PROTOTYPE SISTEM PENYIRAMAN TANAMAN APEL BERBASIS *INTERNET OF THINGS (IOT)*

TUGAS AKHIR

Disusun Untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya

OLEH

YOGA ANDITIA
C2155201003
PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2025

PROTOTYPE SISTEM PENYIRAMAN TANAMAN APEL BERBASIS *INTERNET OF THINGS (IOT)*

TUGAS AKHIR

Disusun Untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangkaraya

OLEH

YOGA ANDITIA
C2155201003
PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
2025

PERSETUJUAN

PROTOTYPE SISTEM PENYIRAMAN TANAMAN APEL BERBASIS *INTERNET OF THINGS (IOT)*

Tugas Akhir Ini Telah Disetujui dan Disahkan

Pembimbing I,	Pembimbing II,
Elia Zakharia, M.T.	Rommi Kaestria, M.Kom.
NIK. 199205262016104	NIK. 198605242011103

Mengetahui
Ketua STMIK Palangkaraya,

Ir. Hj. Siti Maryamah, M.M.
NIK. 196902021995004

PENGESAHAN

PROTOTYPE SISTEM PENYIRAMAN TANAMAN APEL BERBASIS *INTERNET OF THINGS (IOT)*

Tugas Akhir ini Telah Diujikan, Dinilai, dan Disahkan
Oleh Tim Penguji pada Tanggal 28 Mei 2025

Tim Penguji Tugas Akhir :

1. Sam'ani, S.T., M.Kom. ..... 
   Ketua

2. Elok Faiqotul Himmah, S.Si., M.Sc. ..... 
   Sekretaris

3. Ir. Abdul Hadi, S.T., M.Kom. ..... 
   Anggota

4. Elia Zakharia, M.T. ..... 
   Anggota

5. Rommi Kaestria, M.Kom. ..... 
   Anggota

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama Mahasiswa	:	**YOGA ANDITIA**
N I M	:	**C2155201003**

menyatakan bahwa Tugas Akhir dengan judul :

PROTOTYPE* SISTEM PENYIRAMAN TANAMAN APEL BERBASIS *INTERNET OF THINGS (IOT)

adalah hasil karya saya dan bukan merupakan duplikasi sebagian atau seluruhnya dari karya orang lain, kecuali bagian yang sumber informasi dicantumkan.

Pernyataan ini dibuat dengan sebenar-benarnya secara sadar dan bertanggungjawab dan saya bersedia menerima sanksi pembatalan Tugas Akhir apabila terbukti melakukan duplikasi terhadap Tugas Akhir atau karya ilmiah lain yang sudah ada.

Palangka Raya, 28 Mei 2025

Yang Membuat Pernyataan,



YOGA ANDITIA

MOTTO DAN PERSEMBAHAN

"Jangan pernah berhenti ketika gagal, teruslah mengucap syukur dengan apa yang kau miliki sebab orang lain mungkin tak memiliki apa yang kau punya."

Dengan penuh rasa syukur atas berkat dan penyertaan-Nya yang senantiasa menyertai, karya ini saya persembahkan kepada:

1. Kedua orang tua tercinta**,** yang dengan kasih sayang, doa, dukungan, serta semangat yang tak pernah surut, selalu menjadi sumber kekuatan dan inspirasi dalam setiap langkah penulis.
2. Bapak Elia Zakharia, M.T. dan Bapak Rommi Kaestria M.Kom.**,** selaku dosen pembimbing yang dengan kesabaran, ketulusan, dan dedikasi tinggi telah membimbing serta memberikan arahan, saran, dan motivasi selama proses penyusunan Tugas Akhir ini.
3. Rekan-rekan seperjuangan TI-A Angkatan 2021**,** yang telah menjadi bagian dari perjalanan penuh kenangan dan kebersamaan selama masa perkuliahan, serta senantiasa memberikan semangat dalam suka maupun duka.

INTISARI

Yoga Anditia, C2155201003, 2025. *Prototype* Sistem Penyiraman Tanaman Apel Berbasis *Internet of Things (IoT),* Pembimbing I Elia Zakharia, M.T, Pembimbing II Rommi Kaestria, M.Kom.

Budidaya apel di dataran rendah menghadapi tantangan seperti suhu tinggi dan kelembapan udara rendah yang membuat tanah cepat kering. Kondisi ini memerlukan sistem penyiraman yang efisien dan akurat. Penelitian ini bertujuan merancang *prototype* sistem penyiraman otomatis berbasis *Internet of Things (IoT)* untuk memantau dan menyesuaikan kebutuhan air tanaman secara *real-time* agar pertumbuhan apel tetap optimal.

Metode yang digunakan adalah *Research and Development (R&D)*. Sistem dirancang menggunakan NodeMCU ESP8266 yang terhubung dengan *sensor soil moisture*, sensor suhu DS18B20, dan DHT11. Sistem mengendalikan pompa air dan *motor servo* untuk atap otomatis. Data ditampilkan pada LCD dan dikirim ke aplikasi *Blynk* agar pengguna bisa memantau dan mengontrol dari jarak jauh.

Hasil pengujian menunjukkan sistem mampu menyiram otomatis saat tanah kering dan menutup atap saat suhu tinggi. Semua data sensor tampil secara *real-time* di LCD dan *Blynk*. Sistem juga dapat dikendalikan secara manual melalui aplikasi. *Prototype* terbukti efektif menjaga kondisi tanaman, menghemat waktu dan tenaga, serta mendukung pertumbuhan apel secara optimal.

Kata kunci: Internet of Things (IoT), penyiraman otomatis, tanaman apel, sensor kelembapan tanah, NodeMCU ESP8266, aplikasi Blynk.

ABSTRACT

Yoga Anditia, C2155201003, 2025. *Internet of Things (IoT) Based Apple Plant Watering System Prototype, Supervisor I Elia Zakharia, M.T, Supervisor II Rommi Kaestria, M.Kom.*

Apple cultivation in the lowlands faces challenges such as high temperatures and low humidity which cause the soil to dry out quickly. This condition requires an efficient and accurate watering system. This research aims to design a prototype automatic watering system based on the Internet of Things (IoT) to monitor and adjust plant water needs in real-time so that apple growth remains optimal..

The method used is Research and Development (R&D). The system is designed using a NodeMCU ESP8266 connected to a soil moisture sensor, DS18B20 temperature sensor, and DHT11. The system controls water pumps and servo motors for automatic roofing. Data is displayed on the LCD and sent to the Blynk app so users can monitor and control remotely.

Test results show the system is able to water automatically when the ground is dry and close the roof when temperatures are high. All sensor data appears in real-time on the LCD and Blynk. The system can also be controlled manually via the application. The prototype has proven to be effective in maintaining plant conditions, saving time and energy, and supporting optimal apple growth.

Keywords: Internet of Things (IoT), automatic watering, apple plants, soil moisture sensor, NodeMCU ESP8266, Blynk application.

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa atas berkat dan penyertaan-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini yang berjudul "*PROTOTYPE* SISTEM PENYIRAMAN TANAMAN APEL BERBASIS *INTERNET OF THINGS (IOT)*". Tugas akhir ini disusun sebagai salah satu syarat untuk mencapai gelar sarjana strata satu pada Program Studi Teknik Informatika di Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya.

Pada kesempatan ini, penulis juga menyampaikan rasa terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan bimbingan, motivasi, dan semangat dalam menyelesaikan penulisan Tugas Akhir ini, yaitu kepada:

1. **Elia Zakharia, M.T.**, selaku dosen Pembimbing I yang telah meluangkan waktu dan tenaganya untuk memberikan arahan dan bimbingan selama proses penyusunan Tugas Akhir ini.
2. **Rommi Kaestria, M.Kom.**, selaku dosen Pembimbing II yang juga banyak memberikan saran dan masukan berharga dalam penulisan Tugas Akhir ini.
3. Teman-teman kelas dan seluruh rekan mahasiswa yang telah memberikan dukungan, semangat, serta kebersamaan selama masa perkuliahan dan penyelesaian Tugas Akhir ini.

Penulis sangat mengharapkan adanya masukan, saran, dan kritik yang bersifat membangun demi penyempurnaan Tugas Akhir ini. Semoga karya ini dapat memberikan manfaat, tidak hanya bagi penulis sendiri, tetapi juga bagi para pembaca, serta dapat menambah ilmu pengetahuan di bidang yang berkaitan.

Akhir kata, penulis berharap bahwa Tugas Akhir ini dapat menjadi karya yang bermanfaat. Semoga Tuhan Yang Maha Esa senantiasa melimpahkan karunia-Nya kepada kita semua.

