Penerbit Akademia Baru

Journal of Advanced Research in Applied Mechanics

Journal homepage: www.akademiabaru.com/aram.html
ISSN: 2289-7895

Monitoring and Controlling Water Pumping System Using IoT for Agriculture Purpose

Open Access

Mohamad Syafiq Mohd Rafi[1,*], Nur Irwany Ahmad[1], Diyya Hidayah Abd Rahman[1], Nazrul Azril Nazlan[2], Wan Azani Mustafa[1], Habibah Mokhtaruddin[1], Nurhafiza Azizan[1], Norhanisa Kimpol[1]

[1] Department of Electrical Engineering Technology, Faculty of Engineering Technology, Universiti Malaysia Perlis (UniMAP), Perlis, Malaysia
[2] Scenic Drive Sdn Bhd, Bandar Darul Aman, Jitra Kedah, Malaysia

ABSTRACT

The increasing demand for food, both in terms of quantity and quality, has raised the need for intensification and industrialization of the agricultural sector. The amount of worker may lead to poor productivity and quality of agricultural product. Water pumping systems are mostly used by farmers to channel enough water to the crops to maintain the soil moisture. Conventional water pumping system is controlled by manually by farmers. Farmers are hard or even unable to monitor and provide the required conditions for crops at certain time such as during the night or emergency cases. Based on the limitations that have been highlighted previously, this project aims to monitoring and controlling water pumping system using IoT system. This project used Arduino software to program the NodeMCU board to link the board with the cloud system. Then ThingSpeak Cloud is used for data collection and monitoring. The project is used to monitor the temperature, humidity and soil moisture. The Android application is created based on the developer projects. This project used MIT Apps Inventor to create the Android application. Besides, this system also control ‘ON’ and ‘OFF’ water pumping system via smartphone using android interface at a certain range. The performance of the system is analyzed based on specificity, sensitivity and accuracy. The value for specificity is 95%, value for sensitivity is 100% and value for accuracy is 96.67%.

Keywords:
Agriculture, Internet of Things, Cloud, MIT

1. Introduction

The increasing demand for food, both in terms of quantity and quality, has raised the need for intensification and industrialization of the agricultural sector. In order to improve the quality and quantity of products, a never-ending process of technology is implemented and it will keep growing. When designing a product, it is a huge contribution to societies by implementing the latest technology where it can bring a lot of benefits to others. There are different ways to automate the system as it focuses on using smart phone as an automation system interface for this project. Nowadays, most people use a smart phone as it largely fulfills the needs of their users and makes life easier and better. Thus, this project was created in order to demonstrate an agriculture system that allow farmer to monitor and control water pumping system wirelessly via smart phone. This project was aimed to increase farmer convenience as well as enhancing time efficiency.

* *Corresponding author.*
E-mail address: Mohamad Syafiq Mohd Rafi (mr.syafiqrafi95@gmail.com)

This project was focusing on design an agriculture system by using NodeMCU as a controller while smart phones as a switch which it can be control loads from an assured range. The sensors were connected to the controller through relay and smart phone will send the signal to the controller via Wi-Fi module connectivity to “ON” and “OFF” the water pump. An android application and coding for sensors and water pump were developed to automate this system. This system can be applied for any kind of agriculture purpose not only limited for small farm.

2. Problem Formulation

2.1 Performance Evaluation

Two different types of evaluation measures have been used to evaluate the performance of the proposed methods where it is used to assess the classification. The main reason for choosing these measures is that they are standardized, and they will enable the data taken to compare the proposed algorithms with other state of the systems. The classification is implemented on the time taken for water pump to ON and OFF.

2.1.1 Confusion matrix

In the field of artificial intelligence, confusion matrix is a square matrix which contains information about actual and predicted classifications obtained by a classification model. In this matrix, the numbers on the diagonal represent the correct classifications, whereas the off-diagonal numbers correspond to misclassifications. In the real sense of the word, confusion matrix is used to evaluate the performance of the classification and visualize errors for a given category where it also used to compute the accuracy of the classification[1]. Table 1 shows the example of confusion matrix with correctly and incorrectly classified categories for 10 samples where (+) denotes correctly classified and (-) denotes incorrectly classified[2].

Table 1
Example of confusion matrix for 10 samples[2]

		Actual		
	Categories	A	B	C
Predicted	A	6(+)	1(-)	3(-)
	B	0(-)	8(+)	2(-)
	C	1(-)	2(-)	7(+)

2.1.2 Sensitivity and specificity

Sensitivity and specificity are important statistical measures of the performance classification system. These measures have been widely used in medicine, radiology, psychology and other areas for many decades with a good empirical performance. Recently, these measures have become standard tools among machine learning and pattern recognition communities to evaluate the classification system.

