

Pemanfaatan Teknologi *IoT* Pada *Smart Farming Microgreen* dan Akuisisi Data

Rico Wahyu Laksana¹, Bagus Setya Rintyarna², Aji Brahma Nugraha³

^{1,2,3} Teknik Elektro, Fakultas Teknik, Universitas Muhammadiyah Jember
Jl. Karimata no.49, Jember 68121, Indonesia
E-mail: wlrigo360@gmail.com

ABSTRAK

Abstrak - Dalam beberapa tahun belakangan ini, manusia telah disadarkan akan pentingnya menjaga kesehatan, terutama dalam hal pemenuhan kebutuhan nutrisi bagi tubuh, Salah satu jenis pangan yang merupakan sumber vitamin dan mudah dikonsumsi oleh masyarakat adalah tanaman sayur-sayuran, yang dimana merupakan sumber pangan yang banyak mengandung vitamin, mineral dan zat gizi yang dibutuhkan oleh tubuh kita. Dengan pesatnya peningkatan populasi manusia di dunia seiring dengan penyempitan lahan pertanian yang dibutuhkan untuk budidaya tanaman. *Microgreen* hadir sebagai solusi, dengan ukuran tanaman yang kecil lahan yang tidak luas dengan kandungan nutrisi yang lebih tinggi dibanding sayuran dewasa pada umumnya. Pada zaman yang serba canggih sekarang ini, sudah banyak pekerjaan-pekerjaan manusia yang sudah tergantikan dengan teknologi-teknologi modern seperti halnya *Internet of Things (IoT)*. *Internet of Things (IoT)* adalah suatu konsep yang mana objek memiliki kemampuan untuk mengirim data melalui jaringan internet, tidak memerlukan interaksi antar manusia atau manusia dengan komputer. Hanya dengan penggunaan suatu mikrokontrol yang dapat terhubung ke internet seperti ESP32 dengan tambahan sensor-sensor untuk mendeteksi adanya perubahan lingkungan fisik atau kimia dan aktuator-aktuator sebagai manipulator pada lingkungan tersebut. Pada lingkungan buatan untuk *microgreen* ini dibutuhkan parameter nilai suhu, kelembaban udara, dan kelembaban tanah untuk memenuhi kebutuhan dari tanaman *microgreen* itu sendiri.

Kata kunci: *Microgreen, Internet of Things, Smart Farming Microgreen, Microgreen Pakcoy*

ABSTRACT

Abstract - In recent years, humans have been made aware of the importance of maintaining health, especially in terms of meeting the nutritional needs of the body. One type of food that is a source of vitamins and is easily consumed by the public is vegetable plants, which are a large source of food. contains vitamins, minerals, and nutrients needed by our body. With the rapid increase in the world's human population along with the narrowing of agricultural land needed for plant cultivation. Microgreens are present as a solution, with small plant sizes in an area that is not large and a higher nutrient content than mature vegetables in general. In today's sophisticated era, many human jobs have been replaced by modern technologies such as the Internet of Things (IoT). The Internet of Things (IoT) is a concept in which objects have the ability to send data over the internet network without requiring interaction between humans or humans and computers. Only by using a microcontroller that can be connected to the internet, such as the ESP32, with the addition of sensors to detect changes in the physical or chemical environment and actuators as manipulators in that environment. In this artificial environment for microgreens, the parameter values of temperature, air humidity, and soil moisture are needed to meet the needs of the microgreen plants themselves.

Keywords: *Microgreen, Internet of Things, Smart Farming Microgreen, Microgreen Pakcoy*

Copyright © 2023 Jurnal Teknik Elektro dan Komputasi (ELKOM)

1. PENDAHULUAN (10 PT)

Dalam beberapa tahun belakangan ini telah menyadarkan masyarakat tentang betapa pentingnya mengonsumsi vitamin untuk menjaga kesehatan. Salah satu jenis pangan yang merupakan sumber vitamin dan mudah dikonsumsi oleh masyarakat adalah tanaman sayur-sayuran, yang dimana merupakan sumber pangan dengan banyak kandungan vitamin, mineral dan zat gizi yang dibutuhkan oleh tubuh.

Pesatnya meningkatnya populasi manusia di dunia seiring dengan peningkatan kebutuhan lahan sebagai tempat tinggal. Hal ini berdampak terhadap penyempitan lahan pertanian guna budidaya tanaman, yang dalam jangka panjang

akan berdampak pula pada sumber pangan. Kebutuhan akan nutrisi bagi penduduk pun menjadi permasalahan ditengah terus peningkatan pertumbuhan populasi penduduk ini. *Microgreen* merupakan salah satuantisipasi dari permasalahan-permasalahan tersebut.[1]

Dengan kandungan nutrisi yang lebih tinggi dibandingkan dengan jenis sayuran dewasa pada umumnya [2]. Dengan ukurannya yang kecil, tanaman ini tidak perlu lahan yang luas dalam pertumbuhannya.

Pemeliharaan tanaman adalah aspek penting dalam tumbuh kembang tanaman. Bahkan memerlukan pengetahuan terkait karakteristik tanaman tersebut, seperti suhu lingkungan terbaik, kondisi kelembaban udara terbaik, waktu serta dosis pengairan dan pencahayaan. Jika keliru, keoptimalan pertumbuhan dan perkembangannya akan terancam, bahkan hingga bisa layu dan mati.

Pada saat ini, masyarakat di dunia masih banyak yang melakukan pemeliharaan tanaman masih banyak dilakukan secara manual, dan sering terjadi kesalahan dalam prosesnya seperti jadwal yang tidak tepat serta dosis yang tidak sesuai yang menjadikan pertumbuhan kurang efektif. Disamping itu, faktor kelalaian manusia seperti malas dan lupa dikarenakan aktivitas atau hal-hal yang lain menjadikan pemeliharaannya tidak teratur, dan ketika pemilik sedang jauh dari tanaman juga menjadi hambatan yang menyebabkan pemeliharaan menjadi terabaikan.

Berdasarkan pada masalah-masalah diatas, diperlukan sebuah sistem yang dapat diandalkan dalam mengatasi masalah tersebut, seperti sistem pemeliharaan tanaman otomatis dengan sistem kontrol dan *monitoring* pada tanaman tersebut. Sebagai sistem kontrol, dapat mengambil keputusan dan tindakan dalam menentukan kapan waktu yang tepat dalam melakukan pengairan, pencahayaan dan penyesuaian suhu ruang. Sebagai sistem *monitoring*, dapat memantau data dari parameter-parameter diatas yang dapat diakses dengan mudah dari jarak jauh.

Pada penelitian sebelumnya telah dirancang sistem monitoring dan kontrol batas parameter berbasis *IoT* yang dilakukan oleh [3]. Institut Teknologi Nasional, Bandung, yang berjudul "*Rancang Bangun Smart Grow Box Hidroponik untuk Pertumbuhan Tanaman Microgreen Berbasis Internet of Things*". Kita akan membuat suatu sistem yang menggunakan aktuator-aktuator seperti kipas angin dan pompa air yang dapat dikendalikan melalui aplikasi *Blynk* yang terpasang di *smartphone*. Dengan menggunakan metode set point batas parameter lingkungan, kita dapat memonitor dan mengendalikan aktuator-aktuator ini dari jarak jauh melalui internet. Sistem ini dirancang untuk mengatur proses pencahayaan dan pengairan pada pertumbuhan tanaman *microgreen*. Pencahayaan akan berfungsi selama sekitar 12 jam sehari, sementara pengairan akan diatur sesuai dengan batas parameter yang telah ditentukan untuk media tanam. Penelitian ini bermaksud untuk menciptakan dan merancang suatu sistem lingkungan buatan yang memungkinkan pemantauan dan pengendalian parameter lingkungan pertumbuhan tanaman *microgreen* menggunakan *smartphone*. Tujuannya adalah untuk meningkatkan efisiensi dan kemudahan dalam pemeliharaan agar dapat mencapai hasil panen berkualitas tinggi.