Palangkaraya, 17 Mei 2025

Penulis

DAFTAR ISI

PERSETUJUAN iii
PENGESAHAN iv
PERNYATAAN KEASLIAN v
MOTTO DAN PERSEMBAHAN vi
INTISARI vii
ABSTRACT viii
KATA PENGANTAR ix
DAFTAR ISI x
DAFTAR TABEL xii
DAFTAR GAMBAR xiii
DAFTAR LAMPIRAN xv
BAB I PENDAHULUAN 1
1.1 Latar Belakang Masalah 1
1.2 Perumusan Masalah 3
1.3 Batasan Masalah 3
1.4 Tujuan dan Manfaat 4
1.5 Sistematika Penulisan 5
BAB II TINJAUAN PUSTAKA 7
2.1 Kajian Teori 7
2.2 Penelitian Yang Relevan 19
BAB III METODOLOGI PENELITIAN 26
3.1 Tinjauan Umum 26
3.2 Jenis penelitian 27
3.3 Desain Prototype 28
3.4 Instrumen Penelitian 31
3.5 Analisis Kebutuhan 33
3.6 Desain 37
BAB IV HASIL DAN PEMBAHASAN 42
4.1 Hasil 42
4.2 Pembahasan 56

4.3 Penelitian 66
BAB V PENUTUP 73
5.1 Kesimpulan 73
5.2 Saran 74
DAFTAR PUSTAKA 75
LAMPIRAN 78

DAFTAR TABEL

Tabel 2. 1 Penelitian yang Relevan 19
Tabel 3. 1 Perangkat Keras 33
Tabel 3. 2 Diagram Alur 35
Tabel 4. 1 Pengujian Black Box 48
Tabel 4. 2 Tabel Pengamatan 70

DAFTAR GAMBAR

Gambar 2. 1 NodeMCU ESP8266 8
Gambar 2. 2 Sensor Soil Moisture 9
Gambar 2. 3 Temperature DS1820 (Water Resist) 10
Gambar 2. 4 Sensor Humidity DHT11 10
Gambar 2. 5 Baseplate NodeMCU ESP-8266 11
Gambar 2. 6 Liquid Crystal Display (LCD) 12
Gambar 2. 7 Relay Module 1CH 13
Gambar 2. 8 Submersible Water Pump 5v DC 13
Gambar 2. 9 12C/TWL LCD 1602 14
Gambar 2. 10 Motor Servo 15
Gambar 2. 11 Kabel Jumper 15
Gambar 3. 1 Flowchart Prototype 29
Gambar 3. 2 Desain Arsitektur Perangkat 37
Gambar 3. 3 Desain Rangkaian Perangkat Keras 38
Gambar 3. 4 Prototype Sistem Penyiraman Tanaman Apel Berbasis Internet of Thing (IoT) 40
Gambar 4. 1 Apel 18
Gambar 4.2 Implementasi perangkat keras 44
Gambar 4. 3 Prototype Sistem Penyiraman Tanaman Apel Berbasis Internet of Things (IoT) 45
Gambar 4. 4 Proses program 46
Gambar 4. 5 Tampilan awal aplikasi blynk 47
Gambar 4. 6 Tampilan user interface 47
Gambar 4. 7 NodeMCU ESP8266 Berhasil Terkoneksi 49
Gambar 4. 8 Sensor Kelembapan Tanah Berfungsi 50
Gambar 4. 9 Sensor Suhu(Temp) Berfungsi 51
Gambar 4. 10 Sensor Kelebapan Udara(Humidity) Berfungsi 51
Gambar 4. 11 Pengiriman Data Secara Real-Time Berhasil 52
Gambar 4. 12 Kontrol Manual Pompa Berfungsi 53
Gambar 4. 13 Kontrol Manula Buka-Tutup Atap Berfungsi 53
Gambar 4. 14 Mode Pompa Otomatis Berfungsi 54
Gambar 4. 15 Mode Otomatis Buka-Tutup Atap Berfungsi 55
Gambar 4. 16 LCD Berfungsi 55
Gambar 4. 17 Code Program 56
Gambar 4. 18 Konfigurasi Blynk 57
Gambar 4. 19 Library 57
Gambar 4. 20 Pin dan Objek 58
Gambar 4. 21 Variabel Kalibrasi dan Kontrol 59
Gambar 4. 22 WiFi dan Auth 60
Gambar 4. 23 Fungsi Blynk 61
Gambar 4. 24 Setup() 62

Gambar 4. 25 Loop() 64
Gambar 4. 26 Fungsi aturAtap() 66
Gambar 4. 27 Pengamatan Minggu Pertama 67
Gambar 4. 28 Penelitian Minggu Kedua 68
Gambar 4. 29 Penelitian Minggu Ketiga 69
Gambar 4. 30 Before & After 71

DAFTAR LAMPIRAN

Lampiran 1. Surat tugas pembimbing tugas akhir
Lampiran 2. Lembar konsultasi bimbingan Tugas Akhir
Lampiran 3. Surat tugas penguji seminar
Lampiran 4. Berita acara penilaian sidang TA
Lampiran 5. Pengujian sistem Black Box pada Penelitian

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Apel merupakan salah satu jenis buah yang unggul dan sangat digemari dan dikonsumsi masyarakat (Wijaya & Ridwan, 2019). Namun, budidaya apel memerlukan perawatan khusus, terutama dalam hal pengelolaan penyiraman dan pemeliharaan, yang menjadi lebih menantang ketika apel dibudidayakan di dataran rendah. Kondisi lingkungan di dataran rendah, seperti suhu udara yang lebih panas dan kelembapan udara yang lebih rendah, membuat tanah lebih cepat kehilangan air. Hal ini mengharuskan petani untuk mengelola kelembapan tanah dengan cermat, karena penyiraman yang tidak teratur serta penyebaran air yang tidak merata dapat menyebabkan tanaman stres dan pertumbuhan tidak optimal.

Tanaman apel membutuhkan kondisi lingkungan yang spesifik untuk pertumbuhan optimal. Tanah harus memiliki kadar kelembapan yang cukup tanpa menjadi terlalu lembek atau tergenang air. Suhu yang ideal untuk pertumbuhan tanaman apel berkisar antara 16°C hingga 27°C, sementara kelembapan udara yang dibutuhkan berada pada rentang 75-85%. Selain itu, intensitas cahaya matahari yang dibutuhkan oleh tanaman apel adalah sekitar 50-60%, terutama selama masa pembungaan. Ketidaksesuaian kondisi ini, seperti kelembapan tanah yang terlalu rendah, suhu yang terlalu tinggi, atau paparan sinar matahari yang berlebihan, dapat menyebabkan tanaman mengering, pertumbuhan yang tidak optimal (Rahaju et al., 2014).

Di dataran rendah, salah satu tantangan utama adalah bagaimana pengelolaan air dan paparan sinar matahari yang konsisten. Penyiraman yang terlalu banyak dapat menyebabkan tanah tergenang, sedangkan penyiraman yang terlalu sedikit membuat tanaman kekurangan air, keduanya berpotensi menghambat pertumbuhan (Anggara et al., 2017). Oleh karena itu, diperlukan alat penyiraman yang mampu menyesuaikan secara otomatis dengan kondisi kelembapan tanah, sehingga tanaman dapat tumbuh secara optimal. Selain itu, perlindungan terhadap sinar matahari yang berlebih juga menjadi hal yang penting untuk menjaga kesehatan tanaman.

Seiring perkembangan teknologi, penerapan *Internet of Things (IoT)* mulai memberikan solusi yang lebih baik dalam bidang pertanian. Teknologi IoT memungkinkan petani untuk memantau dan mengontrol kondisi tanaman secara *real-time* melalui perangkat yang terhubung ke internet. Dalam budidaya apel, teknologi ini dapat diterapkan pada *prototype* penyiraman otomatis dan kontrol pencahayaan. Dengan memanfaatkan sensor kelembapan tanah, *prototype* dapat mengaktifkan penyiraman otomatis hanya ketika tanaman membutuhkan air, sementara sensor suhu dapat digunakan untuk mengelola intensitas sinar matahari dengan mengaktifkan penutup otomatis jika suhu berlebihan. Penerapan teknologi IoT ini diharapkan mampu mengelola lingkungan tanaman apel di dataran rendah, sehingga tumbuh tetap optimal.

Oleh karena itu saya mengambil Tugas Akhir dengan judul "*Prototype* Sistem Penyiraman Tanaman Apel Berbasis *Internet of Things (IoT)*" karena budidaya apel, khususnya di dataran rendah, menghadapi tantangan besar dalam

pengelolaan kelembapan tanah dan paparan sinar matahari yang tidak konsisten. Kondisi suhu yang lebih panas dan kelembapan udara yang rendah menyebabkan tanah lebih cepat kering, sementara penyiraman yang tidak teratur dapat mengakibatkan stres pada tanaman. Untuk itu, diperlukan sistem yang mampu menyesuaikan kebutuhan air secara otomatis berdasarkan kondisi kelembapan tanah. Dengan memanfaatkan teknologi *Internet of Things (IoT)*, *prototype* ini dirancang untuk mengotomatiskan penyiraman tanaman dan memantau kondisi lingkungan secara *real-time*, sehingga tanaman apel tetap dapat tumbuh optimal meskipun berada di lingkungan yang kurang ideal.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dibuat, maka dirumuskan masalahnya yaitu bagaimana cara merancang *prototype* penyiraman dan buka tutup atap berbasis IoT pada tanaman apel supaya lebih mudah saat melakukan penyiraman.

1.3 Batasan Masalah

Agar penelitian ini dapat lebih fokus dan terarah, beberapa batasan masalah berikut ditetapkan:

a. Penelitian ini berfokus pada tanaman apel di daerah dataran rendah, di mana tantangan utamanya adalah kondisi iklim yang panas dan curah hujan yang relatif rendah, sehingga memerlukan pengelolaan air dan pengelolaan paparan sinar matahari yang yang dapat menghemat waktu dan tenaga.
b. Penelitian hanya berfokus pada penerapan *Prototype* Sistem Penyiraman Tanaman Apel Berbasis IoT .

c. *Prototype* yang dikembangkan menggunakan sensor *Soil Moisture*, sensor suhu *DS1820 Water Resist*, *NodeMCU V3*, dan integrasi IoT.

d. Penelitian ini hanya mencakup *prototype* sistem penyiraman dan *prototype* sistem buka tutup paparan sinar matahari terhadap tanaman.

e. Penelitian ini tidak mencakup aspek lain seperti pemupukan maupun pengendalian hama.

f. Penelitian ini hanya berfokus pada pembibitan tanaman apel.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan penelitian ini adalah merancang *prototype* penyiraman otomatis dan buka tutup pengontrol paparan sinar matahari menggunakan *NodeMCU V3* serta menggunakan sensor *Soil Moisture* untuk mengukur kelembapan tanah.

1.4.2 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat, yaitu :

a. Bagi Penulis

Mengenai Manfaat dari penelitian ini adalah menerapkan ilmu yang didapat diperkuliahan dan mengembangkannya di bidang mikrokontroler.

b. Bagi STMIK Palangkaraya

Dengan adanya penelitian ini kampus dapat memberikan akses kepada generasi berikutnya guna mempelajari lebih lanjut ilmu mengenai penerapan IoT di bidang pertanian, sehingga dapat mendukung proses belajar-mengajar

dan kegiatan riset di masa depan. Selain itu juga, penelitian ini dapat memperkuat posisi kampus sebagai pusat inovasi yang menghasilkan karya ilmiah aplikatif yang bermanfaat bagi masyarakat luas.

c. Bagi Pengguna

Penelitian ini bertujuan menghasilkan *prototype* penyiraman otomatis yang mampu mengoptimalkan penggunaan air dan menjaga kelembapan tanah untuk tanaman apel. Dengan teknologi ini, petani dapat menghemat waktu, tenaga, serta dapat dipantau dari jarak jauh melalui aplikasi *Blynk*. Hal ini sangat mendukung pertumbuhan tanaman apel secara optimal.

