



**SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI**

**PERANCANGAN SISTEM KONTROL DAN *MONITORING*  
TANAMAN UNTUK *SMART GARDEN* BERBASIS *INTERNET*  
*OF THINGS***

**TUGAS AKHIR**

**WAHYU FIRMANSYAH  
0110220139**

**PROGRAM STUDI TEKNIK INFORMATIKA  
DEPOK  
AGUSTUS 2024**



**STT TERPADU  
NURUL FIKRI**

**SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI**

**PERANCANGAN SISTEM KONTROL DAN *MONITORING*  
TANAMAN UNTUK *SMART GARDEN* BERBASIS *INTERNET*  
*OF THINGS***

**TUGAS AKHIR**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana**

**STT NF**  
WAHYU FIRMANSYAH  
0110220139

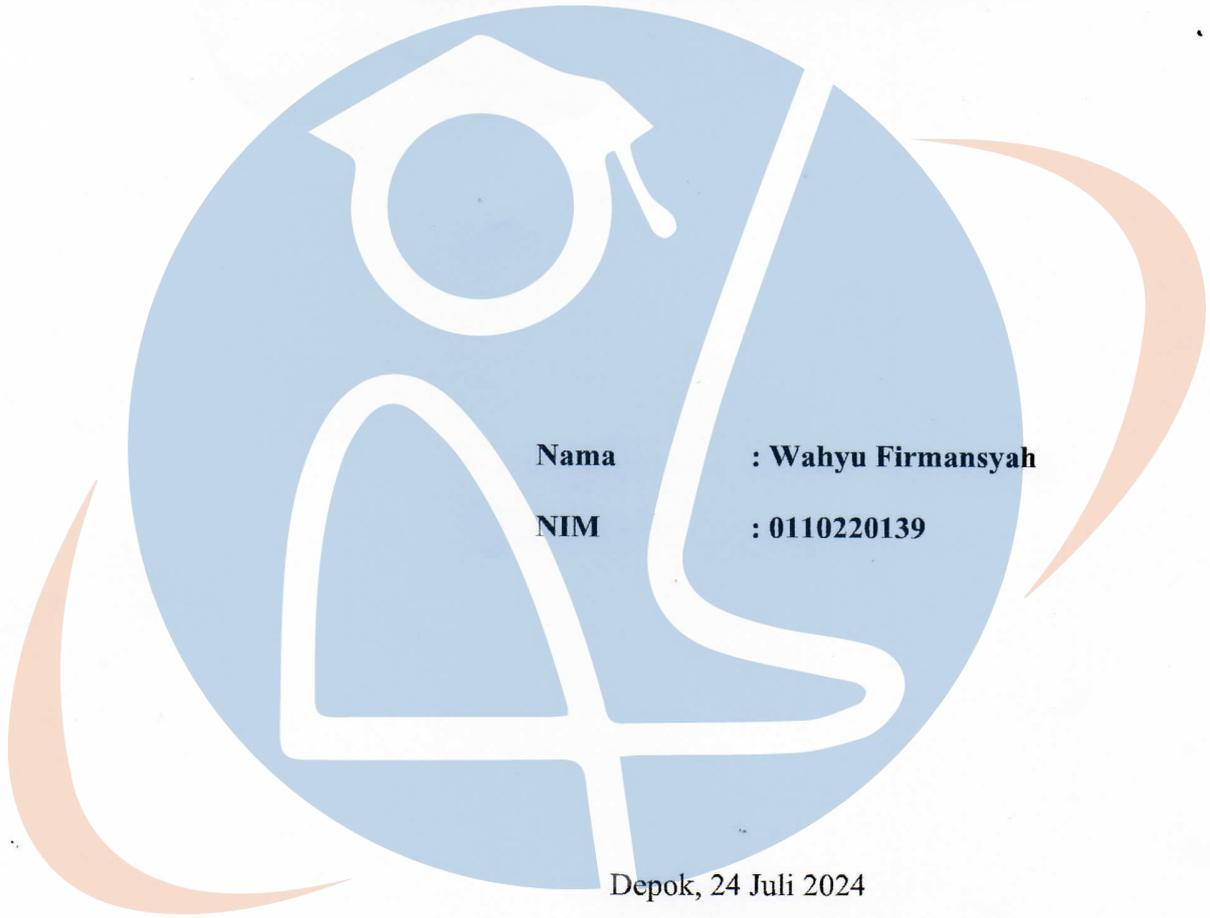
**PROGRAM STUDI TEKNIK INFORMATIKA**

**DEPOK**

**AGUSTUS 2024**

**HALAMAN PERNYATAAN ORISINALITAS**

**Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.**



**Nama : Wahyu Firmansyah**

**NIM : 0110220139**

Depok, 24 Juli 2024

Tanda Tangan

STT - NF

A handwritten signature in black ink, appearing to read 'Wahyu Firmansyah', is written over the 'NF' part of the 'STT - NF' watermark.

Wahyu Firmansyah

## HALAMAN PENGESAHAN

Tugas Akhir ini diajukan oleh :

Nama : Wahyu Firmansyah

NIM : 0110220139

Program Studi : Teknik Informatika

Judul Skripsi : Perancangan Sistem Kontrol dan *Monitoring* Tanaman Untuk  
*Smart Garden* Berbasis *Internet of Things*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana komputer pada Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

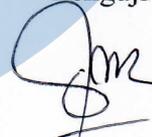
### DEWAN PENGUJI

Pembimbing



(Dr. Lukman Rosyidi, M.T., M.M.)

Penguji



(Dr. Sirojul Munir S.Si., M.Kom)

# STT - NF

Ditetapkan di : Depok

Tanggal : 24 Juli 2024

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Tugas Akhir ini. Penulisan Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Teknik Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT.
2. Orang tua dan semua anggota keluarga yang telah memberikan dorongan baik secara moril maupun materil dalam penyelesaian tugas ini.
3. Bapak Dr. Lukman Rosyidi selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri dan selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
4. Bapak Dr. Sirojul Munir S.Si., M.Kom. selaku Dosen Penguji yang telah memberikan masukan dan saran untuk penyempurnaan Tugas Akhir ini.
5. Ibu Tifani Nabarian, S.Kom., M.T.I. selaku Ketua Program Studi Teknik Informatika dan juga selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama perkuliahan di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
6. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.

Dalam penulisan ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini, dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Akhir kata, penulis berharap Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 24 Juli 2024



Wahyu Firmansyah



STT - NF

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

---

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : Wahyu Firmansyah

NIM : 0110220139

Program Studi : Teknik Informatika

Jenis karya : Tugas Akhir

demikian demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty - Free Right)** atas karya ilmiah saya yang berjudul :

Perancangan Sistem Kontrol dan *Monitoring* Tanaman untuk *Smart Garden* Berbasis *Internet of Things*

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 24 Juli 2024

STT - NF Yang Menyatakan



(Wahyu Firmansyah)

## ABSTRAK

Nama : Wahyu Firmansyah  
NIM : 0110220139  
Program Studi : Teknik Informatika  
Judul : Perancangan Sistem Kontrol dan Monitoring untuk Smart Garden Berbasis Internet of Things

Berkebun telah menjadi tren populer di Indonesia, tidak hanya sebagai hobi tetapi juga sebagai gaya hidup yang lebih sehat. Namun, berkebun secara tradisional seringkali membutuhkan waktu dan tenaga yang besar serta lahan yang luas, yang menjadi penghalang bagi banyak orang untuk bercocok tanam. Penelitian ini bertujuan untuk merancang dan mengembangkan sistem penyiraman otomatis untuk berkebun di halaman rumah yang efisien dan mudah digunakan dengan memanfaatkan teknologi IoT. Metode penelitian yang digunakan adalah Research and Development (R&D), yaitu sebuah metode penelitian yang bertujuan untuk menghasilkan suatu produk tertentu dan menguji efektivitas produk tersebut. Dalam penelitian ini, sistem penyiraman tanaman otomatis dirancang dan dibangun menggunakan ESP8266, sensor DHT11 untuk mengukur suhu dan kelembapan lingkungan, serta sensor soil moisture untuk mengukur kelembapan tanah pada tanaman. Perangkat ini dapat beroperasi dalam dua mode, yaitu mode otomatis dan mode manual. Dalam mode manual, pengguna dapat menggunakan tombol yang tersedia di aplikasi *Blynk* untuk menyalakan atau mematikan pompa sesuai kebutuhan. Hasil dari penelitian ini menunjukkan bahwa sistem penyiraman otomatis berbasis IoT mampu berfungsi dengan baik dalam menjaga kelembapan tanah pada tingkat yang ideal untuk pertumbuhan tanaman. Berdasarkan hasil pengujian, perancangan sistem kontrol dan monitoring untuk *Smart Garden* berbasis *Internet of Things* dapat berfungsi dengan baik sesuai dengan rancangan.

Kata kunci : IoT, *Smart Garden*, penyiraman otomatis ESP8266.

## **ABSTRACT**

*Name : Wahyu Firmansyah*  
*NIM : 0110220139*  
*Study Program : Informatics Engineering*  
*Title : Design of Control and Monitoring System for Smart Garden Based on Internet of Things*

*Gardening has become a popular trend in Indonesia, not only as a hobby but also as a healthier lifestyle. However, traditional gardening often requires a large amount of time and energy as well as a large area of land, which is a barrier for many people to grow crops. This research aims to design and develop an automatic watering system for gardening in the yard that is efficient and easy to use by utilizing IoT technology. The research method used is Research and Development (R&D), which is a research method that aims to produce a certain product and test the effectiveness of the product. In this research, an automatic plant watering system is designed and built using ESP8266, DHT11 sensors to measure the temperature and humidity of the environment, and soil moisture sensors to measure soil moisture in plants. This device can operate in two modes, namely automatic mode and manual mode. In manual mode, users can use the buttons available in the Blynk app to turn the pump on or off as needed. The results of this research show that the IoT-based automatic watering system is able to function well in maintaining soil moisture at an ideal level for plant growth. Based on the test results, the design of the control and monitoring system for smart garden based on the Internet of Things can function properly in accordance with the design.*

*Key words : IoT, Smart Garden, ESP8266 automatic watering.*

STT - NF

## DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS.....	iii
HALAMAN PENGESAHAN.....	iv
KATA PENGANTAR .....	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	vii
ABSTRAK .....	viii
<i>ABSTRACT</i> .....	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL.....	xiv
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan dan Manfaat Penelitian.....	2
1.3.1 Tujuan Penelitian .....	2
1.3.2 Manfaat Penelitian .....	3
1.4 Batasan Masalah.....	3
1.5 Sistematika Penulisan.....	3
<b>BAB 2 KAJIAN LITERATUR.....</b>	<b>5</b>
2.1 Smart Garden.....	5
2.2 Internet of Things .....	5
2.3 Mikrokontroler ESP8266 NodeMCU.....	6
2.4 Sensor Soil Moisture .....	7
2.5 Sensor DHT11 .....	8
2.6 Liquid Crystal Display 16x2 .....	9

2.7	Komponen Elektronik Relay .....	10
2.8	Aplikasi <i>Blynk</i> .....	11
2.9	Software Arduino IDE.....	12
2.10	Penelitian Terkait.....	13
2.11	Posisi Penelitian.....	14
<b>BAB 3</b>	<b>METODOLOGI PENELITIAN .....</b>	<b>16</b>
3.1	Tahapan Penelitian .....	16
3.2	Rancangan Penelitian .....	18
3.2.1	Jenis Penelitian.....	18
3.2.2	Metode Analisis Data.....	18
3.2.3	Metode Pengumpulan Data.....	19
3.2.4	Metode Pengujian.....	20
3.2.5	Metode Implementasi dan Evaluasi .....	21
3.2.6	Lingkungan Pengembangan.....	22
<b>BAB 4</b>	<b>IMPLEMENTASI DAN EVALUASI.....</b>	<b>25</b>
4.1	Perancangan Sistem.....	25
4.1.1	Arsitektur Sistem.....	25
4.1.2	Rangkaian Sistem.....	26
4.1.3	Flowchart Program.....	28
4.2	Implementasi Rancangan Sistem.....	30
4.2.1	Hasil Prototype Sistem.....	30
4.2.2	Kode Program .....	32
4.3	Pengujian Sistem .....	39
4.3.1	Pengujian Hasil Pengukuran .....	41
4.3.2	Pengujian Pengiriman Data.....	46
4.4	Evaluasi Hasil Pengujian.....	48

4.4.1	Evaluasi Hasil Pengukuran .....	48
4.4.2	Analisis Evaluasi Pengiriman Data.....	50
BAB 5	PENUTUP .....	52
5.1	Kesimpulan.....	52
5.2	Saran.....	53
DAFTAR PUSTAKA	.....	54



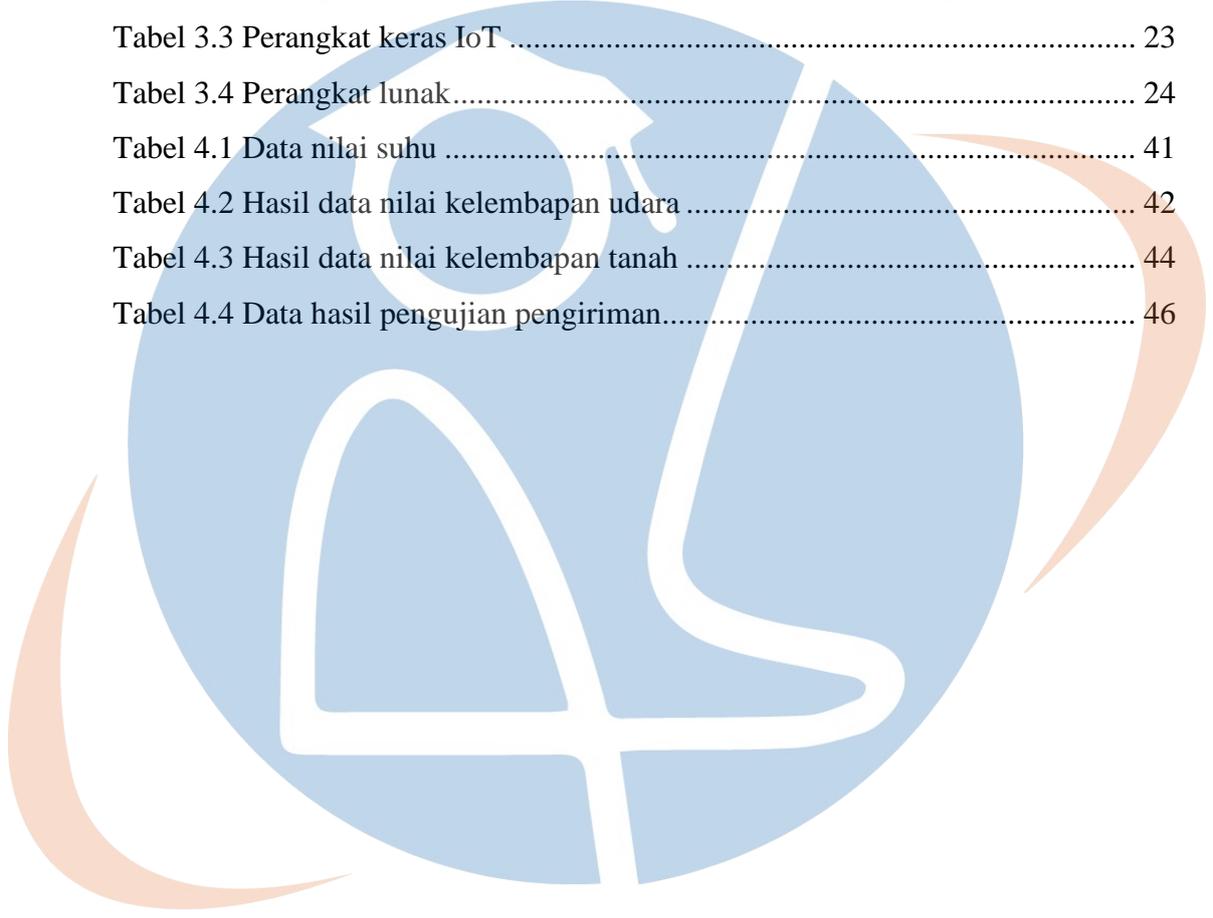
STT - NF

## DAFTAR GAMBAR

Gambar 2.1 Mikrokontroler ESP8266 NodeMCU .....	6
Gambar 2.2 Sensor Soil Moisture v1.2 .....	7
Gambar 2.3 Sensor DHT11 .....	8
Gambar 2.4 Liquid Crystal Display (LCD) dan I2C .....	9
Gambar 2.5 Relay 1 Channel .....	10
Gambar 2.6 Tampilan halaman depan aplikasi Blynk .....	11
Gambar 2.7 Tampilan antarmuka software Arduino IDE .....	12
Gambar 3.1 Diagram alir tahapan penelitian .....	16
Gambar 4.1 Arsitektur sistem .....	25
Gambar 4.2 Rangkaian sistem.....	26
Gambar 4.3 Flowchart Program.....	29
Gambar 4.4 Hasil prototype sistem.....	31
Gambar 4.5 Tampilan sistem keseluruhan .....	31
Gambar 4.6 Kode program Arduino IDE (1).....	32
Gambar 4.7 Kode program Arduino IDE (2).....	32
Gambar 4.8 Kode program Arduino IDE (3).....	33
Gambar 4.9 Kode program Arduino IDE (4).....	34
Gambar 4.10 Kode program Arduino IDE (5).....	35
Gambar 4.11 Kode program Arduino IDE (6).....	35
Gambar 4.12 Kode program Arduino IDE (7).....	36
Gambar 4.13 Kode program Arduino IDE (8).....	37
Gambar 4.14 Kode program Arduino IDE (9).....	38
Gambar 4.15 Dashboard web Blynk.....	40
Gambar 4.16 Dashboard pada aplikasi Blynk .....	40
Gambar 4.17 Grafik nilai suhu.....	42
Gambar 4.18 Grafik nilai kelembapan udara .....	44
Gambar 4.19 Grafik data nilai kelembapan tanah.....	45

## DAFTAR TABEL

Tabel 2.1 Penelitian Terkait .....	13
Tabel 2.2 Posisi Penelitian .....	14
Tabel 3.1 Perangkat keras komputer untuk pengembangan .....	22
Tabel 3.2 Perangkat keras <i>Smartphone Android</i> untuk <i>monitoring</i> .....	23
Tabel 3.3 Perangkat keras IoT .....	23
Tabel 3.4 Perangkat lunak .....	24
Tabel 4.1 Data nilai suhu .....	41
Tabel 4.2 Hasil data nilai kelembapan udara .....	42
Tabel 4.3 Hasil data nilai kelembapan tanah .....	44
Tabel 4.4 Data hasil pengujian pengiriman .....	46



STT - NF

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Berkebun kini telah berkembang menjadi tren populer di Indonesia, tidak hanya sebagai hobi, tetapi juga sebagai gaya hidup yang lebih sehat. Kegiatan ini mencakup berbagai metode, seperti penggunaan media hidroponik, pembuatan taman vertikal, dan perawatan tanaman hias. Dengan berkebun, tidak hanya dapat memproduksi makanan segar sendiri, tetapi juga berkontribusi pada pengurangan jejak karbon dan penciptaan lingkungan yang lebih hijau dan nyaman untuk dihuni.