Dari hasil kajian permasalahan tentang sistem monitor dan kontrol pertumbuhan tanaman *microgreen* yang telah saya amati, maka pada penelitian ini akan digunakan sistem monitor dan kontrol memuat parameter suhu ruangan, kelembaban udara, kelembaban media tanam, kendali ON/OFF pada *Cooling System*, LED, *speaker* dan *water pump*. Serta dapat kita kontrol penjadwalan atas aktuator-aktuator diatas. Pada penelitian ini diberi judul "Pemanfaatan Teknologi *IoT* pada *Smart Farming Microgreen* dan Akuisisi Data".

2. KAJIAN PUSTAKA

2.1. *Microgreen*

Microgreen adalah varietas baru produk sayuran yang semakin populer dari waktu ke waktu. Waktu panennya adalah 7-14 hari setelah penanaman yaitu hingga mulai tumbuhnya daun sejati pertama. Keunikan berbagai warna, tekstur, tampilan visual, aroma, dan rasa yang dimiliki oleh sayuran muda ini menjadi daya tarik utama. Memiliki beberapa manfaat yang baik untuk dikonsumsi, terkandung nutrisi yang lebih tinggi daripada sayuran dewasa [4]. *Microgreen* juga dapat dijual sebagai produk segar yang digunakan dalam salad, *sandwich*, atau sebagai *garnish* pada hidangan makanan [5].

2.2. Tanaman Pakcoy

Pakcoy yang dalam nama latin adalah *Brassica rapa L.* merupakan jenis sayuran dan masih memiliki kesamaan dengan sawi. Tumbuhan pakcoy memiliki akar tunggang dan cabang akar berbentuk elips yang menjulur ke segala arah pada kedalaman 30-50 cm. Daun pakcoy berbentuk lonjong, hijau tua agak mengkilat, daun tidak berbentuk seperti kepala, daun tumbuh agak tegak atau setengah mendatar. Daun tanaman tersusun dalam bentuk spiral yang rapat dan menempel pada batangnya. Tangkai daun berwarna hijau muda, gemuk dan berisi [6]. Memiliki batang jenis semu, dimana pelepah daunnya tumbuh rapat, saling melekat dan tersusun secara teratur. Batang ini berwarna hijau muda dan berfungsi sebagai alat pembentuk dan penopang daun. Selain itu, batangnya berukuran pendek dan beruas-ruas, sehingga tidak terlalu terlihat dengan jelas [7].

Ketinggian tempat penanaman pakcoy mempengaruhi umur panennya, semakin rendah tempat penanaman, waktu panen akan lebih cepat [8]. Biasanya, tanaman pakcoy ditanam di wilayah dataran rendah dengan suhu sekitar 15-30°C. Untuk mendapatkan pertumbuhan yang optimal, pakcoy membutuhkan suhu udara sekitar 19°C - 21°C. Suhu udara ini berpengaruh pada berbagai tahapan pertumbuhan tanaman, seperti pembelahan sel-sel, perkecambahan, pertunasan, pembungaan, dan pemanjangan daun [9]. Kelembaban udara yang ideal bagi pertumbuhan tanaman pakcoy adalah antara 80% - 90%. Namun, jika kelembaban udara melebihi 90%, dampaknya akan merugikan pertumbuhan tanaman. Kelembaban yang tidak sesuai dengan kebutuhan tanaman dapat menyebabkan stomata (pori-pori daun) menutup, mengganggu penyerapan CO₂. Hal ini mengakibatkan masukan gas CO₂ ke dalam daun terhambat, sehingga fotosintesis tanaman menjadi terganggu. Akibatnya, proses

fotosintesis tidak berjalan efisien, dan semua tahapan pertumbuhan pada tanaman mengalami penurunan [10]. Untuk pertumbuhannya, tanah dengan pH antara 6 - 7 adalah yang paling ideal [11].

2.3. ESP32

Pin out pada ESP32 dapat dijadikan sebagai input atau output untuk mengontrol LED, kipas angin DC, atau bahkan menggerakkan motor DC. ESP32 adalah sistem dengan biaya rendah dan memiliki konsumsi daya yang sangat rendah berkat fitur-fitur hemat daya seperti *fine resolution clock gating*, *multiple power modes*, dan *dynamic power scaling* pada *chip (SoC)* dengan kemampuan Wi-Fi dan Bluetooth. ESP32 memiliki inti CPU yang lebih cepat dan kemampuan Wi-Fi yang lebih baik, serta dilengkapi dengan lebih banyak GPIO dan dukungan untuk *Bluetooth Low Energy*. Mikrokontroler ini dirancang untuk perangkat seluler, elektronik yang dapat dipakai, dan aplikasi *IoT (Internet of Things)*. Meskipun ESP32 tidak menggantikan ESP8266, namun membawa peningkatan di berbagai aspek. CPU-nya mirip dengan ESP8266 yaitu Xtensa® LX6 32-bit, namun dengan inti ganda. ESP32 juga dilengkapi dengan ROM 128KB dan SRAM 416K, serta *Flash Memory* (untuk menyimpan program dan data) sebesar 64MB.

2.4. Internet of Things

Internet of Things (IoT) adalah suatu konsep dimana suatu objek memiliki kemampuan untuk mengirim data melalui jaringan internet, tidak memerlukan interaksi antar manusia atau manusia dengan komputer [12]. *IoT* dapat digunakan sebagai alat kontrol, *monitoring*, dan banyak lagi kegunaannya di era modern ini. *IoT* merupakan kemajuan teknologi yang luar biasa karena mampu meningkatkan kualitas hidup melalui pemanfaatan sensor-sensor dan komponen-komponen yang dapat terhubung ke jaringan dan internet, jadi tidak memandang jarak. Prinsip kerja *IoT* adalah setiap objek harus memiliki alamat *Internet Protocol (IP)* yang berfungsi sebagai identitas dalam jaringan, memungkinkan objek tersebut saling berkomunikasi dengan objek lain dalam jaringan yang sama. Selanjutnya, alamat *IP* dalam setiap objek tersebut akan dihubungkan ke jaringan untuk memungkinkan koneksi dan pertukaran data [13]. Dalam artian sederhana perangkat-perangkat yang terkoneksi internet dan dapat dikontrol dengan mudah juga dapat disebut sebagai *IoT*.

2.5. Android

Android adalah sebuah sistem operasi *mobile* yang dikembangkan oleh Google, yang merupakan hasil modifikasi dari versi kernel Linux. Sistem operasi ini dirancang khusus untuk perangkat layar sentuh seperti *smartphone* dan tablet. Antarmuka Android umumnya menggunakan manipulasi langsung dengan gerakan sentuh, seperti geser, ketuk, dan cubit, untuk memanipulasi objek di layar. Aplikasi Android, juga dikenal sebagai "*apps*", berfungsi untuk menambahkan fungsionalitas pada perangkat. Biasanya, aplikasi Android ditulis dalam bahasa pemrograman Java menggunakan *Android Software Development Kit (SDK)*. SDK menyediakan seperangkat alat pengembangan lengkap, termasuk *debugger*, pustaka perangkat lunak, *emulator handset*, dokumentasi, sampel kode, dan *tutorial*. Proses pengembangan aplikasi melibatkan pengelolaan kode, data, dan file untuk membentuk sebuah *Android Package (APK)*, yang merupakan file arsip dengan akhiran *.apk*. Satu file APK berisi semua komponen aplikasi Android dan digunakan oleh perangkat Android untuk menginstal aplikasi. Dalam penelitian ini, sebuah antarmuka untuk mengendalikan *Smart Farming Microgreen* dibuat dalam bentuk aplikasi (perangkat lunak) *Android Mobile Apps*. Untuk membuat antarmuka pengguna tersebut, diperlukan aplikasi IDE (*Integrated Development Environment*) Android Studio, yang dapat diunduh langsung pada website resminya.

2.6. Arduino IDE

Software Arduino IDE (Integrated Development Environment) adalah media pemrograman arduino, dimana *software* ini sering di sebut juga dengan nama "*Sketch*" sebagai pen-debug, pengkompilasi dan dapat menulis programnya. *Software Arduino IDE* dalam menuliskan kode programnya digunakan bahasa *C for Arduino* yaitu bahasa C yang telah ditingkatkan untuk memenuhi kebutuhan *Hardware Arduino*. Kita dapat menulis program sesuai dengan yang kita inginkan, juga pada *software* ini sudah dipermudah dengan *library* yang dapat dengan mudah kita download di menu bagian kiri dan banyak lagi fitur-fitur terbaru yang dapat dengan mudah kita akses seiring dengan perkembangan *software* ini.

