Journal of Computer Engineering, Electronics and Information Technology (COELITE)

Journal homepage: https://ejournal.upi.edu/index.php/COELITE

Verticulture Plant Monitoring System Using Internet of Things Based on Blynk Application

Vion Age Tricahyo, Fatra Nonggala Putra, Muhammad Lutfi Ashari,*

Department of Computer Science, Universitas Nahdlatul Ulama Blitar, Indonesia
*Correspondence: E-mail: vionage@gmail.com

ABSTRACT

The development of technology is now very rapid. One of them is the use of the internet of things in various fields. One of the applications of technology that is developing in the world including in Indonesia is in agriculture. In this field, there is a model of how to grow plants verticulturally that utilizes limited land. In agriculture, the verticulture cropping technique known as vertical cropping requires optimal soil moisture through proper watering. So an automatic monitoring process is needed to measure soil moisture, air humidity to plant temperature. In this study focused on optimizing the automatic monitoring system using the blynk application compared to ordinary monitoring. This research also encourages further exploration of the use of micro controllers and internet of things technology in agriculture.

ARTICLE INFO

Article History:
Submitted/Received 21 Feb 2023
First Revised 05 Mar 2023
Accepted 25 Mar 2023
First Available online 01 Apr 2023
Publication Date 01 Apr 2023

Keyword:
Blynk app,
IoT,
Monitoring system,
NodeMCU,
Vertikultur.

© 2023 Universitas Pendidikan Indonesia

1. INTRODUCTION

Food availability is a staple considering that the longer the population increases. One form of increasing food sufficiency is by utilizing yard land outside the original agricultural land. In urban areas, the problem faced is limited land area [1]. Limited ownership of yard land is a result of the increasing price of land in urban areas. One way of cultivation that the community can apply to limited land conditions is vegetable cultivation with verticulture techniques [2].

In the process of verticulture cultivation, watering is one of the treatments that need to be done[3]. Providing sufficient water is an important factor for plant growth, because water affects soil moisture[4]. Crop production is inseparable from soil moisture which is influenced by water availability where this soil moisture will affect the plant growth process[5]. Not precisely the process of watering plants will cause production results not to be optimal or can even make plants die. Watering is traditionally considered less effective and inefficient because it does not save time and tends to be excessive in meeting water needs. If too much water at the time of watering will result in higher moisture in the soil which causes the plant to rot [6]. To deal with these problems, the development of technology allows us to create a system where the system will work automatically and we can monitor the state of our plants without worrying about plants will lack water or excess water[7].

The technology is the internet of things with the support of three important parts: mechanics, hardware (electronics) and control algorithms. The three parts interact with each other and cannot be separated into one system [8]. One of the electronic circuits that can be used for controllers is a micro controller. A micro controller is a small computer that includes a micro processor, memory, input/output lines, and other devices [9]. One such model is the NodeMCU ESP8266. NodeMCU is a micro controller board designed with ESP8266 Wifi chip that can be used to connect micro controller to Wifi network [10]. Micro controllers can run as desired when a program has been stored in it[11]. In running it, especially in the monitoring system, a sensor is needed[12]. Because the micro controller receives input from sensors, that is, information from the environment, and sends the output signal to the actuator, which can have an effect on the environment[13].

One of the sensors needed in this study is the YL sensor. This sensor is known as a soil moisture sensor[14]. This sensor operates by detecting moisture around the soil by displaying its temperature [15]. Another sensor is the DHT11 sensor which is a sensor module that can detect the temperature and humidity levels of an object with analog voltage output[16]. This information can be processed using a micro controller. DHT11 sensors typically feature accurate calibration for temperature and humidity readings [17]. The calibration data is stored in the OTP program memory, which is also known as the calibration coefficient [18]. Furthermore, the water flow sensor or known as the waterfall sensor. The sensor is made of plastic and has a rotor and Hall Effect sensor. The rotor will rotate as water flows through it[19]. This rotational speed will correspond to the amount of water flowing through the propeller [20]. In addition to sensors, blynk applications and supporters are also needed, power supplies, to several supporting tools to create a framework in building verticulture plant models.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

2. METHODS

In the implementation of system design, tools, meeting functional and non-functional needs, until the implementation of the system researchers use the prototype method. **Figure 1.** shows the diagram flow of the prototype method.