Berkebun secara tradisional seringkali membutuhkan waktu dan tenaga yang besar, serta memerlukan lahan yang terbuka dan luas. Keterbatasan lahan sering menjadi penghalang bagi banyak orang untuk bercocok tanam, sehingga mereka tidak dapat membudidayakan tanaman dengan optimal [1]. Selain itu, tidak semua pemilik tanaman mampu melakukan perawatan secara rutin, karena berbagai alasan yang mungkin menghalangi mereka untuk selalu berada di dekat tanaman mereka, membuat mereka tidak dapat menyiram atau merawat tanaman dengan baik [2].

Di era digital yang serba canggih ini, smartphone tidak hanya sebagai alat komunikasi, tetapi juga telah menjadi alat bantu yang efektif dalam berkebun melalui teknologi *Internet of Things* (IoT). IoT memungkinkan untuk mengendalikan objek dari jarak jauh via internet, termasuk dalam mengoptimalkan penggunaan peralatan listrik dan elektronik. Dengan IoT, berkebun menjadi lebih mudah dan efisien, terutama dalam hal penyiraman tanaman. Tanaman memerlukan air untuk tumbuh dan melakukan fotosintesis, serta faktor-faktor lain seperti suhu dan kelembaban tanah. Tanpa pemenuhan kebutuhan ini, tanaman bisa layu dan mati.

Saat ini kenyamanan dan keefisienan waktu serta tenaga menjadi pertimbangan utama dalam melakukan aktivitas sehari - hari. Oleh karena itu, salah satu untuk mengatasi kesulitan yang dipaparkan di atas adalah dengan menggunakan metode penyiraman otomatis yang memanfaatkan *Internet of Things*.

Dengan IoT, tanaman bisa dikontrol secara otomatis dan *monitoring* secara jarak jauh dengan menggunakan *handphone*. Penggunaan IoT dalam berkebun juga membantu dalam menghemat air, karena penyiraman dapat dilakukan secara lebih terukur dan efisien. Ini tidak hanya menguntungkan bagi pengguna tetapi juga bagi lingkungan, karena mengurangi pemborosan sumber daya alam. Dengan demikian, teknologi penyiraman otomatis yang memanfaatkan IoT menjadi pilihan yang cerdas, yang memungkinkan bisa menjaga tanaman tetap hidup dan subur dengan usaha yang minimal.

## **1.2 Rumusan Masalah**

Dari latar belakang diatas permasalahan yang akan dibahas pada penelitian ini adalah :

1. Bagaimana merancang sistem penyiraman otomatis untuk berkebun di halaman rumah?
2. Bagaimana sistem smart garden mendeteksi tingkat kelembapan tanah dan memutuskan kapan perlu melakukan penyiraman untuk menjaga tanah pada tingkat kelembapan yang optimal?

## **1.3 Tujuan dan Manfaat Penelitian**

### **1.3.1 Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah :

1. Merancang dan mengembangkan sistem penyiraman otomatis untuk berkebun di halaman rumah dengan memanfaatkan teknologi *Internet of Things* (IoT) yang efisien dan mudah digunakan oleh pemilik rumah.
2. Membuat sebuah algoritma atau logika keputusan yang menerima data dari sensor kelembapan tanah, dan menggunakan informasi tersebut untuk menentukan waktu dan jumlah penyiraman yang tepat, dengan tujuan menjaga kelembapan tanah pada tingkat yang ideal untuk pertumbuhan tanaman.

### 1.3.2 Manfaat Penelitian

Adapun Manfaat dari penelitian ini adalah :

1. Hal ini bisa menjadi jawaban atas permasalahan terbatasnya ketersediaan lahan milik masyarakat perkotaan untuk melakukan kegiatan berkebun.
2. Membantu meningkatkan efisiensi berkebun untuk masyarakat yang memiliki keterbatasan waktu.

### 1.4 Batasan Masalah

Mengingat banyaknya permasalahan yang ada dalam penulisan tugas akhir ini, penulis hanya akan membahas :

1. Perancangan dan pembuatan alat ini menggunakan *mikrokontroler ESP8266 NodeMCU*.
2. Sistem ini dibatasi hanya untuk skala kecil seperti kebun rumah tangga dan bukan untuk skala komersial atau skala industri.
3. Fokus utama sistem ini adalah untuk penyiraman otomatis dan *monitoring* kondisi tanaman seperti kelembapan tanah dan pengukuran suhu.
4. Penelitian ini dilakukan di lokasi rumah penulis yaitu di Sawangan Depok.

### 1.5 Sistematika Penulisan

Untuk memudahkan penyusunan tugas akhir, penulis memberikan sistematika sebagai berikut :

## BAB I PENDAHULUAN

Bab I dalam penelitian ini menguraikan tentang latar belakang yang menjadi dasar dari masalah penelitian yang berkaitan dengan tantangan memiliki ruang yang terbatas untuk aktivitas berkebun. Dari situasi ini, akan diidentifikasi rumusan masalah tentang pertanyaan-pertanyaan penelitian yang spesifik muncul dari konteks latar belakang tersebut. Untuk memastikan penelitian ini tetap terarah, akan ditetapkan batasan-batasan masalah yang akan memfokuskan pembahasan pada aspek-aspek yang relevan dengan topik penelitian. Tujuan utama dari penelitian ini adalah untuk

memberikan jawaban atas pertanyaan-pertanyaan dari rumusan masalah dan untuk menghasilkan manfaat yang dapat menjadi solusi praktis dalam mengatasi masalah ruang terbatas untuk berkebun. Bab ini juga akan menyajikan sistematika penulisan yang terstruktur untuk memandu penyusunan laporan penelitian secara sistematis.

## BAB II KAJIAN LITERATUR

Bab II berisi teori-teori yang mendasari penelitian ini, yaitu tentang *ESP8266 MCU* dan sensor-sensor yang mendukung untuk penelitian ini. Serta membandingkan penelitian yang sudah dilakukan oleh penulis dengan penelitian lain sesuai dengan konteks yang dibahas.

## BAB III METODE PENELITIAN

Bab III berisi tentang penjelasan mengenai jenis penelitian yang dilakukan oleh penulis, metode analisis data untuk mengetahui analisis apa yang akan digunakan, selanjutnya metode pengumpulan data yang digunakan oleh peneliti untuk mengumpulkan data, tahapan penelitian mencakup langkah-langkah penelitian yang digambarkan dalam diagram alir dan metode pengujian.

## BAB IV IMPLEMENTASI DAN EVALUASI

Bab IV menguraikan rencana dan pelaksanaan penelitian. Ini termasuk detail tentang bagaimana sistem kontrol dan *monitoring* tanaman *Smart Garden* dengan teknologi IoT, yang memanfaatkan aplikasi *Blynk* untuk pemantauan, akan dibangun dan diuji. Bab ini juga akan menggambarkan pengumpulan data, serta metode analisis dan evaluasi data tersebut. Dari analisis ini, penulis akan menentukan seberapa sukses penelitian berdasarkan hasil yang diperoleh.

## BAB V KESIMPULAN DAN SARAN

Bab ini merupakan tahap akhir dari penulisan tugas akhir yang berisi kesimpulan dan saran.

## BAB 2 KAJIAN LITERATUR

### 2.1 Smart Garden

*Smart garden* merupakan salah satu aplikasi dari *Internet of Things* (IoT) yang dapat diterapkan di area terbuka seperti taman kota atau ruang hijau publik, di mana teknologi digunakan untuk meningkatkan manajemen dan pemeliharaan [3]. Sistem ini memungkinkan pengelola untuk memonitor kondisi tanaman secara *real-time* selama periode waktu tertentu, sehingga memudahkan dalam menentukan langkah pemeliharaan yang perlu diambil.

*Smart garden* dirancang sebagai sistem otomatis untuk penyiraman tanaman yang menggunakan *mikrokontroler ESP8266*, bersama dengan sensor kelembapan tanah dan sensor suhu. Untuk menciptakan *smart garden* ini, beberapa komponen seperti pompa air, modul *mikrokontroler ESP8266*, sensor kelembapan, sensor suhu, modul *relay*, dan layar LCD 16x2 digunakan dalam perakitan sistem.

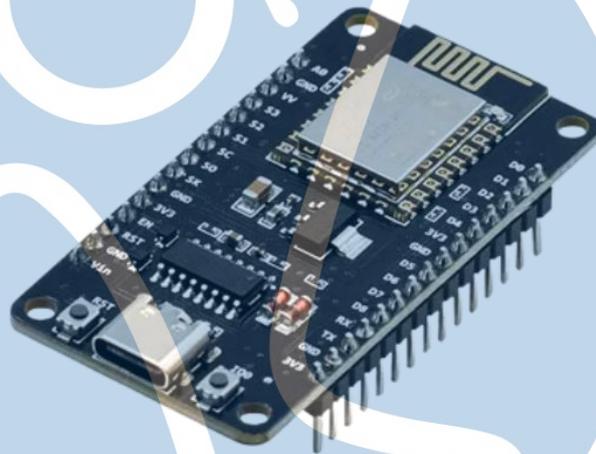
Pembangunan sistem *smart garden* berkontribusi pada peningkatan nilai ekonomi pertanian dan pendapatan petani, serta mendukung penggabungan antara pariwisata dan pelestarian lingkungan. Selain itu, keberadaan *smart garden* berperan dalam menurunkan emisi karbon, penghematan penggunaan air, serta berpotensi meningkatkan penyerapan karbon oleh vegetasi dan tanah. Sistem ini beroperasi berdasarkan tingkat kelembapan tanah, jika tanah terdeteksi kering, maka sistem akan mengaktifkan penyiraman tanaman secara otomatis. Sebaliknya, jika tanah telah mencapai tingkat kelembapan yang diinginkan, sistem akan menghentikan penyiraman dan menutup sendiri tanpa campur tangan manual [4].

### 2.2 Internet of Things

Konsep *Internet of Things* (IoT), yang pertama kali diperkenalkan oleh Kevin Ashton pada tahun 1999, kini telah menjadi istilah yang luas digunakan. Ashton, seorang inovator asal Inggris, turut serta dalam pendirian Auto-ID Center di Massachusetts Institute of Technology (MIT). IoT merujuk pada ide atau sistem di

mana objek dapat mentransfer data melalui jaringan, seringkali menggunakan konektivitas nirkabel, tanpa memerlukan intervensi langsung dari manusia atau komputer. Saat ini, IoT telah mengalami ekspansi dan evolusi yang signifikan [5]. Dengan demikian, secara umum IoT dapat didefinisikan sebagai sebuah jaringan objek cerdas yang luas dan terintegrasi yang mampu mengelola dan bertukar informasi serta data secara otomatis, serta bereaksi terhadap kondisi atau perubahan yang terjadi di lingkungan sekitarnya [6].

### 2.3 Mikrokontroler ESP8266 NodeMCU

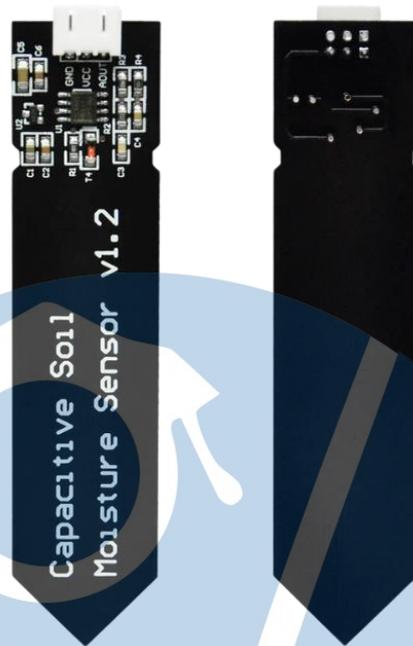


Gambar 2.1 Mikrokontroler ESP8266 NodeMCU

(Sumber: Darmawan, 2022) [4]

Kebanyakan orang yang baru memulai proyek IoT memilih *Mikrokontroler ESP8266 NodeMCU* sebagai papan pengembangan untuk proyek mereka, seperti yang diperlihatkan pada Gambar 2.1. Alasan utamanya adalah karena harganya yang terjangkau dan sifatnya yang praktis. Perangkat ini sudah siap pakai dan mudah dihubungkan langsung ke *port USB* komputer atau laptop untuk pemrograman tanpa memerlukan perangkat tambahan seperti USB to Serial. *NodeMCU* dilengkapi dengan modul *Wi-Fi* yang sudah terintegrasi pada papan sirkuitnya, sehingga tidak memerlukan tambahan modul *Wi-Fi* eksternal untuk terhubung ke jaringan *Wi-Fi*. Dengan banyaknya pin *input/output* (I/O) yang tersedia, *NodeMCU* dapat dengan mudah dikembangkan menjadi perangkat atau sistem *monitoring* dan kontrol untuk berbagai proyek IoT [7].

## 2.4 Sensor Soil Moisture



Gambar 2.2 Sensor *Soil Moisture v1.2*

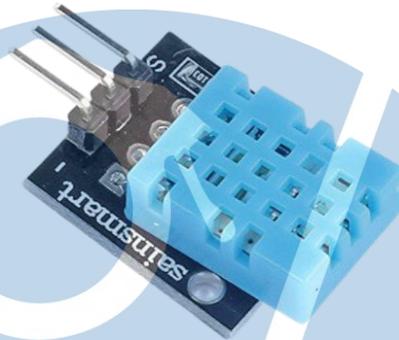
(Sumber: Effendi, 2022) [8]

Sensor kelembapan tanah yang digunakan dalam penelitian ini adalah Capacitive Soil Moisture Sensor v1.2, yang terlihat pada Gambar 2.2. Sensor ini adalah tipe sensor analog yang dirancang khusus untuk mengukur kelembapan tanah dan dibuat dari material yang tahan terhadap karat, menjadikannya pilihan yang cocok untuk aplikasi di lingkungan yang lembab atau basah [3]. Sensor kelembapan tanah ini menggunakan dua probe yang berfungsi untuk mengirimkan arus listrik melalui tanah dan kemudian mengukur resistansi yang dihasilkan untuk menentukan tingkat kelembapan. Air dalam tanah akan menurunkan resistansi karena air memudahkan aliran listrik (menunjukkan resistansi rendah), sedangkan tanah kering akan menawarkan resistansi yang lebih tinggi karena kurangnya air yang membuatnya lebih sulit untuk menghantarkan listrik [9].

Menurut data kelembapan tanah ideal untuk berbagai tanaman, tanah kering memiliki kelembapan antara 0%-40%, kelembapan ideal berada di kisaran 40%-60%, dan tanah basah memiliki kelembapan antara 60%-100%. Dengan demikian, penggunaan sensor ini memungkinkan untuk secara akurat mengidentifikasi kondisi tanah sebagai kering, ideal, atau basah berdasarkan resistansi yang diukur.

Informasi ini kemudian dapat digunakan untuk mengoptimalkan sistem *Smart Garden* dan pemeliharaan tanaman [10]. Namun, dalam penelitian ini, pengukuran kelembapan tanah hanya difokuskan pada kategori tanah kering dan tanah basah.

## 2.5 Sensor DHT11



Gambar 2.3 Sensor *DHT11*

(Sumber: Omega, 2023) [1]

Pada Gambar 2.3 memperlihatkan modul Sensor *DHT11* yang digunakan untuk mengukur suhu dan kelembapan di lingkungan sekitar. Modul ini memberikan output berupa sinyal tegangan analog yang dapat diinterpretasikan oleh *mikrokontroler* [11]. *DHT11* terkenal dengan fitur kalibrasi yang sangat akurat, dimana koefisien kalibrasi disimpan dalam memori program *One-Time Programmable* (OTP). Hal ini memungkinkan sensor untuk memberikan pembacaan yang konsisten karena setiap kali suhu atau kelembapan diukur, modul ini akan membaca koefisien kalibrasi yang relevan [12]. Sensor ini menghasilkan data dengan resolusi 14 bit untuk suhu dan 12 bit untuk kelembapan.

## 2.6 Liquid Crystal Display 16x2



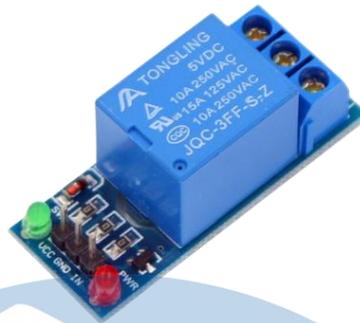
Gambar 2.4 *Liquid Crystal Display (LCD) dan I2C*

(Sumber: Darmawan, 2022) [4]

*Liquid Crystal Display (LCD)* yang ditunjukkan pada Gambar 2.4 merupakan tipe tampilan elektronik yang tercipta melalui teknologi logika CMOS. Tampilan ini tidak mengeluarkan cahaya sendiri, melainkan dengan memantulkan cahaya dari lingkungan sekitar melalui *front-lit* atau dengan mengirimkan cahaya melalui sumber cahaya dari belakang (*back-lit*). Fungsinya adalah untuk menunjukkan informasi yang bisa berupa karakter, huruf, angka, atau gambar [13]. *Liquid Crystal Display (LCD)* dengan konfigurasi 16x2 dapat menunjukkan sejumlah total 32 karakter, yang dibagi ke dalam dua baris, dengan setiap baris mampu menampilkan hingga 16 karakter.

Dengan integrasi modul I2C, para pengembang dan desainer perangkat dapat menghubungkan LCD ini ke *mikrokontroler* dengan lebih sederhana dan efisien, memanfaatkan hanya 4 konektor I2C, yang mempermudah proses komunikasi data dan mengurangi jumlah pin yang diperlukan untuk konektivitas antara LCD dan *mikrokontroler* [14].

