. She was appointed as assistant professor in the Electrical Engineering Department of Benha Faculty of Engineering of Benha University. Her research interests and publications span a relatively good spectrum that includes hardware description languages, programming languages, Embedded Systems, IOT, Image processing, Computer Architecture, and artificial intelligence.

Adly S. Tag Eldien received the B.Sc., M.Sc., and Ph.D. degrees from Benha University, Egypt, in 1984, 1989, and 1993, respectively. He was the Ex-Head of the Network and Information Center, Benha University. He is currently serving as a Professor with the Department of Electrical Engineering, Benha University. His research interests include robotics, networks, and mobile communications.