2.7. Capacitive Soil Moisture Sensor

Sensor merupakan perangkat yang digunakan untuk mendeteksi perubahan dalam lingkungan fisik atau kimia. Sensor mengubah variabel keluarannya menjadi sinyal listrik melalui suatu komponen yang disebut *transduser*. Saat ini, sensor-sensor ini telah dikembangkan dalam ukuran yang sangat kecil, bahkan mencapai orde nano meter. Ukuran yang sangat kecil ini memberikan kemudahan penggunaan dan juga menghemat energi. *Capacitive Soil Moisture Sensor* adalah sebuah perangkat sensor yang digunakan untuk mengukur kelembaban tanah. Sensor ini beroperasi berdasarkan prinsip kapasitansi, sirkuit tercetak yang terdapat pada sensor tersebut dilapisi dengan lapisan pernis untuk menghindari korosi. Sensor ini menghasilkan keluaran berupa tegangan analog dengan rentang nilai 1,2 hingga 2,5V. Penerapan *capacitive soil moisture sensor* untuk penyiraman otomatis sudah dapat dengan mudah dihubungkan dengan *microcontroller* [14].

2.8. Sensor DHT22

Pengukuran suhu (*temperature*) dan kelembaban udara (*humidity*) merupakan parameter yang sering digunakan dalam proses akuisisi data, terutama dalam bidang pertanian. Sensor memiliki peran penting dalam proses ini, karena bertugas mengubah kuantitas data dari alam menjadi bentuk digital yang dapat diproses oleh komputer. Akurasi hasil pengukuran sensor juga menentukan sejauh mana hasil yang diperoleh sesuai dengan pengukuran yang sebenarnya melalui instrumen ukur. Dalam proses akuisisi data, sensor dapat berbentuk

komponen diskrit atau terintegrasi dalam satu rangkaian chip IC. Biasanya, untuk mengukur satu jenis parameter, digunakan satu jenis sensor. Meskipun demikian, ada juga sensor yang mampu mengukur dua atau lebih parameter secara simultan, seperti DHT22 sebagai contohnya. Pada sistem yang akan kita buat dalam penelitian ini menggunakan sensor DHT22, dimana yang mana sering menjadi pilihan dalam pengukuran suhu dan kelembaban udara dengan keluaran digital [15]. Dengan kualitas pembacaan yang baik, dapat terbukti dari respon proses akuisisi data yang cepat dan tepat dengan harga yang relatif murah dibanding alat *thermohygrometer*. Menggunakan kapasitor dan termistor untuk mengukur udara disekitarnya [16].

2.9. Blynk Apps

Blynk adalah sebuah aplikasi yang dirancang untuk digunakan pada berbagai perangkat. Aplikasi ini memiliki tampilan yang menarik untuk produk terhubung atau proyek *IoT*. *Blynk* adalah pembuat aplikasi tanpa perlu menulis kode yang sederhana namun kuat juga bisa dengan mudah membuat prototipe, menyebarkan, dan mengelola perangkat elektronik yang terhubung dalam berbagai skala, mulai dari proyek pribadi hingga implementasi jutaan produk. Jika kita menghubungkan hardware kita seperti ESP32, Arduino, Raspberry Pi, Seeed, Particle, SparkFun, Adafruit, TI, dan berbagai perangkat lainnya ke cloud menggunakan *Blynk*. Kita dapat menggunakan *widget-widget* untuk membangun antarmuka pengguna. Antarmuka ini memungkinkan untuk memvisualisasikan data dari sensor dan mengontrol berbagai perangkat elektronik.

3. METODE PENELITIAN

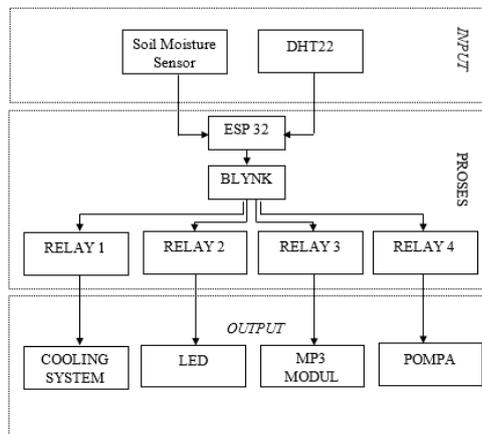
Dengan mengumpulkan referensi-referensi terkait menganalisa bagaimana pemanfaatan teknologi *IoT* pada *smart farming microgreen* dapat berjalan dengan baik, serta dengan melakukan pengujian sampai mendapat hasil dan data yang diinginkan.

3.1 Metodologi Penelitian

Metodologi perancangan sistem akan terbagi menjadi dua bagian, yakni perancangan *hardware* dan perancangan *software*. Berikut adalah perancangan hardware dan software dari sistem tersebut.

3.2 Perancangan Hardware

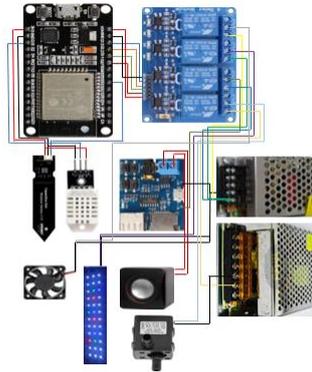
3.2.1 Diagram Blok Sistem



Gambar 1. Diagram blok sistem

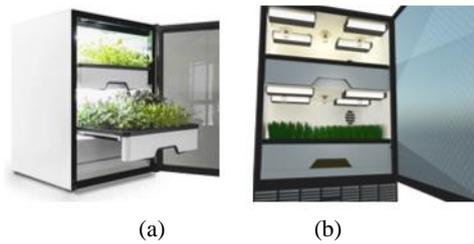
Sistem yang dirancang membutuhkan perangkat berupa *soil moisture sensor* dan DHT22 sebagai *Input*, mikrokontroler ESP32 dan *Blynk* sebagai proses, aktuator-aktuator sebagai *output* untuk memanipulasi parameter lingkungan sesuai dengan yang dibutuhkan tanaman.

3.2.2 Desain Skematik



Gambar 2. Desain skematik alat

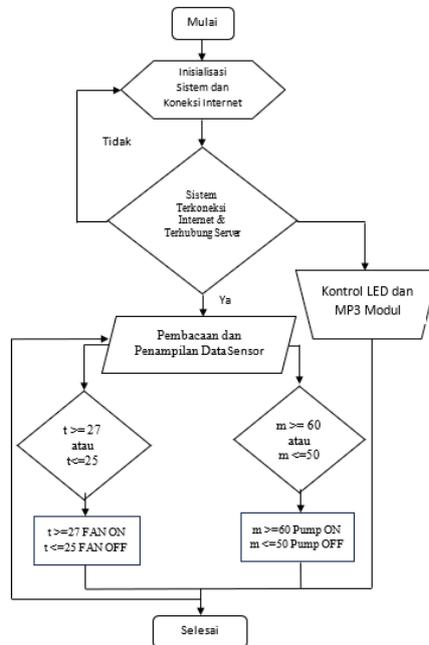
3.2.3 Desain Alat



Gambar 3. (a) Gambar alat tampak samping, (b) tampak dalam

3.3 Perancangan Software

3.3.1 Flowchart Sistem

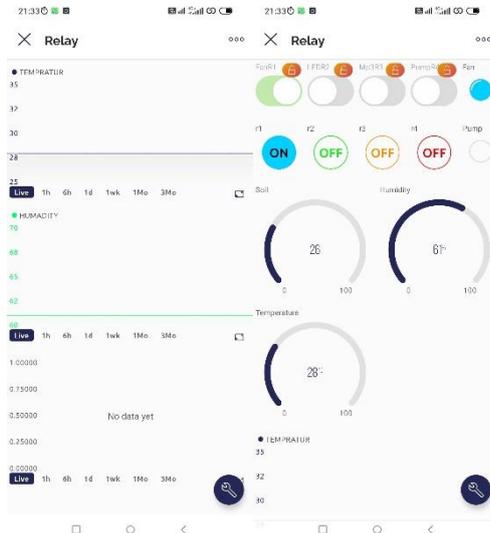


Gambar 4. Flowchart sistem

Pertama yakni inisiasi sistem atau ESP32 beserta koneksi internet. Jika sistem sudah terkoneksi ke internet dan terhubung ke server *Blynk*, ESP32 akan menerima data sensor yang di *Inputkan* dari sensor suhu dan kelembaban untuk diproses dengan batasan-batasan kondisi nilai sensor seperti yang ada pada *flowchart* diatas, yaitu jika suhu (t) lebih atau sama dengan 28°C maka *Fan Cooler* dan *Fan Airflow* akan menyala, kemudian jika suhu sudah memenuhi batas yang telah ditentukan, yakni kurang dari atau sama dengan 26°C maka *Fan Cooler* dan *Fan Airflow* akan mati sampai batas yang ditentukan agar suhu tidak melebihi dari batas parameter yang