The measures are described by assuming that a confusion matrix for a binary classifier as shown in Table 2. In this table, the value is determined and confirmed by the experiment which usually based on human expert refers as the actual value. In contrast, the predicted value represents a value that is obtained by a classifier model[3].

Table 2
Typical Confusion Matrix for a Binary Classifier[3]

		Actual	
		Positives	Negatives
Predicted Value	Positives	True Positive (TP)	False Positive (FP)
	Negatives	False Negative (FN)	True Negative (TN)

The confusion matrix for a binary classifier contains four parts as follows:

$$\textbf{Sensitivity} = \text{TPR} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \text{ x } 100\% \quad (1)$$

$$\textbf{Specificity} = \text{TNR} = \frac{\text{TN}}{\text{TN+FP}} \text{ x } 100\% \quad (2)$$

$$\textbf{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \text{ x } 100\% \quad (3)$$

TP: The actual value is positive, and the classifier predicted it as positive.
FN: The actual value is positive, but the classifier incorrectly predicted it as negative.
FP: The actual value is negative, however the classifier incorrectly predicted it as positive.
TN: The actual value is negative, and the classifier predicted it as negative.

3. Methodology

3.1 Software Configuration

This part covered on developing Android application, ThingSpeak and TinkerCAD.

3.1.1 Developing android application

MIT App Inventor is a mobile developer app for Android that using drag and drop visual programming tool for building and designing an app. This app provided a new method for developers to design, create, and use personally meaningful mobile technology solutions in endlessly unique situations. This app focused on the logic programming to develop an Android application rather than using the syntax C# for the coding language. Thus, it is easy to understand and troubleshooting for developer since it used a logic block diagram connection for its coding language.

The android application for smart phone is developed by using MIT Apps Inventor is an open source web application that originally provided by Google. This application is divided into two parts which are designer editor and block editor. The first part is a designer part where it is contain an elements such as buttons, labels, list pickers, images and many more and it is coupling with the mobile device features such as NFC, Bluetooth, GPS and others for developer to design the apps. Figure 1 shows the designer editor part in MIT Apps Inventor. Figure 2 shows the block editor part for MIT application start up page.

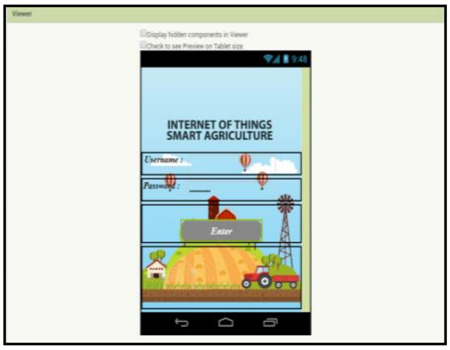


Fig. 1. Designer Editor Part in MIT Apps Inventor

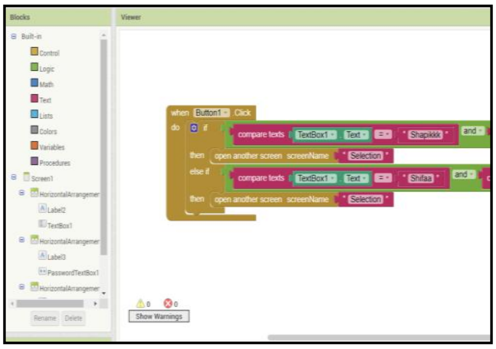
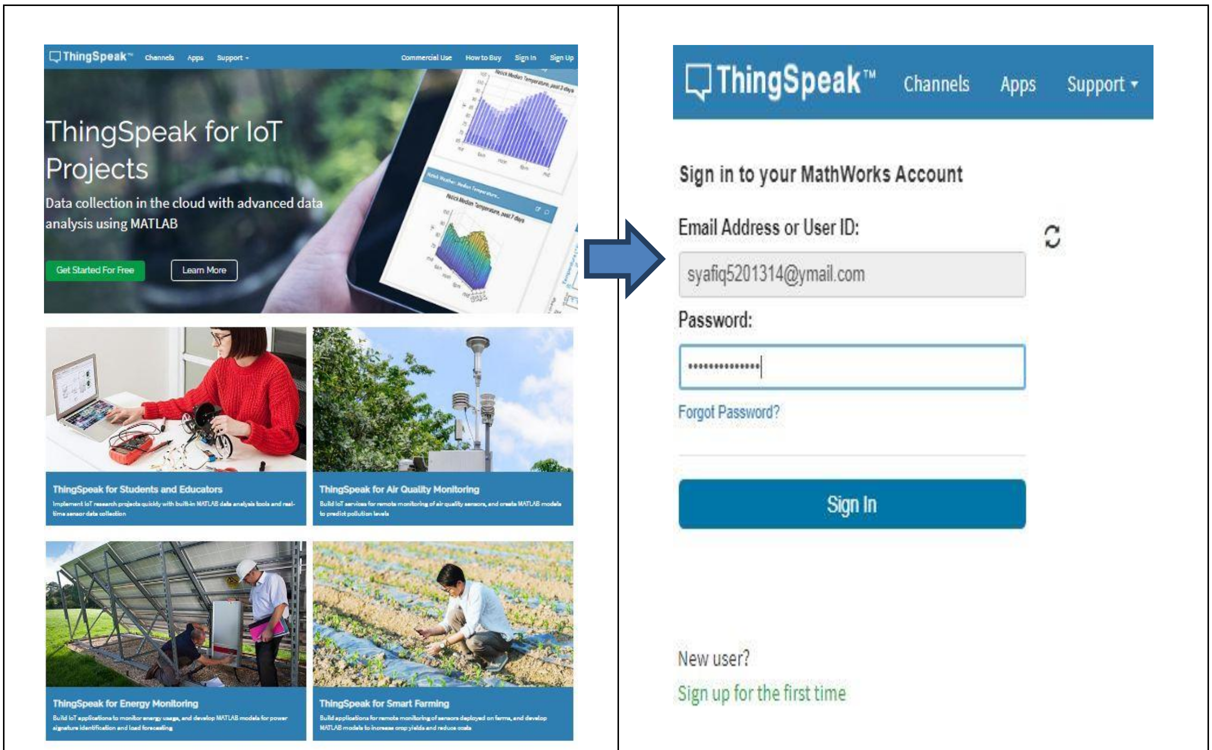


