Farm Monitoring System

Jethro P. Moleño[1], Eldrick John A. Aguado[2], Gian Carlo R. Oraa[3], Jocelyn F. Villaverde[4]

School of Electrical, Electronics and Computer Engineering, Mapúa University
Muralla St. Intramuros, Manila 1002, Philippines

[1]jpmoleno@mymail.mapua.edu.ph

[2]ejaaguado@mymail.mapua.edu.ph

[3]gcroraa@mymail.mapua.edu.ph

[4]javillaverde@mymail.mapua.edu.ph

***Abstract*— Nowadays, innovation in the agriculture aspect has increased using various techniques and methods. By utilizing different newer and modern methods in harvesting crops, the productivity of farms has grown. These methods include applying smart sensors and different farm monitoring systems to achieve a sustainable and healthy environment for the crops to grow. In this paper, the researcher proposes a portable system that monitors the plant's relative humidity, water level, temperature, light intensity, and height, which is then analyzed and processed to predict the estimated growth time for the Pechay seedling. The monitoring parameter is then displayed in ThingSpeak and the packaged application created in MATLAB. Based on the results gathered, the predictive model accumulated an acceptable value of RMSE, which is 0.45, and was used for testing. Moreover, the researchers tested the model using the partitioned from the imported dataset, which has a difference of 0.42, indicating that the predictive model used was accurate and feasible.**

***Index Terms*—** *agriculture, MATLAB, monitoring system, sensor, ThingSpeak*

I. Introduction

The world has concluded that technology will drive the pace required to keep up with the food demand of such a vast population. Farm monitoring system solution is that technology—innovative Farming IoT Solutions or Smart Agriculture in short. IoT will play a vital role in this; with the help of sensors and data analytics, farmers will gain priceless data of what the ever-changing climate will bring and have actionable insights. New technologies like automated irrigation & fertilizer schedule will push them towards achieving optimal growth from their farms.

People are looking for ways to start a monitoring system that is easily accessible and portable features that can be used for small-scale farming, which can be done in gardens. The concept of the Internet of things (IoT), introduced in 1999, has been proved to be one of the emerging technologies of recent times that have a slated effect on every industry and sector, including agriculture and farming. IoT-oriented or smart farming is now the future with promising applications and solutions, including farm vehicle tracking, livestock monitoring, storage monitoring, crops/plants monitoring, and more. Smart farming promises greater efficiency, resources, and stock reduction, less human intervention, automation, data-driven processes, increased production, water conservation, real-time data and production insights and accurate farm and field evaluation, and many more. The system's objective is to design a farm monitoring system that daily checks the plant's status (parameters like height, water levels, temperature, humidity, and light intensity). It can predict the number of days needed for seedling transfer based on the parameters acquired by the plant. It also has a feature wherein the user can access the status of the plant through an application and in the ThingSpeak server. Putjaika et al. [1] proposed an intelligent farming (IF) system. IF is the technology that uses loT and intelligent farming concepts to help farmers monitor and sense helpful information from their farms to help improve quality and product quantity. Manjula et al. [3] built architecture for crop yield prediction besides proposing a new methodology that combines the usage of vegetation indices derived from remote sensing images and other attributes. James et al. [5] did a cost-effective system that receives data about the plants' conditions from various sensors in the system. The sensors used are DHT-11, SHT-11, and a 'Height Measuring Apparatus' consisting of magnetic switches on a vertical pipe and a Neodymium magnet piece. The system works based on communication between the farmer's mobile phone and a Raspberry Pi.

II. Methodology

This chapter discusses the development and the process of designing the prototype for the farm monitoring system. The researchers followed specific step-by-step procedures in developing and designing the prototype used by the system. Moreover, the researchers implemented IoT technology for data gathering and MATLAB software for predictive model analytics.