1.5 Sistematika Penulisan

Adapun sistematika penulisan dalam penulisan tugas akhir ini terdiri dari beberapa bab dan masing-masing bab membahas dan menguraikan pokok permasalahan yang berbeda, sebagai gambaran disini penulis menyertakan garis-garis besarnya yaitu :

BAB I PENDAHULUAN

Bab ini berisikan latar belakang masalah, rumusan masalah, Batasan masalah, tujuan dan manfaat, serta sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisikan tentang tinjauan pustaka yang diambil dari penelitian yang relevan beserta susunan kajian teori yang disesuaikan dengan tema Tugas Akhir.

BAB III METODOLOGI PENELITIAN

Bab ini berisikan tentang tahapan yang dilakukan peneliti dalam mengumpulkan informasi atau data yang dibutuhkan.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisikan tentang hasil dari penelitian dan pembahasan darihasil yang didapat.

BAB V KESIMPULAN DAN SARAN

Bab ini berisikan tentang kesimpulan dan saran yang diberikan untuk pengembangan selanjutnya.

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Teori

2.1.1 Prototype

Model *Prototype* didefinisikan sebagai model atau metode yang digunakan untuk merancang perangkat lunak tanpa menentukan persyaratan input, pemrosesan, dan output secara rinci karena pengembang atau pemrogram tidak terlalu yakin tentang efesiensi algoritma, penyesuaian sistem operasi, atau format tampilan antarmuka (Meisak et al., 2022).

2.1.2 Internet of Things (IoT)

Internet of Things (IoT) adalah sistem komputasi di mana perangkat yang saling terhubung, termasuk mesin mekanik dan digital, benda, hewan, atau individu disertai pengidentifikasi unik. Sistem ini memungkinkan transfer data lewat jaringan dengan tidak butuh adanya interaksi langsung antar orang dengan orang atau orang dengan komputer (Isnawati & Ali, 2024).

2.1.3 NodeMCU ESP8266

NodeMCU adalah pengembangan dari ESP8266 yang dilengkapi dengan *firmware* berbasis *e-Lua*. Modul ini memiliki *port micro USB* yang berfungsi sebagai sumber daya sekaligus untuk proses pemrograman. *NodeMCU* juga dilengkapi dengan dua tombol *push*, yaitu tombol *reset* dan *flash*. Bahasa pemrograman utama yang digunakan adalah *Lua*, yang merupakan bagian dari paket ESP8266. Meskipun logika dan struktur pemrograman *Lua* mirip dengan C, kedua bahasa ini memiliki perbedaan pada *syntax*. Jika memilih menggunakan

bahasa *Lua*, pengembang dapat memanfaatkan *tool* seperti *Lua Loader* atau *Lua Uploader*.

Selain itu, *NodeMCU* juga mendukung pemrograman menggunakan *Arduino IDE* setelah melakukan penyesuaian pada *Board Manager*. Sebelum digunakan, *NodeMCU* harus di-*flash* agar kompatibel dengan *tool* yang diinginkan. Jika menggunakan *Arduino IDE*, *firmware* yang sesuai adalah keluaran *AiThinker* yang mendukung *AT Command*. Namun, jika memakai *tool loader*, *firmware* yang digunakan adalah *firmware* khusus *NodeMCU* (Ummah, 2019).



Gambar 2. 1 NodeMCU ESP8266
Sumber gambar (Ummah,2019)

2.1.4 Sensor Soil Moisture

Sensor kelembapan tanah kapasitif (*Capacitive Soil Moisture Sensor*) adalah sensor untuk mendeteksi kelembapan tanah, sensor ini dapat diintegrasikan dengan mikrokontroler seperti *Arduino*. Sensor kelembapan tanah ini umumnya dimanfaatkan pada sistem penyiraman otomatis atau untuk memantau kelembapan tanah tanaman secara *offline* maupun *online* (Suska, 2016).

Sensor kelembapan tanah bekerja dengan mendeteksi tingkat kelembapan tanah melalui perubahan tegangan *output*. Ketika tanah dalam kondisi kering,

tegangan *output* sensor akan meningkat. Sebaliknya, jika tanah terdeteksi basah, tegangan *output* akan menurun. Tegangan *output* ini dapat dimonitor melalui serial monitor pada *Arduino IDE* dalam bentuk nilai analog, berkisar antara 600 hingga 2900. Nilai analog tersebut kemudian digunakan sebagai indikator untuk menentukan tingkat kelembapan tanah.



Gambar 2. 2 Sensor Soil Moisture
Sumber gambar (Suska,2016)

2.1.5 Sensor Temperature DS1820 (Water Resist)

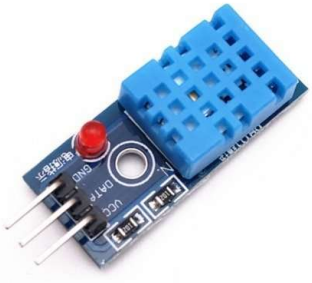
Sensor suhu merupakan komponen yang dapat mengubah kalor menjadi besaran listrik sehingga dapat mendeteksi gejala perubahan suhu pada objek tertentu. Sensor suhu mengukur besaran energi panas/dingin yang dihasilkan oleh suatu objek sehingga memungkinkan kita mengetahui atau mendeteksi gejala perubahan suhu tersebut dalam bentuk keluaran analog maupun digital (Hariono et al., 2021).



Gambar 2. 3 Temperature DS1820 (Water Resist)
Sumber gambar (Rozaq & DS, 2017)

2.1.6 Sensor Humidity DHT11

DHT11 adalah salah satu sensor yang dapat mengukur dua parameter lingkungan sekaligus, yakni suhu dan kelembapan udara *(humidity)*. Dalam sensor ini terdapat sebuah *thermistor* tipe *NTC (Negative Temperature Coefficient)* untuk mengukur suhu, sebuah sensor kelembapan tipe resisitif dan sebuah mikrokontroller 8-bit yang mengolah kedua sensor tersebut dan mengirim hasilnya ke pin output dengan format *single-wire bidirectional* (kabel tunggal dua arah). Jadi walaupun kelihatannya kecil, DHT11 ini melakukan fungsi yang cukup kompleks dengan ukuran komponen yang kecil (Setyaningsih et al., 2023).

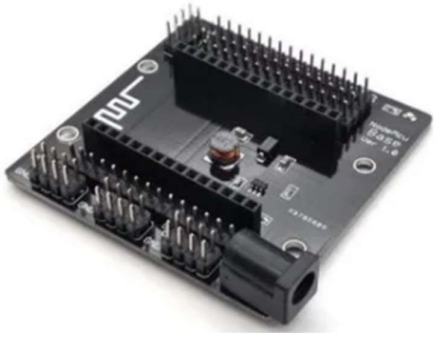


Gambar 2. 4 Sensor Humidity DHT11

Sumber gambar (Setyaningsih et al., 2023)

2.1.7 Baseplate NodeMCU ESP-8266

Baseplate Node MCU ESP-8266 sendiri merupakan sebuah *board* yang difungsikan sebagai papan sirkuit yang dapat menghubungkan semua pin-pin yang terdapat pada *board Node MCU ESP-8266* agar dapat terhubung dengan sebuah komponen lain agar dapat rapi dan terstruktur. *Baseplate Node MCU ESP- 8266* sendiri memiliki sebuah *port DC Jack 6V - 24VDC* sebagai *input* catu daya nya agar dapat berfungsi. *Baseplate* sendiri biasanya memiliki pin-pin yang sama seperti modul yang akan dipasangkan dengan nya, hanya saja memiliki jumlah lebih banyak dari masing masing pin *module* yang terpasang (Khaledi et al., 2022).

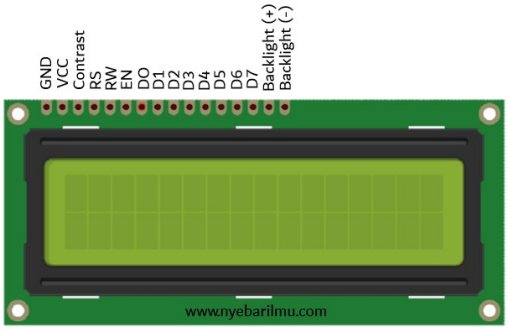


Gambar 2. 5 Baseplate NodeMCU ESP-8266
Sumber gambar (Khaledi et al., 2022)

2.1.8 Liquid Crystal Display (LCD)

Display elektronik adalah salah satu komponen eletronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. *LCD (Liquid Crystal Display)* adalah salah satu jenis display elektronik yang dibuat menggunakan teknologi *CMOS logi*c yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau

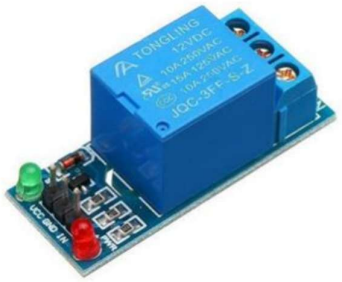
mentransmisikan cahaya dari back-lit. *LCD* ini berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka, ataupun grafik. Prinsip kerja *LCD* 16x2 adalah dengan menggunakan lapisan film yang berisi kristal cair dan diletakkan di antara dua lempeng kaca yang telah dipasang elektroda logam transparan. Saat tegangan dicatukan pada beberapa pasang elektroda, molekul-molekul kristal cair akan menyusun agar cahaya yang mengenainya akan diserap. Dari hasil penyerapan cahaya tersebut akan terbentuk huruf, angka, atau gambar sesuai bagian yang diaktifkan (Mindasari et al., 2022).