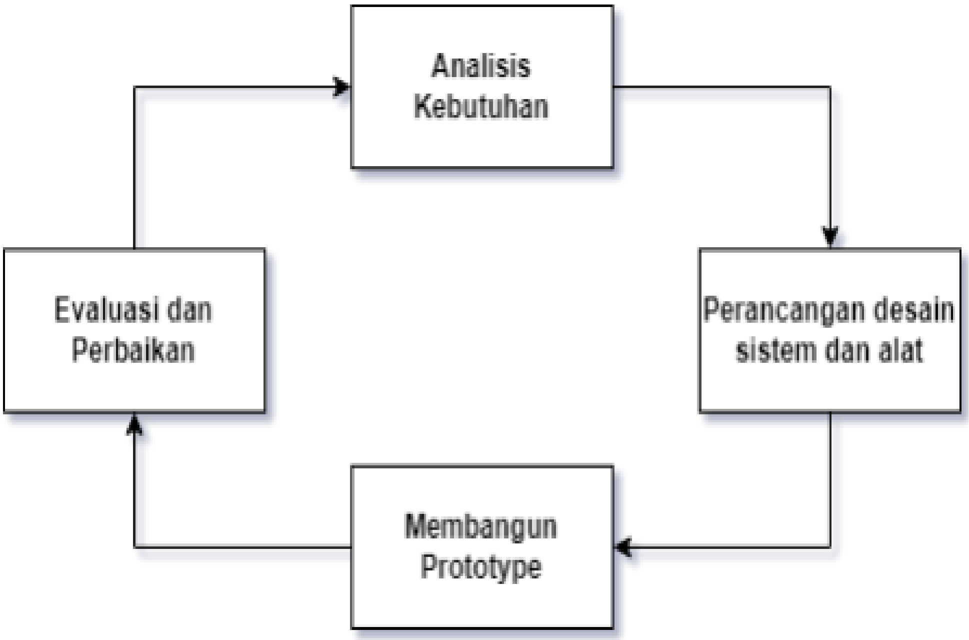


Figure 1. Prototype method.

2.1. Needs analysis

In this analysis, the author determines what needs are needed for the process of making the system in this study. Needs including functional and non-functional needs fall into this stage. The functional need in this study is that the system made must be able to read soil moisture, air humidity and temperature data on the media. Non-functional needs will be provided software and supporting hardware, internet access and location as well as ease of support for system use.

2.2 Design of systems and tools

In the design of systems and tools, researchers design systems and design tools to be built. In the process of making a hardware circuit system design and also a prototype architecture design on the system. The stage begins with designing the interface according to the functional needs of the software. This display design uses mock-up service provider software and uses flowcharts as system design. The interface design consists of mobile and web equipped with database design. Before designing an interface, the workflow of the system work flowchart must first exist. So that when designing the interface it will be easier to identify what kind of features are in the system later. **Figure 2.** shows the system workflow consisting of application and sensor needs.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

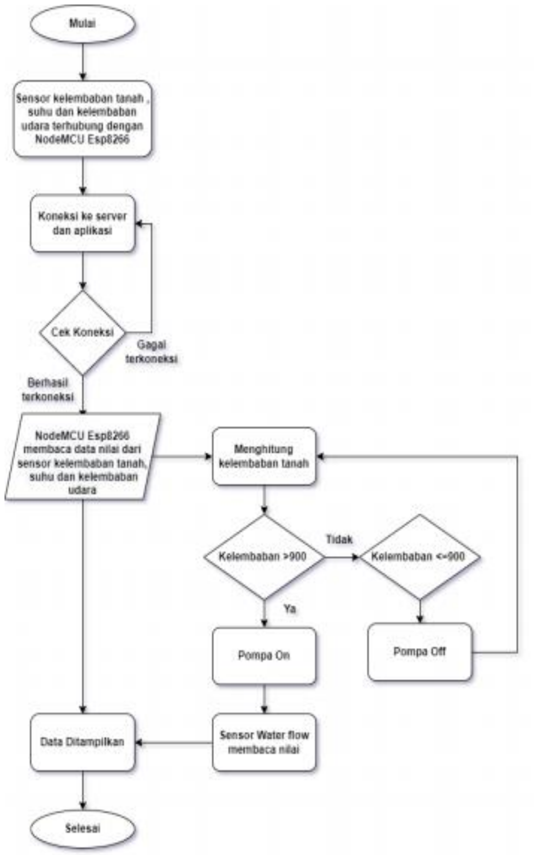


Figure 2. System workflow.