A. Conceptual Framework

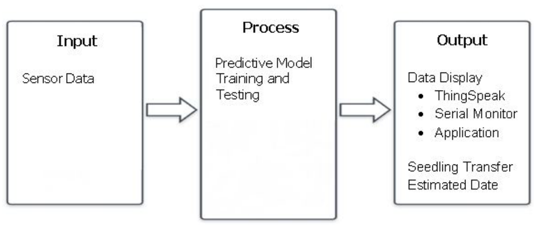

Fig. 1 Conceptual Framework

The conceptual framework above shows the input, process, and output of the device. Different sensors are used to accumulate the data needed for the prediction analysis. The researcher used a photoresistor to acquire light intensity data, a DHT11 temperature sensor for both temperature and humidity sensing, a water level sensor for measuring the plant's moisture through water levels, and an ultrasonic sensor for measuring the plant's height. Moreover, the imported data will be used to determine the appropriate predictive model, determining the estimated number of days for seedling transfer.

B. Block Diagram

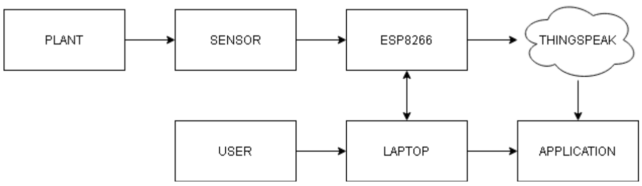

Fig.2 Block Diagram

Fig 2. illustrates the block diagram for the monitoring system. In the figure, the ESP8266 acquires the sensor values from the plant. These values are then transmitted to both the ThingSpeak server and Arduino's serial monitor, which are displayed for the user. Furthermore, the laptop or computer unit can troubleshoot the microcontroller if there is an error in value or the program. Also, the laptop can be used to estimate the number of days using the packaged application created in MATLAB. The application gathers the recent data from the ThingSpeak server and uses it to predict the number of days left before the seedling can be transferred. Moreover, the application can be opened using MATLAB software.

C. Schematic Diagram

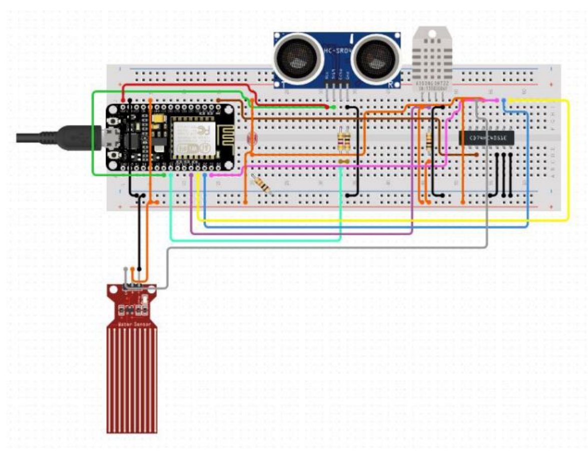

Fig. 3 Schematic Diagram

The schematic is shown in Fig. 3 was made using the circuit.io website. The analog sensors needed a multiplexer (CD74HC4051E) since there is only one analog input. Furthermore, the sensors used are photoresistors for light intensity reading, which measures the amount of resistance when a light source is placed near it. Also, the water level sensor was used as an alternative for soil moisture sensor for water readings in the soil. Although the soil moisture sensor is better for this scenario, the researchers chose the water level sensor to reduce the design cost. As for temperature and relative humidity measurement, the researcher used a DHT11 temperature sensor to read both parameters. As for determining the plant height, image capture was considered. However, the researchers used an HC-SR04 ultrasonic sensor for determining the plant's height to reduce some processing. Some resistors were used in some sensors as part of the circuit.