dibutuhkan. Begitu pula dengan pompa, jika kelembaban media tanam lebih dari sama dengan 63% (kering) maka pompa akan menyala, setelah mencapai batas parameter yang ditentukan yakni kurang dari sama dengan 43% (basah) maka pompa akan mati. Untuk penggunaan LED dan MP3 Modul akan menyala secara manual atau sesuai dengan yang telah kita jadwalkan.

3.3.2 Desain Software



Gambar 5. Desain software

Desain *software* diatur sedemikian rupa agar dapat memudahkan pengguna dalam mengontrol dan *monitoring*, juga sudah tersedia widget *Superchart* yang dapat menyimpan hasil pembacaan data sensor dalam bentuk grafik. Grafik tersebut juga memungkinkan pengguna melihat hasil pembacaan sensor dalam kurun waktu tertentu.

4. HASIL DAN PEMBAHASAN

Pada bab ini dilakukan pengujian dengan tujuan mengetahui hasil dari perancangan yang telah dibuat pada bab 3. Pengujian ini terbagi menjadi beberapa bagian, mulai pengujian sensor yang digunakan hingga pengujian alat keseluruhan. Dari hasil pengujian-pengujian ini dapat dianalisa performa-performa dari tiap-tiap rangkaian yang saling terhubung hingga menjadi *box Smart Farming Microgreen*. Pengujian keseluruhan berguna untuk mengetahui kinerja sistem secara menyeluruh apakah sistem sudah berjalan sesuai dengan baik seperti yang diharapkan.

4.1 Pengujian Power Supply 12V dan 5V

Pengujian *power supply* atau catu daya ini bertujuan untuk mengetahui kinerja dari *power supply* apakah sesuai dengan spesifikasinya dengan mengukur *input* dan *output* dari *power supply* sendiri dan juga dengan menggunakan avometer digital. Berikut gambar dan hasil pengujiannya.

4.2 Tabel 1. Tabel uji *power supply* 12V

No.	Parameter	Tanpa beban		Dengan beban		ΔUkur		Ket
		V	I(mA)	V	I	V	I	
1	Input	236,1	7,10	235	7,09	1,1	0,01	Berhasil
2		236,2	7,07	235,4	7,10	0,8	-0,03	Berhasil
3		236,1	7,08	235,2	7,09	0,9	-0,01	Berhasil
4		236,1	7,09	235,2	7,10	0,9	-0,01	Berhasil
5		236,1	7,06	235,3	7,11	0,8	-0,05	Berhasil
6	Σ Input	1180,6	35,4	1176,1	35,49	4,5	0,11	Berhasil
7	Rata2 Input	236,12	7,08	235,22	7,098	0,9	0,022	Berhasil
8	Output	12,15	0,98	12	0,94	0,15	0,04	Berhasil
9		12,15	0,99	11,9	0,95	0,25	0,04	Berhasil
10		12,15	1,00	12,02	0,96	0,13	0,04	Berhasil
11		12,15	1,01	11,88	0,97	0,27	0,04	Berhasil
12		12,15	1,00	12,02	0,97	0,13	0,03	Berhasil
13	Σ Output	60,75	4,98	59,82	4,79	0,93	0,19	Berhasil
14	Rata2 Output	12,15	0,996	11,964	0,958	0,186	0,038	Berhasil

Error pada Power Supply 12V

$$\text{Error V Input} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,9}{236,12} 100\% = 0,38\%$$

$$\text{Error I Input} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,022}{7,08} 100\% = 0,31\%$$

$$\text{Error V output} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,186}{12,15} 100\% = 1,5\%$$

$$\text{Error V output} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,038}{0,996} 100\% = 3,81\%$$

Tabel 2. Tabel uji power supply 5V

No.	Parameter	Tanpa beban		Dengan beban		ΔUkur		Ket
		V	I	V	I	V	I	
1	Input	231,7	7,11	231,2	7,08	0,5	0,02	Berhasil
2		231,6	7,10	230,9	7,09	0,7	0,01	Berhasil
3		231	7,09	230,6	7,08	0,4	0,01	Berhasil
4		230,6	7,09	230,7	7,09	0,1	0	Berhasil
5		231	7,09	230,7	7,08	0,3	0,01	Berhasil
6	Σ Input	1155,9	35,48	1154,1	35,42	2	0,05	Berhasil
7	Rata2 Input	231,18	7,096	230,82	7,084	0,4	0,01	Berhasil
8	Output	5,39	0,33	5,38	0,20	0,01	0,13	Berhasil
9		5,40	0,33	5,38	0,25	0,02	0,07	Berhasil
10		5,40	0,33	5,38	0,28	0,02	0,05	Berhasil
11		5,39	0,33	5,38	0,16	0,01	0,17	Berhasil
12		5,39	0,33	5,38	0,19	0,01	0,14	Berhasil
13	Σ Output	26,97	1,65	26,9	1,08	0,07	0,56	Berhasil
14	Rata2 Output	5,394	0,33	5,38	0,216	0,014	0,112	Berhasil

Error pada Power Supply 5V

$$\text{Error V Input} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,4}{231,18} 100\% = 0,17\%$$

$$\text{Error I Input} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,01}{7,096} 100\% = 0,14\%$$

$$\text{Error V output} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,014}{5,394} 100\% = 0,25\%$$

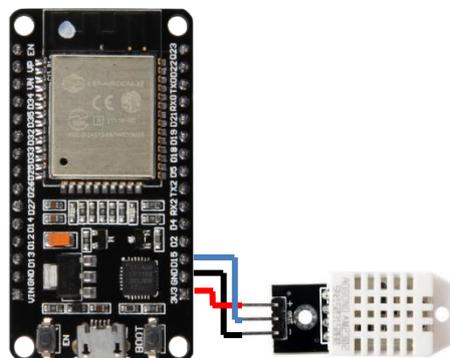
$$\text{Error V output} = \frac{\Sigma \Delta U_{ukur}}{n} 100\% = \frac{0,112}{0,33} 100\% = 0,33\%$$

Dari data hasil uji diatas, didapat nilai error yang sangat kecil, dapat kita ambil kesimpulan bahwa kinerja dari power supply tersebut normal dan dapat bekerja dengan baik.