Fig. 2. Block Editor Part in MIT Apps Inventor

3.1.2 ThingSpeak

In this part, all the data from sensors is sent to the cloud through Wi-Fi. The cloud that has been used in this project is ThingSpeak. ThingSpeak is an IoT analytic platform service from MathWorks, the maker of MATLAB and Simulink. ThingSpeak provides instant visualizations of data from sensor and shown in line graph. Figure 3 shows the visualizations of data in ThingSpeak Cloud.



1st interface 2nd interface

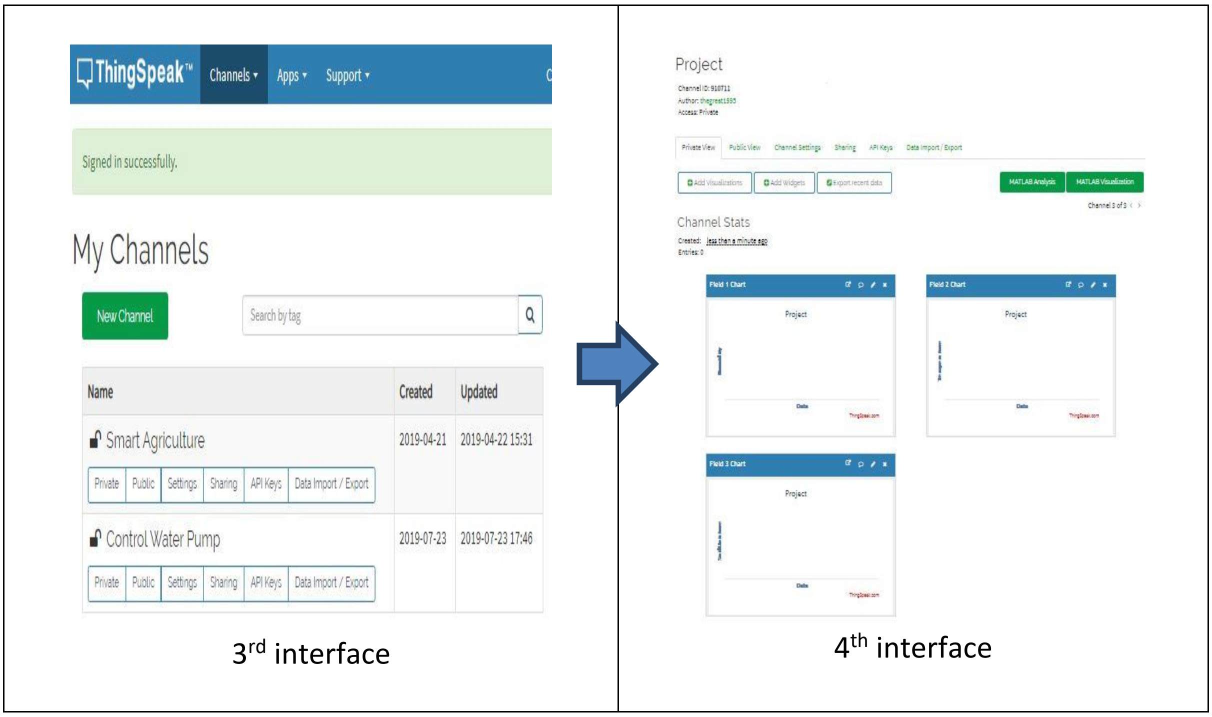


Fig. 3. Interface and Visualization of Data in ThingSpeak Cloud

3.1.3 TinkerCAD

TinkerCAD software is used for designing this project. Figure 4 shows four different views of the project design from TinkerCAD. The design of the prototype is presented in 3D view.