When data is displayed in the application, the application must contain the contents of the sensor consisting of soil moisture, temperature value and air humidity value of a medium. The sensor will later be connected to the Android mobile which will then be displayed to the screen through the intermediary blynk app. **Figure 3.** shows the system architecture of how the blynk app connects with sensors.

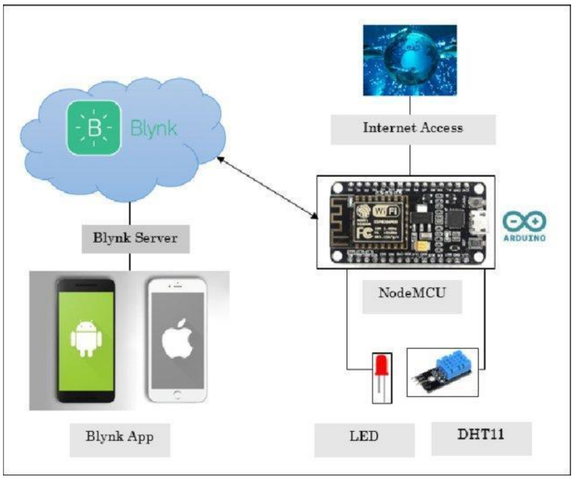


Figure 3. Bylnk app.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

After knowing the workflow of the system, proceed to know the flow of flowcharts and mobile interfaces. In the android flowchart, the system must display the main monitoring menu containing the results of sensor data. Mobile will read and display data through the mobile phone sent by sensors. The design of the mobile interface must match the state of the existing flowchart flow. The design should describe and display the actual state of the system. **Figure 4.** shows flowchart for design and mobile interface that will later be implemented in building a prototype.

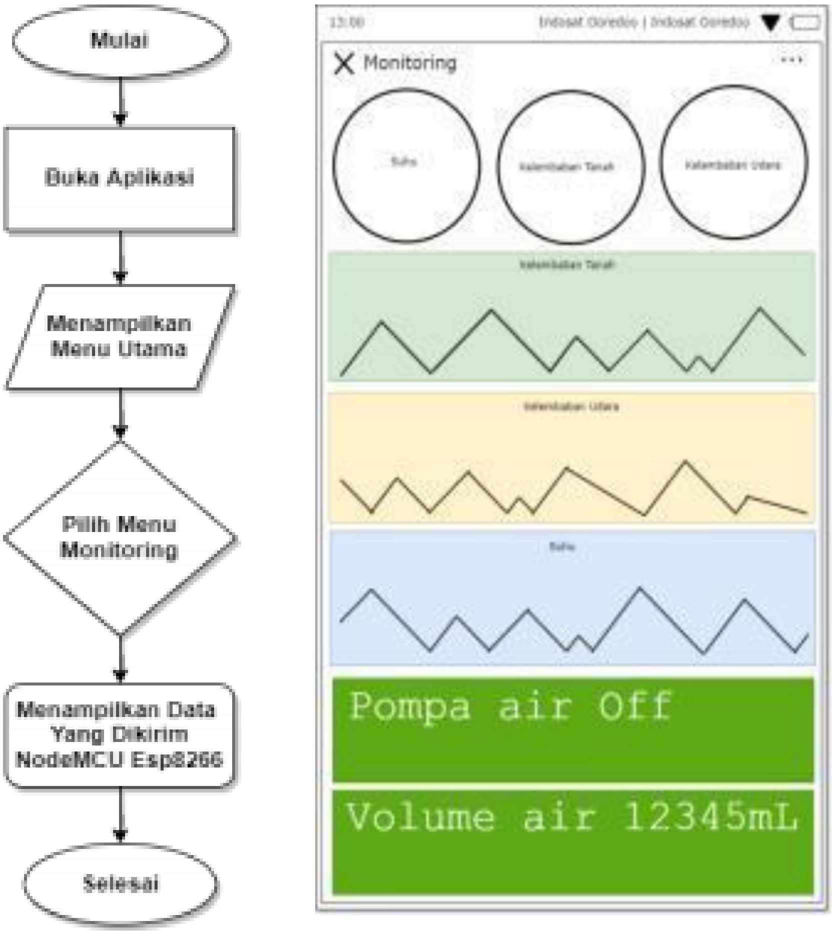


Figure 4. (Left) Android system workflow. (Right) Android interface.