4.3 Pengujian dan running sensor DHT22

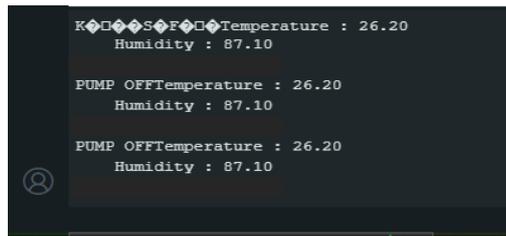
Pada pengujian dan running sensor DHT22 dilakukan beberapa tahapan untuk memastikan akurasi pembacaan sensor sesuai jika dibandingkan dengan alat ukur standart yaitu seperti dibawah ini:

- a. Running sensor DHT22 bertujuan untuk mengetahui apakah sensor DHT22 sudah dapat mengukur suhu dengan baik atau tidak, dan mengetahui apakah sensor DHT22 sudah dapat mengirim data dengan benar ke ESP32 yang kemudian akan ditampilkan pada Blynk.
- b. Skema running sensor DHT22:



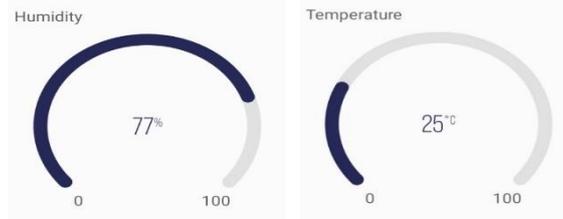
Gambar 6. Skema rangkaian sensor DHT22

- c. Output running sensor DHT22
 Didapat tampilan output Running sensor DHT22 pada “Serial Monitor” Software Arduino IDE seperti dibawah ini.



Gambar 7. Tampilan *output* pada serial monitor Arduino IDE

Tampilan *output running* sensor DHT22 pada aplikasi *Blynk* seperti dibawah ini.



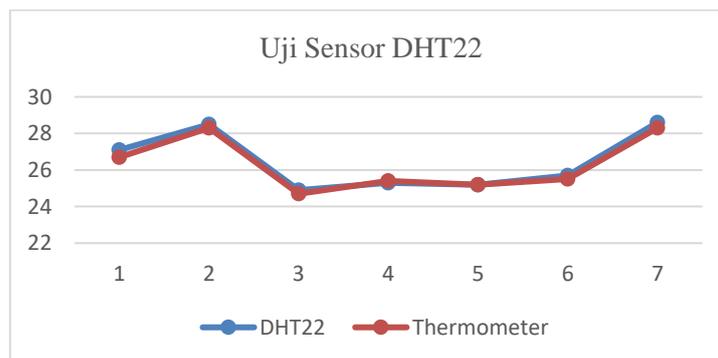
Gambar 8. Tampilan *output* pada Blynk

Gambar-gambar diatas merupakan *output running* sensor DHT22 yang merupakan hasil dari *upload coding* program ke mikrokontrol ESP32, yang mana dapat kita atur sesuai dengan keinginan kita pada program kita di *software* Arduino IDE.

- d. Tabel pengujian akurasi suhu dengan DHT22 dan Thermometer Digital

Tabel 3. Tabel uji banding akurasi suhu

No.	Parameter	Input t (°C)	Output sensor	Thermometer	ΔUkur	Ket
1	DHT22	26,7°C	27,1°C	26,7°C	0,4	Berhasil
2		28,3°C	28,5°C	28,3°C	0,2	Berhasil
3		24,7°C	24,9°C	24,7°C	0,2	Berhasil
4		25,4°C	25,3°C	25,4°C	0,1	Berhasil
5		25,2°C	25,2°C	25,2°C	0	Berhasil
6	Jumlah	130,3	131	130,3	0,9	Berhasil
7	Rata-rata	26,06	26,2	26,06	0,18	Berhasil



Gambar 9. Grafik pengujian sensor DHT22 dan Thermometer Digital

$$\text{Error} = \frac{\sum \Delta U_{kur}}{n} 100\% = \frac{0,18}{26,06} 100\% = 0,6\%$$

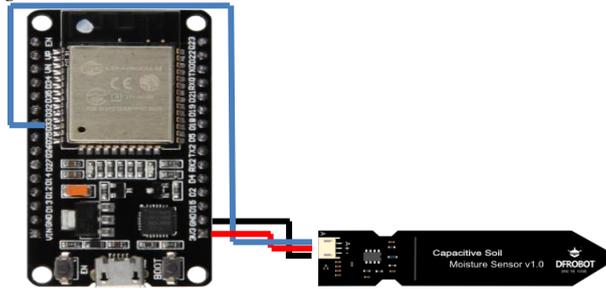
Dari data yang kita peroleh dari pengujian perbandingan nilai suhu DHT22 dan *Thermometer Digital* didapat data seperti tabel dan grafik diatas, diperoleh nilai error yang kecil dengan selisih angka yang tidak begitu jauh atau bisa ditoleransi.

4.4 Pengujian dan *running soil moisture sensor*

Pada pengujian dan *running soil moisture sensor* ini dilakukan beberapa tahapan dan tujuan yaitu seperti dibawah ini:

- a. Tujuan pengujian dan *running soil moisture sensor* ini dilakukan untuk mengetahui apakah sensor bekerja dengan baik dan dapat menampilkan data sesuai yang diinginkan.

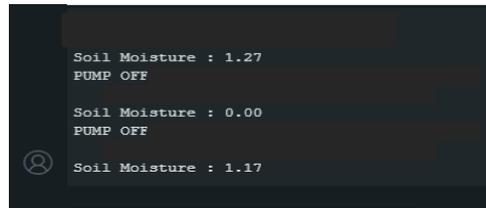
b. Skema *running* Soil Moisture Sensor:



Gambar 10. Skema rangkaian soil moisture sensor

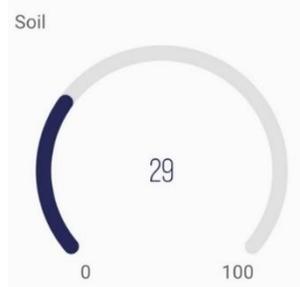
c. *Output Running Soil Moisture Sensor*

Didapat tampilan *output running* Soil Moisture Sensor pada Serial Monitor Software Arduino IDE seperti dibawah ini.



Gambar 11. Tampilan *output* pada serial monitor Arduino IDE

Tampilan *output running* Soil Moisture Sensor pada aplikasi *Blynk* seperti dibawah ini.

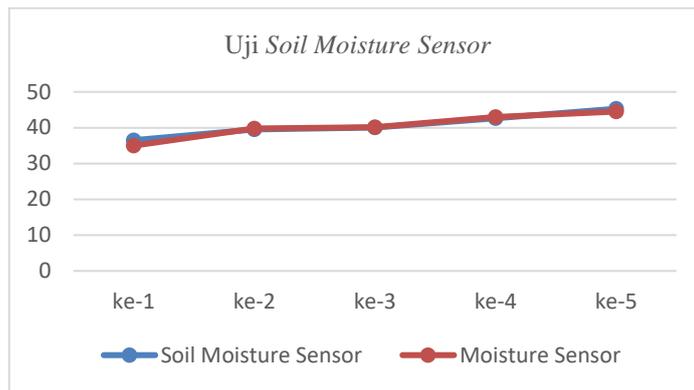


Gambar 12. Tampilan output pada *Blynk*

Gambar-gambar diatas merupakan *output running* Soil Moisture Sensor yang sesuai dengan program *coding* pada *software* Arduino IDE yang sudah kita *upload*.

d. Berikut tabel pengujian akurasi kelembaban tanah dengan *Soil Moisture Sensor* dan *Moisture Sensor*

No.	Parameter	Input %(Hg)	Output Sensor	Moisture Sensor	ΔUkur	Ket
1.	Soil Moisture Sensor	35	36,51	35	1,51	Berhasil
2.		39,8	39,54	39,8	0,26	Berhasil
3.		40,1	40,02	40,1	0,08	Berhasil
4.		43	42,59	43	0,41	Berhasil
5.		44,5	45,29	44,5	0,79	Berhasil
6.	Jumlah	202,4	203,95	202,4	3,05	Berhasil
7.	Rata-rata	40,48	40,79	40,48	0,31	Berhasil



Gambar 13. Grafik hasil pengukuran *soil moisture sensor* dan *moisture sensor*

$$\text{Error} = \frac{\sum \Delta U_{kur}}{n} 100\% = \frac{0,31}{40,48} 100\% = 0,76\%$$

Dari data hasil uji pada tabel dan grafik diatas, didapat hasil yang cukup memuaskan, selisih nilai yang terbaca oleh sensor dengan *moisture sensor* tidak jauh dan nilai error yang didapat juga cukup kecil.