## 2.7 Komponen Elektronik Relay



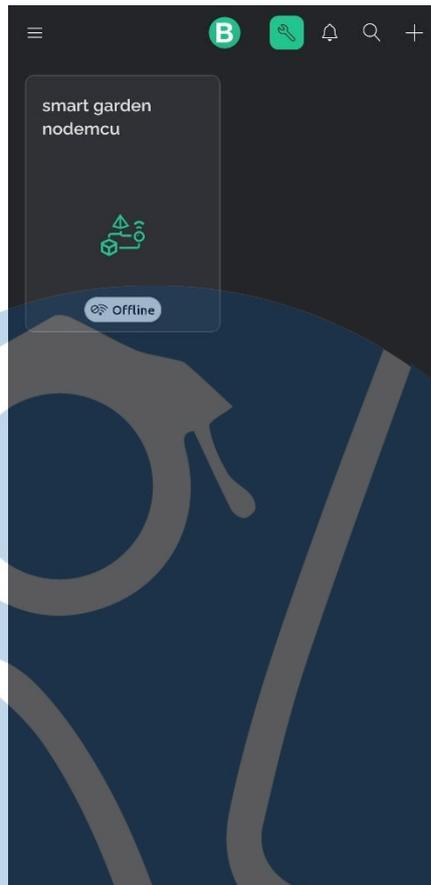
Gambar 2.5 *Relay* 1 Channel

(Sumber: Darmawan, 2022) [4]

*Relay* memiliki kemampuan untuk mengontrol aliran listrik, memungkinkan atau menghentikan pasokan listrik ke berbagai perangkat elektronik, sebagaimana diperlihatkan pada Gambar 2.5. Rangkaian *driver relay* ini dibuat untuk bekerja sesuai dengan instruksi yang diberikan oleh *mikrokontroler*, yang mengirimkan sinyal-sinyal khusus untuk mengatur operasi *relay* tersebut [15]. Kontak pada *relay* terbuka atau tertutup karena adanya gaya magnet yang dihasilkan oleh arus listrik dalam kumparan. Perbedaan utama antara *relay* dan sakelar manual terletak pada metode pengoperasiannya. *Relay* beralih posisi secara otomatis melalui sinyal listrik, sementara sakelar dioperasikan secara fisik oleh tangan pengguna [1].

STT - NF

## 2.8 Aplikasi *Blynk*



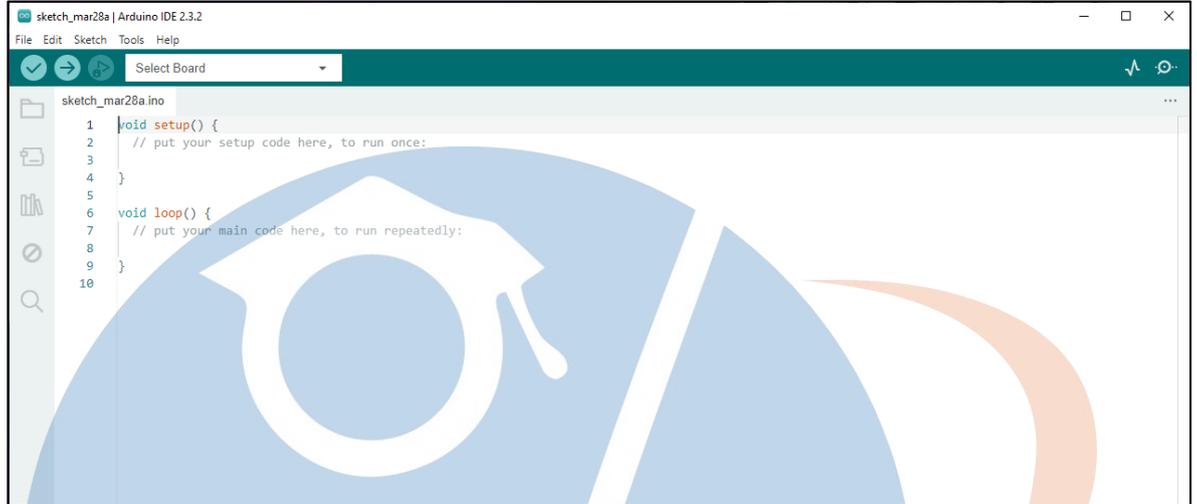
Gambar 2.6 Tampilan halaman depan aplikasi *Blynk*

(Sumber: Dokumen Pribadi)

Aplikasi *Blynk* merupakan aplikasi yang tersedia untuk perangkat *IOS* dan *Android*, dirancang untuk memungkinkan pengguna mengontrol modul seperti *Arduino*, *Raspberry Pi*, *ESP8266*, dan perangkat serupa lainnya secara *remote* melalui koneksi internet [16]. Aplikasi *Blynk* yang ditampilkan pada Gambar 2.6, kompatibel dengan berbagai macam perangkat keras, asalkan perangkat tersebut memiliki kemampuan untuk terhubung ke jaringan *Wi-Fi*, yang diperlukan untuk memfasilitasi komunikasi antara aplikasi dan *hardware* yang digunakan [17]. Aplikasi *Blynk* terdiri dari tiga elemen kunci. Pertama, *Blynk App*, yang memungkinkan pengguna untuk merancang antarmuka pengguna dengan berbagai *widget* yang tersedia. Kedua, *Blynk Server*, yang mengelola semua komunikasi antara perangkat *smartphone* pengguna dan perangkat keras. Dan ketiga, *Blynk*

*Libraries*, yang memfasilitasi pertukaran data antara server *Blynk* dan proses *input* serta *output* pada perangkat keras [18].

## 2.9 Software Arduino IDE



Gambar 2.7 Tampilan antarmuka *software Arduino IDE*

(Sumber: Dokumen Pribadi)

*Arduino IDE* merupakan lingkungan pengembangan yang berfungsi sebagai editor teks untuk menulis, membuka, mengedit, dan memverifikasi kode, serta mengunggahnya ke papan *Arduino*. Antarmuka pengguna dari *Arduino IDE*, yang menampilkan area untuk penulisan kode, opsi untuk kompilasi, dan pengunggahan ke perangkat keras, dapat dilihat pada Gambar 2.7. Program yang dibuat untuk *Arduino* dikenal sebagai "*sketch*," yang merupakan file kode sumber *Arduino* dengan ekstensi *.ino* [18]. Bahasa pemrograman yang digunakan oleh *Arduino* pada dasarnya adalah C, yang merupakan bahasa pemrograman tingkat tinggi yang populer dan luas digunakan oleh para *programmer*. *Arduino IDE* telah berkembang menjadi *platform* yang sangat diutamakan oleh berbagai profesional, tidak hanya karena adanya beragam modul pendukung seperti *sensor*, *monitor*, pembaca, dan lain-lain, tetapi juga karena sifat *open source* dari baik perangkat keras maupun perangkat lunaknya. Ketersediaan kode sumber yang terbuka ini memungkinkan pengguna untuk memodifikasi, menyesuaikan, dan membagikan perangkat keras dan perangkat lunak sesuai dengan kebutuhan mereka, yang menambah daya tarik *Arduino* di kalangan penggemar dan profesional [19].

## 2.10 Penelitian Terkait

Dalam bagian ini, akan ditampilkan sebuah tabel yang merangkum berbagai penelitian terkait yang telah dijalankan sebelumnya.

Tabel 2.1 Penelitian Terkait

(Sumber: Dokumen Pribadi)

No	Nama dan Tahun	Judul	Hasil Penelitian
1	Panji Hidayatullah, Mira Orisa, Ali Mahmudi, 2022	Desain dan Implementasi Sistem Pemantauan dan Pengendalian Tanaman Hidroponik Menggunakan Teknologi <i>Internet of Things</i> (IoT) [7]	Melakukan penyiraman tanaman otomatis yang dikontrol di aplikasi <i>Whatsapp</i> .
2	Muhammad Irsyam, Alamsyahzali Tanjung, 2019	Sistem Pengairan Tanaman Otomatis yang Terintegrasi dengan Aplikasi <i>Telegram</i> [20]	Melakukan Penyiraman tanaman otomatis yang dikontrol dengan aplikasi <i>Telegram</i> .
3	Joni Eka Candra, Algifanri Maulana, 2019	Implementasi Sensor Kelembapan Tanah dalam Perancangan Sistem Irigasi Tanaman Secara Otomatis [9]	Melakukan penyiraman tanaman otomatis dengan aplikasi <i>Arduino IDE</i> .

Tabel 2.1 menguraikan berbagai penelitian yang telah dilakukan sebelumnya dengan fokus yang serupa, yaitu penerapan teknologi *Internet of Things* (IoT) dalam sistem irigasi. Tabel tersebut mengkaji sistem kontrol dan pemantauan pertumbuhan tanaman. Penelitian-penelitian ini mengindikasikan bahwa proses penyiraman tanaman dapat dioptimalkan melalui berbagai teknologi yang memungkinkan kontrol dan penjadwalan otomatis. Secara umum, penelitian ini menyoroti kemungkinan integrasi teknologi IoT dalam sektor pertanian atau kebun untuk memperbaiki proses penyiraman dengan berbagai metode kontrol, yang masing-masing memiliki keuntungan dan aplikasi khususnya.

## 2.11 Posisi Penelitian

Dalam bagian ini, akan ditampilkan sebuah tabel yang memuat daftar penelitian yang telah dilaksanakan serta posisi penelitian yang sedang dikerjakan oleh penulis dalam konteks penelitian-penelitian tersebut.

Tabel 2.2 Posisi Penelitian

(Sumber: Dokumen Pribadi)

No	Nama penelitian, judul dan tahun penelitian	Mikrokontroler yang dipakai	Tipe sensor yang dipakai	Platform
1	Panji Hidayatullah, Mira Orisa, Ali Mahmudi. Desain dan Implementasi Sistem Pemantauan dan Pengendalian Tanaman Hidroponik Menggunakan Teknologi Internet of Things (IoT), 2022 [7]	ESP8266 dan <i>Arduino Uno R3</i>	Sensor DHT11, sensor pH, sensor TDS (Total Dissolved Solid), sensor LDR (Light Dependent Resistor), sensor Ultrasonik HC-SR04	<i>Whatsapp</i>
2	Muhammad Irsyam, Alamsyahzali Tanjung. Sistem Pengairan Tanaman Otomatis yang Terintegrasi dengan Aplikasi Telegram, 2019 [20]	<i>Arduino Mega 2560</i> dan <i>NodeMCU</i>	Sensor <i>soil moisture</i>	<i>Telegram</i>
3	Joni Eka Candra, Algifanri Maulana. Implementasi Sensor Kelembapan Tanah dalam Perancangan	<i>Arduino Uno</i>	Sensor <i>soil moisture</i>	<i>Arduino IDE</i>

	Sistem Irigasi Tanaman Secara Otomatis, 2019 [9]			
4	Wahyu Firmansyah. Perancangan Sistem Kontrol dan <i>Monitoring</i> Tanaman untuk <i>Smart Garden</i> Berbasis <i>Internet of Things</i> , 2024	ESP8266	Sensor DHT11, sensor <i>soil moisture</i>	<i>Blynk</i>

Tabel 2.2 memaparkan berbagai riset yang telah dan sedang dilaksanakan, termasuk penelitian yang dilakukan oleh penulis sendiri. Faktor pembeda dari setiap studi tercermin dalam kolom-kolom tabel, menyoroti perbedaan dalam penggunaan *mikrokontroler*, jenis sensor, dan platform yang digunakan. Meskipun setiap penelitian berfokus pada pemantauan kondisi tanaman, yang umumnya melibatkan sensor untuk mendeteksi kelembapan tanah dan suhu, terdapat variasi pada platform yang digunakan untuk menampilkan data dari hasil pemantauan tersebut.

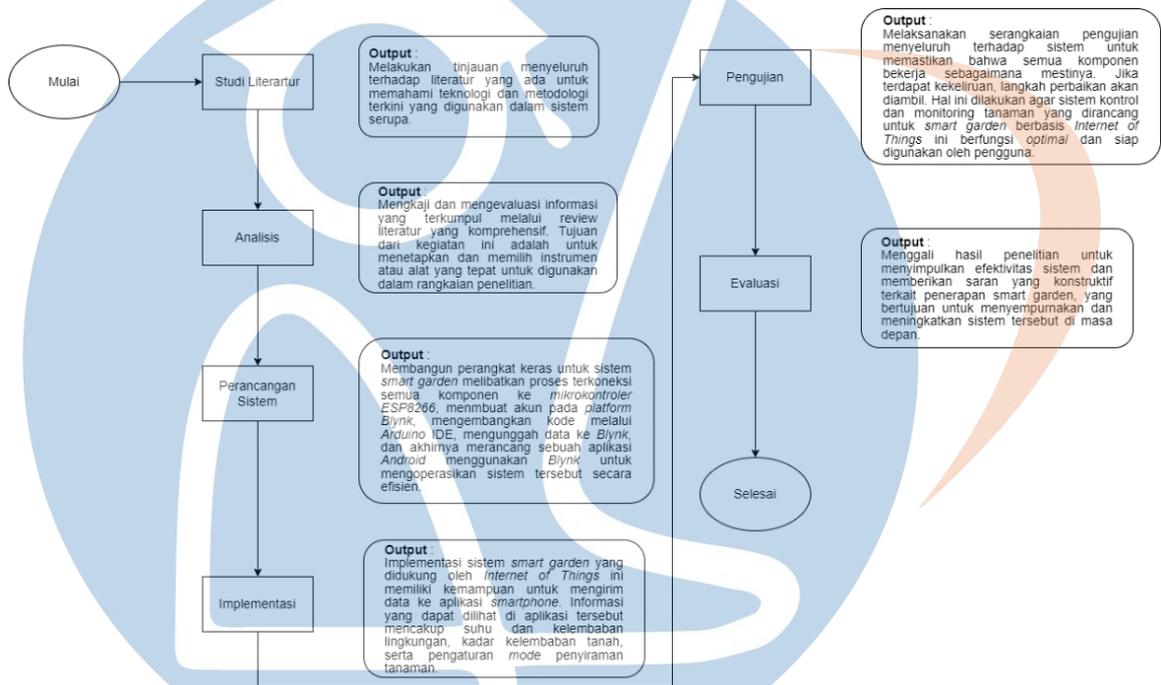
STT - NF

## BAB 3

### METODOLOGI PENELITIAN

#### 3.1 Tahapan Penelitian

Penelitian yang berjudul "Perancangan Sistem Kontrol dan *Monitoring* Tanaman untuk *Smart Garden* Berbasis *Internet of Things*" ini mencakup beberapa langkah utama, yang digambarkan dalam diagram alir berikut ini:



Gambar 3.1 Diagram alir tahapan penelitian

(Sumber: Dokumen Pribadi)

Pada Gambar 3.1 menggambarkan diagram alir yang mencerminkan serangkaian proses dan langkah-langkah yang diambil oleh penulis dalam penelitian ini.

Tahap awal dari penelitian ini diawali dengan proses studi literatur yang komprehensif. Penulis menghabiskan waktu untuk mencari, mengumpulkan, dan mempelajari berbagai sumber ilmiah, termasuk jurnal, artikel, buku, serta paper konferensi yang relevan dengan topik "Perancangan Sistem Kontrol dan *Monitoring* Tanaman untuk *Smart Garden* Berbasis *Internet of Things*". Tujuan dari studi literatur ini adalah untuk mendapatkan pemahaman mendalam mengenai

konsep-konsep dasar, metodologi yang telah ada, serta temuan terkini yang berkaitan dengan sistem kontrol dan *monitoring* berbasis IoT.

Berdasarkan pemahaman yang mendalam yang diperoleh dari studi literatur, penulis melanjutkan ke tahap analisis data dan informasi yang telah dikumpulkan. Langkah ini sangat penting karena membantu dalam mempersiapkan alat dan bahan dengan lebih efektif untuk digunakan dalam penelitian. Penulis memulai dengan mengidentifikasi spesifikasi teknis dari komponen-komponen yang akan digunakan, termasuk sensor untuk pengukuran suhu, kelembaban, dan parameter lain yang relevan dengan pertumbuhan tanaman.

Selanjutnya dalam tahap perancangan sistem kontrol dan *monitoring smart garden*, penulis mengintegrasikan sensor suhu, kelembaban tanah, cahaya, dan modul penyiraman dengan *mikrokontroler ESP8266*, memastikan kompatibilitas dan efisiensi komunikasi antar komponen. Menggunakan *Arduino IDE*, penulis mengembangkan kode untuk mengumpulkan dan memproses data sensor, serta merancang antarmuka pengguna pada aplikasi *Blynk* untuk pemantauan *real-time*.

Setelah tahap perancangan sistem, implementasi sistem *smart garden* menggunakan aplikasi *Blynk* memungkinkan pengguna untuk mengontrol dan memantau tanaman melalui *smartphone*. Aplikasi ini diintegrasikan dengan *mikrokontroler ESP8266*, dan *dashboard* yang ramah pengguna dikembangkan untuk visualisasi data dan operasi kontrol. Serangkaian pengujian fungsional yang ketat dijalankan untuk memastikan keakuratan sensor dan komunikasi yang lancar antara perangkat keras dan perangkat lunak. Kesalahan yang teridentifikasi selama pengujian diatasi melalui iterasi desain dan pemrograman, mengarah pada peningkatan sistem yang berkelanjutan. Implementasi ini juga termasuk pengujian lapangan untuk mengevaluasi performa sistem dalam kondisi nyata, memastikan sistem berfungsi dengan optimal dan sesuai dengan kebutuhan pengguna.

Setelah melewati tahap implementasi dan pengujian yang menunjukkan sistem *Smart Garden* beroperasi secara efisien tanpa hambatan signifikan, penulis akan menyusun kesimpulan dari hasil penelitian yang mencakup evaluasi kinerja keseluruhan sistem, analisis keakuratan data dari sensor, dan kehandalan operasi kontrol melalui aplikasi *Blynk*. Dalam kesimpulan ini, penulis juga akan

mengemukakan rekomendasi yang dirancang untuk mengoptimalkan penerapan teknologi *smart garden*. Rekomendasi ini akan berfokus pada potensi peningkatan efisiensi dalam konsumsi air dan energi, peningkatan otomatisasi, serta pengembangan fitur keamanan untuk melindungi sistem dari gangguan eksternal, sehingga memberikan panduan yang berharga bagi peneliti atau praktisi yang berkeinginan untuk menerapkan atau mengembangkan *smart garden* di masa depan.