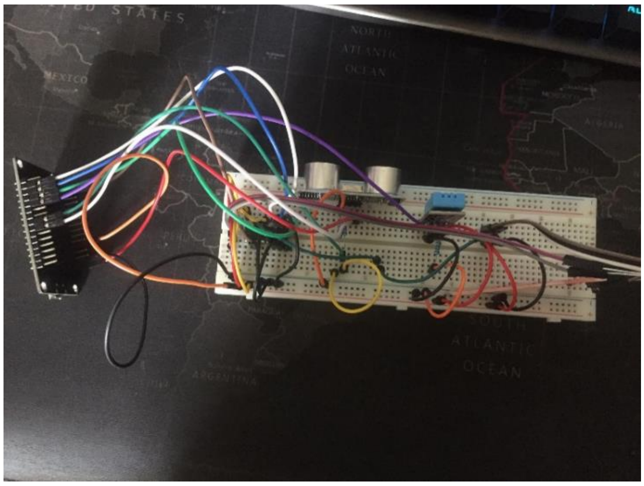

Fig. 4 Breadboard Implementation

After working with the schematic diagram, the following procedure was the breadboard application. As shown in Fig. 4, the breadboard implementation was accomplished using the components and the jumper wires. Compared to the schematic diagram, one of the resistors in the ultrasonic sensor was removed since upon testing the circuit, if it is working, it is not needed. The circuit is still working even though it was removed.

D. Main Flowchart

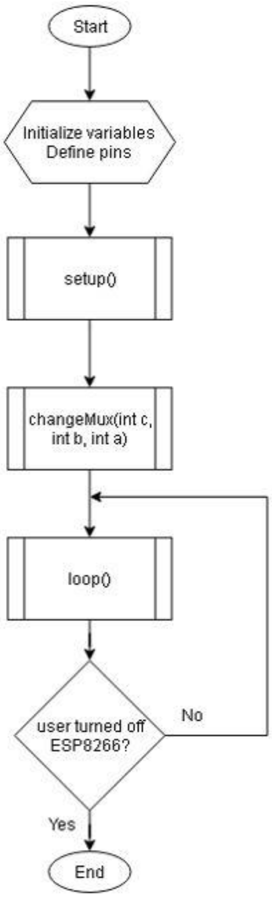

Fig. 5 Main Flowchart

The following procedure was constructing the main flow of the program for the circuit. Fig. 5 illustrates the main flowchart of the system of circuit. Firstly, the program initializes the variables and pins for the sensors in the ESP8266 microcontroller. Then the program ensures that it establishes the connection with the Internet to send the data to ThingSpeak, which is in the setup function. The program proceeds with the changeMux function, which switches the inputs of the multiplexer to accumulate data to both analog sensors, the water level, and the photoresistor. Subsequently, the loop function wherein the data gathering is done will be executed. After gathering the data, it will be sent to the ThingSpeak every 30 seconds. This function will be executed until the user turns off the microcontroller, and the program will end.

E. Prototype Design

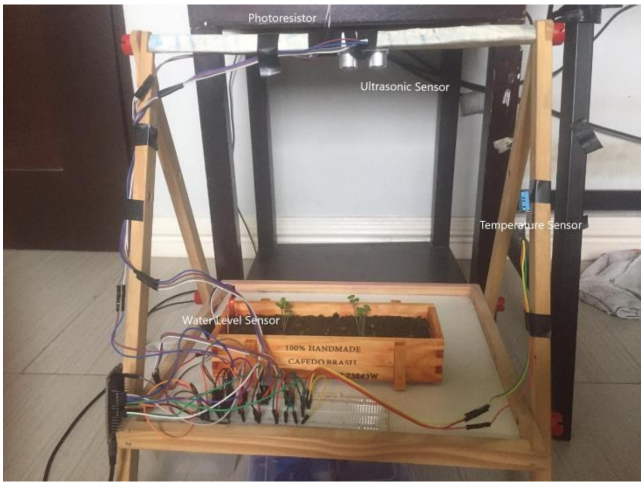

Fig. 6 Prototype Implementation

The figure above shows the implementation of the design prototype. The researchers placed a plant box where the Pechay plant grows, and both ultrasonic sensor and photoresistor were placed above. The researchers ensured the ultrasonic sensor was placed directly above the plant to determine the height. Initially, the ultrasonic sensor was tested to determine the initial height reading (in cm) wherein which will be subtracted to the height reading when the plant grows. On the other hand, the water level sensor was placed in the plant box. The researchers buried the water level sensor to the ground until it reached the bottom part of the box. The temperature and humidity reading was placed where the breeze came for the temperature and humidity reading to be accurate. Finalizing the prototype, the researchers used wood frames for the foundation of the prototype.