Followed by the hardware design stage that combines all electronic components into one unit. Hardware design starts from assembling each component and continues to connect between a series of components with cables. **Figure 5.** shows the hardware design that will be implemented.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

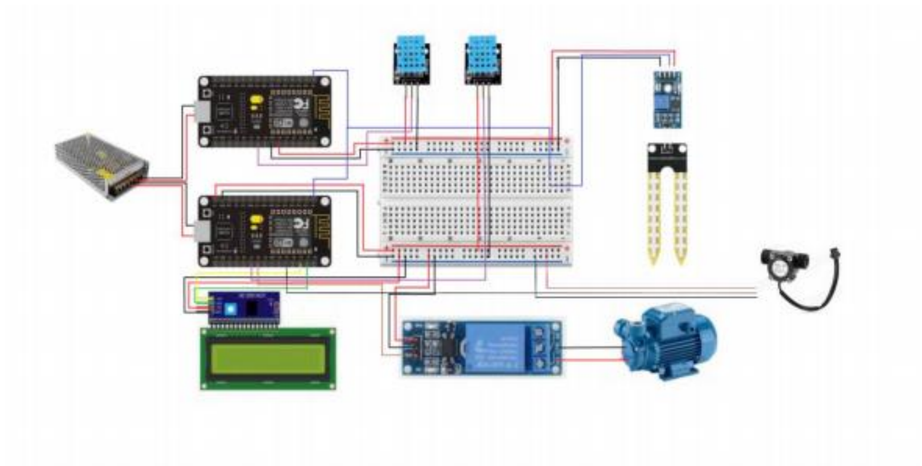


Figure 5. Hardware design.

Figure 6. shows the design of the skeleton of verticulture plants. At this stage combines the hardware design with the addition of a framework from the verticulture model plant.

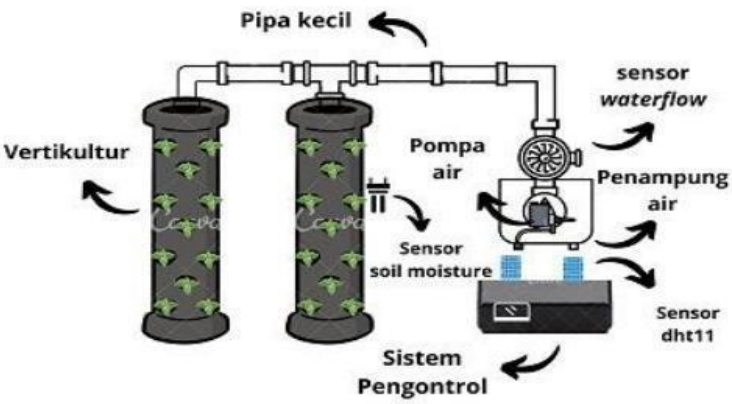


Figure 6. System architecture prototype.

2.3 Build a Prototype

This stage is a step to change from the design stage to the stage of building systems and assembling on hardware. At this stage, researchers first implement hardware components according to the needs and designs that have been designed before. After implementing the hardware components and completing the system prototype including the database in it, proceed with integrating software and hardware.

2.4 Evaluation

After all the previous stages have been carried out, the next stage is to evaluate through testing software and hardware systems. Whether the system is up and running or not. And it is connected to one another or not. This process is also the process of implementing a system that is ready to operate. So that if the system is not ready or there are still problems, immediate repairs will be made.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

3. RESULTS AND DISCUSSION

Research on prototype automatic sprinkler systems in plant cultivation verticulture based on the Internet of Things requires implementation on hardware, verticulture plant frameworks, and software. The following are the implementation results of all three.

3.1. Hardware Implementation

Hardware implementation is the execution process of the hardware design process into a system so that it can work as expected. **Figure 7.** shows the implementation of the system.

Figure 7. Hardware implementation.

Picture in **Figure 7.** is the last appearance of an automatic sprinkler system consisting of several components that have been made together, including soil moisture sensors, dht11 sensors, water flow sensors as data senders, 2 NodeMCU as system control centers, power supply as power supply, and others.

3.2. Implementation of System Architecture

The implementation of this prototype uses verticulture techniques which use 2 4 cm paralon pipes with water pipes on top. This pipe will later be given a water flow sensor that will calculate how much water flows through it. The water reservoir on this prototype is a glass water reservoir with a size of 50x30x20 cm which will be inserted 1 water pump into it. **Figure 8.** shows the whole implementation of the system along with the material that used in verticulture techniques.