## **3.2 Rancangan Penelitian**

### **3.2.1 Jenis Penelitian**

Studi ini menerapkan pendekatan penelitian pengembangan (R&D) merupakan metode yang melibatkan serangkaian langkah yang sistematis, mulai dari identifikasi potensi dan permasalahan, rancangan produk, validasi desain, pengujian prototipe, revisi berdasarkan hasil uji, hingga pengujian dalam skenario nyata dan persiapan untuk produksi skala besar [21]. Sebagai langkah awal, dilakukan analisis kebutuhan yang mendalam untuk memastikan bahwa produk yang akan dikembangkan tidak hanya memenuhi kebutuhan pengguna tetapi juga beroperasi dengan efektivitas tinggi. Metode R&D ini umumnya diaplikasikan dalam bidang-bidang seperti sains, teknik, dan teknologi informasi, terutama dalam konteks pendidikan, untuk menghasilkan produk yang inovatif dan praktis. Penelitian ini mengadopsi metode R&D karena tujuan utamanya adalah untuk **mengembangkan** sebuah perangkat yang berfungsi untuk kontrol dan *monitoring* tanaman dalam sistem *Smart Garden* yang terintegrasi dengan teknologi *Internet of Things* (IoT).

### **3.2.2 Metode Analisis Data**

Dalam penelitian ini, penulis menggunakan metode analisis data kuantitatif untuk mengevaluasi efektivitas sistem penyiraman otomatis dalam *Smart Garden*. Data yang relevan dikumpulkan secara *real-time* menggunakan sensor kelembapan tanah dan sensor suhu dan kelembapan udara DHT11. Untuk mengaktifkan mekanisme penyiraman otomatis, sistem ini memanfaatkan *relay* yang terhubung

dengan *water pump*. Semua data yang terkumpul ini kemudian diintegrasikan dan dipantau melalui aplikasi *Blynk*, yang menyediakan antarmuka pengguna yang intuitif dan memungkinkan visualisasi data secara langsung dalam bentuk angka dan grafik.

Untuk membuat sistem penyiraman tanaman otomatis yang baik, penulis mulai dengan mengumpulkan informasi tentang seberapa basah tanahnya. Informasi ini dikumpulkan terus-menerus sehingga penulis bisa mengerti seberapa sering dan berapa lama penulis perlu menyiram tanaman. Sistem penyiraman akan mulai bekerja sendiri jika sensor di dalam tanah merasakan bahwa tanah sudah mulai kering dan membutuhkan air, sesuai dengan apa yang tanaman butuhkan.

Penggunaan aplikasi *Blynk* tidak hanya memudahkan proses *monitoring* kondisi tanaman secara *real-time*, tetapi juga memungkinkan peneliti untuk mengumpulkan dan mengolah data besar dengan cepat dan efisien. Dengan demikian, penelitian ini diharapkan untuk menghasilkan temuan yang dapat memberikan wawasan tentang efektivitas *Smart Garden* dalam mengatur penyiraman tanaman secara otomatis, menjaga kelembapan tanah pada level optimal, dan berkontribusi pada pengelolaan sumber daya air yang lebih berkelanjutan.

### **3.2.3 Metode Pengumpulan Data**

Dalam proses pembangunan sistem ini, penulis menggunakan berbagai metode penelitian dan pengumpulan data yang diperlukan untuk menyelesaikan tugas dengan hasil yang diharapkan. Metode penelitian yang diterapkan adalah sebagai berikut:

1. Kajian literatur, yakni proses mengeksplorasi dan memahami literatur yang berkaitan, termasuk buku dan sumber-sumber terkait lainnya, untuk mengumpulkan teori-teori yang dapat mendukung penyelesaian isu yang sedang diteliti.
2. Pengamatan langsung (observasi), yaitu melakukan pengamatan secara langsung kondisi nyata di lapangan. Pengamatan ini membantu peneliti untuk

memahami aspek-aspek praktis dan operasional dari sistem yang dirancang. Observasi ini juga akan mencakup pengujian komponen-komponen IoT seperti sensor kelembapan tanah, sensor suhu, dan sistem penyiraman otomatis untuk memastikan bahwa mereka bekerja sesuai dengan spesifikasi.

3. Eksperimen, yaitu langkah penting dalam penelitian ini untuk menguji dan memvalidasi fungsionalitas serta efektivitas dari sistem kontrol dan *monitoring* tanaman yang dirancang untuk *Smart Garden* berbasis *Internet of Things* (IoT). Eksperimen akan dilakukan dengan menggunakan *prototype Smart Garden* yang telah dibuat. Tanaman akan ditempatkan dalam sistem dan dipantau menggunakan berbagai sensor IoT untuk mengumpulkan data yang akan ditampilkan di aplikasi *Blynk*.

#### **3.2.4 Metode Pengujian**

Metode pengujian yang akan digunakan dalam penelitian ini adalah *functional testing*. Dalam tahap pengembangan sistem kontrol dan *monitoring* tanaman untuk *Smart Garden* berbasis *Internet of Things* (IoT), *functional testing* menjadi metode utama untuk memastikan bahwa setiap komponen sistem beroperasi sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini akan melibatkan beberapa langkah yang terstruktur untuk mengumpulkan data yang relevan dan akurat.

Prosedur pengujian akan dimulai dengan komponen individu. Sensor DHT11, yang mengukur suhu dan kelembapan udara, akan diuji untuk memastikan pembacaannya akurat dan konsisten di berbagai kondisi lingkungan. Data akan dicatat dan dibandingkan dengan standar keakuratan yang telah ditetapkan. Sensor kelembapan tanah akan diuji untuk memverifikasi responsivitasnya terhadap perubahan kelembapan tanah, dengan data yang dikumpulkan mencakup waktu respons dan tingkat keakuratan dalam mendeteksi kondisi tanah yang berbeda.

Selanjutnya, *relay* dan pompa air akan diuji untuk mengevaluasi kinerja dalam mengontrol aliran air sesuai dengan instruksi yang diberikan oleh sistem. Pengujian ini akan mencakup pengukuran waktu yang diperlukan untuk *relay* mengaktifkan dan menonaktifkan pompa air, serta pengujian keandalan operasi berulang.

Setelah pengujian komponen individu, pengujian integrasi sistem akan dilakukan untuk menilai bagaimana komponen-komponen ini bekerja bersama sebagai satu kesatuan. Ini akan melibatkan simulasi skenario nyata dimana sensor akan mengirimkan data ke aplikasi *Blynk*, dan sistem akan membuat keputusan berdasarkan data tersebut untuk mengaktifkan atau menonaktifkan *relay* dan pompa air. Pengujian ini akan diuji coba dalam 3 hari di rumah penulis.

Semua data yang dikumpulkan selama pengujian ini akan dicatat dengan rinci, termasuk parameter pengujian, kondisi lingkungan, dan hasil yang diamati. Data ini akan dianalisis untuk mengevaluasi kinerja sistem dan untuk mengidentifikasi area potensial perbaikan.

Dalam proses pengujian, instrumen yang digunakan adalah data yang terkumpul diambil selama periode uji coba sistem. Untuk menentukan tingkat keberhasilan penelitian, data yang terkumpul ini kemudian dianalisis menggunakan rumus yang telah ditetapkan, yang mengukur efektivitas sistem. Adapun rumus untuk menentukan tingkat keberhasilan adalah:

$$\text{Tingkat keberhasilan} = \frac{\text{jumlah berhasil}}{\text{jumlah percobaan}} \times 100\% \quad (3.1)$$

Dengan menggunakan rumus ini, penulis dapat dengan mudah menghitung efisiensi operasional sistem dengan membagi jumlah berhasil di mana sistem bekerja sebagaimana mestinya dengan jumlah percobaan yang diuji, kemudian mengalikan hasilnya dengan 100 untuk mendapatkan persentase.

### **3.2.5 Metode Implementasi dan Evaluasi**

Dalam rangka mencapai tujuan penelitian ini, metode implementasi yang dipilih adalah melalui pengembangan prototipe. Prototipe merupakan implementasi awal yang telah direalisasikan dan memerlukan proses evaluasi berkelanjutan untuk mengidentifikasi serta memperbaiki aspek-aspek yang belum optimal[22]. Proses ini melibatkan perancangan, pembuatan, dan pengujian awal dari model kerja yang dapat menunjukkan prinsip-prinsip operasional dan fungsional dari sistem yang diusulkan. Pendekatan pengembangan prototipe ini memungkinkan penelitian untuk mengevaluasi secara empiris aspek-aspek teoritis yang telah dijabarkan, serta

memberikan wawasan praktis mengenai implementasi dan operasionalisasi dari konsep yang diteliti.

### 3.2.6 Lingkungan Pengembangan

Untuk memastikan kelancaran dan efektivitas dalam proses pengembangan penelitian, diperlukan berbagai perangkat pendukung. Berikut ini adalah perangkat yang digunakan dalam penelitian ini adalah:

#### A. Perangkat keras komputer

Bagian ini mengulas komponen-komponen perangkat keras yang diperlukan dalam prototipe *Smart Garden* yang dirancang untuk sistem penyiraman otomatis. Informasi lebih lanjut mengenai perangkat keras yang spesifik tertera dalam Tabel 3.1.

Tabel 3.1 Perangkat keras komputer untuk pengembangan  
(Sumber: Dokumen Pribadi)

No	Perangkat Keras	Spesifikasi
1	Processor	Ryzen 5 3600 3.6GHz
2	Memori	16 GB
3	Penyimpanan	512 GB
4	Monitor	24 Inch, resolusi 1920x1080 pixel

#### B. Perangkat keras *Smartphone Android*

Bagian ini menguraikan komponen-komponen perangkat keras yang diterapkan dalam sistem *monitoring* tanaman yang mengintegrasikan teknologi *Internet of Things* (IoT). Informasi lebih lanjut mengenai perangkat keras yang spesifik tertera dalam Tabel 3.2.

Tabel 3.2 Perangkat keras *Smartphone Android* untuk *monitoring*

(Sumber: Dokumen Pribadi)

No	Perangkat Keras	Spesifikasi
1	Processor	Dimensity 8300-Ultra
2	Memori	12 GB
3	Penyimpanan	512 GB

### C. Perangkat keras IoT

Perangkat Internet of Things (IoT) terdiri atas komponen-komponen seperti *mikrokontroler*, berbagai jenis sensor, dan modul-modul tambahan. Rincian spesifikasi dari perangkat IoT ini dapat dilihat pada Tabel 3.3.

Tabel 3.3 Perangkat keras IoT

(Sumber: Dokumen Pribadi)

No	Perangkat Keras	Keterangan
1	ESP822	Mikrokontroler
2	<i>Relay</i> Module 1 Channel	Kontrol peralatan ON – OFF
3	LCD 16x2	Display sistem
4	Sensor DHT11	Monitoring suhu dan kelembapan
5	Sensor Soil Moisture	Monitoring kelembapan tanah
6	Water Pump	Memompa air

### D. Perangkat lunak

Dalam mengoperasikan sistem penyiraman tanaman otomatis yang mengadopsi teknologi IoT, diperlukan serangkaian perangkat lunak tertentu. Informasi lebih detail mengenai perangkat lunak yang diintegrasikan dalam IoT ini dapat ditemukan dalam tabel 3.4.

Tabel 3.4 Perangkat lunak

(Sumber: Dokumen Pribadi)

No	Perangkat Lunak	Keterangan
1	Arduino IDE	Pengkodean Smart Garden
2	Bahasa Pemrograman	C/C++
3	<i>Blynk</i>	Kontrol penyiraman tanaman



STT - NF

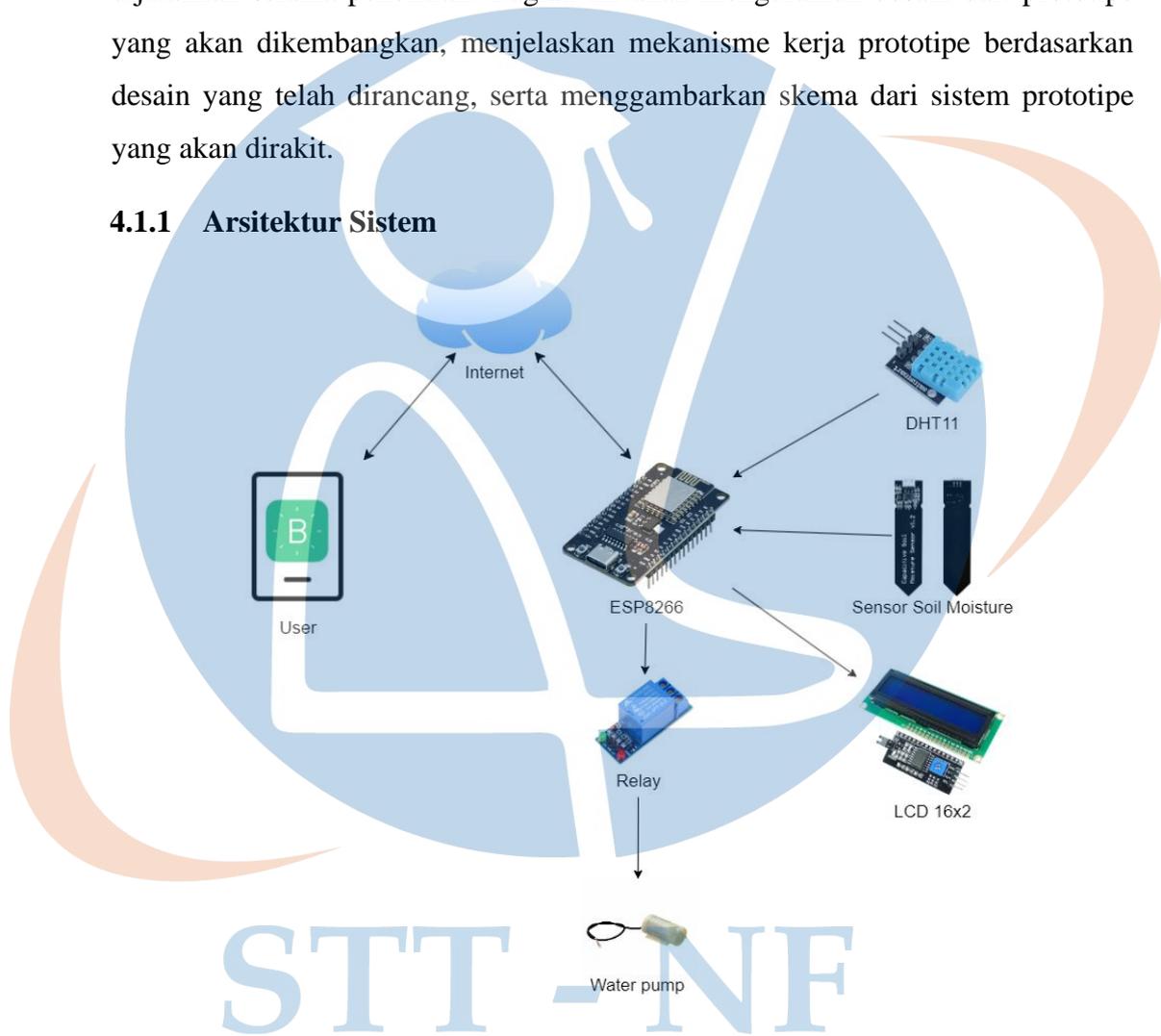
## BAB 4

### IMPLEMENTASI DAN EVALUASI

#### 4.1 Perancangan Sistem

Bagian rancangan penelitian ini menjelaskan tentang strategi yang akan dijalankan selama penelitian. Bagian ini akan menguraikan desain dari prototipe yang akan dikembangkan, menjelaskan mekanisme kerja prototipe berdasarkan desain yang telah dirancang, serta menggambarkan skema dari sistem prototipe yang akan dirakit.

##### 4.1.1 Arsitektur Sistem



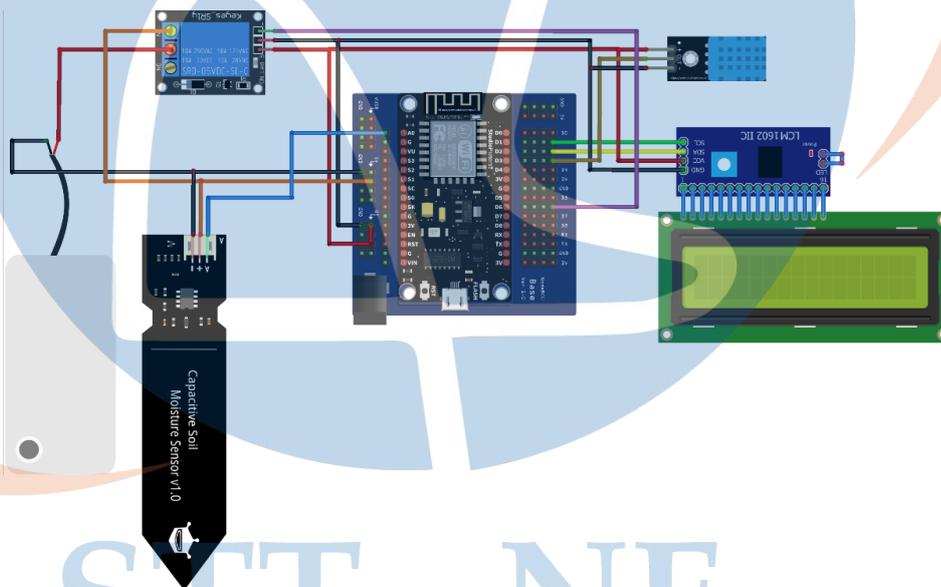
Gambar 4.1 Arsitektur sistem

(Sumber: Dokumen Pribadi)

Pada gambar 4.1 desain arsitektur sistem ini merinci bagaimana data *input*, proses, dan hasil *output* akan ditangani oleh sistem pemantauan tanaman yang sedang dikembangkan. Modul ESP8266 memerlukan sumber daya 5V untuk mengoperasikan rangkaian perangkat termasuk pompa air. Saat sistem diaktifkan,

berbagai sensor akan mulai mengambil data yang selanjutnya ditampilkan pada layar LCD. Sensor kelembapan tanah bertugas memeriksa kondisi kelembapan untuk menentukan apakah pompa air perlu diaktifkan atau tidak, berdasarkan pembacaan yang diperoleh dari sensor tersebut. *relay* akan beraksi jika kondisi tanah terdeteksi kering, mengirim sinyal *low* yang menyebabkan pompa air aktif. Sebaliknya, jika tanah cukup lembap, *relay* akan mengirim sinyal *high* yang mematikan pompa air. Informasi yang dihimpun oleh sensor ini kemudian dikirimkan ke aplikasi *Blynk* melalui koneksi WIFI dari ponsel yang telah tersambung. Data dari sensor akan ditampilkan dalam bentuk visual pada *dashboard* aplikasi *Blynk*, memungkinkan pemantauan kondisi tanaman secara *real-time*.