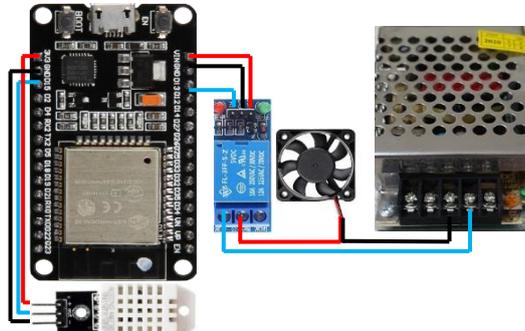
4.5 Pengujian dan *running* mikrokontrol

Pengujian mikrokontrol ini bertujuan untuk mengecek apakah mikrokontrol yang kita gunakan ada error atau semacamnya yang akan mengganggu kinerja sistem secara keseluruhan. Berikut merupakan beberapa komponen yang termasuk dalam pengujian ini.

4.5.1 *Cooling System*

Pengujian *cooling system* ini bertujuan untuk memastikan apakah relay yang terhubung pada *cooling system* dapat bekerja dengan baik dengan beberapa kondisi sesuai batas parameter pada nilai yang terbaca oleh sensor DHT22 yang sudah kita atur, dibawah ini tahap pengujian *cooling system* tersebut:

- a. Skema *running Cooling System*



Gambar 14. Skema rangkaian *Cooling System*

- b. *Output running Cooling System*

Didapat tampilan *output running Cooling System* pada Serial Monitor Software Arduino IDE seperti dibawah ini.

```
FAN ON
PUMP OFF
Temperature : 27.50
Humidity : 75.20
Soil Moisture : 25.67
FAN ON
PUMP OFF
Temperature : 27.50
Humidity : 75.20
Soil Moisture : 29.23
FAN ON
PUMP OFF
Temperature : 27.50
```

Gambar 15. Tampilan *output* pada serial monitor Arduino IDE

Tampilan *output Running Cooling System* pada aplikasi *Blynk* seperti dibawah ini.

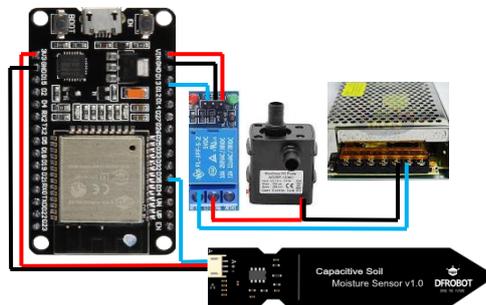


Gambar 16. Tampilan *output* pada *Blynk*

4.5.2 Pompa Air

Pada pengujian ini, dimaksudkan untuk mengetahui apakah pompa dapat berjalan sesuai yang diharapkan. Pompa disini akan diberi beberapa batas set parameter untuk ON dan OFF-nya, yakni sesuai dengan nilai yang terbaca oleh *soil moisture sensor*. Jika nilai sensor mencapai batas bawah kelembaban media tanam, maka pompa akan aktif, dan jika telah mencapai batas atasnya pompa akan mati, begitu juga seterusnya. Berikut beberapa tahap dan hal yang perlu dipersiapkan untuk proses pengujian ini.

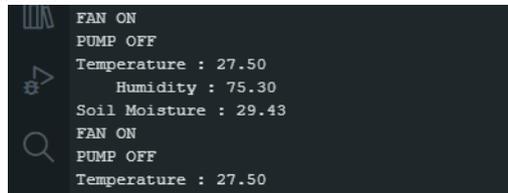
- a. Skema *running* Pompa Air



Gambar 17. Skema rangkaian Pompa Air

- b. *Output running* Pompa Air

Didapat tampilan *output running* Pompa Air pada “Serial Monitor” *Software* Arduino IDE seperti dibawah ini.



Gambar 18. Tampilan *output* pada serial monitor Arduino IDE

Tampilan *output running* Pompa Air pada aplikasi *Blynk* seperti dibawah ini.

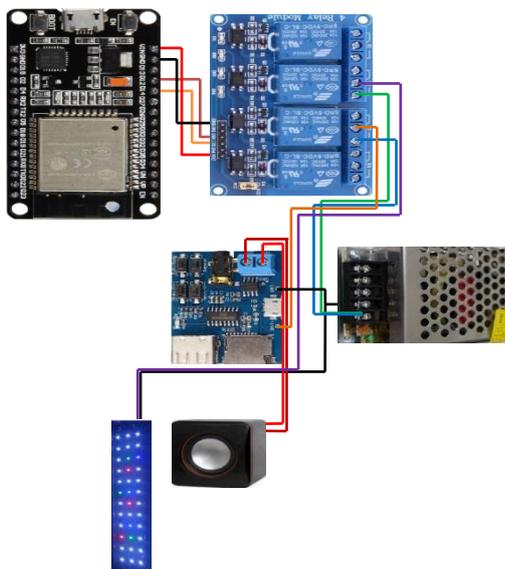


Gambar 19. Tampilan *output* pada *Blynk*

4.5.3 LED dan MP3 Module

Pengujian LED dan MP3 ini bertujuan untuk mengetahui apakah keduanya dapat bekerja dengan sebagaimana mestinya. Kedua aktuator ini bekerja sesuai dengan *schedule* yang kita atur pada “Automation” pada menu *Blynk*. *Scheduling* bertujuan untuk mengatur waktu hidup dan mati pada LED dan *MP3 Module*. Pengujian ini dilakukan untuk mengetahui apakah *scheduling* ini dapat berjalan dengan baik pada waktu yang sudah kita atur.

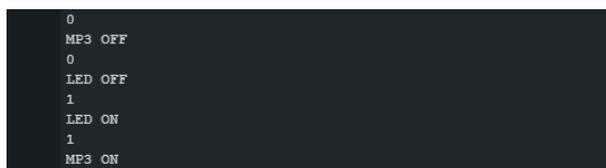
a. Skema *running* LED dan MP3 Module



Gambar 20. Skema rangkaian LED dan MP3 Module

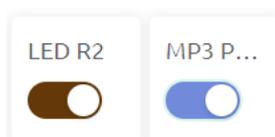
b. *Output running* LED dan MP3 Module

Didapat tampilan *output running* LED dan *MP3 Module* pada “Serial Monitor” *Software* Arduino IDE seperti dibawah ini.



Gambar 21. Tampilan *output* pada serial monitor Arduino IDE

Tampilan *output running* LED dan *MP3 Module* pada aplikasi *Blynk* seperti dibawah ini.



Gambar 22. Tampilan *output* pada Blynk

4.5.4 Hasil pengujian mikrokontrol dan IoT

Dalam pengujian ini mencakup keseluruhan sistem yang digunakan, seberapa bagus performa kinerja sistem dapat bekerja

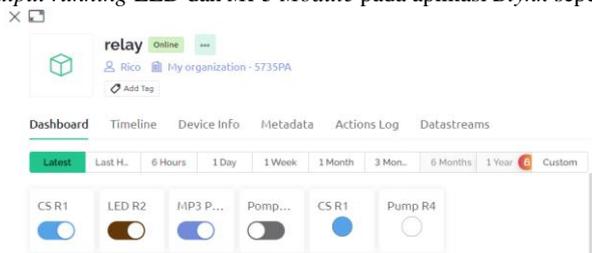
a. *Output running* mikrokontrol dan IoT

Didapat tampilan *output running* LED dan *MP3 Module* pada Serial Monitor *Software* Arduino IDE seperti dibawah ini.



Gambar 23. Tampilan *output* keseluruhan pada serial monitor Arduino IDE

Tampilan *output running* LED dan *MP3 Module* pada aplikasi *Blynk* seperti dibawah ini.