Figure 8. Implementation of system architecture.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

The architecture of the system has been tested several times until the system is actually running validly. The system that is already running will be connected to the android system and the sensor test results will be displayed periodically. So there will be a test between each sensor that will be displayed for comparison.

3.3. Sensor Testing

Sensor testing is carried out with the aim of knowing whether the sensor used in the system can read data properly or not. Sensor testing is carried out by connecting all sensors to Nodemcu with the help of jumper cables. After all connected then open the ArduinoIDE software and select the serial monitor on the toolbar. Sensor testing is carried out on all sensors which include temperature sensors, air humidity sensors, soil moisture sensors, and water flow sensors, sensor test results are seen in the following figure and table.

a. Waterflow Sensor Accuracy Testing

In this test using a waterflow sensor as a counter for the discharge of water coming out of the pump. This test is carried out with the aim of determining the level of accuracy resulting from measuring water discharge without using a monitoring system and measuring water discharge using a monitoring system. This measurement is done by pumping water from the water reservoir through the waterflow sensor into a measuring cup as much as 1000 ml then matching the results of the two measurements. **Table 1.** shows the test results of the waterflow sensor.

Table 1. Waterflow sensor test results

No	Plan (ml)	Test result			
		With sensor (ml)	Error (%)	without sensor (ml)	Error (%)
1	1000	990	1 %	877	12,3 %
2	1000	1010	1 %	1217	21,7 %
3	1000	980	2 %	979	2,1 %
4	1000	1010	1 %	1057	5,7 %
5	1000	1010	1 %	1128	12,8 %
6	1000	950	5 %	1036	3,6 %
7	1000	1040	4 %	1130	13,0 %
8	1000	1020	2 %	1147	14,7 %
9	1000	990	1 %	866	13,4 %
10	1000	990	1 %	1145	14,5 %
11	1000	1000	0 %	1012	1,2 %
12	1000	1010	1 %	995	5 %
13	1000	990	1 %	997	3 %
14	1000	1000	0 %	987	1,3 %
15	1000	1000	0 %	1082	8,2 %
16	1000	990	1 %	979	2,1 %
17	1000	1030	3 %	989	1,1 %
18	1000	980	2 %	937	6,3 %
19	1000	1000	0 %	1119	11,9 %
20	1000	990	1 %	1173	17,3 %
Result		**Error Average (%)**	**1,4 %**	**Error Average (%)**	**8,2%**

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

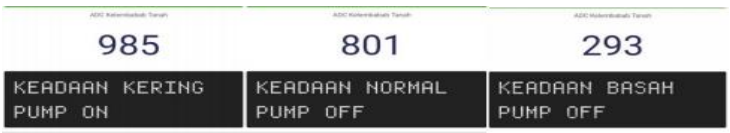
From **Table 1.**, it can be seen the results of testing waterflow sensors without a monitoring system and with a monitoring system on water with a volume of 1000 ml has an average percentage of error in discharge, which is 1.4% with sensors and 8.2% without sensors. From the table, it can be concluded that the accuracy obtained from testing the waterflow sensor to measure water discharge in measuring water discharge using a monitoring system is 98.6% and without a monitoring system is 91.8% in measuring water discharge using a monitoring system. This shows that in addition to spending energy and time to water, tests without sensors have more errors than tests using sensors.

b. Relay and pump testing

Relay and pump testing is carried out by connecting the relay to a pre-programmed NodeMCU digital pin. When soil moisture shows different ADCs then NodeMCU through digital pins will also give commands to different relays. This test can be seen directly and can also be seen through the LCD on the Android application. **Table 2.** shows the test results of the relay according to the values that read by the sensor.

Table 2. Relay and pump test results

Sensor	ADC	Circumstances	Reason	Effect	Relay
Soil moisture sensor	>900	Dry soil	Water-deficient soil can cause plants to lack nutrients and slow down the process of photosynthesis.	Make the plant wither, dry the tips of the leaves, it will even make the plant die	On
	>=600and <=900	Normal Soil	Sufficient soil will need water, not less water or excess water.	Make plants grow well and lush due to the process of photosynthesis	Off
	<600	Wet soil	Too much water soil can cause the condition of the roots to lack oxygen.	Makes the roots die, the leaves turn yellow and the stems become flabby	Off



From **Table 2.**, it can be concluded that the relay and pump testing on the automatic sprinkler system runs according to the planned soil moisture test data.