Gambar 2. 6 Liquid Crystal Display (LCD)
Sumber gambar (Satriyo Cahya Rachmanda & Aprilianto, 2023)

2.1.9 Relay Module 1CH

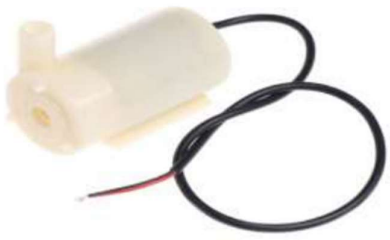
Relay merupakan komponen elektronika berupa saklar dan terdiri dari 2 bagian utama yaitu *Elektromagnet (coil)* dan *mekanikal* (seperangkat kontak Saklar/Switch). Komponen elektronika ini menggunakan prinsip elektromagnetik untuk menggerakan saklar sehingga dengan arus listrik yang kecil *(low power)* dapat menghantarkan listrik yang bertegangan lebih tinggi. *Relay* pada perancangan ini berfungsi sebagai saklar untuk menghidupkan dan mematikan sensor (Setyaningsih et al., 2023).



Gambar 2. 7 Relay Module 1CH
Sumber gambar (Setyaningsih et al., 2023)

2.1.10 Pompa DC Mini

Menurut (Moch. Bakhrul Ulum, Moch. Lutfi,Arif Faizin 2022) *Mini Submersible Water pump* adalah motor pompa air celup yang berukuran kecil. Pompa air mini ini biasa digunakan untuk akuarium, kolam ikan, *hidroponik*, robotika atau proyek dalam pembuatan aplikasi yang berbasis mikrokontroller. Pompa air mini *subersible* ini menggunakan motor *DC brushless* dan bekerja dengan tegangan *DC* 5V 120L/jam, kelebihan dari pompa air mini ini adalah tidak berisik saat digunakan dan aman saat bekerja di air. Pompa air ini nanti nya akan berfungsi sebagai pemindah air dari bak penampungan air ke tanaman yang akan disiram, pompa ini akan bekerja atau aktif ketika sensor kelembaban tanah mendeteksi tanah kering (Surapati & Anwar, 2022).



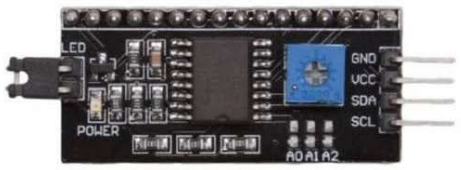
Gambar 2. 8 Submersible Water Pump 5v DC

Sumber gambar (Surapati & Anwar, 2022)

2.1.11 *I2C/TWI LCD 1602*

Merupakan modul yang dipakai untuk mengurangi penggunaan kaki di *LCD* 1602. Modul ini memiliki 4 Pin yang akan dihubungkan ke Arduino (Satriyo Cahya Rachmanda & Aprilianto, 2023).

.



Gambar 2. 9 12C/TWL LCD 1602
Sumber gambar (Satriyo Cahya Rachmanda & Aprilianto, 2023)

2.1.12 Motor Servo

Motor Servo merupakan motor yang mampu bekerja secara dua arah, *motor servo* bekerja dengan sistem *closed feedback* dimana posisi dari *motor servo* akan diinformasikan kembali ke rangkaian kontrol yang ada didalam *motor servo*. *Motor servo* terdiri dari 19 Jurnal Infotel Vol.5 No.2 November 2013 sebuah motor, rangkaian *gear*, *potensiometer*, serta rangkaian *control*. *Potensiometer* pada *motor servo* berfungsi sebagai penentu batas sudut dari putaran *servo*. *Motor servo* biasanya hanya bergerak mencapai sudut tertentu saja (Rinaldy et al., 2013).



Gambar 2. 10 Motor Servo
Sumber gambar (Rinaldy et al., 2013)

2.1.13 Kabel *Jumper*

Kabel *jumper* adalah kabel elektrik yang memiliki pin konektor di setiap ujungnya dan memungkinkanmu untuk menghubungkan dua komponen yang melibatkan *Arduino* tanpa memerlukan *solder*. Intinya kegunaan kabel *jumper* ini adalah sebagai konduktor listrik untuk menyambungkan rangkaian listrik. Biasanya kabel *jumper* digunakan pada *breadboard* atau alat *prototyping* lainnya agar lebih mudah untuk mengutak-atik rangkaian. Konektor yang ada pada ujung kabel terdiri atas dua jenis yaitu konektor jantan *(male connector)* dan konektor betina *(female connector)* (Nur Alfan & Ramadhan, 2022).



Gambar 2. 11 Kabel Jumper
Sumber gambar (Nur Alfan & Ramadhan, 2022)

2.1.14 Arduino IDE

IDE itu merupakan kependekan dari *Integrated Development Environment*, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui *software* inilah *Arduino* dilakukan pemrograman untuk melakukan fungsifungsi yang dibenamkan melalui *sintaks* pemrograman. *Arduino* menggunakan bahasa pemrograman sendiri yang menyerupai bahasa *C*. Bahasa pemrograman *Arduino (Sketch)* sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler *Arduino* telah ditanamkan suatu program bernama *Bootlader* yang berfungsi sebagai penengah antara *compiler Arduino* dengan mikrokontroler. (Septian & Aji, 2018).

2.1.15 Aplikasi Blynk

Blynk Merupakan *platform* sistem operasi *iOS* maupun *Android* sebagai kendali pada modul *Arduino*, *Raspberry Pi*, ESP8266 dan Perangkat sejenis lainnya melalui internet. Penggunan aplikasi *Blynk* sangat mudah, untuk penggunaannya dapat menggunakan *android* maupun *ios*. Aplikasi Blynk tidak terikat dengan komponen atau *chip* manapun, namun harus mendukung board dengan memiliki akses wifi untuk dapat berkomunikasi dengan *hardware* yang digunakan. Aplikasi Blynk memiliki 3 komponen utama, yaitu *Aplikasi*, *Server*, dan *Libraries*. Blynk Server berfungsi untuk menangani semua komunikasi diantara *smartphone* dan *hardware* (Syukhron, 2021).

2.1.16 Aplikasi Fritzing

Fritzing adalah suatu software atau perangkat lunak gratis yang digunakan oleh desainer, seniman, dan para penghobi elektronika untung perancangan berbagai peralatan elektronika. antarmuka *fritzing* dibuat seinteraktif dan semudah mungkin agar bisa digunakan oleh orang yang minim pengetahuannya tentang simbol dari perangkat elektronika. Di dalam *fritzing* sudah terdapat skema siap pakai dari berbagai mikrokontroller arduino serta shieldnya. Software ini memang khusus dirancang untuk perancangan dan pendokumentasian tentang produk kreatif yang menggunakan mikrokontroller arduino (Ahmad fatoni,Dhany Dwi Nugroho, 2015).

2.1.17 Tanaman Apel

Tanaman apel (*Malus sylvestris Mill*) merupakan tanaman buah tahunan yang secara alami tumbuh optimal di daerah beriklim subtropis, dan di Indonesia umumnya dibudidayakan di dataran tinggi. Kondisi ideal untuk pertumbuhan tanaman apel meliputi suhu antara 16°C hingga 27°C, kelembapan udara 75–85%, serta intensitas cahaya matahari sekitar 50–60% per hari. Namun, bibit apel tetap mampu tumbuh dengan baik pada ketinggian 300 mdpl dengan suhu 25–27°C dan kelembapan 70% melalui perlakuan zat pengatur tumbuh alami (cytokinin) yang berasal dari bahan seperti bonggol pisang, jagung, tomat, dan air kelapa (Prasetyo & Rahaju, 2012). Penelitian tersebut menunjukkan bahwa meskipun tanaman apel berada di lingkungan yang tidak sepenuhnya ideal seperti di dataran rendah, pertumbuhannya masih dapat dioptimalkan melalui perlakuan agronomis tertentu. Dengan demikian, temuan ini mendukung pengembangan budidaya apel di dataran

rendah, selama pengelolaan lingkungan khususnya kelembapan tanah dan suhu dapat dikendalikan dengan baik, misalnya melalui teknologi seperti *Internet of Things (IoT)* yang memungkinkan pengaturan penyiraman dan pencahayaan secara otomatis.



Gambar 4. 1 Apel
(adar BakhshBaloch, 2017)

2.2 Penelitian Yang Relevan

Tabel 2. 1 Penelitian yang Relevan

No	Penulis / Tahun	Topik Penelitia	Metode	Pembahasan	Hasil
1.	(Tullah et al., 2019)	Sistem Penyiraman Tanaman Otomatis Berbasis Mikrokontroler Arduino Uno Pada Toko Tanaman Hias Yopi	Metode yang digunakan dalam penelitian ini meliputi desain sistem otomatis yang memanfaatkan sensor kelembaban tanah dan suhu udara yang terhubung ke mikrokontroler Arduino Uno. Sensor kelembaban tanah berfungsi untuk mengukur tingkat kelembapan, dan data yang diperoleh ditampilkan pada *LCD* serta dikirimkan melalui SMS kepada pengguna. Sistem ini diusulkan untuk menggantikan metode penyiraman konvensional yang dilakukan secara manual, dengan tujuan	Dalam pembahasan, penelitian ini mengidentifikasi kelemahan dari sistem penyiraman konvensional yang masih bersifat manual dan tidak efisien, karena penyiraman dilakukan berdasarkan jadwal tanpa mempertimbangkan kebutuhan air tanaman. Dengan penerapan sistem otomatis, penyiraman dapat dilakukan berdasarkan data *real-time* dari sensor, sehingga tanaman mendapatkan air yang cukup tanpa pemborosan. Selain itu, sistem ini juga	Hasil penelitian menunjukkan bahwa sistem penyiraman otomatis dapat meningkatkan efisiensi dalam pengelolaan air untuk tanaman. Dengan adanya sistem ini, pengguna dapat lebih mudah dalam memantau dan merawat tanaman, serta mengurangi frekuensi intervensi manual. Penelitian ini merekomendasikan penerapan sistem penyiraman otomatis untuk meningkatkan produktivitas dan efektivitas dalam budidaya tanaman, terutama di kebun kecil dan sedang.