F. Training Model Using Predictive Analytics

The following procedure is to determine the predictive model which fits the system. Firstly, the data from ThingSpeak was exported in a CSV file. This raw data was pre-processed and arranged using MATLAB software. Furthermore, the data was partitioned using the cv partition, as shown in Fig. 8. The data set accumulated in 14 days with an estimated 23,000 entries was partitioned into two structures: 70% for training and 30% for testing. After that, the processed data was imported into the regression learner in MATLAB in a new session. Then the imported data was trained using cross-validation, which avoids overfitting to reduce errors.

Fig. 7 Data Plot from ThingSpeak

```
>> cv = cvpartition(feeds9.field6,"Holdout",0.3);
Warning: Ignoring rows in GROUP with missing values.
> In internal.stats.cvpartitionInMemoryImpl (line 141)
In cvpartition (line 175)
>> dataTrain = feeds9(cv.training,:);
>> dataTest = feeds9(cv.test,:);
```

Fig. 8 Cv Partition Command

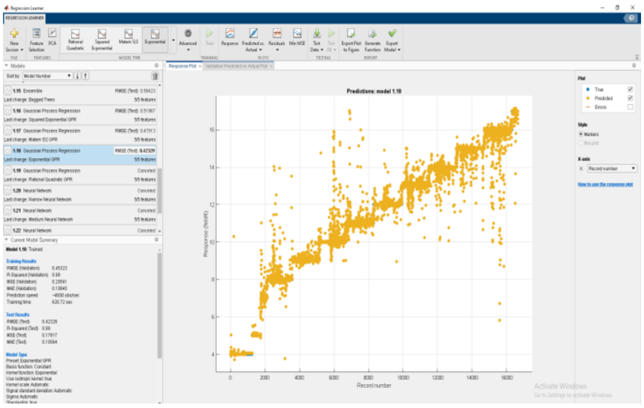

Fig. 9 Screenshot of Predicted Values vs. Actual Values in Model Training

The imported data was trained with all the models available in MATLAB, and the regression app was executed. The researchers found out that the Exponential Gaussian Process Regression is the best predictive model for our system since it has the lowest RMSE (Root mean squared error), 0.45, as shown in Fig. 9. The determined predictive model was then exported using the application, and the struct was created.

G. Application Development

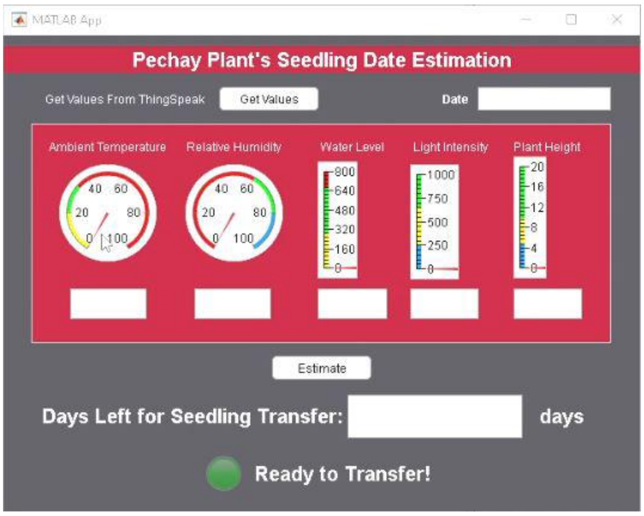

Fig. 10 Farm Monitoring System Application

The application development, made in MATLAB software, was designed using the app designer shown in Fig. 10. Firstly, the design layout was made using the components available by the designer. Display gauges were included to visualize the values clearly, and data are displayed below. After designing the application, the code was implemented. The data consisting of ambient temperature, relative humidity, water level, light intensity, the height of the plant, and the date from ThingSpeak was acquired using the Read ThingSpeak command. However, it only reads the recent data from ThingSpeak. Then the function created for predicting the model was used to predict the number of days the plant has grown. This value will be subtracted to the estimated number of days before the seedling can be transferred, around fifteen days.