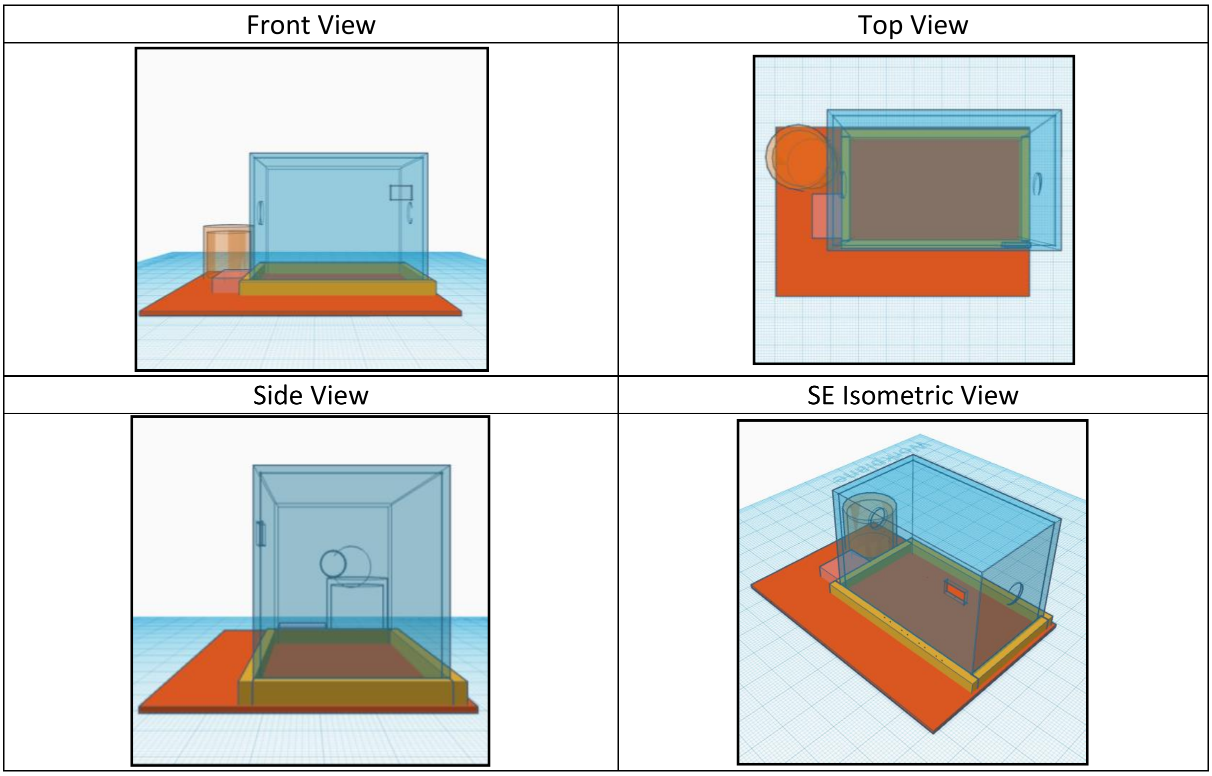


Fig. 4. Four Different Views of Project in TinkerCAD

3.1.2 Hardware construction
3.1.2.1 Monitoring and controlling water pumping system end product

Figure 5 shows the monitoring and controlling water pumping system end product in four different perspectives.

Front View	Top View
	
Side View	SE Isometric View
	

Fig. 5. Four Different Perspective of Project End Product

4. Result and Discussion
4.1 Data from ThingSpeak

The experiment was conduct to test the sensor performance with commercialized product. The value of temperature and humidity from this project is compared with digital hygrometer. The data collected for every 30 minutes from 8.00 am until 5.30 pm per day. In order to get an accurate value of sensor data, the data is collected for every 30 minutes because NodeMCU will send data to

ThingSpeak cloud for every 15 seconds. The data collected random for 7 days from 23 October 2019 to 5 November 2019.

4.1.1 Assessment 1: Day 1

From the data collected, line graph is used to show the sensor data against time. Table 3 shows the graph of sensor data against time. Table 4 shows the percentage error between IoT result and digital hygrometer on day 1 that recorded on 23 October 2019. From the data, average percentage error of temperature is approximately 0.82% while percentage error of humidity is 4.8%.