No	Penulis / Tahun	Topik Penelitia	Metode	Pembahasan	Hasil
			untuk meningkatkan efisiensi dan efektivitas dalam penyiraman tanaman.	menyediakan tampilan informasi melalui LCD dan SMS, yang memungkinkan pengguna untuk memantau kondisi tanaman dari jarak jauh.	
2.	(Effendi et al., 2022)	Perancangan sistem penyiraman tanaman otomatis menggunakan sensor kelembapan tanah berbasis IoT	Dalam penelitian ini, metode yang digunakan adalah *Rapid Application Development (RAD)*, yang terdiri dari beberapa tahapan, yaitu *Requirement Planning, Design Workshop*, *Fase Instruction*, dan *Implementation. Pada tahap Requirement Planning*, peneliti mengidentifikasi kebutuhan sistem untuk merancang solusi penyiraman tanaman otomatis. Selanjutnya, pada *Design Workshop*,	Penelitian ini berfokus pada pengembangan sistem penyiraman tanaman otomatis yang memanfaatkan teknologi *Internet of Things (IoT)*. Sistem ini menggunakan sensor kelembapan tanah untuk mendeteksi kadar air dan mengontrol pompa air secara otomatis melalui aplikasi Blynk. Dengan demikian, petani dapat memantau dan mengatur penyiraman tanaman secara *real-time* melalui *smartphone*, yang	Hasil dari penelitian menunjukkan bahwa sistem yang dirancang berhasil melakukan pengujian sensor kelembapan tanah dan modul *relay*. Pengujian sensor menunjukkan bahwa sistem dapat mengukur tingkat kelembapan tanah dengan akurat, dan pompa air dapat diaktifkan atau dinonaktifkan berdasarkan data yang diterima dari sensor. Hasil pengujian ini membuktikan efektivitas sistem penyiraman otomatis dalam menjaga kelembapan tanaman, sehingga dapat meningkatkan efisiensi dalam perawatan tanaman

No	Penulis / Tahun	Topik Penelitia	Metode	Pembahasan	Hasil
			dilakukan perancangan perangkat keras dan perangkat lunak yang diperlukan. *Fase Instruction* meliputi pembuatan program menggunakan Arduino IDE dan pengujian alat, sedangkan *Implementation* adalah penerapan sistem yang telah dirancang.	mengurangi kebutuhan akan penyiraman manual yang memakan waktu dan tenaga	
3.	(Ardiansah et al., 2023)	Sistem Penyiraman Otomatis Pada Tanaman dengan Monitoring Berbasis *IoT (Internet of Things*	Dalam penelitian ini, metode yang digunakan adalah pengembangan sistem penyiraman otomatis berbasis *Internet of Things (IoT)* untuk tanaman cabai (Capsicum annum L). Sistem ini dirancang menggunakan NodeMCU sebagai otak dari sistem, yang terhubung dengan sensor kelembaban tanah, pompa air mini, dan LCD untuk	Pembahasan dalam penelitian ini mencakup perancangan perangkat keras dan perangkat lunak. Perangkat keras terdiri dari komponen seperti sensor kelembaban, pompa air, dan modul LCD yang diintegrasikan dalam sebuah wadah plastik. Sementara itu, perangkat lunak dirancang menggunakan aplikasi	Hasil penelitian menunjukkan bahwa sistem berfungsi dengan baik, di mana penyiraman otomatis terjadi ketika kelembaban tanah kurang dari 60%. Pada kondisi kelembaban antara 60% hingga 70%, penyiraman otomatis tidak aktif, tetapi dapat diaktifkan secara manual melalui aplikasi. Jika kelembaban tanah lebih dari 70%, baik penyiraman otomatis maupun manual tidak berjalan. Selain itu, sistem juga mengirimkan email untuk memberikan informasi

No	Penulis / Tahun	Topik Penelitia	Metode	Pembahasan	Hasil
			menampilkan informasi kondisi tanah. Data dari sensor dipantau melalui aplikasi Blynk, yang juga memungkinkan pengguna untuk mengontrol pompa secara manual.	Blynk untuk menampilkan data kelembaban tanah dan mengontrol pompa. Pengujian sistem dilakukan untuk memastikan bahwa penyiraman otomatis berfungsi dengan baik berdasarkan kondisi kelembaban tanah yang terdeteksi.	tentang waktu dan kondisi kelembaban saat penyiraman dilakukan, yang menunjukkan efektivitas sistem dalam mendukung pengembangan smart farming yang lebih luas
4.	(Chandra, Subandi, 2023)	*Prototype* Sistem Alat Penyiraman Tanaman Cabai Otomatis Berbasis Web Menggunakan Mikrokontroler Nodemcu Esp8266	Penelitian ini menggunakan metode *prototype*, yang memungkinkan pengembangan desain sistem secara cepat dan bertahap untuk memenuhi kebutuhan pengguna. Metode ini bertujuan untuk menciptakan rancangan desain produk yang dapat dievaluasi dan	Sistem ini memanfaatkan NodeMCU ESP8266 sebagai mikrokontroler yang berfungsi untuk membaca data dari sensor kelembapan tanah dan suhu. Data yang diperoleh kemudian dikirim ke server, dan mikrokontroler mengendalikan pompa air melalui *relay* untuk melakukan penyiraman	Hasil dari perancangan alat menunjukkan bahwa sistem dapat secara efektif menghidupkan pompa air ketika kelembapan tanah berada di bawah 60%. Hal ini membantu petani dalam mengoptimalkan penyiraman tanaman cabai, meningkatkan efisiensi penggunaan air, dan mendukung pertumbuhan tanaman yang lebih baik.

No	Penulis / Tahun	Topik Penelitia	Metode	Pembahasan	Hasil
			diimplementasikan dengan efektif.	otomatis ketika kelembapan tanah terdeteksi di bawah batas yang ditentukan.	
5.	(Sumiati & Santoso, 2023)	Perancangan Alat Penyiraman Tanaman Krisan Otomatis Menggunakan Nodemcu Berbasis *Internet Of Things(Iot)*	Penelitian ini menggunakan metode deskriptif dan pengembangan dengan pendekatan *waterfall* untuk merancang alat penyiraman tanaman otomatis berbasis NodeMCU dan *Internet of Things (IoT)*. Proses penelitian meliputi analisis kebutuhan perangkat keras dan perangkat lunak, perancangan sistem, implementasi alat, serta pengujian untuk memastikan alat berfungsi dengan baik. Analisis dilakukan untuk memahami interaksi antar komponen, sedangkan	Pembahasan dalam penelitian ini menekankan pentingnya pengelolaan air dalam pertanian. Alat penyiraman otomatis dirancang untuk mengatasi masalah penyiraman manual yang seringkali tidak efisien. Dengan memanfaatkan sensor kelembaban tanah, alat ini dapat mendeteksi kebutuhan air tanaman secara real-time dan memberikan air sesuai kebutuhan, sehingga menghindari kondisi tanaman yang kekurangan atau kelebihan air.	Hasil penelitian menunjukkan bahwa alat penyiraman tanaman otomatis ini berhasil berfungsi dengan baik, mampu menyiram tanaman secara otomatis, dan dapat diakses serta dikendalikan melalui internet. Alat ini meningkatkan efisiensi dalam pengairan dan memberikan kemudahan bagi petani dalam merawat tanaman mereka, yang pada akhirnya dapat meningkatkan produktivitas pertanian.

No	Penulis / Tahun	Topik Penelitia	Metode	Pembahasan	Hasil
			perancangan mencakup penggabungan komponen seperti NodeMCU, sensor kelembaban tanah, *relay*, dan pompa air.		

2.3 Perbedaan dengan Penelitian Sebelumnya

Penelitian ini memiliki perbedaan yang menonjol dibandingkan dengan penelitian sebelumnya, baik dari segi objek, fitur sistem, maupun pendekatan teknologinya. Penelitian ini secara khusus difokuskan pada tanaman apel yang dibudidayakan di dataran rendah, yang memiliki tantangan lingkungan seperti suhu tinggi dan kelembapan udara rendah, berbeda dengan penelitian sebelumnya yang umumnya membahas tanaman hias, cabai, atau krisan. Selain menerapkan sistem penyiraman otomatis, penelitian ini juga mengembangkan fitur tambahan berupa sistem buka-tutup atap otomatis berdasarkan suhu, untuk melindungi tanaman dari paparan sinar matahari berlebih—fitur yang belum ditemukan dalam penelitian terdahulu. Sistem ini menggunakan kombinasi sensor soil moisture, suhu DS18B20, dan kelembapan udara DHT11, serta mendukung pemantauan dan pengendalian secara otomatis maupun manual melalui aplikasi Blynk dan LCD. Kontribusi penulis dalam penelitian ini adalah menghadirkan prototype berbasis IoT yang mampu memberikan kemudahan bagi pengguna dalam memantau kondisi tanaman dari jarak jauh, menghemat waktu dan tenaga dalam perawatan tanaman, serta menjaga lingkungan tumbuh tanaman apel agar tetap optimal di wilayah dengan kondisi iklim yang kurang ideal.

BAB III

METODOLOGI PENELITIAN

3.1 Tinjauan Umum

Budidaya tanaman apel di dataran rendah menghadapi tantangan lingkungan yang cukup kompleks, seperti suhu tinggi dan kelembapan udara yang rendah, yang mempercepat penguapan air tanah dan mengganggu kestabilan kelembapan. Tanaman apel sendiri membutuhkan kondisi lingkungan yang ideal, termasuk kelembapan tanah yang cukup, suhu antara 16°C–27°C, dan intensitas cahaya 50%–60%. Penyiraman yang tidak teratur serta paparan cahaya berlebih dapat menyebabkan tanaman stres dan pertumbuhan terganggu. Untuk menjawab permasalahan tersebut, pemanfaatan teknologi *Internet of Things (IoT)* menjadi solusi yang potensial, di mana sistem mampu memantau dan mengontrol penyiraman serta perlindungan tanaman secara otomatis dan *real-time*. Dengan integrasi sensor kelembapan tanah dan suhu pada mikrokontroler yang terhubung ke aplikasi pemantauan, sistem ini diharapkan mampu menciptakan kondisi tumbuh yang optimal, mengurangi kebutuhan intervensi manual, serta memberikan kemudahan dan efisiensi dalam pengelolaan tanaman apel di wilayah dengan iklim yang kurang ideal.