III. Results and Discussion

1. Testing Description

The researchers use MATLAB software in the testing process for the monitoring system. The trained model was used to test the remaining 30% of the data partitioned from the initial dataset. The test data was simulated using the test data option in the regression app estimateSeedDate function, which gets the sensor values and outputs the estimated date using the trained model. Also, the researchers tested the package application for further testing.

2. Testing Results

The details regarding the test results statistics and the deployed model are shown in the figure. The researchers can assess that based on the parameters shown, and it can be used as a model for the system. The test results consist of statistical values like RMSE, R-Squared, MSE, and MAE. The testing results showed an RMSE value of 0.42, meaning the difference between actual and predicted values are minimal. Also, the RMSE value is closer to zero, and the R-Squared Value is close to 1, which means that the model is accurate.

The validation plot was constructed upon testing the data in the regression app, as shown in Fig. 11. Moreover, the figure showed that most of the responses are near the perfect prediction line. The plot indicates that the trained model is feasible and fits the system.

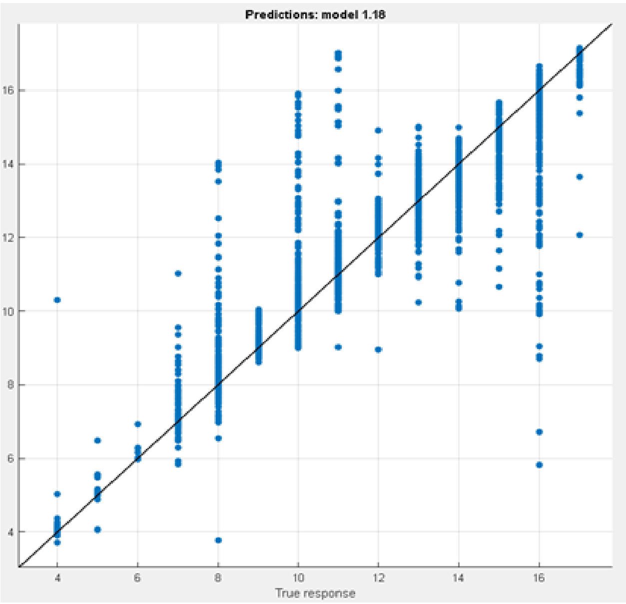

Fig.11 Test Data Validation Plot

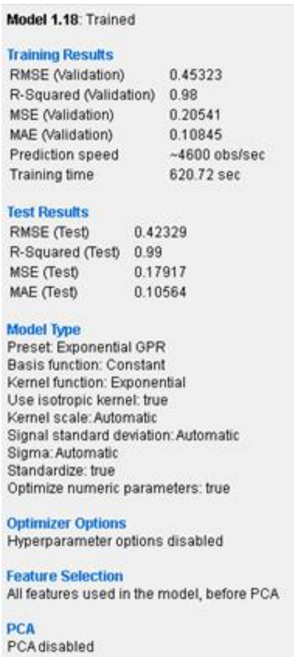

Fig.12 Test Results

Another testing was conducted using the struct model from the estimateSeedDate function. Based on Table 1, the test data showed the date and time recorded, entry id, sensor values, and the estimated days (both predictive and actual value) of the recent data acquired. Also, the units of measurement are shown in each sensor value. However, there are no included units in the data gathered at the water level sensor and photoresistor since it is a raw value from the sensors and is therefore relative. Based on the results gathered, the predicted values are less than one and are near the actual value, indicating that the predicted model is feasible and accurate.

TABLE 1
Recent Data Tested using the Predictive Model

Ambient Temp (in °C)	Relative humidity (in % RH)	Water Level	Light Intensity	Plant Height (in cm)	Actual Value (in days)	Predicted Value (in days)
28	80	496	72	7	17	16.9928
28	80	490	72	7	17	16.9999
28	80	496	72	7	17	16.9999
28	80	496	72	7	17	16.9999
28	80	496	72	7	17	16.9999
28	80	490	66	7	17	16.9999
28	80	496	74	7	17	16.9999
28	80	490	66	7	17	16.9999
28	80	497	74	7	17	16.9999
28	80	497	74	7	17	16.9989
28	80	497	74	7	17	16.9928

Lastly, the recent data was tested by running the packaged application in MATLAB. As shown in Fig. 11, the estimated number of days left before the seedling can be transferred is 0 days since the predicted date from the function is subtracted to 15. Also, it can be observed that the green lamp below is glowing, which indicates that the seedlings can be transferred into a new pot.