Table 3
Graph of Sensor Data against Time

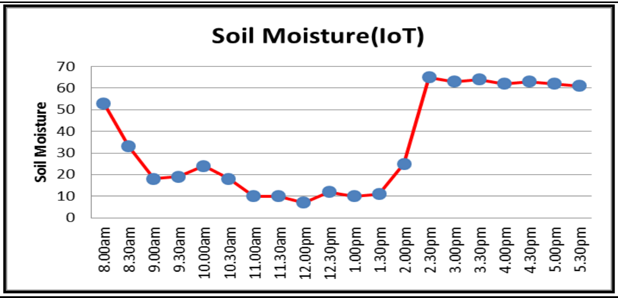
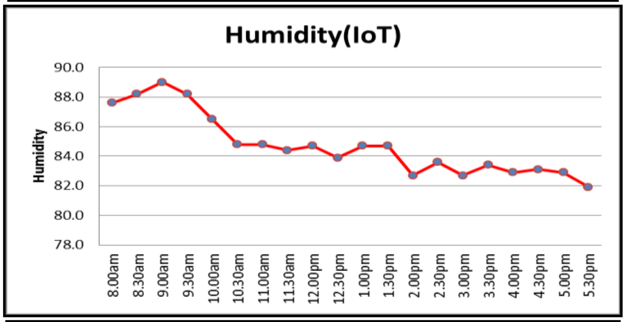
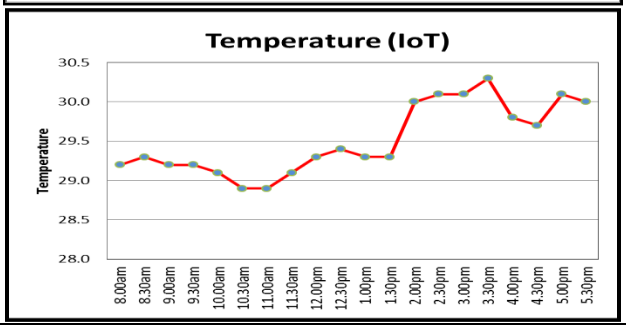
Sensor	Line Graph
Soil Moisture	
Humidity	
Temperature	

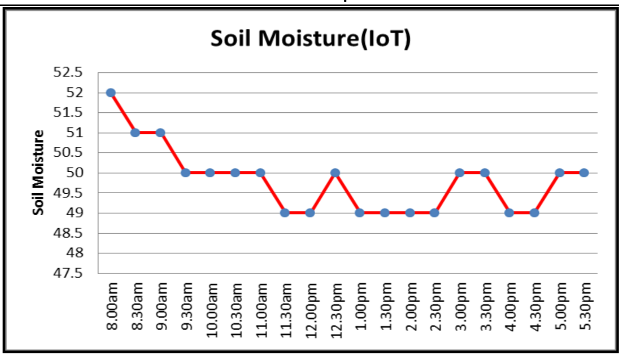
Table 4
Percentage Error of Day 1

Time	Percentage Error of Temperature (%)	Percentage Error of Humidity (%)
8.00am	0.68	4.1
8.30am	1.02	3.6
9.00am	0.68	3.4
9.30am	0.68	3.6
10.00am	0.69	4.0
10.30am	2.08	4.5
11.00am	2.08	4.5
11.30am	1.37	5.2
12.00pm	1.02	5.5
12.30pm	1.36	4.6
1.00pm	1.02	5.5
1.30pm	1.02	5.5
2.00pm	1.00	5.7
2.30pm	0.33	4.3
3.00pm	0.33	5.7
3.30pm	0.99	5.3
4.00pm	0.67	5.9
4.30pm	0.67	4.9
5.00pm	0.33	5.9
5.30pm	1.00	4.8
Average	0.82	4.8

4.1.2 Assessment 1: Day 2

Table 5 shows the graph of sensor data against time. Table 6 shows the percentage error between IoT result and digital hygrometer on day 2 recorded on 24 October 2019. From the data, average percentage of error of temperature is at 1.53%. However, the average percentage of error of humidity is at 2.9%.

Table 5
Graph of Sensor Data against Time

Sensor	Line Graph
Soil Moisture	

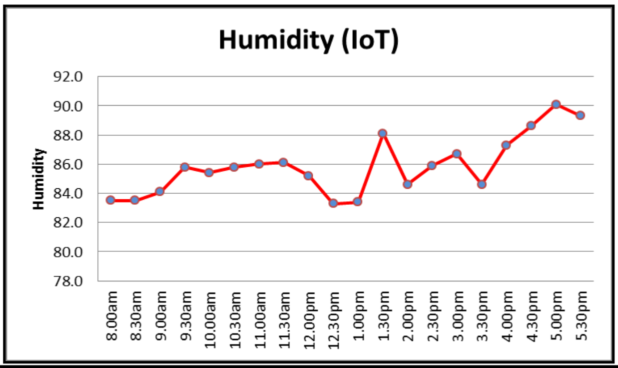
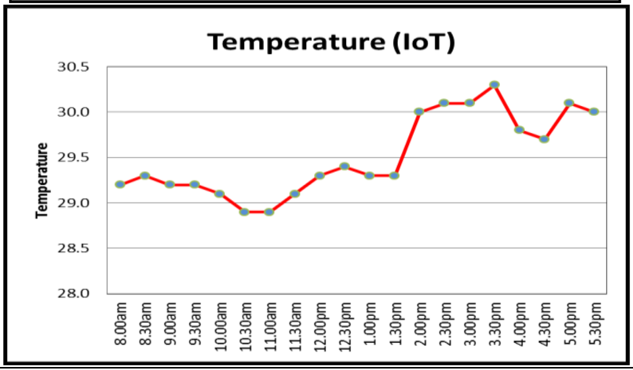
Humidity	
Temperature	