c. Testing real time monitoring

At this stage, testing is carried out on the monitoring system to find out whether this system can display time in real-time or close to real-time with a slight delay. This test was carried out by observing 20 data which was then displayed on the homepage of the android application. **Table 3.** shows the test results with the average delay time of the system.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

Table 3. Test results and the delay time average

No	Temperature	Air humidity	Soil moisture	Time	Delay (seconds)	Real-time description
1	26.20	69	711	23:31:10	1	fix
2	26.20	69	716	23:31:12	2	delay
3	26.20	69	715	23:31:14	2	delay
4	26.20	69	711	23:31:15	1	fix
5	26.20	69	716	23:31:16	1	fix
6	26.20	69	717	23:31:18	2	delay
7	26.20	69	712	23:31:19	1	fix
8	26.20	69	717	23:31:21	2	delay
9	26.20	69	717	23:31:23	2	delay
10	26.20	69	712	23:31:24	1	fix
11	26.20	69	717	23:31:26	2	delay
12	26.20	69	718	23:31:28	2	delay
13	26.20	69	714	23:31:29	1	fix
14	26.20	69	716	23:31:31	2	delay
15	26.20	69	716	23:31:33	2	delay
16	26.20	69	711	23:31:34	1	fix
17	26.20	69	711	23:31:35	1	fix
18	26.20	69	712	23:31:37	2	delay
19	26.20	69	712	23:31:39	2	delay
20	26.20	69	708	23:31:40	1	fix
		Delay average			1,55 seconds	

From **Table 3.**, it was found that the monitoring system in this study can be said to be close to real-time with the results of the time delay on the Android application is an average of 1.55 seconds.

3.4. Implementation of the android system

The implementation of the android application is the final result after carrying out the application design process along with the application interface. The display of the android application is useful as a medium for exposing information in the form of unit data, graphics or information in the form of text. The data can be displayed in the android application when the data that has been uploaded by NodeMCU is stored on the server. The display on this android application can be seen when the user has an account and also has access to open the application that has been created.

From the sensor test above, the sensor test results will be sent and received, then will be displayed by the android system. The display of data corresponds to the sensor and the results to be displayed are in real time. **Figure 9.** shows the implementation of the android application using Blynk that described the data view.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

Figure 9. Android implementation.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

In tests carried out on the monitoring system on the homepage of the application built, the android application successfully displays the data that has been received from the micro controller. **Table 4.** shows the test results of the android application using the Black Box software testing method.

Table 4. Android testing.

Variable Test Data	Result	
	Expected Result	Testing Result
Temperature monitoring graph	Display average temperatures	Successes [√] It didn't work [ ]
Air humidity monitoring graph	Display air humidity graph	Successes [√] It didn't work [ ]
Soil moisture monitoring graph	Display soil moisture graph	Successes [√] It didn't work [ ]
Water discharge data monitoring to android	Displays information in the form of water discharge text	Successes [√] It didn't work [ ]

4. CONCLUSION

In the research of automatic sprinkler systems on verticultural plant cultivation based on the Internet of Things with the prototype method, a prototype of the skeleton architecture of verticultural plants and automatic sprinklers was successfully made. In addition, the monitoring system has been tested on the android application through the blynk app application. The tests that have been carried out resulted in 98.6% for water discharge monitoring measurements and a delay of 1.55 seconds for real time testing.

6. AUTHORS' NOTE

The authors declare that there is no conflict of interest regarding the publication of this article. Authors confirmed that the paper was free of plagiarism.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

7. REFERENCES

[1] Kusumo, R. A. B., Sukayat, Y., Heryanto, M. A., dan Wiyono, S. N. (2020). Budidaya sayuran dengan teknik vertikultur untuk meningkatkan ketahanan pangan rumah tangga di perkotaan. *Dharmakarya: Jurnal Aplikasi Ipteks untuk Masyarakat*, *9*(2), 89-92.

[2] Hendri, M., Rozirwan, R., Apri, R., and Handayani, Y. (2018). Intensification of seaweed cultivation euchema cottonii with verticulture method in the water of Kelagian Island, Lampung bay. *International Journal of Marine Science*, *8*(14), 114-126

[3] Ariati, P. E. P. (2017). Produksi beberapa tanaman sayuran dengan sistem vertikultur di lahan pekarangan. *Jurnal Agrimeta*, *7*(13). 76-86

[4] Wigmore, O., Mark, B., McKenzie, J., Baraer, M., and Lautz, L. (2019). Sub-metre mapping of surface soil moisture in proglacial valleys of the tropical Andes using a multispectral unmanned aerial vehicle. *Remote Sensing of Environment*, *222*(1), 104-118.