Kajian ini memberikan dasar ilmiah untuk *prototype* penyiraman otomatis berbasis IoT yang *inovatif* dan *aplikatif*, sehingga dapat mendukung kemajuan teknologi di bidang pertanian, khususnya tanaman apel. Selain itu, penelitian ini diharapkan dapat memberikan manfaat nyata bagi petani.

3.2 Jenis penelitian

Jenis penelitian yang digunakan dalam penelitian ini adalah penelitian kuantitatif dengan metode *Research and Development (R&D)*. Metode R&D bertujuan untuk mengembangkan dan menghasilkan produk atau prototipe sistem penyiraman tanaman apel berbasis *Internet of Things (IoT)* yang aplikatif dan terukur secara kuantitatif. Model R&D yang digunakan mengacu pada langkah-langkah pengembangan *Borg and Gall* yang telah dimodifikasi sesuai dengan kebutuhan penelitian ini.

Penelitian ini melalui beberapa tahapan, yaitu identifikasi potensi dan permasalahan pada budidaya apel di dataran rendah, pengumpulan data melalui pengamatan dan studi literatur, desain dan validasi *prototype*, uji coba lapangan, revisi produk, hingga finalisasi *prototype* yang siap digunakan. Pendekatan kuantitatif diterapkan dengan mengumpulkan data numerik melalui serangkaian eksperimen dan pengukuran menggunakan sensor kelembapan tanah dan sensor suhu yang terintegrasi dengan sistem IoT.

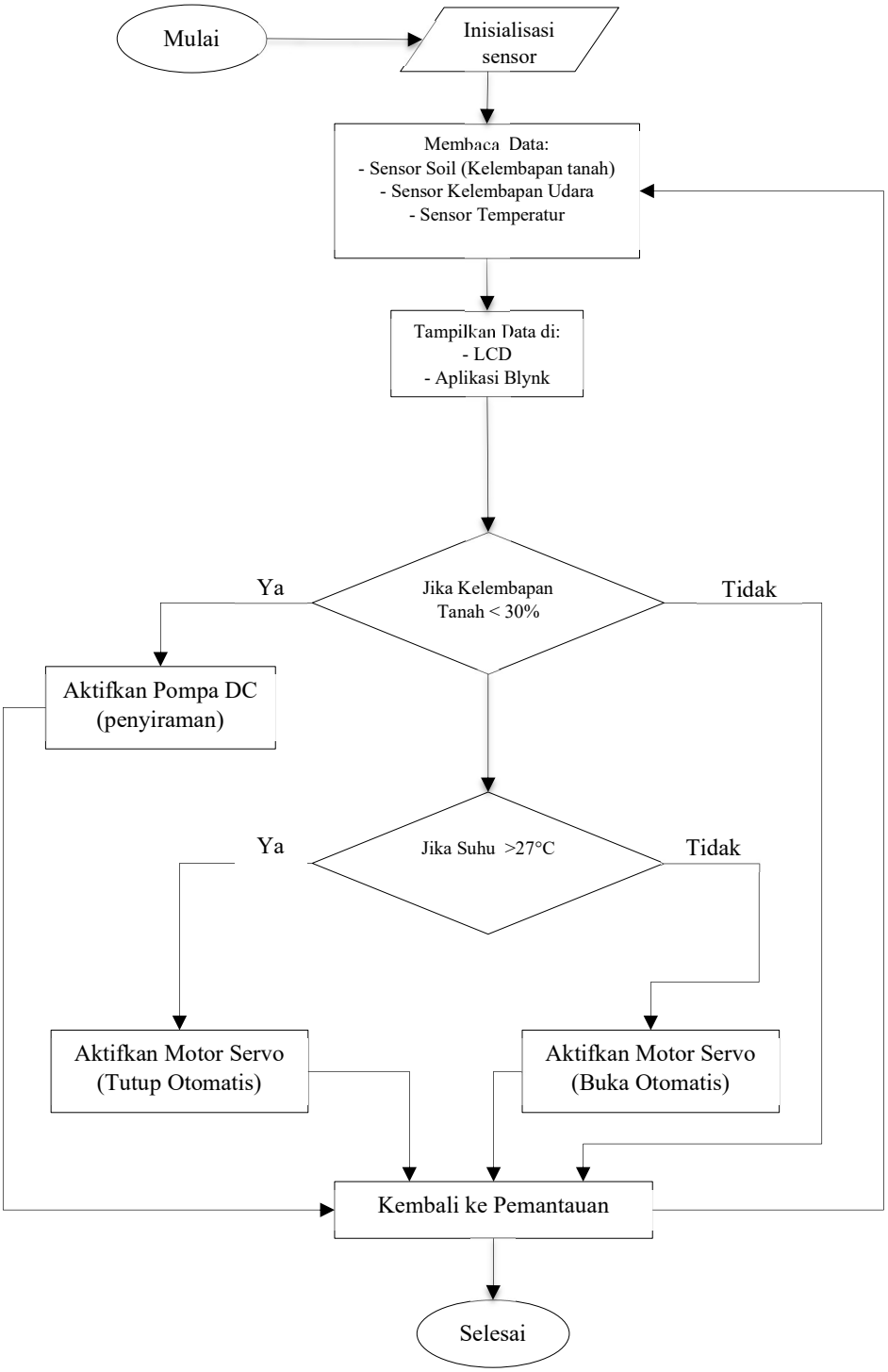
Data yang diperoleh dari sensor digunakan untuk mengendalikan sistem penyiraman otomatis dan mekanisme buka tutup atap secara otomatis berdasarkan kondisi lingkungan yang terdeteksi. Analisis kuantitatif dilakukan terhadap data sensor untuk mengevaluasi efektivitas sistem dalam menjaga kelembapan tanah dalam rentang optimal dan mengatur intensitas cahaya yang diterima tanaman apel. Dengan pendekatan ini, diharapkan *prototype* yang dihasilkan tidak hanya fungsional, tetapi juga terbukti efektif berdasarkan data kuantitatif dalam

membantu pengelolaan tanaman apel secara efisien, baik dari segi waktu maupun tenaga.

3.3 Desain Prototype

3.3.1 Flowchart

IoT memegang peran utama dalam *prototype* ini sebagai penghubung antara perangkat sensor, aktuator, dan aplikasi pemantauan. Teknologi IoT memungkinkan sistem untuk membaca data lingkungan dari sensor kelembapan tanah, kelembapan udara dan suhu suhu secara *real-time*, serta mengontrol pompa penyiraman dan sistem buka-tutup otomatis sesuai kondisi yang terdeteksi. Dengan menggunakan aplikasi *Blynk*, petani dapat memantau dan mengontrol sistem ini melalui perangkat seluler mereka kapan saja. Desain *prototype* yang digunakan adalah *flowchart* dimana dapat mempermudah alur dalam *prototype* sistem penyiraman tanaman apel berbasis *internet of thing (IoT)*, berikut dapat dilihat pada gambar 3.1.



Gambar 3. 1 Flowchart Prototype

Dari Gambar 3.1 diatas menjelaskan bahwa :

a. Proses dimulai dengan inisialisasi sensor untuk mempersiapkan sistem dalam mengambil data. Sensor-sensor yang digunakan adalah sensor kelembapan tanah, sensor kelembapan udara dan sensor temperatur. Sensor-sensor ini akan digunakan untuk mengukur kondisi lingkungan yang mempengaruhi pertumbuhan tanaman.

b. Setelah sensor diinisialisasi, sistem mulai membaca data dari masing-masing sensor yang terpasang. Sensor kelembapan tanah mengukur kadar kelembapan tanah, sensor kelembapan udara mengukur tingkat kelembapan di sekitar tanaman dan sensor temperatur mengukur suhu udara sekitar. Data tersebut diperlukan untuk menganalisis apakah kondisi lingkungan mendukung pertumbuhan tanaman.

c. Data yang terkumpul kemudian ditampilkan untuk pemantauan. Data akan ditampilkan pada *LCD* yang terpasang pada *prototype* sehingga pengguna dapat melihatnya secara langsung. Selain itu, aplikasi *Blynk* juga digunakan untuk memungkinkan pemantauan data secara jarak jauh melalui *smartphone*, memberikan fleksibilitas bagi pengguna dalam mengawasi kondisi tanaman kapan saja.

d. Salah satu parameter penting dalam *prototype* ini adalah kelembapan tanah. Jika kelembapan tanah terdeteksi kurang dari 30%, ini menunjukkan bahwa tanaman memerlukan penyiraman. Dalam hal ini, pompa DC akan diaktifkan untuk menyirami tanaman. Jika kelembapan tanah lebih dari 45%, tidak ada

tindakan penyiraman yang dilakukan, karena kondisi tanah dianggap cukup lembap.

e. Setelah itu sistem memeriksa suhu lingkungan. Jika suhu udara lebih dari 27°C, kondisi ini dapat mengganggu pertumbuhan tanaman. Dalam hal ini, motor servo diaktifkan untuk menutup atap otomatis, guna menjaga suhu tetap stabil. Namun, jika suhu berada dalam rentang 16°C hingga 27°C, motor servo akan diaktifkan untuk membuka atap otomatis, sehingga suhu tetap ideal untuk pertumbuhan tanaman.

f. Setelah semua proses di atas selesai, sistem kembali ke tahap pemantauan untuk terus memantau kondisi lingkungan secara *real-time*. Dengan begitu, sistem dapat merespons perubahan kondisi dengan cepat dan otomatis.

g. Proses monitoring dan pengendalian selesai, dan sistem siap untuk memulai siklus pemantauan baru. Proses ini berulang terus menerus untuk memastikan kondisi tanaman tetap optimal.