#### 4.1.2 Rangkaian Sistem



Gambar 4.2 Rangkaian sistem

(Sumber: Dokumen Pribadi)

Pada gambar 4.2 rangkaian sistem ini terdiri dari beberapa komponen utama yang saling terhubung untuk membentuk sistem penyiraman tanaman otomatis berbasis mikrokontroler ESP8266 NodeMCU. Berikut adalah penjelasan detail mengenai setiap komponen yang ada pada skematik:

## 1. Mikrokontroler ESP8266 NodeMCU

Mikrokontroler ESP8266 NodeMCU berperan sebagai pusat pengendalian sistem. Mikrokontroler ini dilengkapi dengan modul Wi-Fi, memungkinkan sistem untuk terhubung ke jaringan internet. Pin VCC dan GND pada NodeMCU digunakan untuk menyuplai daya ke seluruh sistem.

## 2. Sensor Kelembaban Tanah

Sensor ini berfungsi untuk mengukur kelembaban tanah. Sensor memiliki tiga pin utama: VCC (tegangan suplai), GND (*ground*), dan AOUT (output analog). Pin VCC sensor dihubungkan ke pin 5V pada NodeMCU, pin GND dihubungkan ke pin GND pada NodeMCU, dan pin AOUT dihubungkan ke pin A0 pada NodeMCU. Data kelembaban tanah yang diperoleh dari sensor ini digunakan oleh mikrokontroler untuk menentukan kapan pompa air harus diaktifkan.

## 3. Relay

*Relay* berfungsi sebagai saklar elektronik yang dikendalikan oleh mikrokontroler untuk mengontrol perangkat eksternal, seperti pompa air. *Relay* memiliki pin VCC, GND, dan IN. Pin VCC *relay* dihubungkan ke pin 5V pada NodeMCU, pin GND dihubungkan ke pin GND pada NodeMCU, dan pin IN dihubungkan ke pin D6 pada NodeMCU. *Relay* ini akan mengaktifkan pompa air berdasarkan perintah dari mikrokontroler.

## 4. Pompa Air

Pompa air dikendalikan oleh *relay* dan digunakan untuk menyiram tanaman. Kabel daya dari pompa air dihubungkan ke *relay*, sedangkan kabel lainnya dihubungkan ke pin GND. Aktivasi pompa air tergantung pada kondisi kelembaban tanah yang dipantau oleh sensor kelembaban tanah.

## 5. Sensor DHT11

Sensor DHT11 digunakan untuk mengukur suhu dan kelembaban udara di sekitar tanaman. Sensor ini memiliki pin VCC, GND, dan Data. Pin VCC sensor dihubungkan ke pin 5V pada NodeMCU, pin GND dihubungkan ke pin GND pada NodeMCU, dan pin Data dihubungkan ke pin D3 pada NodeMCU. Data suhu dan kelembaban udara yang diperoleh dari sensor ini dapat ditampilkan pada LCD.

## 6. LCD 16x2 dengan Modul I2C

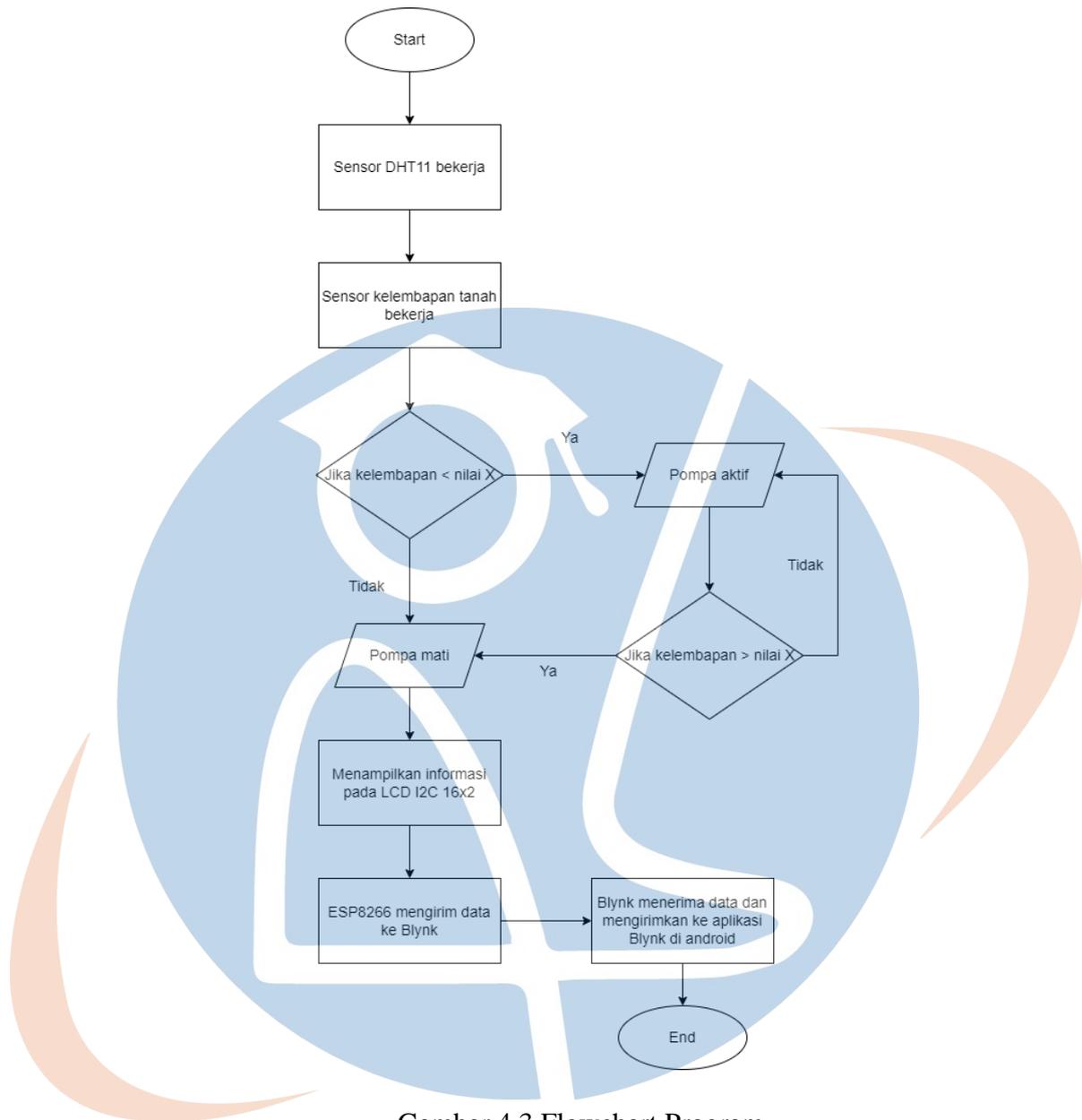
LCD 16x2 digunakan untuk menampilkan informasi seperti status kelembaban tanah, suhu, dan kelembaban udara secara real-time. LCD ini dilengkapi dengan modul I2C untuk memudahkan komunikasi dengan mikrokontroler. Pin VCC modul I2C dihubungkan ke pin 5V pada NodeMCU, pin GND dihubungkan ke pin GND pada NodeMCU, pin SDA dihubungkan ke pin D2 pada NodeMCU, dan pin SCL dihubungkan ke pin D1 pada NodeMCU.

## 7. *Blynk*

*Blynk* adalah platform IoT yang memungkinkan kontrol dan pemantauan perangkat dari jarak jauh melalui aplikasi *mobile*. Dalam proyek ini, *Blynk* digunakan untuk memonitor dan mengendalikan sistem penyiraman tanaman otomatis secara *real-time* melalui internet. Mikrokontroler ESP8266 NodeMCU terhubung ke aplikasi *Blynk* menggunakan koneksi Wi-Fi. Data dari sensor kelembaban tanah, suhu, dan kelembaban udara dikirimkan ke aplikasi *Blynk*, sehingga pengguna dapat memantau kondisi tanaman dari jarak jauh. Selain itu, pengguna juga dapat mengaktifkan atau menonaktifkan pompa air melalui aplikasi *Blynk*.

### 4.1.3 Flowchart Program

Flowchart program menggambarkan urutan proses yang menguraikan bagaimana program yang diunggah ke Arduino IDE dijalankan. Alur ini dibangun secara bertahap, mulai dari tahap awal mengunggah program hingga berhasil menampilkan data dari sensor ke aplikasi *Blynk*. Flowchart tersebut memberikan visualisasi langkah demi langkah dari eksekusi program yang akan berlangsung. Detail lebih lanjut mengenai flowchart program tersebut dapat dilihat dalam ilustrasi yang disediakan di bawah.



Gambar 4.3 Flowchart Program

(Sumber: Dokumen pribadi)

Pada gambar 4.3 yang menampilkan Flowchart di atas menggambarkan alur kerja sistem penyiraman tanaman otomatis berbasis mikrokontroler ESP8266 yang terintegrasi dengan aplikasi *Blynk*. Proses dimulai dengan mengaktifkan sensor DHT11 untuk mengukur suhu dan kelembapan udara di sekitar tanaman. Selanjutnya, sensor kelembapan tanah diaktifkan untuk mengukur tingkat kelembapan tanah.

Setelah data dari sensor kelembapan tanah diperoleh, sistem akan memeriksa apakah kelembapan tanah kurang dari nilai ambang batas yang telah ditentukan (nilai X). Jika kelembapan tanah kurang dari nilai X, pompa air akan diaktifkan untuk menyiram tanaman. Pompa akan terus aktif hingga kelembapan tanah melebihi nilai X. Jika kondisi kelembapan tanah sudah melebihi nilai X, pompa air akan dimatikan.

Selanjutnya, informasi mengenai kondisi suhu dan kelembapan dari sensor DHT11 dan sensor kelembapan tanah akan ditampilkan pada layar LCD I2C 16x2. Mikrokontroler ESP8266 kemudian mengirimkan data yang diperoleh dari sensor ke platform *Blynk*. Aplikasi *Blynk* yang terinstal pada perangkat Android akan menerima data tersebut, memungkinkan pengguna untuk memantau kondisi lingkungan tanaman secara real-time melalui perangkat mereka.

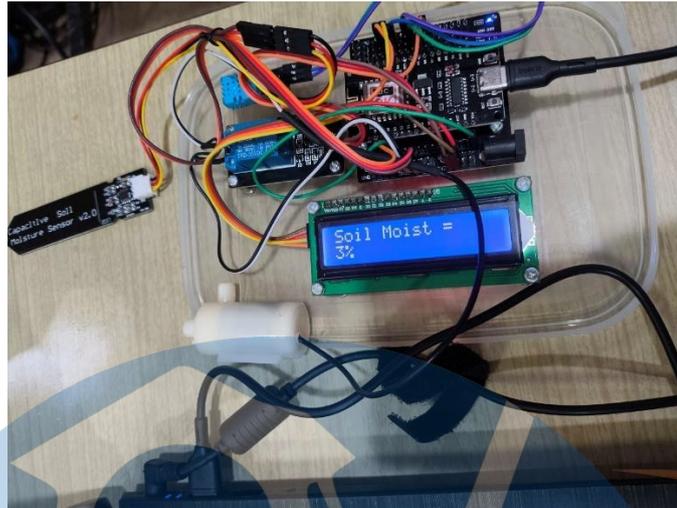
Proses ini berulang secara terus-menerus, memastikan bahwa tanaman mendapatkan penyiraman yang cukup berdasarkan kondisi kelembapan tanah yang terpantau secara otomatis.

## **4.2 Implementasi Rancangan Sistem**

Implementasi rancangan sistem merujuk pada penerapan praktis dari konsep penelitian yang telah dirancang sebelumnya. Terdapat dua aspek utama dalam implementasi ini: pertama adalah pengembangan kode program yang dijalankan melalui Arduino IDE, dan kedua adalah pembuatan prototipe fisik yang sesuai dengan skema rangkaian yang telah dirancang. Informasi lebih rinci tentang proses implementasi ini diuraikan dalam penjelasan yang akan mengikuti.

### **4.2.1 Hasil Prototipe Sistem**

Pada bagian ini merupakan hasil prototipe yang didasarkan pada rangkaian sistem. Berikut hasil prototipe yang dapat dilihat pada gambar dibawah ini.



Gambar 4.4 Hasil prototype sistem

(Sumber: Dokumen pribadi)



Gambar 4.5 Tampilan sistem keseluruhan

(Sumber: Dokumen pribadi)

Gambar 4.5 menggambarkan desain dan konfigurasi rangkaian yang akan dibuat untuk sistem penyiraman otomatis tanaman. Sementara itu, Gambar 4.6 menunjukkan sistem penyiraman otomatis yang telah terpasang pada tanaman. Dalam sistem ini, terdapat sebuah selang yang berfungsi untuk menyemprotkan air ke tanaman. Selanjutnya, ada pompa air yang beroperasi dengan tenaga listrik, yang arusnya disuplai melalui *relay*. *Relay* ini dioperasikan oleh rangkaian driver *relay* yang mengontrol pompa air dengan dua kondisi: "HIGH" yang mengaktifkan pompa air dan "LOW" yang mematikannya. Selama proses pengujian, sistem akan

memutuskan apakah kondisi tanah basah atau kering. Jika kelembaban tanah di bawah 40% yang menandakan tanah kering, maka *relay* akan memberi instruksi untuk menghidupkan pompa air, sehingga air akan mengalir melalui selang atau pipa. Di sisi lain, jika sensor kelembaban mendeteksi kelembaban tanah di atas 40%, yang menandakan tanah sudah basah, maka *relay* akan secara otomatis mematikan pompa air dan mengembalikannya ke posisi awal.

#### 4.2.2 Kode Program

Berikut ini adalah penulisan kode program pada Arduino IDE yang digunakan penulis untuk membuat rancangan penelitian.

```
1 #define BLYNK_TEMPLATE_ID "TMPL6Y8ndhcwN"  
2 #define BLYNK_TEMPLATE_NAME "smart garden"  
3 #define BLYNK_AUTH_TOKEN "ja16L0DJZwuzqmyZVf7eKJw26htK7L72"  
.
```

Gambar 4.6 Kode program Arduino IDE (1)

(Sumber: Dokumen pribadi)

Pada Gambar 4.6 menampilkan kode program tersebut adalah konfigurasi awal untuk menggunakan layanan *Blynk* dalam proyek IoT. Ini mencakup definisi untuk ID template, nama template, dan token otentikasi yang digunakan untuk mengidentifikasi dan mengamankan koneksi antara perangkat keras dan server *Blynk*. Dengan konfigurasi ini, perangkat IoT dapat terhubung ke aplikasi *Blynk* dan berinteraksi dengan fitur-fitur yang telah ditetapkan dalam template "*smart garden*".

```
5 #include <LiquidCrystal_I2C.h>  
6 #include <DHT.h>  
7 #include <Blynk.h>  
8 #include <ESP8266WiFi.h>  
9 #include <BlynkSimpleEsp8266.h>
```

Gambar 4.7 Kode program Arduino IDE (2)

(Sumber: Dokumen pribadi)

Pada gambar 4.7 menampilkan kode program yang bertujuan untuk mengatur mikrokontroler ESP8266 dalam proyek IoT (*Internet of Things*) yang

melibatkan beberapa komponen hardware dan software. Program ini memungkinkan ESP8266 untuk membaca data suhu dan kelembaban dari sensor DHT, menampilkan data tersebut pada layar LCD menggunakan antarmuka I2C, dan mengirimkan data ke aplikasi *Blynk* melalui koneksi WiFi. Dengan menggunakan pustaka yang diimpor, program ini dapat mengintegrasikan semua komponen tersebut secara efisien, sehingga data lingkungan dapat dipantau secara *real-time* melalui aplikasi *mobile Blynk*. Ini memungkinkan pengguna untuk memonitor dan mengontrol perangkat keras dari jarak jauh dengan mudah.

```
11 #define BLYNK_PRINT Serial
12 #define DHTPIN D3
13 #define DHTTYPE DHT11
14 #define pump D6
15
16 LiquidCrystal_I2C lcd(0x27, 16, 2);
17 DHT dht(DHTPIN, DHTTYPE);
```

Gambar 4.8 Kode program Arduino IDE (3)

(Sumber: Dokumen pribadi)

Pada gambar 4.8 menampilkan kode program untuk menetapkan beberapa definisi dan menginisialisasi objek yang digunakan untuk mengontrol perangkat keras dalam proyek IoT dengan mikrokontroler ESP8266. Program ini menentukan bahwa output dari platform *Blynk* akan dikirim ke port serial untuk tujuan *debugging*. Sensor DHT, yang digunakan untuk mengukur suhu dan kelembaban, dihubungkan ke pin D3 dari ESP8266 dan tipe sensor yang digunakan adalah DHT11. Selain itu, pompa air dihubungkan ke pin D6. Objek untuk mengontrol layar LCD menggunakan antarmuka I2C diinisialisasi dengan alamat I2C 0x27 dan ukuran layar 16x2. Objek untuk sensor DHT juga diinisialisasi dengan parameter pin dan tipe sensor yang sesuai. Secara keseluruhan, kode ini menyiapkan berbagai komponen yang diperlukan untuk proyek IoT yang melibatkan pengukuran suhu dan kelembaban, kontrol pompa air, dan tampilan informasi pada layar LCD.

```

19  const int AirValue = 790;
20  const int WaterValue = 420;
21  int soilMoistureValue = 0;
22  int soilmoist= 0;
23  int buttonState;
24  int humidity,temperature,fp,sistem;
25
26  // set point
27  int SP_LOW=40;
28  int SP_HIGH=60;

```

Gambar 4.9 Kode program Arduino IDE (4)

(Sumber: Dokumen pribadi)

Pada gambar 4.9 menampilkan Kode program untuk menginisialisasi beberapa variabel yang digunakan dalam proyek IoT untuk mengukur dan mengontrol kelembaban tanah serta memantau kondisi lingkungan. Nilai konstanta AirValue dan WaterValue digunakan sebagai referensi untuk mengkalibrasi sensor kelembaban tanah, di mana AirValue menunjukkan nilai pembacaan sensor saat tanah kering, dan WaterValue menunjukkan nilai saat tanah basah. Variabel soilMoistureValue dan soilmoist akan menyimpan pembacaan aktual dari sensor kelembaban tanah dan nilai kelembaban tanah yang telah diolah.