[5] Sudarmaji, A., Supriyo, H., and Widodo, A. (2020). Simple parallel probe as soil moisture sensor for sandy land in tropical-coastal areas. *Pertanika Journal of Science and Technology*, *28*(3), 829-838.

[6] Fuadi, S., dan Candra, O. (2020). Prototype alat penyiram tanaman otomatis dengan sensor kelembaban dan suhu berbasis arduino. *JTEIN: Jurnal Teknik Elektro Indonesia*, *1*(1), 21-25.

[7] Rajakumar, G., Sankari, M. S., Shunmugapriya, D., and Maheswari, S. U. (2018). IoT based smart agricultural monitoring system. *Asian Journal of Applied Science and Technology (AJAST)*, *2*(2), 474-480.

[8] Goap, A., Sharma, D., Shukla, A. K., and Krishna, C. R. (2018). An IoT based smart irrigation management system using machine learning and open source technologies. *Computers and Electronics in Agriculture*, *155*(1), 41-49.

[9] Abdullah, A., Cholish, C., dan Haq, M. Z. (2021). Pemanfaatan IoT (internet of things) dalam monitoring kadar kepekatan asap dan kendali camera tracking. *CIRCUIT: Jurnal Ilmiah Pendidikan Teknik Elektro*, *5*(1), 86-92.

[10] Arini, D., dan Kumara, P. W. (2019). Robot line follower berbasis mikrokontroller arduino uno atmega328. *INFORMANIKA*, *5*(1). 18-25

[11] Dalimunte, B., dan Sitorus, P. (2021). Pengembangan prototype traffic light mikrokontroler berbasis rduino mega pada mata pelajaran teknik pemrograman mikroprosesor dan mikrokontroler di SMK negeri 1 Percut Sei Tuan. *JEVTE: Journal of Electrical Vocational Teacher Education*, *1*(1), 10-17.

[12] Raut, M., Shete, G., Shinde, V., and Suryawanshi, A. (2019). Automatic mini CNC machine for pcb drawing using arduino. *International Research Journal of Engineering and Technology (IRJET)*, *6*(6), 310-314.

[13] Maroli, A., Narwane, V. S., and Gardas, B. B. (2021). Applications of IoT for achieving sustainability in agricultural sector: a comprehensive review. *Journal of Environmental Management, 298*(1), 1-8.

[14] Purwanti, S., Febriani, A., Mardeni, M., and Irawan, Y. (2021). Temperature monitoring system for egg incubators using raspberry pi3 based on internet of things (IoT). *Journal of Robotics and Control (JRC)*, *2*(5), 349-352.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149

[15] Bhatnagar, V., Chandra, R., and Prasad, J. (2019). Soil moisture sensors for sustainable irrigation: comparison and calibration. *International Journal of Sustainable Agricultural Management and Informatics*, *5*(1), 25-36.

[16] Mardika, A. G., dan Kartadie, R. (2019). Mengatur kelembaban tanah menggunakan sensor kelembaban tanah YL-69 berbasis arduino pada media tanam pohon gaharu. *JoEICT (Journal of Education and ICT)*, *3*(2), 130-14.

[17] Srivastava, D., Kesarwani, A., and Dubey, S. (2018). Measurement of temperature and humidity by using arduino tool and DHT11. *International Research Journal of Engineering and Technology (IRJET)*, *5*(12), 876-878.

[18] Shrestha, R. (2019). Study and control of DHT11 using atmega328P microcontroller. *International Journal of Scientific and Engineering Research*, *10*(4),518-521.

[19] Rangan, A. Y., Yusnita, A., dan Awaludin, M. (2020). Sistem monitoring berbasis internet of things pada suhu dan kelembaban udara di laboratorium kimia XYZ. *Jurnal E-Komtek*, *4*(2), 168-183.

[20] Harija, H., George, B., and Tangirala, A. K. (2021). A cantilever-based flow sensor for domestic and agricultural water supply system. *IEEE Sensors Journal*, *21*(23), 27147-27156.

DOI: https://doi.org/10.17509/coelite.v2i1.57108
p- ISSN 2829-4157 e- ISSN 2829-4149