Gambar 24. Tampilan *output* mikrokontrol pada *Blynk*

b. Hasil pengujian mikrokontrol dan *IoT*

Tabel 4. Tabel uji mikrokontrol

No	Parameter	Input					Drv		CS		Output Led		Mp3		Pump		Ket
							ON	OFF	ON	OFF	ON	OFF	ON	OFF	ON	OFF	
1	ESP32	0	0	0	0	0	√		√		√		√		√	Brhsl	
2	ESP32	0	0	0	0	1	√		√		√		√		√	Brhsl	
3	ESP32	0	0	0	1	0	√		√		√	√			√	Brhsl	
4	ESP32	0	0	0	1	1	√		√		√	√			√	Brhsl	
5	ESP32	0	0	1	0	0	√		√	√			√		√	Brhsl	
6	ESP32	0	0	1	0	1	√		√	√	√			√		Brhsl	
7	ESP32	0	0	1	1	0	√		√	√	√		√		√	Brhsl	
8	ESP32	0	0	1	1	1	√		√	√	√		√		√	Brhsl	
9	ESP32	0	1	0	0	0	√	√			√		√		√	Brhsl	
10	ESP32	0	1	0	0	1	√	√			√		√		√	Brhsl	
11	ESP32	0	1	0	1	0	√	√			√	√			√	Brhsl	
12	ESP32	0	1	0	1	1	√	√			√	√			√	Brhsl	
13	ESP32	0	1	1	0	0	√	√		√			√		√	Brhsl	
14	ESP32	0	1	1	0	1	√	√		√			√		√	Brhsl	
15	ESP32	0	1	1	1	0	√	√		√		√			√	Brhsl	
16	ESP32	0	1	1	1	1	√	√		√		√			√	Brhsl	
17	ESP32	1	0	0	0	0	√		√		√		√		√	Brhsl	
18	ESP32	1	0	0	0	1	√		√		√		√		√	Brhsl	
19	ESP32	1	0	0	1	0	√		√		√	√			√	Brhsl	
20	ESP32	1	0	0	1	1	√		√		√	√			√	Brhsl	
21	ESP32	1	0	1	0	0	√		√	√			√		√	Brhsl	
22	ESP32	1	0	1	0	1	√		√	√			√		√	Brhsl	
23	ESP32	1	0	1	1	0	√		√	√		√			√	Brhsl	
24	ESP32	1	0	1	1	1	√		√	√		√			√	Brhsl	
25	ESP32	1	1	0	0	0	√	√			√		√		√	Brhsl	
26	ESP32	1	1	0	0	1	√	√			√		√		√	Brhsl	
27	ESP32	1	1	0	1	0	√	√			√	√			√	Brhsl	
28	ESP32	1	1	0	1	1	√	√			√	√			√	Brhsl	
29	ESP32	1	1	1	0	0	√		√		√		√		√	Brhsl	
30	ESP32	1	1	1	0	1	√		√		√		√		√	Brhsl	
31	ESP32	1	1	1	1	0	√		√		√		√		√	Brhsl	
32	ESP32	1	1	1	1	1	√		√		√		√		√	Brhsl	

Tabel 5. Tabel uji komunikasi *IoT*

No	Parameter	Input t (s)		Output t (s)		ΔUkur t (s)		Ket
		Log in	Log in	Active	Log out	Active	Log out	
1.		30:15.75	30:15.75	30:16.35	30:16.71	00:00.60	00:00.96	Brhsl
2.		32:01.42	32:01.42	32:03.23	32:03.45	00:00.80	00:02.03	Brhsl
3.	CS	34:37.55	34:37.55	34:38.25	34:39.24	00:00.70	00:01.69	Brhsl
4.		37:24.71	37:24.71	37:25.38	37:26.27	00:00.67	00:01.56	Brhsl
5.		39:47.56	39:47.56	39:48.26	39:48.53	00:00.70	00:00.97	Brhsl
6.	Σ Rata2					00:03.47	00:07.21	Brhsl
7.						00:00.69	00:01.44	Brhsl
8.		43:01.95	43:01.95	43:04.65	43:05.03	00:00.56	00:03.08	Brhsl
9.		45:19.86	45:19.86	45:20.42	45:24.44	00:00.56	00:04.58	Brhsl
10.	LED	48:52.52	48:52.52	48:53.56	48:57.65	00:01.04	00:05.13	Brhsl
11.		49:39.78	49:39.78	49:40.21	49:44.98	00:00.43	00:05.20	Brhsl
12.		51:21.34	51:21.34	51:22.07	51:26.83	00:00.73	00:05.49	Brhsl
13.	Σ Rata2					00:03.32	00:23.48	Brhsl
14.						00:00.66	00:04.69	Brhsl
15.		54:47.40	54:47.40	54:47.92	54:49.51	00:00.52	00:02.11	Brhsl
16.		55:57.64	55:57.64	55:57.95	55:59.81	00:00.31	00:02.17	Brhsl
17.	Mp3	57:20.31	57:20.31	57:20.63	57:21.94	00:00.32	00:01.63	Brhsl
18.		58:46.66	58:46.66	58:47.31	58:47.47	00:00.65	00:00.81	Brhsl
19.		59:53.47	59:53.47	59:54.60	59:54.89	00:01.20	00:01.42	Brhsl
20.	Σ Rata2					00:03.00	00:08.14	Brhsl
22.						00:00.60	00:01.62	Brhsl
21.		01:06.46	01:06.46	01:06.73	01:07.08	00:00.27	00:00.62	Brhsl
22.		03:23.56	03:23.56	03:24.64	03:24.86	00:01.08	00:01.30	Brhsl
22.	Pump	04:53.13	04:53.13	04:54.98	04:55.37	00:00.85	00:02.24	Brhsl
23.		06:52.67	06:52.67	06:53.18	06:55.45	00:00.51	00:00.78	Brhsl
24.		08:01.16	08:01.16	08:03.00	08:03.50	00:00.84	00:02.34	Brhsl
25.	Σ Rata2					00:03.55	00:07.28	Brhsl
26.						00:00.71	00:01.45	Brhsl

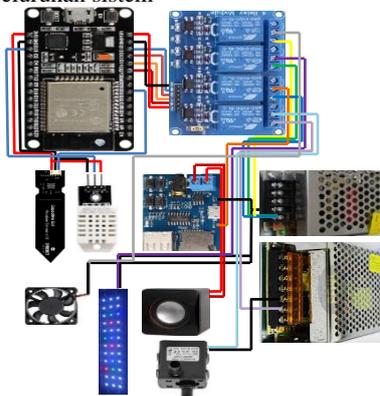
Pengujian diatas tidak mencantumkan jam, hanya mencantumkan menit, detik dan mili detik, dan dilakukan mulai jam 12.30 secara bergiliran. Dari tabel diatas dapat disimpulkan bahwa respon waktu komunikasi *IoT* sudah tergolong baik. Penggunaan cocopeat juga memberi pengaruh yang baik pada sistem tersebut, yang mana daya simpan airnya yang besar dan juga tahan lama, jadi pompa tidak mudah menyala dalam waktu yang singkat.

Kelemahan penggunaan *Automation* pada aplikasi *Blynk* yaitu jika suatu saat koneksi sistem ke internet terputus akan mempengaruhi pada kinerja sistem seperti halnya salah satu aktuator yang diatur menyala pada waktu tertentu namun pada waktu menyala koneksi internet tiba-tiba terputus, maka aktuator tersebut akan melewati *schedule* tersebut dan tidak aktif saat sistem kembali terkoneksi ke internet. Begitupun saat koneksi terputus juga salah satu aktuator yang seharusnya sudah menyala tiba-tiba tidak merespon batas set parameternya, seperti *cooling system* yang tidak menyala saat suhu mencapai titik 24°C tanggal 23 Juli pada 01.30AM.