Selain itu, buttonState digunakan untuk menyimpan status tombol (mungkin untuk kontrol manual atau mode operasi), sementara humidity dan temperature menyimpan data dari sensor DHT mengenai kelembaban dan suhu udara. Variabel fp dan sistem mungkin digunakan untuk fungsi tambahan dalam sistem, seperti pengaturan parameter atau status sistem secara keseluruhan.

Dua variabel SP\_LOW dan SP\_HIGH mendefinisikan titik setel (set point) untuk kontrol kelembaban tanah, dengan SP\_LOW sebagai batas bawah dan SP\_HIGH sebagai batas atas. Ketika kelembaban tanah turun di bawah SP\_LOW, sistem mungkin mengaktifkan pompa untuk menambah air, dan ketika kelembaban mencapai SP\_HIGH, sistem mungkin mematikan pompa. Ini memastikan bahwa kelembaban tanah berada dalam rentang yang diinginkan untuk pertumbuhan tanaman yang optimal.

```

30 // TOKEN BLYNK
31 char auth[] = "ja16LODJZwuzqmyZVf7eKJw26htK7L72";
32 // JARINGAN WIFI
33 char ssid[] = "POCO"; // Nama Hotspot/WiFi
34 char pass[] = "wahyu1234"; // Password

```

Gambar 4.10 Kode program Arduino IDE (5)

(Sumber: Dokumen pribadi)

Pada gambar 4.10 menampilkan kode program berisi informasi yang diperlukan untuk menghubungkan mikrokontroler ESP8266 ke platform *Blynk* dan jaringan WiFi. Variabel “auth” menyimpan token autentikasi yang unik untuk aplikasi *Blynk*, yang memungkinkan mikrokontroler untuk berkomunikasi dengan server *Blynk* dan aplikasi seluler yang terkait. Variabel “ssid” dan “pass” menyimpan nama dan kata sandi jaringan WiFi yang akan digunakan oleh ESP8266 untuk terhubung ke internet. Dengan menghubungkan ke jaringan WiFi yang ditentukan, mikrokontroler dapat mengirim dan menerima data ke dan dari aplikasi *Blynk*, memungkinkan pemantauan dan pengendalian perangkat dari jarak jauh melalui aplikasi seluler.

```

36 BLYNK_WRITE(V4){
37   buttonState = param.asInt();
38   if(buttonState==HIGH){
39     sistem=1;//automatic
40     fp=0;
41     delay(10);
42   }
43   else if(buttonState==LOW){
44     sistem=0;//manual
45     delay(10);
46   }
47 }
48 //=====
49 BLYNK_WRITE(V5){
50   buttonState = param.asInt();
51   if(sistem==0){
52     if(buttonState==LOW){
53       digitalWrite(pump,HIGH);
54       delay(10);
55     }
56     else if(buttonState==HIGH){
57       digitalWrite(pump,LOW);
58       delay(10);
59     }
60   }
61 }

```

Gambar 4.11 Kode program Arduino IDE (6)

(Sumber: Dokumen pribadi)

Pada gambar 4.11 menampilkan kode program bagian dari integrasi dengan platform *Blynk* yang mengontrol mode operasi dan pengendalian pompa air dalam sistem IoT. Fungsi `BLYNK_WRITE(V4)` dan `BLYNK_WRITE(V5)` menangani input dari widget Blynk yang terhubung ke pin virtual V4 dan V5, masing-masing.

Fungsi pertama (`BLYNK_WRITE(V4)`) digunakan untuk menentukan mode operasi sistem, apakah otomatis atau manual, berdasarkan status tombol yang diterima dari aplikasi *Blynk*. Jika tombol pada pin V4 berada dalam posisi tinggi, sistem beralih ke mode otomatis (`sistem=1`). Sebaliknya, jika tombol berada dalam posisi rendah, sistem beralih ke mode manual (`sistem=0`).

Fungsi kedua (`BLYNK_WRITE(V5)`) digunakan untuk mengontrol pompa air saat sistem berada dalam mode manual. Jika sistem berada dalam mode manual (`sistem==0`), status tombol pada pin V5 akan menentukan apakah pompa air diaktifkan atau dinonaktifkan. Jika tombol pada pin V5 berada dalam posisi rendah, pompa akan dinyalakan (`digitalWrite(pump, HIGH)`). Jika tombol berada dalam posisi tinggi, pompa akan dimatikan (`digitalWrite(pump, LOW)`).

```
63 void setup(){
64   lcd.begin(16,2);
65   lcd.init();
66   lcd.backlight();
67   lcd.clear();
68   lcd.print(" Smart Garden ");
69   lcd.setCursor(0, 1);
70   lcd.print(" NodeMCU V3 ");
71   Serial.begin(9600);
72   dht.begin();
73   pinMode(pump,OUTPUT);
74   digitalWrite(pump,HIGH);
75   delay(1200);
76   Blynk.begin(auth, ssid, pass);
77   delay(1500);
78 }
```

Gambar 4.12 Kode program Arduino IDE (7)

(Sumber: Dokumen pribadi)

Pada gambar 4.12 menampilkan kode program yang berisi fungsi `setup()` yang digunakan untuk inisialisasi perangkat keras dan perangkat lunak dalam sistem IoT berbasis NodeMCU V3. Pada awalnya, fungsi ini menginisialisasi dan

mengatur layar LCD, menampilkan pesan "*Smart Garden*" di baris pertama dan "NodeMCU V3" di baris kedua untuk memberi tahu bahwa sistem telah menyala. Selanjutnya, komunikasi serial dimulai untuk tujuan debugging, dan sensor DHT diinisialisasi untuk mulai membaca data suhu dan kelembaban.

Pompa air diatur sebagai *output* dan dinyalakan sebentar untuk memastikan siap beroperasi. Kemudian, fungsi ini menginisialisasi koneksi *Blynk* dengan menggunakan token autentikasi dan kredensial WiFi yang telah ditentukan sebelumnya (auth, ssid, pass). Setelah koneksi *Blynk* berhasil, sistem akan siap untuk beroperasi, menerima dan mengirim data antara perangkat keras dan aplikasi *Blynk*. Penundaan (*delay*) digunakan pada berbagai titik untuk memastikan bahwa setiap komponen memiliki cukup waktu untuk inisialisasi yang benar sebelum melanjutkan ke langkah berikutnya.

```
80 void loop(){
81   float temperature = dht.readTemperature();
82   float humidity = dht.readHumidity();
83
84   lcd.clear();
85   lcd.setCursor(0,0);
86   lcd.print("Temp =");
87   lcd.print(temperature);
88   lcd.print("C");
89
90   lcd.setCursor(0,1);
91   lcd.print("Hum =");
92   lcd.print(humidity);
93   lcd.print("%");
94   delay(2000);
95
96   int soilMoistureValue = analogRead(A0);
97   int soilmoist = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
98   if(soilmoist >= 100){
99     | soilmoist=100;
100  }
101  else if(soilmoist <= 0){
102    | soilmoist=0;
103  }
104
105  lcd.clear();
106  lcd.setCursor(0,0);
107  lcd.print("Soil Moist =");
108  lcd.setCursor(0, 1);
109  lcd.print(soilmoist);
110  lcd.print("% ");
```

Gambar 4.13 Kode program Arduino IDE (8)

(Sumber: Dokumen pribadi)

Pada gambar 4.13 menampilkan kode program yang berisi logika utama yang berjalan terus-menerus untuk mengukur dan menampilkan data lingkungan dalam sistem IoT. Dalam *loop* ini, suhu dan kelembaban diukur menggunakan sensor DHT, dan hasilnya ditampilkan pada layar LCD. Setiap pembacaan suhu dan

kelembaban ditampilkan selama 2 detik sebelum dihapus untuk menampilkan informasi berikutnya.

Setelah menampilkan suhu dan kelembaban, fungsi ini membaca nilai kelembaban tanah dari sensor yang terhubung ke pin analog A0. Nilai ini kemudian dipetakan ke dalam persentase menggunakan nilai referensi untuk tanah kering dan basah (AirValue dan WaterValue). Persentase kelembaban tanah ini kemudian dikoreksi untuk memastikan berada dalam rentang 0 hingga 100%.

Hasilnya kemudian ditampilkan di layar LCD, memberikan pengguna informasi real-time tentang kelembaban tanah. Dengan *loop* ini, sistem secara terus-menerus memperbarui informasi lingkungan dan menampilkannya pada layar LCD untuk pemantauan yang mudah.

```
112 //-----control
113 if(sistem==1){
114   if((soilmoist<SP_LOW)&&(fp==0)){
115     digitalWrite(pump,LOW);
116     lcd.setCursor(7,1);
117     lcd.print("Pump: ON");
118     fp=1;
119   }
120   else if((soilmoist>SP_HIGH)&&(fp==1)){
121     digitalWrite(pump,HIGH);
122     lcd.setCursor(7,1);
123     lcd.print("Pump: OFF");
124     fp=0;
125   }
126 }
127
128 Blynk.virtualWrite(V1, temperature);
129 Blynk.virtualWrite(V2,soilmoist);
130 Blynk.virtualWrite(V3,humidity);
131
132 delay(2000);
133 }
```

Gambar 4.14 Kode program Arduino IDE (9)

(Sumber: Dokumen pribadi)

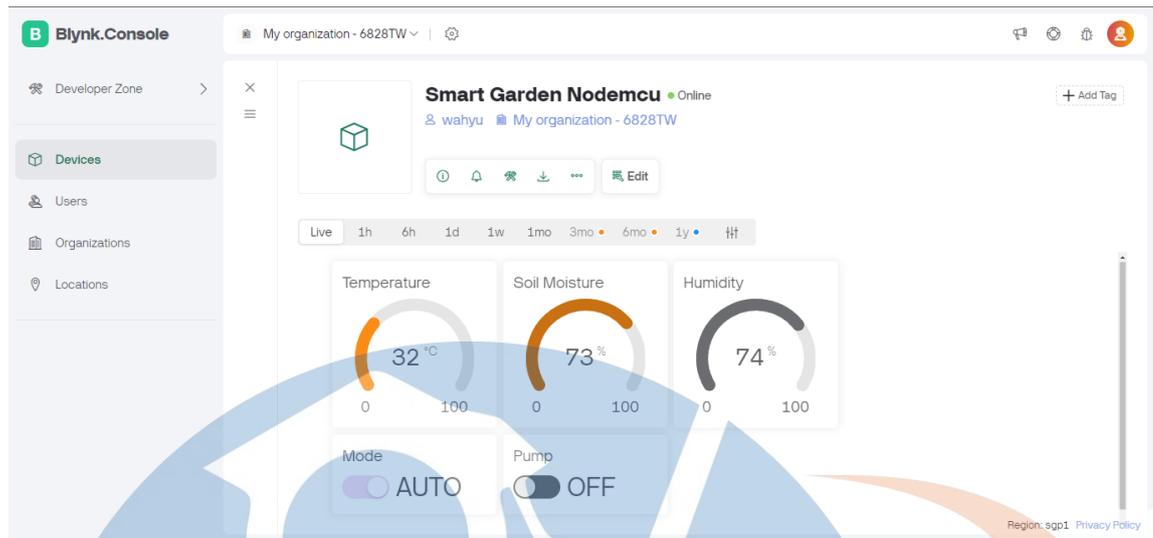
Pada gambar 4.14 menampilkan kode program yang bertujuan untuk mengontrol pompa air dan memantau kondisi tanah serta lingkungan pada sistem monitoring tanaman yang terhubung dengan platform *Blynk*. Pada awalnya, kode tersebut memeriksa apakah sistem dalam keadaan aktif (`sistem==1`). Jika ya, maka program akan memeriksa kelembaban tanah (`soilmoist`) dan status pompa air (`fp`).

Jika kelembaban tanah lebih rendah dari set point kelembaban rendah (SP\_LOW) dan pompa air belum menyala ( $fp==0$ ), maka pompa akan diaktifkan (`digitalWrite(pump,LOW)`) untuk menyirami tanaman, status pompa diubah menjadi menyala ( $fp=1$ ), dan informasi "Pump: ON" ditampilkan pada LCD. Sebaliknya, jika kelembaban tanah melebihi set point kelembaban tinggi (SP\_HIGH) dan pompa sedang menyala ( $fp==1$ ), maka pompa akan dimatikan (`digitalWrite(pump,HIGH)`), status pompa diubah menjadi mati ( $fp=0$ ), dan informasi "Pump: OFF" ditampilkan pada LCD.

Setelah itu, kode mengirimkan data suhu (temperature), kelembaban tanah (soilmoist), dan kelembaban udara (humidity) ke server *Blynk* menggunakan perintah `Blynk.virtualWrite()`, yang memungkinkan data tersebut ditampilkan dalam aplikasi *Blynk* untuk pemantauan jarak jauh. Setiap pengiriman data ini dilakukan pada pin virtual yang berbeda (V1, V2, V3) di dalam aplikasi *Blynk*. Terakhir, ada jeda dua detik (`delay(2000)`) sebelum siklus program diulang, yang memberikan waktu bagi mikrokontroler untuk menyelesaikan pengiriman data sebelum membaca sensor kembali atau melakukan aksi selanjutnya. Ini membantu dalam mengurangi beban pada mikrokontroler serta menghindari pompa air beroperasi terus-menerus tanpa jeda.

### **4.3 Pengujian Sistem**

Dalam bagian ini akan disajikan hasil dari pengujian prototipe yang telah dilakukan oleh peneliti di rumahnya yang terletak di Sawangan, Depok. Tempat pengujian dipilih di rumah penulis karena memungkinkan pemantauan yang konsisten dan intervensi cepat jika diperlukan selama proses pengujian dan lingkungan rumah penulis memberikan kondisi nyata yang relevan untuk menguji fungsionalitas dan keandalan prototipe dalam skenario penggunaan sehari-hari. Sebagai contoh, gambar di bawah ini menunjukkan data yang dikirim ke platform *Blynk*.



Gambar 4.15 Dashboard web *Blynk*

(Sumber: Dokumen pribadi)



Gambar 4.16 Dashboard pada aplikasi *Blynk*

(Sumber: Dokumen pribadi)

Pada gambar 4.15 dan gambar 4.16 merupakan hasil dari pengiriman data di web maupun di aplikasi *Blynk*. Apabila seluruh data berhasil dikirim pada *Blynk*, maka program akan menampilkan nilai dari sensor DHT11 dan sensor Soil Moisture.

### 4.3.1 Pengujian Hasil Pengukuran

Pengujian ini dilakukan di halaman rumah penulis selama tiga hari. Selama periode ini, 40 data dikumpulkan dengan interval waktu satu jam. Hasil pengujian ini akan menyajikan data mengenai suhu, kelembapan udara, dan kelembapan tanah yang diperoleh selama pengujian.

#### 1. Hasil data nilai suhu

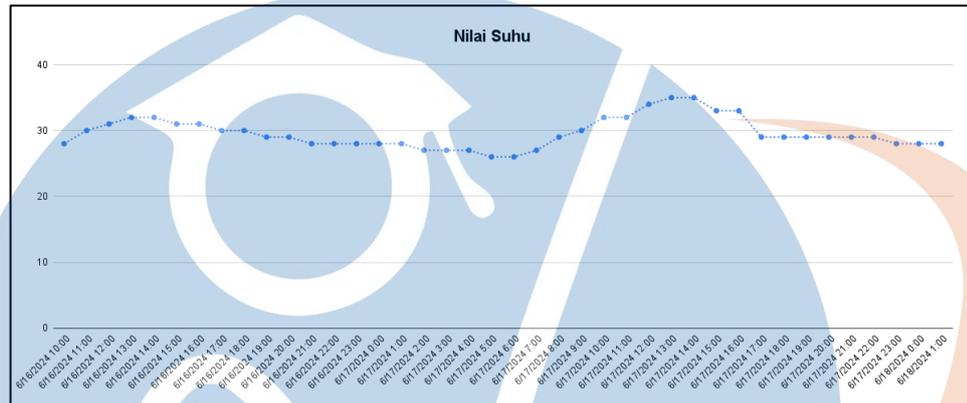
Bagian ini akan menyajikan data mengenai nilai suhu yang diperoleh selama proses pengujian alat.

Tabel 4.1 Data nilai suhu

(Sumber: Dokumen pribadi)

Waktu	Suhu (°C)	Status	Waktu	Suhu (°C)	Status
6/16/2024 10:00	28	Sesuai	6/17/2024 6:00	26	Sesuai
6/16/2024 11:00	30	Sesuai	6/17/2024 7:00	27	Sesuai
6/16/2024 12:00	31	Sesuai	6/17/2024 8:00	29	Sesuai
6/16/2024 13:00	32	Sesuai	6/17/2024 9:00	30	Sesuai
6/16/2024 14:00	32	Sesuai	6/17/2024 10:00	32	Sesuai
6/16/2024 15:00	31	Sesuai	6/17/2024 11:00	32	Sesuai
6/16/2024 16:00	31	Sesuai	6/17/2024 12:00	34	Sesuai
6/16/2024 17:00	30	Sesuai	6/17/2024 13:00	35	Sesuai
6/16/2024 18:00	30	Sesuai	6/17/2024 14:00	35	Sesuai
6/16/2024 19:00	29	Sesuai	6/17/2024 15:00	33	Sesuai
6/16/2024 20:00	29	Sesuai	6/17/2024 16:00	33	Sesuai
6/16/2024 21:00	28	Sesuai	6/17/2024 17:00	29	Sesuai
6/16/2024 22:00	28	Sesuai	6/17/2024 18:00	29	Sesuai
6/16/2024 23:00	28	Sesuai	6/17/2024 19:00	29	Sesuai
6/17/2024 0:00	28	Sesuai	6/17/2024 20:00	29	Sesuai
6/17/2024 1:00	28	Sesuai	6/17/2024 21:00	29	Sesuai
6/17/2024 2:00	27	Sesuai	6/17/2024 22:00	29	Sesuai
6/17/2024 3:00	27	Sesuai	6/17/2024 23:00	28	Sesuai
6/17/2024 4:00	27	Sesuai	6/18/2024 0:00	28	Sesuai
6/17/2024 5:00	26	Sesuai	6/18/2024 1:00	28	Sesuai

Berdasarkan data nilai suhu yang diperoleh dari tabel 4.1, dapat disimpulkan bahwa fluktuasi suhu menunjukkan pola yang konsisten, dengan suhu cenderung lebih tinggi pada siang hari dan lebih rendah pada malam hari dan dini hari. Data ini dikumpulkan selama tiga hari, dari 6/16/2024 hingga 6/18/2024, dan menunjukkan variasi yang dapat diprediksi berdasarkan waktu hari.