3.4 Instrumen Penelitian

Pengumpulan data dalam penelitian ini dilakukan dengan menggunakan berbagai metode penelitian, seperti pengamatan, kepustakaan, eksperimen, dan dokumentasi, dengan penjelasan sebagai berikut:

a. Pengamatan

Pada tahap pengamatan, peneliti melakukan pengamatan langsung terhadap tanaman apel untuk memahami kebutuhan kelembapan tanah yang optimal serta pola pertumbuhannya. Pengamatan ini mencakup analisis kondisi lingkungan tempat tumbuh tanaman, seperti kelembapan tanah, suhu, dan

intensitas cahaya. Data yang terkumpul dari pengamatan ini menjadi dasar untuk merancang sistem monitoring yang dapat mendukung pertumbuhan tanaman apel dengan baik.

b. Kepustakaan

Peneliti mempelajari literatur dari buku, jurnal, dan artikel ilmiah yang membahas tentang tanaman apel, kelembapan tanah, serta teknologi *Internet of Things* (IoT) yang digunakan dalam pemantauan pertanian. Kajian pustaka ini bertujuan untuk memberi pemahaman mengenai teori dan prinsip dasar yang relevan dengan penelitian, serta teknologi yang dapat diterapkan dalam sistem monitoring tanaman apel.

c. Eksperimen

Pada tahap eksperimen, peneliti melakukan uji coba terhadap perangkat IoT yang digunakan untuk memantau kelembapan tanah. Peneliti mengatur sensor kelembapan tanah, sensor suhu, sensor kelembapan udara, dan sensor cahaya untuk mengumpulkan data tentang kondisi lingkungan tanaman apel. Eksperimen ini mencakup pengujian keakuratan perangkat dalam berbagai kondisi lingkungan untuk memastikan hasil yang konsisten dan akurat. Selain itu, peneliti juga menguji kelayakan perangkat keras yang digunakan dalam sistem. Teknologi IoT memainkan peran kunci dalam eksperimen ini dengan memungkinkan pengumpulan data lingkungan yang konsisten dan akurat dari berbagai sensor yang terhubung ke mikrokontroler NodeMCU ESP8266. Data tersebut kemudian dikirimkan secara *real-time* ke aplikasi Blynk untuk dianalisis, sehingga peneliti dapat mengevaluasi

efektivitas sistem penyiraman otomatis dan kontrol pencahayaan. Dengan IoT, penelitian ini tidak hanya memberikan data yang valid tetapi juga menciptakan solusi yang aplikatif untuk mendukung produktivitas tanaman apel di dataran rendah.

d. Dokumentasi

Tahap dokumentasi dilakukan untuk mencatat setiap langkah dalam proses penelitian, mulai dari pengaturan perangkat hingga hasil eksperimen. Semua temuan dan proses yang dilalui selama penelitian akan didokumentasikan dengan rinci, termasuk spesifikasi teknis perangkat yang digunakan dan hasil analisis pengukuran. Dokumentasi ini penting untuk memastikan bahwa penelitian dapat diulang atau dievaluasi di masa depan, serta untuk memberikan gambaran lengkap tentang sistem yang dikembangkan.

3.5 Analisis Kebutuhan

3.5.1 Analisis Kebutuhan *Hardware*

Dalam penelitian ini penulis menggunakan perangkat keras *(hardware)* dua jenis yaitu *laptop* dan *handphone* dapat dilihat pada Tabel 3.1

Tabel 3. 1 Perangkat Keras

Spesifikasi Laptop		**Spesifikasi Handphone**	
Type	ASUS P1512CEA	*Type*	iPhone 8 Plus
Processor	*11th Gen Intel®Core™ i5-1135G7*	*Processor*	A11 Bionic
Operating System	*Windows 11 Home Single Language 64-bit*	*iOS Version*	16.7.10

Spesifikasi Laptop		Spesifikasi Handphone	
VGA	Intel® Iris® Xe Graphics	*Storage*	256 GB
RAM	8 GB	RAM	3 GB
SSD	256 GB		

3.5.2 Analisis Kebutuhan *Software*

Perangkat lunak *(software)* yang digunakan dalam pembuatan *prototype* ini adalah sebagai berikut :

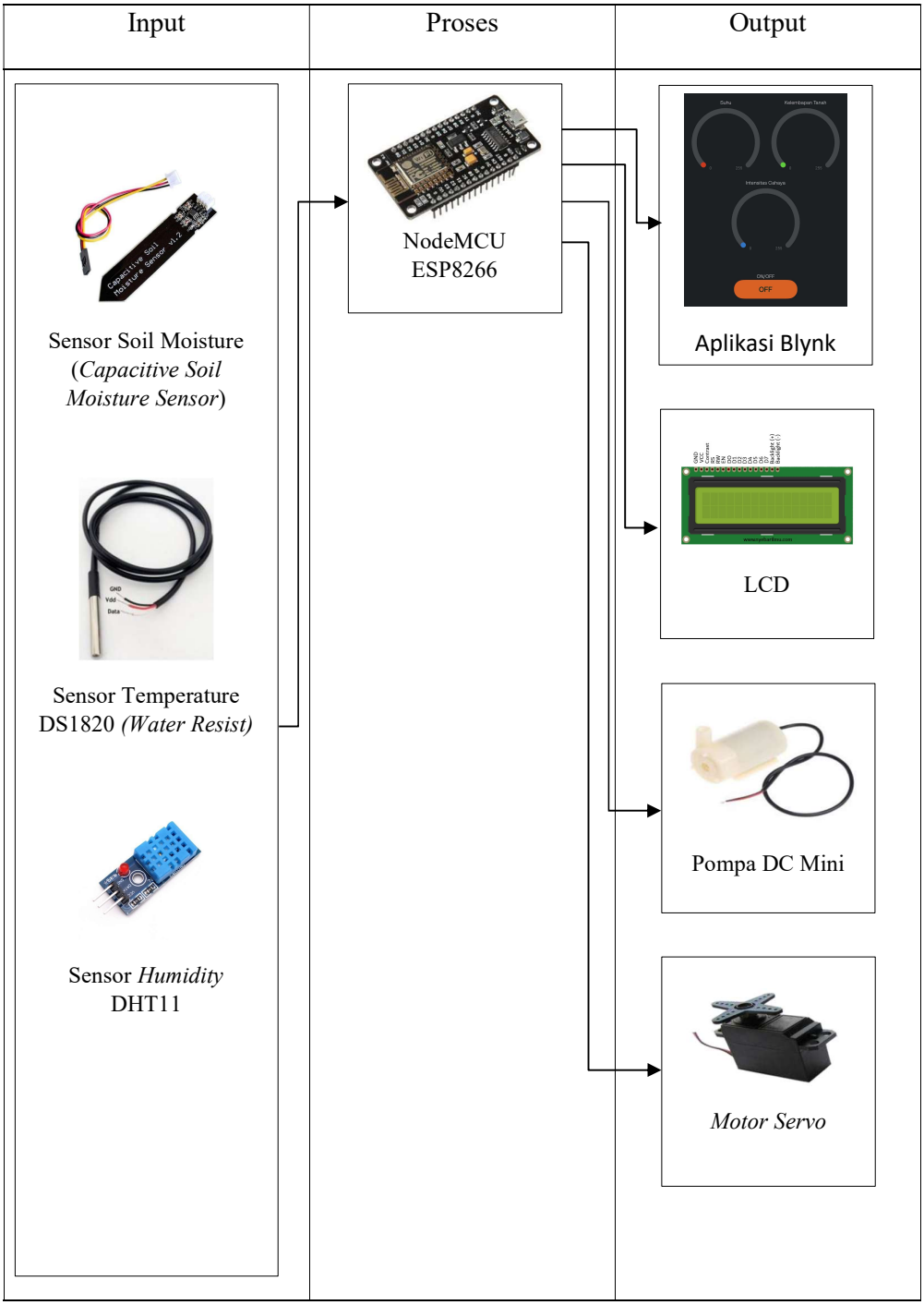
a. Windows 11 *Home Single Language 64-bit* sebagai sistem operasi

b. Arduino IDE sebagai pemrograman pada NodeMCU ESP8266

c. Blynk untuk mengontrol dan menapilkan hasil dari pembacaan sensor

3.5.3 Analisis Proses

Tabel 3. 2 Diagram Alur



Pada tabel 3.3 menjelaskan bahwa *prototype* ini terdiri dari tiga komponen utama, yaitu *Input*, Proses, dan *Output*. Bagian *Input* mencakup beberapa sensor, yaitu *Capacitive Soil Moisture Sensor* untuk mengukur kelembapan tanah, *DS1820* untuk mengukur suhu lingkungan dan *DHT11* untuk memantau kelembapan udara. Data dari sensor ini sangat penting untuk mengetahui apakah kondisi lingkungan sesuai dengan kebutuhan ideal tanaman apel, yaitu suhu 16-27°C serta kelembapan udara 75-85%.

Semua data dari sensor dikirimkan ke *NodeMCU ESP8266* sebagai pusat pemrosesan dalam bagian Proses. *NodeMCU* mengolah data ini untuk menentukan tindakan yang diperlukan, seperti mengaktifkan pompa penyiraman jika tanah kering atau mengontrol *motor servo* untuk menutup atau membuka penutup otomatis berdasarkan intensitas cahaya. Selain itu, *NodeMCU* juga bertugas untuk mengirimkan data yang telah diproses ke bagian *Output*, yaitu *Liquid Crystal Display (LCD)* dan aplikasi *Blynk*. *LCD* menampilkan data kondisi lingkungan secara langsung di perangkat keras, sementara aplikasi *Blynk* memungkinkan pengguna untuk memantau data sensor secara *real-time* melalui perangkat seluler.