4.6 Running keseluruhan sistem

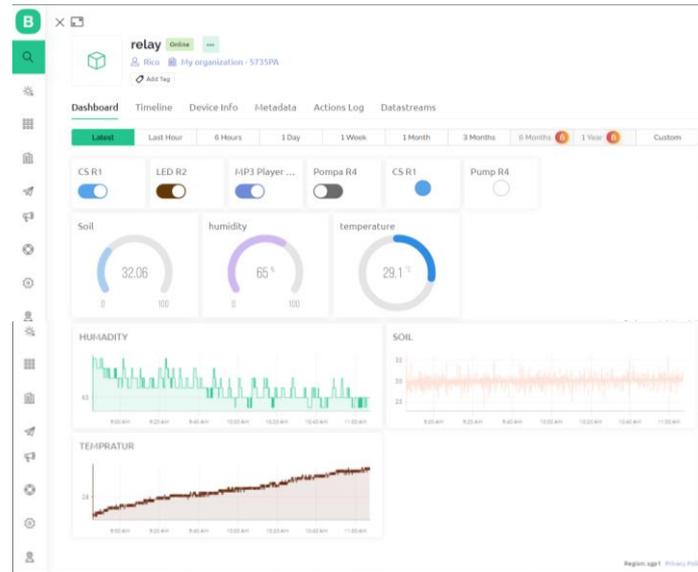
Pada pengujian kali ini bertujuan untuk mengetahui apakah sistem secara keseluruhan dapat berjalan dengan baik atau tidak, mulai dari pembacaan sensor suhu, kontrol manual pada aktuator, begitupula otomatisasi pada aktuator. Berikut merupakan beberapa hal yang perlu dipersiapkan dalam pengujian ini.

a. Skema rangkaian *running* keseluruhan sistem



Gambar 25. Skema rangkaian keseluruhan sistem

- b. Tampilan *output* dan pengujian keseluruhan sistem pada *Blynk*



Gambar 26. Tampilan *output* keseluruhan sistem pada *Blynk*

5. KESIMPULAN

Berdasarkan hasil penelitian diatas, maka dapat diambil kesimpulan sebagai berikut:

1. Dalam perancangan sistem kendali otomatis sekaligus monitoring sensor pada Smart Farming Microgreen, kita dapat menyiapkan berbagai referensi yang sekiranya kita butuhkan dari segi rangkaian, coding, maupun desain pada *Blynk*. Siapkan semua komponen yang kita butuhkan terlebih dahulu, dan buatlah coding pada Software Arduino IDE mengikuti berbagai referensi yang ada dan tidak lupa menambahkan set batas parameter pada sensor menggunakan logika “if”. Lalu desain laman *Blynk* sesuai dengan yang kita inginkan.
2. Akuisisi Data berbasis IOT yakni menggunakan salah satu widget pada Aplikasi *Blynk* yang disebut “*SuperChart*”, widget tersebut akan terhubung ke *Blynk Cloud* yang berfungsi untuk menyimpan data yang diperoleh oleh sensor yang mana dapat kita lihat secara live maupun atur *time range*-nya sesuai keinginan kita.
3. Untuk pengujian data yang diperoleh kita dapat mengeceknya secara langsung pada diagram lingkaran nilai suhunya karena akan terhubung langsung pada *SuperChart*, kita juga dapat melakukan akuisisi data pada aktuator-aktuator yang kita gunakan dengan *download report* pada opsi di device *Blynk* yang kita gunakan. Data yang kita dapat akan berbentuk Excel dengan *Time Range* pengambilan data yang dapat kita atur dalam menu *DataStream* -> *Advance Setting*.

Untuk pengembangan sistem ke depannya kita dapat menghemat biaya yang digunakan untuk *upgrade* aplikasi *Blynk* kita dengan menambahkan RTC untuk membuat *Automation* pada aktuator dan Akuisisi Datanya, karena seperti yang kita ketahui bahwa kita harus berlangganan di *Blynk Plus* untuk dapat menggunakan widget dan *download report* pada aplikasi *Blynk*.

Juga *sprayer/sprinkler* yang kita pakai dapat diganti dengan *Mist Maker* yang dapat kita tempatkan pada wadah kecil dibagian bawah *microgreen*. Yang mana di tempat tersebut merupakan lokasi pembuangan air dari *peltier* yang selalu meneteskan air es dalam beberapa saat ketika menyala.

REFERENSI

- [1] V. Febriani, E. Nasrika, T. Munasari, Y. Permatasari, and T. Widiatningrum, “Analisis Produksi Microgreens Brassica oleracea Berinovasi Urban Gardening Untuk Peningkatan Mutu Pangan Nasional,” *Journal of Creativity Student*, vol. 2, no. 2, pp. 58–66, 2017.
- [2] C. F. Weber, “Broccoli microgreens: A mineral-rich crop that can diversify food systems,” *Front Nutr*, vol. 4, p. 7, 2017.
- [3] R. H. Hilmy, R. Susana, and F. Hadiatna, “Rancang bangun Smart Grow Box hidroponik untuk pertumbuhan tanaman microgreen berbasis Internet of Things,” *Power Elektronik: Jurnal Orang Elektro*, vol. 10, no. 2, pp. 41–47, 2021.
- [4] H. Valupi, “Pertumbuhan Dan Hasil Microgreens Beberapa Varietas Pakcoy (Brassica rapa. L) Pada Media Tanam Yang Berbeda,” in *Prosiding Seminar Nasional Pertanian, 2022*, pp. 1–13.
- [5] C. Kaiser and M. Ernst, “babyveggies,” *Center for Crop Diversification*, vol. 1, no. 1, pp. 1–3, 2017.

- [6] R. Darmawan *et al.*, “Pendampingan Pembuatan Pupuk Cair Berbasis Organik dan Aplikasinya Terhadap Tanaman Uji Secara Hidroponik,” *Sewagati*, vol. 6, no. 2, pp. 136–146, 2022.
- [7] R. Idrus, “PENGARUH DERAJAT KEMIRINGAN PIPA HIDROPONIK PORTABLE TERHADAP PERTUMBUHAN TANAMAN PAKCOY,” Thesis, PROGRAM STUDI TEKNIK PERTANIAN JURUSAN TEKNOLOGI PERTANIAN, Mataram, 2021.
- [8] D. I. Yama and H. Kartiko, “Pertumbuhan dan kandungan klorofil pakcoy (*Brassica rappa* L) pada beberapa konsentrasi AB Mix dengan sistem wick,” *J Teknol*, vol. 12, no. 1, pp. 21–30, 2020.
- [9] R. Megasepta, “PENGARUH SUHU PIROLISIS BIOCHAR TONGKOL JAGUNG DAN DOSIS PUPUK UREA TERHADAP PERTUMBUHAN DAN HASIL TANAMAN PAKCOY (*Brassica rapa* L.)” Thesis, JURUSAN TEKNIK PERTANIAN FAKULTAS PERTANIAN, Lampung, 2021.
- [10] W. Noor, “RESPONS TANAMAN SAWI (*Brassica juncea* L.) TERHADAP JENIS KOMPOS CAIR DAN DOSIS PUPUK NITROGEN,” Thesis, EKOLAH TINGGI ILMU PERTANIAN, Metro, 2019.
- [11] N. Khoiriyah and A. Nugroho, “Pengaruh konsentrasi dan frekuensi aplikasi pupuk organik cair pada tanaman pakcoy (*Brassica rapa* L.) varietas flamingo,” *Jurnal Protan*, vol. 6, no. 8, pp. 1875–1883, 2018.
- [12] M. R. Fahlevi and H. Gunawan, “Perancangan Sistem Pendeteksi Banjir Berbasis Internet of Things,” *It (Informatic Technique) Journal*, vol. 8, no. 1, pp. 23–29, 2021.
- [13] W. Wilianto and A. Kurniawan, “Sejarah, cara kerja dan manfaat internet of things,” *Matrix: Jurnal Manajemen Teknologi Dan Informatika*, vol. 8, no. 2, pp. 36–41, 2018.
- [14] A. A. Wicaksana, A. T. Mulyani, N. S. M. Sukmah, A. A. Rahma, Y. T. B. Tarigan, and D. S. Y. Kartika, “Penerapan Teknologi Tepat Guna Penyiraman Otomatis Menggunakan Capacitive Soil Moisture Sensor pada Taman Tanaman Obat Keluarga (TOGA) Desa Gedangan,” *JURNAL PENGABDIAN MASYARAKAT INDONESIA*, vol. 2, no. 2, pp. 150–163, 2023.
- [15] A. H. Saptadi, “Perbandingan Akurasi Pengukuran Suhu dan Kelembaban Antara Sensor DHT11 dan DHT22,” *Jurnal Infotel*, vol. 6, no. 2, pp. 49–56, 2014.
- [16] T. P. Satya, U. Y. Oktawati, I. Fahrurrozi, and H. Prisyanti, “Analisis Akurasi Sistem sensor DHT22 berbasis Arduino terhadap Thermohygrometer Standar,” *Jurnal Fisika dan aplikasinya*, vol. 16, no. 1, pp. 40–45, 2020.