Gambar 4.17 Grafik nilai suhu

(Sumber: Dokumen pribadi)

Pada gambar 4.17 diperlihatkan suhu tertinggi yang tercatat adalah 35°C, yang terjadi pada 6/17/2024 pukul 14:00, sementara suhu terendah yang tercatat adalah 26°C, yang terjadi pada beberapa waktu, termasuk 6/17/2024 pukul 5:00 dan 6/18/2024 pukul 5:00. Informasi ini dapat membantu dalam mengoptimalkan pengaturan sistem penyiraman otomatis untuk memastikan efisiensi dan efektivitas yang lebih tinggi.

## 2. Hasil data nilai kelembapan udara

Bagian ini akan menyajikan data mengenai nilai kelembapan udara yang diperoleh selama proses pengujian alat.

Tabel 4.2 Hasil data nilai kelembapan udara

(Sumber: Dokumen pribadi)

Waktu	Kelembapan (%)	Status
6/16/2024 10:00	89	Sesuai
6/16/2024 11:00	81	Sesuai
6/16/2024 12:00	78	Sesuai
6/16/2024 13:00	76	Sesuai
6/16/2024 14:00	76	Sesuai
6/16/2024 15:00	77	Sesuai
6/16/2024 16:00	77	Sesuai
6/16/2024 17:00	79	Sesuai
6/16/2024 18:00	84	Sesuai
6/16/2024 19:00	86	Sesuai
6/16/2024 20:00	86	Sesuai
6/16/2024 21:00	91	Sesuai
6/16/2024 22:00	92	Sesuai
6/16/2024 23:00	90	Sesuai
6/17/2024 0:00	90	Sesuai
6/17/2024 1:00	91	Sesuai
6/17/2024 2:00	93	Sesuai
6/17/2024 3:00	94	Sesuai
6/17/2024 4:00	94	Sesuai
6/17/2024 5:00	94	Sesuai

Waktu	Kelembapan (%)	Status
6/17/2024 6:00	93	Sesuai
6/17/2024 7:00	92	Sesuai
6/17/2024 8:00	88	Sesuai
6/17/2024 9:00	82	Sesuai
6/17/2024 10:00	76	Sesuai
6/17/2024 11:00	72	Sesuai
6/17/2024 12:00	64	Sesuai
6/17/2024 13:00	59	Sesuai
6/17/2024 14:00	60	Sesuai
6/17/2024 15:00	67	Sesuai
6/17/2024 16:00	71	Sesuai
6/17/2024 17:00	80	Sesuai
6/17/2024 18:00	84	Sesuai
6/17/2024 19:00	83	Sesuai
6/17/2024 20:00	86	Sesuai
6/17/2024 21:00	88	Sesuai
6/17/2024 22:00	89	Sesuai
6/17/2024 23:00	90	Sesuai
6/18/2024 0:00	91	Sesuai
6/18/2024 1:00	93	Sesuai

Berdasarkan data kelembapan yang diperoleh dari tabel 4.2, dapat disimpulkan bahwa kelembapan udara mengalami variasi yang cukup signifikan selama periode pengujian. Kelembapan berkisar antara 59% hingga 94%. Fluktuasi kelembapan menunjukkan pola yang konsisten, dengan kelembapan cenderung lebih tinggi pada pagi hari dan malam hari, serta lebih rendah pada siang hari. Data ini dikumpulkan selama tiga hari, dari 6/16/2024 hingga 6/18/2024, dan menunjukkan variasi yang dapat diprediksi berdasarkan waktu hari.



Gambar 4.18 Grafik nilai kelembapan udara

(Sumber: Dokumen pribadi)

Berdasarkan gambar 4.18 nilai kelembapan tertinggi yang tercatat adalah 94%, yang terjadi pada beberapa waktu, termasuk 6/17/2024 pukul 2:00 dan 6/17/2024 pukul 4:00. Sementara itu, kelembapan terendah yang tercatat adalah 59%, yang terjadi pada 6/17/2024 pukul 13:00. Dengan demikian, data kelembapan yang tercatat menunjukkan bahwa kondisi lingkungan secara umum berada dalam batas yang sesuai untuk tanaman, meskipun terdapat fluktuasi harian yang perlu dipantau untuk memastikan tanaman mendapatkan kelembapan yang cukup sepanjang waktu.

### 3. Hasil data nilai kelembapan tanah

Bagian ini akan menyajikan data mengenai nilai kelembapan tanah yang diperoleh selama proses pengujian alat.

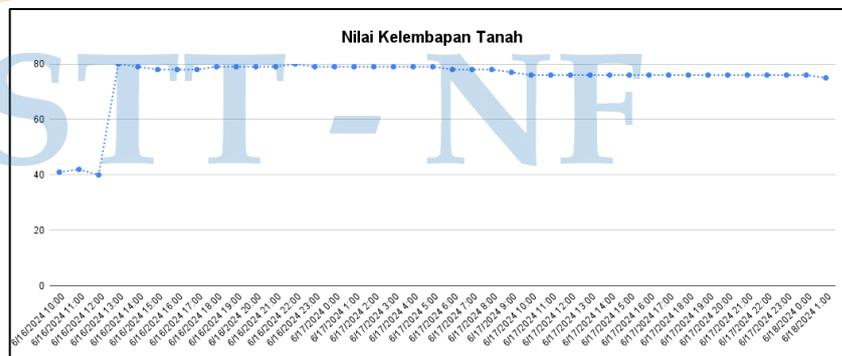
Tabel 4.3 Hasil data nilai kelembapan tanah

(Sumber: Dokumen pribadi)

Waktu	Kelembapan (%)	Status	Waktu	Kelembapan (%)	Status
6/16/2024 10:00	41	Sesuai	6/17/2024 6:00	78	Sesuai
6/16/2024 11:00	42	Sesuai	6/17/2024 7:00	78	Sesuai
6/16/2024 12:00	40	Sesuai	6/17/2024 8:00	78	Sesuai
6/16/2024 13:00	80	Sesuai	6/17/2024 9:00	77	Sesuai
6/16/2024 14:00	79	Sesuai	6/17/2024 10:00	76	Sesuai

6/16/2024 15:00	78	Sesuai	6/17/2024 11:00	76	Sesuai
6/16/2024 16:00	78	Sesuai	6/17/2024 12:00	76	Sesuai
6/16/2024 17:00	78	Sesuai	6/17/2024 13:00	76	Sesuai
6/16/2024 18:00	79	Sesuai	6/17/2024 14:00	76	Sesuai
6/16/2024 19:00	79	Sesuai	6/17/2024 15:00	76	Sesuai
6/16/2024 20:00	79	Sesuai	6/17/2024 16:00	76	Sesuai
6/16/2024 21:00	79	Sesuai	6/17/2024 17:00	76	Sesuai
6/16/2024 22:00	80	Sesuai	6/17/2024 18:00	76	Sesuai
6/16/2024 23:00	79	Sesuai	6/17/2024 19:00	76	Sesuai
6/17/2024 0:00	79	Sesuai	6/17/2024 20:00	76	Sesuai
6/17/2024 1:00	79	Sesuai	6/17/2024 21:00	76	Sesuai
6/17/2024 2:00	79	Sesuai	6/17/2024 22:00	76	Sesuai
6/17/2024 3:00	79	Sesuai	6/17/2024 23:00	76	Sesuai
6/17/2024 4:00	79	Sesuai	6/18/2024 0:00	76	Sesuai
6/17/2024 5:00	79	Sesuai	6/18/2024 1:00	75	Sesuai

Berdasarkan data dalam tabel 4.3 kelembapan tanah yang diberikan, dapat disimpulkan bahwa tanah pada umumnya berada dalam kondisi basah selama periode pengujian. Hanya satu kali tercatat kelembapan tanah berada di bawah 40%, yaitu pada 6/16/2024 pukul 12:00 dengan nilai 40%, menunjukkan bahwa tanah berada dalam kondisi kering pada waktu tersebut. Semua pengukuran lainnya menunjukkan kelembapan tanah di atas 40%, yang berarti tanah berada dalam kondisi basah.



Gambar 4.19 Grafik data nilai kelembapan tanah

(Sumber: Dokumen pribadi)

Berdasarkan grafik dari gambar 4.19 pada 6/17/2024 dan 6/18/2024, nilai kelembapan tanah berkisar antara 76% hingga 79%, yang menunjukkan bahwa tanah tetap dalam kondisi basah sepanjang hari. Tidak ada nilai kelembapan yang tercatat di bawah 40% pada tanggal-tanggal tersebut, sehingga tidak ada indikasi tanah kering. Secara keseluruhan, data menunjukkan bahwa tanah sebagian besar berada dalam kondisi basah selama periode pengamatan, dengan hanya tiga kali kejadian tanah kering yang tercatat pada 6/16/2024 pukul 10:00 sampai 12:00. Hal ini menunjukkan bahwa sistem penyiraman atau kondisi lingkungan secara umum berhasil menjaga kelembapan tanah dalam batas yang diinginkan untuk mendukung pertumbuhan tanaman.

#### 4.3.2 Pengujian Pengiriman Data

Pada bagian ini akan menjelaskan mengenai pengujian pengiriman data dalam sistem monitoring dan kontrol tanaman yang menggunakan platform *Blynk*. Pengujian ini bertujuan untuk memastikan bahwa data dari sensor kelembapan tanah dan sensor DHT11 dapat dikirim dan diterima dengan baik oleh aplikasi *Blynk*.

Tabel 4.4 Data hasil pengujian pengiriman

(Sumber: Dokumen pribadi)

No	Waktu Diterima	DHT11		Kelembapan Tanah	Relay	Status
		Suhu	Kelembapan			
1	6/16/2024 10:00	28	89	41	OFF	Berhasil
2	6/16/2024 11:00	30	81	42	OFF	Berhasil
3	6/16/2024 12:00	31	78	40	ON	Berhasil
4	6/16/2024 13:00	32	76	80	OFF	Berhasil
5	6/16/2024 14:00	32	76	79	OFF	Berhasil
6	6/16/2024 15:00	31	77	78	OFF	Berhasil
7	6/16/2024 16:00	31	77	78	OFF	Berhasil
8	6/16/2024 17:00	30	79	78	OFF	Berhasil
9	6/16/2024 18:00	30	84	79	OFF	Berhasil
10	6/16/2024 19:00	29	86	79	OFF	Berhasil

11	6/16/2024 20:00	29	86	79	OFF	Berhasil
12	6/16/2024 21:00	28	91	79	OFF	Berhasil
13	6/16/2024 22:00	28	92	80	OFF	Berhasil
14	6/16/2024 23:00	28	90	79	OFF	Berhasil
15	6/17/2024 0:00	28	90	79	OFF	Berhasil
16	6/17/2024 1:00	28	91	79	OFF	Berhasil
17	6/17/2024 2:00	27	93	79	OFF	Berhasil
18	6/17/2024 3:00	27	94	79	OFF	Berhasil
19	6/17/2024 4:00	27	94	79	OFF	Berhasil
20	6/17/2024 5:00	26	94	79	OFF	Berhasil
21	6/17/2024 6:00	26	93	78	OFF	Berhasil
22	6/17/2024 7:00	27	92	78	OFF	Berhasil
23	6/17/2024 8:00	29	88	78	OFF	Berhasil
24	6/17/2024 9:00	30	82	77	OFF	Berhasil
25	6/17/2024 10:00	32	76	76	OFF	Berhasil
26	6/17/2024 11:00	32	72	76	OFF	Berhasil
27	6/17/2024 12:00	34	64	76	OFF	Berhasil
28	6/17/2024 13:00	35	59	76	OFF	Berhasil
29	6/17/2024 14:00	35	60	76	OFF	Berhasil
30	6/17/2024 15:00	33	67	76	OFF	Berhasil
31	6/17/2024 16:00	33	71	76	OFF	Berhasil
32	6/17/2024 17:00	29	80	76	OFF	Berhasil
33	6/17/2024 18:00	29	84	76	OFF	Berhasil
34	6/17/2024 19:00	29	83	76	OFF	Berhasil
35	6/17/2024 20:00	29	86	76	OFF	Berhasil
36	6/17/2024 21:00	29	88	76	OFF	Berhasil
37	6/17/2024 22:00	29	89	76	OFF	Berhasil
38	6/17/2024 23:00	28	90	76	OFF	Berhasil
39	6/18/2024 0:00	28	91	76	OFF	Berhasil
40	6/18/2024 1:00	28	93	75	OFF	Berhasil

Berdasarkan data tabel 4.4 dapat dilihat bahwa dari total 40 entri data yang dicatat, seluruh pengiriman data berhasil diterima dengan status "Berhasil". Hal ini menunjukkan bahwa sistem transmisi data bekerja dengan baik dan dapat

diandalkan untuk memonitor kondisi lingkungan dan status sistem penyiraman secara *real-time*. Keberhasilan penerimaan data ini sangat penting untuk memastikan bahwa informasi yang diterima oleh pengguna adalah akurat dan tepat waktu, sehingga tindakan yang diperlukan dapat diambil segera. Analisis waktu pengiriman dan penerimaan data menunjukkan bahwa data dikirim dan diterima dengan interval yang konsisten, yaitu setiap jam. Hal ini menunjukkan bahwa sistem pemantauan dan pengiriman data diatur dengan baik untuk memberikan pembaruan secara berkala. Konsistensi ini penting untuk memastikan bahwa tidak ada data yang hilang atau gagal diterima, yang menunjukkan bahwa sistem memiliki tingkat keandalan yang tinggi dalam pengiriman data. Secara keseluruhan, data yang diterima menunjukkan konsistensi dalam pengukuran parameter lingkungan dan status *relay*. Tidak ada data yang hilang atau gagal diterima, yang menunjukkan bahwa sistem memiliki tingkat keandalan yang tinggi dalam pengiriman data.

#### **4.4 Evaluasi Hasil Pengujian**

Pada bagian ini akan dibahas evaluasi dari hasil pengujian yang telah dilakukan terhadap sistem monitoring dan kontrol tanaman berbasis IoT dengan menggunakan platform *Blynk*.

##### **4.4.1 Evaluasi Hasil Pengukuran**

Data hasil pengukuran menunjukkan bahwa sistem monitoring dan kontrol tanaman berhasil mengukur suhu, kelembapan tanah, dan kelembapan udara secara konsisten. Pada pengukuran suhu yang diukur berkisar antara 26°C hingga 35°C. Suhu tertinggi tercatat pada 6/17/2024 pukul 13:00 dan 14:00 dengan nilai 35°C, sementara suhu terendah tercatat pada 6/17/2024 pukul 4:00 dengan nilai 26°C. Semua pengukuran suhu memiliki status "Sesuai", menunjukkan bahwa nilai yang diukur berada dalam rentang yang diharapkan.

Pada pengukuran kelembapan udara yang diukur berkisar antara 64% hingga 94%. Kelembapan udara terendah tercatat pada 6/17/2024 pukul 12:00 dengan nilai 64%, sementara kelembapan tertinggi tercatat pada 6/17/2024 pukul 4:00 dengan nilai 94%. Semua pengukuran kelembapan udara juga memiliki status "Sesuai".

Pada pengukuran kelembapan tanah yang diukur berkisar antara 40% hingga 80%. Kelembapan tanah terendah tercatat pada 6/16/2024 pukul 12:00 dengan nilai 40%, sementara kelembapan tertinggi tercatat pada 6/16/2024 pukul 10:00 dengan nilai 41%. *Relay* yang mengontrol pompa air menunjukkan bahwa pompa air hanya aktif satu kali, yaitu pada 6/16/2024 pukul 12:00 ketika kelembapan tanah mencapai nilai 40%. Semua pengukuran kelembapan tanah juga memiliki status "Sesuai".

Untuk menghitung tingkat keberhasilan, kita menggunakan rumus:

$$\text{Tingkat keberhasilan} = \frac{\text{jumlah berhasil}}{\text{jumlah percobaan}} \times 100\%$$

Dari pengukuran jumlah suhu yang berhasil adalah 40 dari 40 percobaan, sehingga tingkat keberhasilannya adalah:

$$\text{Tingkat keberhasilan suhu} = \frac{40}{40} \times 100\% = 100\%$$

Dari pengukuran jumlah kelembapan udara yang berhasil adalah 40 dari 40 percobaan, sehingga tingkat keberhasilannya adalah:

$$\text{Tingkat keberhasilan kelembapan udara} = \frac{40}{40} \times 100\% = 100\%$$

Dari pengukuran jumlah kelembapan tanah yang berhasil adalah 40 dari 40 percobaan, sehingga tingkat keberhasilannya adalah:

$$\text{Tingkat keberhasilan kelembapan tanah} = \frac{40}{40} \times 100\% = 100\%$$

Berdasarkan hasil perhitungan, tingkat keberhasilan pengukuran suhu, kelembapan udara, dan kelembapan tanah adalah 100%. Hal ini menunjukkan bahwa sistem penyiraman otomatis bekerja dengan sangat baik dalam mengukur parameter lingkungan yang diperlukan untuk menentukan kapan penyiraman tanaman harus dilakukan. Semua pengukuran yang dilakukan sesuai dengan rentang yang diharapkan dan tidak ada kesalahan atau kegagalan dalam pengukuran.

Keberhasilan pengukuran dalam sistem penyiraman otomatis ini dapat diatributkan kepada beberapa faktor kunci yang berkontribusi secara signifikan terhadap kinerja yang optimal. Pertama, kualitas sensor yang digunakan, yaitu sensor DHT11 untuk mengukur suhu dan kelembapan udara serta sensor kelembapan tanah, memainkan peran penting. Sensor-sensor ini dikenal memiliki akurasi dan reliabilitas yang tinggi, sehingga mampu memberikan hasil pengukuran yang konsisten dan akurat. Kedua, kalibrasi yang tepat dari sensor-sensor tersebut sebelum digunakan memastikan bahwa hasil pengukuran selalu berada dalam rentang yang diharapkan. Kalibrasi yang baik adalah langkah krusial untuk memastikan bahwa sensor dapat memberikan data yang akurat dan dapat diandalkan.

#### 4.4.2 Analisis Evaluasi Pengiriman Data

Fenomena pengiriman data dalam sistem monitoring dan kontrol tanaman ini menunjukkan bahwa setiap data yang dikirimkan dari sensor ke platform *Blynk* berhasil diterima dengan baik. Data yang dikirim mencakup suhu, kelembapan udara, kelembapan tanah, dan status *relay*. Pengiriman data dilakukan secara periodik untuk memastikan bahwa kondisi lingkungan dan status sistem dapat dipantau secara *real-time*. Dari table 4.4 yang diberikan, terlihat bahwa data dikirim dan diterima dengan konsisten setiap jam, menunjukkan bahwa sistem pengiriman data berfungsi dengan baik tanpa adanya jeda atau kehilangan data.