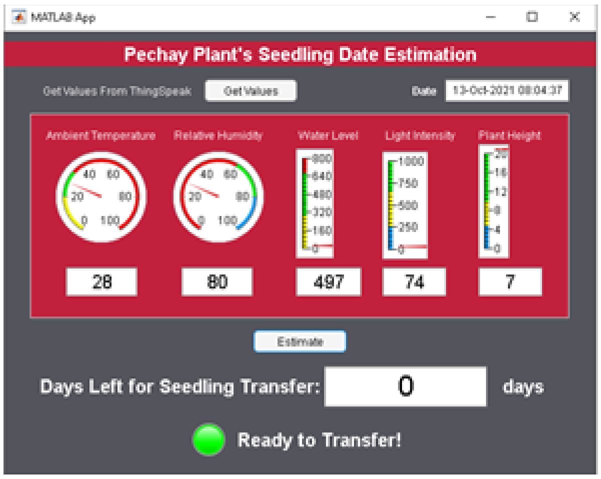

Fig.11 Screenshot of Recent Data from the Application

IV. CONCLUSION AND FUTURE WORKS

The farm monitoring system, designed using ESP8266 module and MATLAB software, executed its functionality well. Moreover, the system successfully gathered data through the sensors and displayed these values in the ThingSpeak server. Based on the results, the system accurately predicted the number of days left before seedling transfer using the predicted model, which has an RMSE of 0.45. The predicted number of days left was shown in the package application created using MATLAB's app designer and the estimatedSeedDate function that the researchers executed.

The research promotes urban farming, which will contribute to some of the food shortages in the country. Also, the paper encourages the community to grow their food since the system will guide these users to monitor and check their plant's health. Further innovations in the system include improving the predictive model to provide more accurate results, optimizing the model until the complete harvest of the plant, applying different plants to grow and test as part of data gathering, and adding automation to the system to maintain the condition of the plant.

REFERENCES

[1] Narayut Putjaikal, Sasimanee Phusael, Anupong ChenIml,Dr.Phond Phunchongharnl, and Dr.Khajonpong Akkarajitsakup, A Control System in an Intelligent Farming by using Arduino Technology, Fifth ICT international Student Project Conference (ICT-ISPC), 2016

[2] Ms. G. Rekha M.E1, S. Muthu Selvi, Arduino Interface with Smart Farming System, International Journal Of Engineering And Computer Science ISSN:2319- 7242,2017, Page No. 20521-20526

[3] Aakunuri Manjula; G. Narsimha, XCYPF: A flexible and extensible framework for agricultural Crop Yield Prediction, IEEE 9th International Conference on Intelligent Systems and Control (ISCO), 2015, Pages: 1 – 5

[4] Grossi, Marco; Lakhiar, Imran Ali; Jianmin, Gao; Syed, Tabinda Naz; Chandio, Farman Ali; Buttar, Noman Ali; Qureshi, Waqar Ahmed, Monitoring and Control Systems in Agriculture Using Intelligent Sensor Techniques: A Review of the Aeroponic System; 1687-725X; https://doi.org/10.1155/2018/8672769.

[5] J. James and M. P. Maheshwar, "Plant growth monitoring system, with dynamic user-interface," in 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), pp. 1–5, Agra, India, December 2016

[6] J. P. P. Banjao, K. S. Villafuerte and J. F. Villaverde, "Development of Cloud-Based Monitoring of Abiotic Factors in Aquaponics using ESP32 and Internet of Things," 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 2020, pp. 1-6, DOI: 10.1109/HNICEM51456.2020.9400083.