Table 6
Percentage Error of Day 2

Time	Percentage Error of Temperature (%)	Percentage Error of Humidity (%)
8.00am	2.78	1.8
8.30am	2.77	1.8
9.00am	2.77	2.5
9.30am	2.75	3.3
10.00am	1.03	2.8
10.30am	1.02	3.3
11.00am	1.70	3.5
11.30am	1.35	3.6
12.00pm	1.36	2.6
12.30pm	1.01	1.6
1.00pm	1.01	1.7
1.30pm	1.02	3.5
2.00pm	1.71	1.9
2.30pm	0.70	3.4
3.00pm	1.06	3.1
3.30pm	1.06	4.3
4.00pm	1.76	2.6
4.30pm	0.74	4.1
5.00pm	1.49	3.4
5.30pm	1.48	3.7
Average	1.53	2.9

4.1.3 Assessment 1: Day 3

Table 7 shows the graph of sensor data against time. Table 8 shows the percentage error between IoT result and digital hygrometer on day 3 recorded on 29 October 2019. From the data, average percentage of temperature is at 0.92% while average percentage of humidity is at 1.2%.

Table 7
Graph of Sensor Data against Time

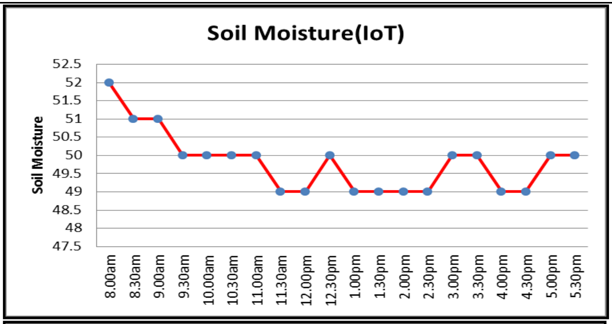
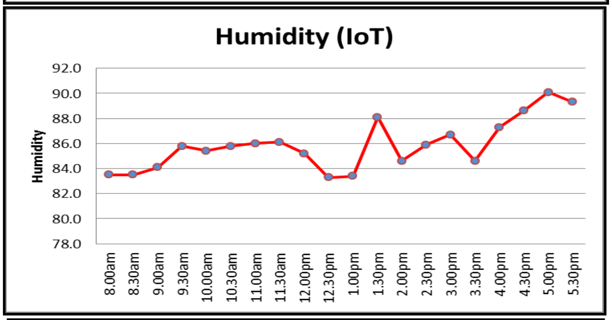
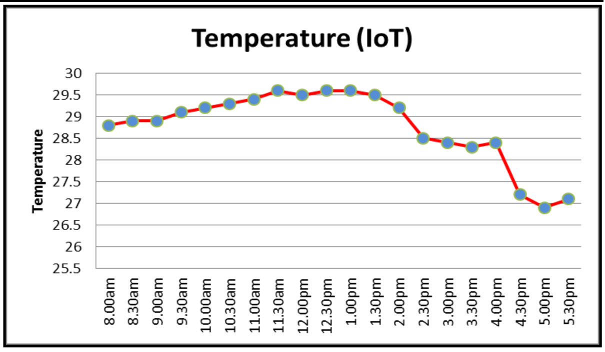
Sensor	Line Graph
Soil Moisture	
Humidity	
Temperature	

Table 8
Percentage Error Day 3

Time	Percentage Error of Temperature (%)	Percentage Error of Humidity (%)
8.00am	0.71	2.2
8.30am	0.71	1.2
9.00am	0.36	1.3
9.30am	0.71	1.2
10.00am	0.71	0.2
10.30am	0.70	0.8
11.00am	1.05	1.5
11.30am	1.72	1.5
12.00pm	1.02	1.4
12.30pm	1.02	2.2
1.00pm	1.68	0.7
1.30pm	1.36	1.4
2.00pm	1.07	0.6
2.30pm	1.08	0.7
3.00pm	1.08	1.2
3.30pm	0.36	1.2
4.00pm	0.36	1.6
4.30pm	0.35	1.7
5.00pm	1.05	0.8
5.30pm	1.40	1.0
Average	0.92	1.2

4.2 Summary of Sensor Data from IoT and Digital Hygrometer

Generally, errors are classified into three types which are systematic errors, random errors and blunders errors. This system may have systematic errors in which errors may occur as the sensors maybe poorly calibrated before taking result. The sensors itself has 1% to 5% of tolerance. The digital hygrometer also has its tolerance of 5%. To sum up, this project efficiency is accepted as the sensors and digital hygrometer has its tolerance. Table 9 shows the summary average of percentage error of humidity and temperature. Figure 6 shows the bar graph of average of percentage error from Day 1 until Day 7.