Untuk menghitung tingkat keberhasilan pengiriman data, kita perlu menghitung jumlah total pengiriman data dan jumlah pengiriman data yang berhasil. Berdasarkan tabel yang diberikan, terdapat total 40 entri pengiriman data, dan semua entri menunjukkan status "Berhasil".

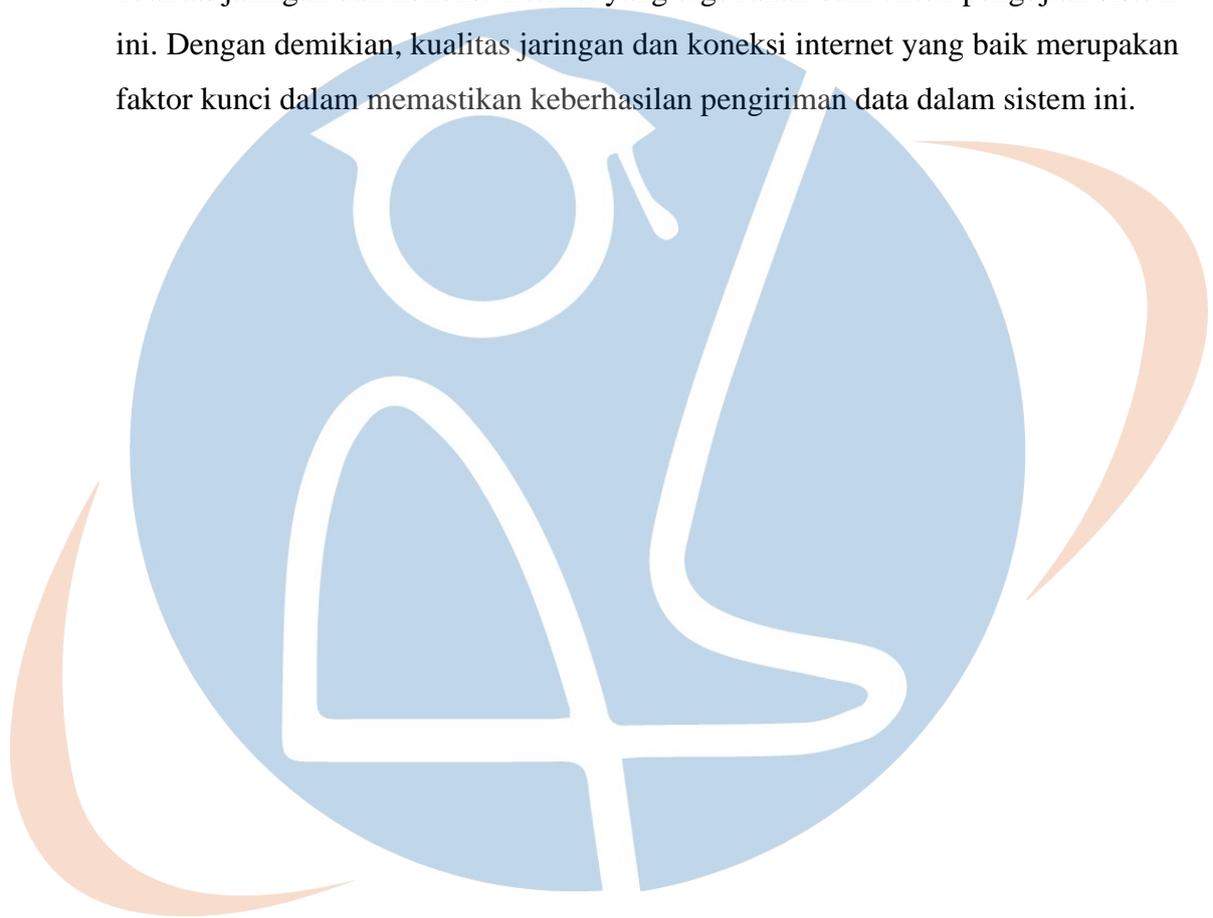
$$\text{Tingkat keberhasilan} = \frac{\text{jumlah berhasil}}{\text{jumlah percobaan}} \times 100\%$$

$$\text{Tingkat keberhasilan} = \frac{40}{40} \times 100\% = 100\%$$

Dari analisis data yang diberikan, dapat disimpulkan bahwa tingkat keberhasilan pengiriman data mencapai 100%. Ini menunjukkan bahwa sistem

pengiriman data dalam sistem penyiraman otomatis ini berfungsi dengan sangat baik, tanpa adanya kegagalan dalam pengiriman data selama periode pengujian yang tercatat dalam tabel.

Keberhasilan pengiriman data dalam sistem monitoring dan kontrol tanaman ini yang mencapai tingkat keberhasilan 100%, menunjukkan bahwa kualitas jaringan dan koneksi internet yang digunakan baik untuk pengujian sistem ini. Dengan demikian, kualitas jaringan dan koneksi internet yang baik merupakan faktor kunci dalam memastikan keberhasilan pengiriman data dalam sistem ini.



STT - NF

## BAB 5 PENUTUP

### 5.1 Kesimpulan

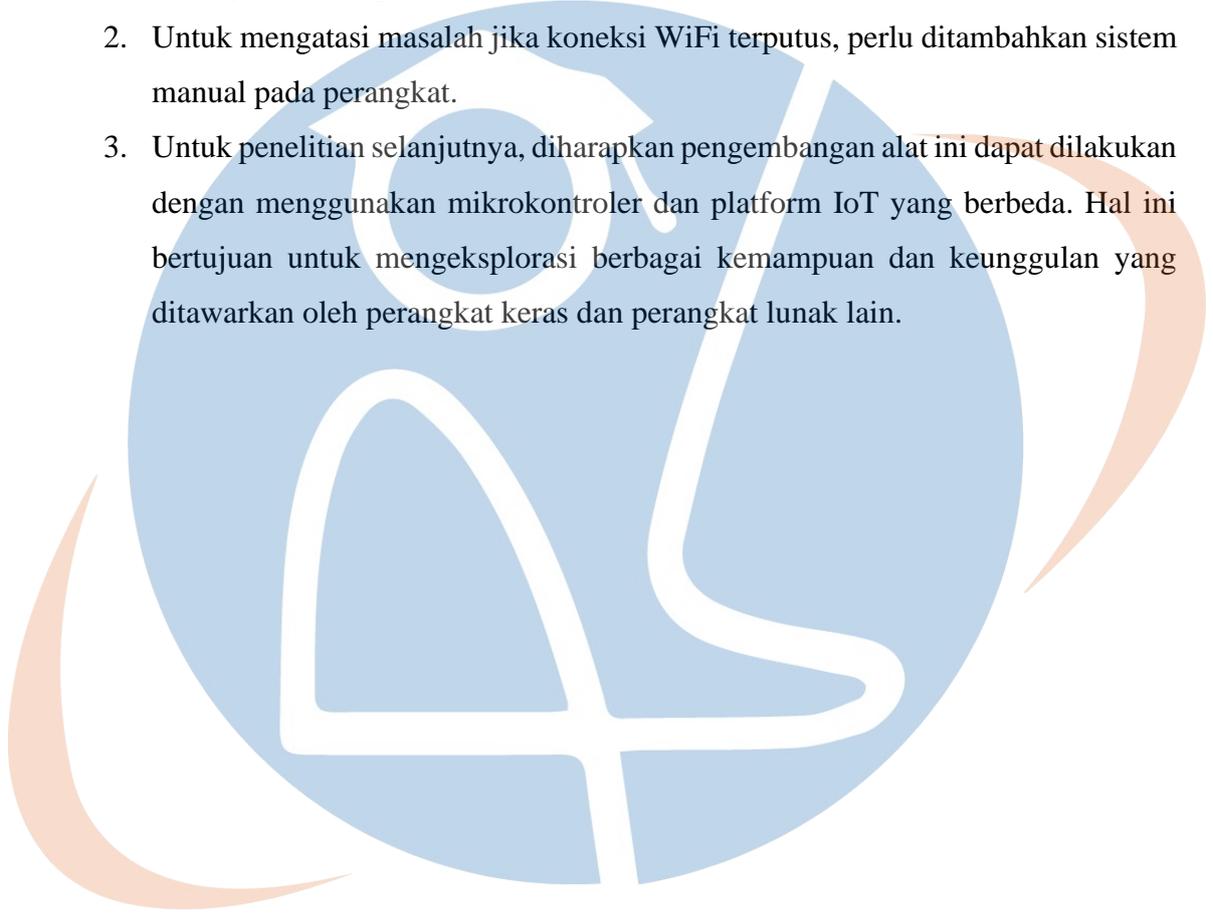
Dalam penelitian ini, penulis telah berhasil merancang dan mengimplementasikan sebuah sistem kontrol dan monitoring tanaman untuk smart garden berbasis *Internet of Things* (IoT). Sistem ini bertujuan untuk memudahkan proses penyiraman tanaman secara otomatis dengan memanfaatkan berbagai komponen elektronik dan platform IoT. Berikut adalah kesimpulan yang dapat diambil berdasarkan rumusan masalah yang telah diajukan:

1. Sistem penyiraman otomatis yang efektif untuk digunakan di halaman rumah telah dapat dirancang menggunakan ESP8266 sebagai mikrokontroler yang terhubung dengan sensor kelembapan tanah dan sensor DHT11 untuk mengukur suhu dan kelembapan udara. Data yang diperoleh dari sensor-sensor ini kemudian dikirimkan ke platform IoT *Blynk* untuk monitoring dan pengendalian lebih lanjut. Sistem ini juga dilengkapi dengan *relay* yang berfungsi untuk mengontrol pompa air yang akan menyiram tanaman secara otomatis berdasarkan kondisi kelembapan tanah yang terdeteksi. Dengan demikian, sistem ini mampu mengotomatisasi proses penyiraman tanaman, sehingga pengguna tidak perlu melakukan penyiraman secara manual.
2. Sistem *smart garden* mampu mendeteksi tingkat kelembapan tanah secara *real-time* menggunakan sensor kelembapan tanah yang mengirimkan data ke mikrokontroler ESP8266 yang telah ditanamkan program untuk prosedur penyiraman, sehingga jika tingkat kelembapan tanah berada di bawah ambang batas yang telah ditentukan, mikrokontroler akan mengaktifkan *relay* untuk menyalakan pompa air dan tanaman akan disiram hingga mencapai tingkat kelembapan yang optimal, serta sebaliknya jika tingkat kelembapan tanah sudah cukup, maka sistem akan mematikan pompa air untuk menghindari penyiraman yang berlebihan.

## 5.2 Saran

Berdasarkan hasil penelitian dan implementasi sistem kontrol dan monitoring tanaman untuk smart garden berbasis Internet of Things (IoT), berikut adalah beberapa saran untuk pengembangan lebih lanjut:

1. Penambahan sumber daya cadangan untuk memastikan sistem tetap berjalan meskipun terjadi pemadaman listrik.
2. Untuk mengatasi masalah jika koneksi WiFi terputus, perlu ditambahkan sistem manual pada perangkat.
3. Untuk penelitian selanjutnya, diharapkan pengembangan alat ini dapat dilakukan dengan menggunakan mikrokontroler dan platform IoT yang berbeda. Hal ini bertujuan untuk mengeksplorasi berbagai kemampuan dan keunggulan yang ditawarkan oleh perangkat keras dan perangkat lunak lain.



STT - NF

## DAFTAR PUSTAKA

- [1] V. Omega *et al.*, “Smart Garden Berbasis Internet of Things,” *Jtim*, vol. 6, no. 1, pp. 36–42, 2023.
- [2] J. Andika, E. Permana, and S. Attamimi, “Perancangan Sistem Otomatisasi dan Monitoring Perangkat Perawatan Tanaman Hias Berbasis Internet of Things,” *J. Teknol. Elektro*, vol. 13, no. 2, p. 100, 2022, doi: 10.22441/jte.2022.v13i2.007.
- [3] M. Saiqul Umam, S. Adi Wibowo, and Y. Agus Pranoto, “Implementasi Protokol Mqtt Pada Aplikasi Smart Garden Berbasis Iot (Internet of Things),” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 1, pp. 899–906, 2023, doi: 10.36040/jati.v7i1.6131.
- [4] I. W. B. Darmawan, I. N. S. Kumara, and D. C. Khrisne, “Smart Garden Sebagai Implementasi Sistem Kontrol Dan Monitoring Tanaman Berbasis Teknologi Cerdas,” *J. SPEKTRUM*, vol. 8, no. 4, p. 161, Jan. 2022, doi: 10.24843/SPEKTRUM.2021.v08.i04.p19.
- [5] A. Prihanto, N. Rachmawati, and A. Prapanca, “Smart Garden Automation Dengan Memanfaatkan Teknologi Berbasis Internet Of Things (IoT),” *J. Inf. Eng. Educ. Technol.*, vol. 5, no. 2, pp. 55–60, 2021, doi: 10.26740/jieet.v5n2.p55-60.
- [6] A. P. S. P. Kristyawan, R. Mustaricihie, and L. A. Wardoyo, “Review Artikel: Pengaplikasian Internet of Things (IoT) dalam Manufaktur Industri Farmasi di Era Industri 4.0,” *J. Farmaka*, vol. 20, no. 1, pp. 105–112, 2022.
- [7] P. Hidayatullah, M. Orisa, and A. Mahmudi, “Rancang Bangun Sistem Monitoring Dan Kontrol Tanaman Hidroponik Berbasis Internet of Things (Iot),” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 6, no. 2, pp. 1200–1207, 2023, doi: 10.36040/jati.v6i2.5433.
- [8] N. Effendi, W. Ramadhani, and F. Farida, “Perancangan Sistem Penyiraman Tanaman Otomatis Menggunakan Sensor Kelembapan Tanah Berbasis IoT,”

- J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 3, no. 2, pp. 91–98, 2022, doi: 10.37859/coscitech.v3i2.3923.
- [9] J. E. Candra and A. Maulana, “Penerapan Soil Moisture Sensor Untuk Desain System Penyiram Tanaman Otomatis,” *Pros. Semin. Nas. Ilmu Sos. dan Teknol.*, no. September, pp. 109–114, 2019.
- [10] L. A. Y. Merbawani, M. Rivai, and H. Pirngadi, “Sistem Monitoring Profil Kedalaman Tingkat Kelembapan Tanah Berbasis IoT dan LoRa,” *J. Tek. ITS*, vol. 10, no. 2, 2021, doi: 10.12962/j23373539.v10i2.68613.
- [11] A. Y. Rangan, Amelia Yusnita, and Muhammad Awaludin, “Sistem Monitoring berbasis Internet of things pada Suhu dan Kelembaban Udara di Laboratorium Kimia XYZ,” *J. E-Komtek*, vol. 4, no. 2, pp. 168–183, 2020, doi: 10.37339/e-komtek.v4i2.404.
- [12] I. Nurpriyanti, “Otomatisasi sensor dht11 sebagai sensor suhu dan kelembapan pada hidroponik berbasis arduino uno R3 untuk tanaman kangkung,” *J. Teknol. dan Terap. Bisnis*, vol. 3, no. 1, pp. 40–45, 2020.
- [13] M. Natsir, D. B. Rendra, and A. D. Y. Anggara, “Implementasi IOT Untuk Sistem Kendali AC Otomatis Pada Ruang Kelas di Universitas Serang Raya,” *J. PROSISKO (Pengembangan Ris. dan Obs. Rekayasa Sist. Komputer)*, vol. 6, no. 1, pp. 69–72, 2019.
- [14] Ahmad Hunaepi, Ahmad Roihan, and Ahmad Nurtursina, “Perancangan Sistem Kehadiran Pendidik Dan Tenaga Kependidikan Berbasis Mikrokontroler Esp32Cam,” *J. Sist. Inf. dan Teknol.*, vol. 3, no. 2, pp. 61–67, 2023, doi: 10.56995/sintek.v3i2.60.
- [15] I. G. Friansyah, Safe’I, and D. F. Waidah, “Implementasi Sistem Bluetooth Menggunakan Android Dan Arduino Untuk Kendali Peralatan Elektronik,” *J. TIKAR*, vol. 2, no. 2, pp. 121–127, 2021.
- [16] M. Artiyasa, A. Nita Rostini, Edwinanto, and Anggy Pradifita Junfithrana, “Aplikasi Smart Home Node Mcu Iot Untuk Blynk,” *J. Rekayasa Teknol. Nusa Putra*, vol. 7, no. 1, pp. 1–7, 2021, doi: 10.52005/rekayasa.v7i1.59.

- [17] I. Syukhron, "Penggunaan Aplikasi Blynk untuk Sistem Monitoring dan Kontrol Jarak Jauh pada Sistem Kompos Pintar berbasis IoT," *Electrician*, vol. 15, no. 1, pp. 1–11, 2021, doi: 10.23960/elc.v15n1.2158.
- [18] R. P. Gozal, A. Setiawan, and H. Khoswanto, "Aplikasi SmartRoom Berbasis Blynk untuk Mengurangi Pemakaian Tenaga Listrik," *J. Infra*, vol. 8, no. 1, pp. 39–45, 2020, [Online]. Available: <https://publication.petra.ac.id/index.php/teknik-informatika/article/view/9753>
- [19] S. P. Santoso and F. Wijayanto, "Rancang Bangun Akses Pintu Dengan Sensor Suhu Dan Handsanitizer Otomatis Berbasis Arduino," *J. Elektro*, vol. 10, no. 1, pp. 1–31, 2022.
- [20] M. Irsyam and A. Tanjung, "Sistem Otomasi Penyiraman Tanaman Berbasis Telegram," *Sigma Tek.*, vol. 2, no. 1, p. 81, 2019, doi: 10.33373/sigma.v2i1.1834.
- [21] K. Imtihan and L. Mutawalli, "Penerapan Research and Development (R&D) Dalam Membangun Alat Penyiraman Tanaman Otomatis Berbasis Arduino," *J. Manaj. Inform. ...*, vol. 5, 2022, [Online]. Available: <http://e-journal.stmiklombok.ac.id/index.php/misi/article/view/582%0Ahttp://e-journal.stmiklombok.ac.id/index.php/misi/article/download/582/175>
- [22] R. Damara Ritonga and H. Munandar, "Penerapan Iot Pada Prototype Smarthome Dengan Kontrol Website," *Semin. Nas. Mhs. Fak. Teknol. Inf. Jakarta-Indonesia*, no. September, pp. 1246–1255, 2022.