Table 9
Summary of Average of Percentage Error

Date	Average percentage error temperature (%)	Average percentage error humidity (%)
23 October 2019	0.82	4.8
24 October 2019	1.53	2.9
29 October 2019	0.92	1.2
30 October 2019	0.96	1.4
2 November 2019	0.66	2.5
4 November 2019	0.82	3.0
5 November 2019	0.50	3.2

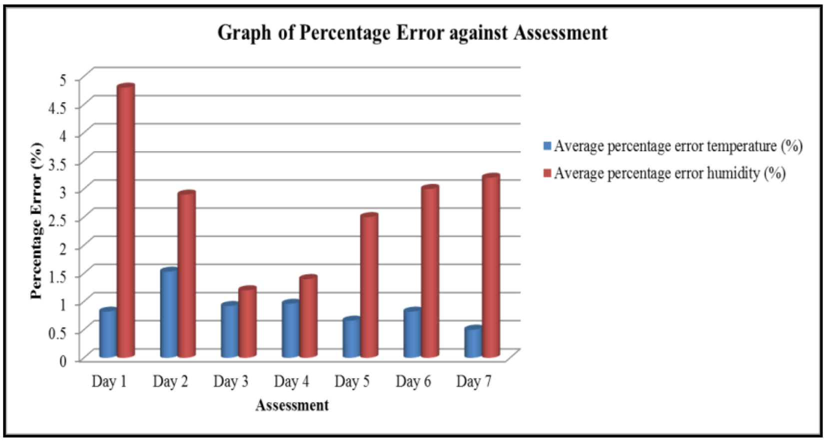


Fig. 6. Graph of Percentage Error against Assessment

4.3 Performance of Controlling Water Pumping System based on Accuracy, Sensitivity and Specificity

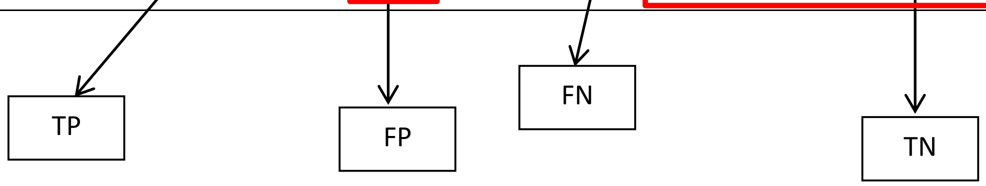
The experiment was conducted to test the performance of water pumping system. In order to evaluate the accuracy, sensitivity and specificity of the product performance, a confusion matrix approach was applied. The range selected to test product performance are 20 meters, 40 meters and 50 meters respectively. The range selected based on the distance of Wi-Fi modem from NodeMCU to test the Wi-Fi strength based on different range. The water pumping system will be tested on and off for 10 times repetitively for each range. Table 10 shows the result of the performance of water pumping system. Then, a confusion matrix for binary classifier is used to determine the product performance by calculating the collected data in Table 11.

Table 10
Confusion Matrix

Repetitive	Distance (meter)		
	20	40	50
1	✓	✓	✓
2	✓	✓	✓
3	✓	✓	✓
4	✓	✓	×
5	✓	✓	✓
6	✓	✓	×
7	✓	✓	✓
8	✓	✓	✓
9	✓	✓	×
10	✓	✓	✓

Table 11
Confusion Matrix for a Binary Classifier

	Predict		
Actual	**20**	**40**	**50**
20	10	0	0
40	0	10	0
50	1	2	7



From the result above, confusion matrix for a binary classifier the data can be extract and calculated to get the sensitivity, specificity and accuracy value of the system. The value for true positive, TP is 10 and false negative, FN is 0. Then, the value for true negative, TN is the sum of 10, 0, 2, and 7, while false positive, FP is equal to 1. The result of sensitivity, specificity and accuracy are then recorded in Table12. Based on the result, the value of sensitivity, specificity and accuracy are 100%, 95% and 96.67% respectively. The accuracy of the system is 96.67% as it shows an excellence result for Wi-Fi module accuracy in transmitting and receiving signal between NodeMCU controller and Android application smartphone.

Table 12
Result of Sensitivity, Specificity and Accuracy

Description	Specificity(%)	Sensitivity(%)	Accuracy(%)
Result	95	100	96.67

5. Conclusion

The objectives of this project are achieved where firstly to develop a system for monitoring temperature, humidity and soil moisture using ThingSpeak. The result from ThingSpeak is presented in line graph. Next, develop a system controlling water pumping system using Maasachusetts Institute of Technology (MIT) Apps Inventor. An application is created by using MIT for monitoring and controlling water pumping system. The sensor reading of each sensor will be displayed inside the Apps. Then, the performance of water pumping system using IoT for agriculture purpose is analyzed based on specificity, sensitivity and accuracy. The Wi-Fi module able to receive and transmit signal between NodeMCU and smartphone to switch on and off the load within 50 meters range. The application is named as Smart Agriculture System. All the objectives achieved.

References

[1] J. Brownlee, "What is a Confusion Matrix in Machine Learning," *Machinelearningmastery.Com*, 2016.
[2] K. M. Ting, "Confusion Matrix," in *Encyclopedia of Machine Learning and Data Mining*, 2017.
[3] Ohsaki, Miho, Peng Wang, Kenji Matsuda, Shigeru Katagiri, Hideyuki Watanabe, and Anca Ralescu. "Confusion-matrix-based kernel logistic regression for imbalanced data classification." *IEEE Transactions on Knowledge and Data Engineering* 29, no. 9 (2017): 1806-1819.
[4] Prathibha, S. R., Anupama Hongal, and M. P. Jyothi. "IoT based monitoring system in smart agriculture." In *2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, pp. 81-84. IEEE, 2017.
[5] Sreekantha, D. K., and A. M. Kavya. "Agricultural crop monitoring using IOT-a study." In *2017 11th International Conference on Intelligent Systems and Control (ISCO)*, pp. 134-139. IEEE, 2017.
[6] Hasan, Siti Zuhaili, and Sarah Aziz. "Kedah Water Resources Enactment 2008 for Sustainable Agriculture Development." *Pertanika Journal of Tropical Agricultural Science* 41, no. 1 (2018).
[7] K. M. Fotouhi, "Microcontroller controlled walking robot," *Comput. Educ. J.*, 2016.
[8] D. Dhuri, A. Gaushal, R. Sawant, T. Renose, Mrs. Mansi, "Speaking Hand using Raspberry Pi 3," *Int. J. Trend Sci. Res. Dev.*, 2018.
[9] Pawar, Prajakta A. "Heart rate monitoring system using IR base sensor & Arduino Uno." In *2014 Conference on IT in Business, Industry and Government (CSIBIG)*, pp. 1-3. IEEE, 2014.
[10] M. Wolfson and D. Felker, "Developing with Android Studio," *Andriod Dev. Tools Essent. Android Stud. to Zipalign*, 2015.
[11] Dickson, Jared. "Xamarin Mobile Development." (2013).
[12] DiMarzio, Jerome. *Beginning Android Programming with Android Studio*. John Wiley & Sons, 2016.
[13] Android Studio, "Android Studio," *Android Studio :The Official IDE for Android*. 2016.
[14] HuiRu Shih, P. E., Jacqueline M. Jackson, Cassandra L. Hawkins Wilson, and Pao-Chiang Yuan. "Using MIT app inventor in an emergency management course to promote computational thinking." *age* 24 (2014): 1.
[15] Ortega García, Antonio, Antonio Ruiz-Martínez, and Rafael Valencia-García. "Using App Inventor for creating apps to support m-learning experiences: A case study." *Computer Applications in Engineering Education* 26, no. 3 (2018): 431-448.
[16] Raut, R., H. Varma, C. Mulla, and Vijaya Rahul Pawar. "Soil Monitoring, Fertigation, and Irrigation System Using IoT for Agricultural Application." In *Intelligent Communication and Computational Technologies*, pp. 67-73. Springer, Singapore, 2018.
[17] Rau, Amogh Jayaraj, Jairam Sankar, Ashok R. Mohan, Deepti Das Krishna, and Jimson Mathew. "IoT based smart irrigation system and nutrient detection with disease analysis." In *2017 IEEE Region 10 Symposium (TENSYMP)*, pp. 1-4. IEEE, 2017.
[18] Baranwal, Tanmay, and Pushpendra Kumar Pateriya. "Development of IoT based smart security and monitoring devices for agriculture." In *2016 6th International Conference-Cloud System and Big Data Engineering (Confluence)*,

pp. 597-602. IEEE, 2016.

[19] Khan, Shuraia. "Wireless sensor network based water well management system for precision agriculture." In *2016 26th International Telecommunication Networks and Applications Conference (ITNAC)*, pp. 44-46. IEEE, 2016.

[20] Kamilaris, Andreas, and Andreas Pitsillides. "Mobile phone computing and the internet of things: A survey." *IEEE Internet of Things Journal* 3, no. 6 (2016): 885-898.