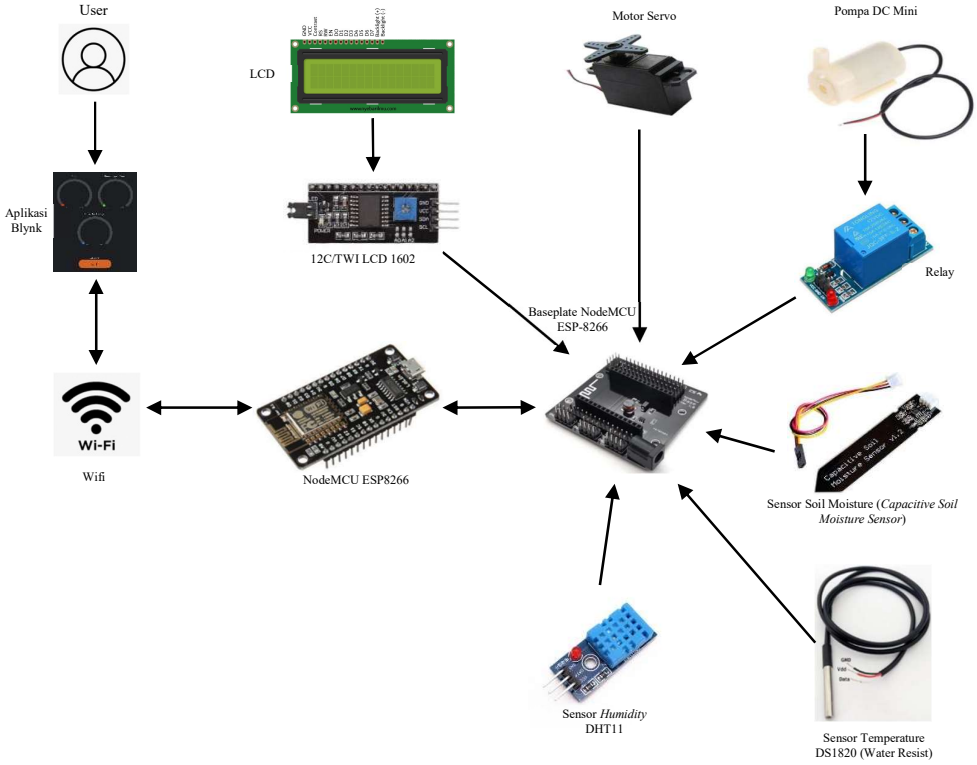
Bagian *Output* juga mencakup perangkat pengendali seperti pompa *DC mini* dan *motor servo*. Pompa *DC mini* secara otomatis menyiram tanaman ketika sensor kelembapan mendeteksi kondisi tanah kering, sedangkan *motor servo* mengatur pembukaan dan penutup untuk menjaga intensitas cahaya tetap ideal. Dengan alur kerja ini, *prototype* ini mampu mengoptimalkan kondisi lingkungan secara otomatis untuk mendukung pertumbuhan tanaman apel. Semua komponen bekerja secara

bersamaan, sehingga pengguna dapat memantau kondisi secara *real-time* dari jarak jauh, atau mengontrol sistem langsung di lokasi.

3.6 Desain

3.6.1 Desain Arsitektur Perangkat

Berikut merupakan desain arsitektur dari *prototype* sistem penyiraman tanaman apel berbasis *Internet of Thing (IoT)*:



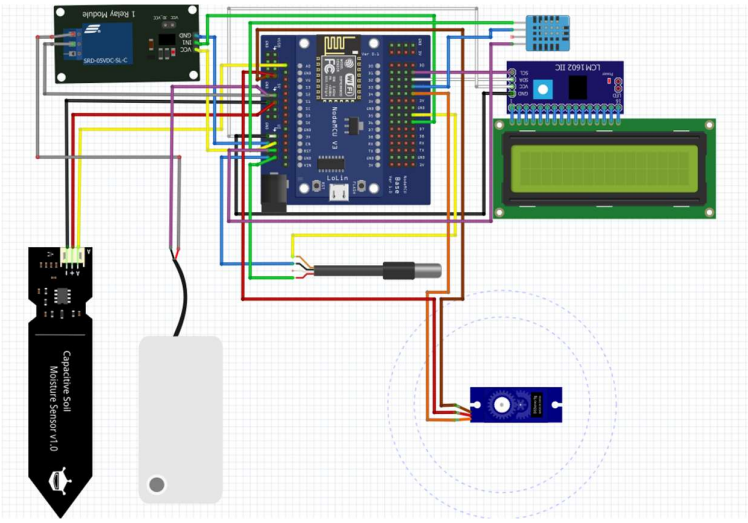
Gambar 3. 2 Desain Arsitektur Perangkat

Desain arsitektur perangkat pada Gambar 3.2 menggunakan *NodeMCU ESP8266* sebagai pusat pengendali yang menerima data dari berbagai sensor, yaitu

sensor soil moisture untuk mendeteksi kelembapan tanah, sensor temperatur *DS18B20* untuk mengukur suhu lingkungan, sensor kelembapan udara *DHT11* untuk memantau tingkat serta kelembapan udara untuk mengukur suhu. Data dari sensor-sensor ini diproses oleh *NodeMCU* dan ditampilkan secara *real-time* melalui *LCD I2C 1602* serta aplikasi Blynk yang terhubung melalui Wi-Fi, memungkinkan pemantauan jarak jauh. Berdasarkan data yang diperoleh, *NodeMCU* mengontrol pompa *DC mini* untuk menyiram tanaman secara otomatis ketika kelembapan tanah rendah dan *motor servo* untuk membuka atau menutup pelindung tanaman sesuai suhu. Dengan arsitektur ini, *prototype* ini dapat menjaga kondisi lingkungan tanaman apel tetap optimal dalam hal kelembapan tanah, suhu dan kelembapan udara.

3.6.1 Desain Rangkaian Perangkat Keras

Berikut adalah desain perangkat keras *prototype* sistem penyiraman tanaman apel berbasis *Internet of Thing (IoT)*:



Gambar 3. 3 Desain Rangkaian Perangkat Keras

Gambar 3.3 menunjukkan seluruh komponen hardware yang telah dikonfigurasikan dengan *NodeMCU ESP8266*. Selain itu, terdapat beberapa komponen lain yang ditampilkan pada gambar tersebut, yaitu:

a) *NodeMCU ESP8266*

b) *Baseplate NodeMCU ESP-8266*

c) Sensor *Soil Moisture*

d) Sensor *Temperature* DS1820 *(Water Resist)*

e) Sensor *Humidity* DHT11

f) *12C/TWL LCD 1602*

g) *Relay Module* 1CH

h) Pompa *DC Mini*

i) *Motor Servo*

j) Kabel *Jumper*

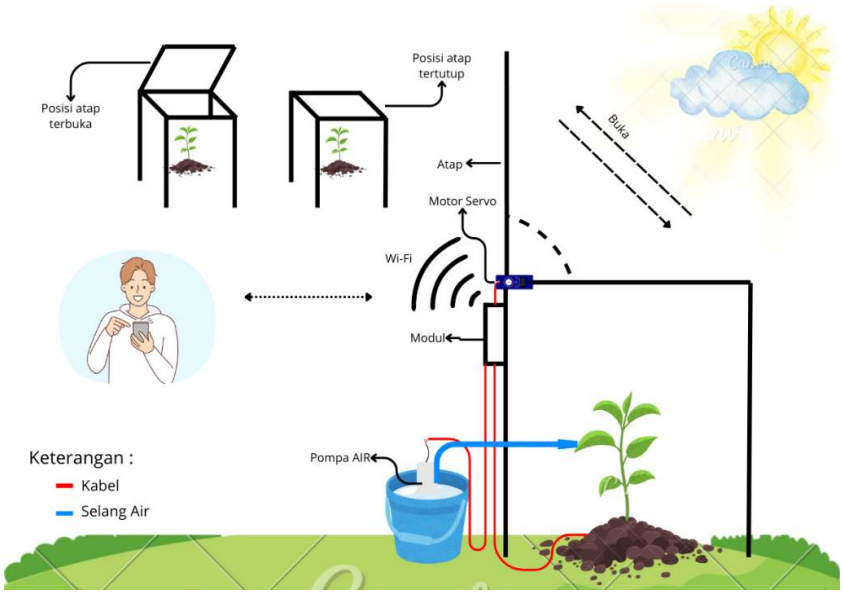
Sensor kelembapan tanah ditempatkan pada tanah di sekitar tanaman apel untuk mendeteksi tingkat kelembapan tanah. Sensor ini mengukur kadar air dalam tanah dan menghasilkan data berupa tegangan yang berkorelasi dengan tingkat kelembapan tanah. Data tersebut dikirimkan ke *NodeMCU ESP8266* melalui *breadboard* untuk diproses. Jika kelembapan tanah berada di bawah batas yang telah ditentukan, pompa *DC mini* akan diaktifkan melalui modul *relay*

untuk menyiram tanaman. Sensor suhu *DS18B20* ditempatkan di lingkungan sekitar tanaman untuk memantau suhu udara, sedangkan sensor kelembapan udara *(DHT11)* digunakan untuk mengukur tingkat kelembapan udara. Data suhu dan kelembapan udara ini dikirimkan ke *NodeMCU* untuk diproses lebih lanjut. Selain

itu, sensor Sensor *Temperature* digunakan untuk mendeteksi suhu. Jika suhu terlalu panas atau dingin, *motor servo* akan menggerakkan pelindung tanaman secara otomatis untuk menyesuaikan kondisi suhu sesuai kebutuhan. Semua data dari sensor-sensor tersebut ditampilkan secara lokal melalui *LCD I2C 1602*, serta dikirimkan ke aplikasi Blynk melalui koneksi *Wi-Fi* pada *NodeMCU ESP8266*. Aplikasi Blynk memungkinkan pengguna untuk memantau kondisi tanaman secara *real-time* dari jarak jauh, termasuk data kelembapan tanah, suhu dan kelembapan. Dengan demikian, *prototype* ini mendukung pengelolaan lingkungan tanaman apel secara otomatis untuk memastikan kondisi yang optimal bagi pertumbuhannya.

3.6.2 Desain Prototype

Berikut merupakan desain *prototype* sistem penyiraman tanaman apel berbasis *Internet of Thing (IoT)*:



Gambar 3. 4 Prototype Sistem Penyiraman Tanaman Apel Berbasis Internet of Thing (IoT)

Pada Gambar 3.4 merupakan *prototype* sistem penyiraman tanaman apel berbasis *Internet of Thing (IoT)* untuk mengoptimalkan pertumbuhan tanaman apel. *Prototype* ini menggunakan *NodeMCU ESP8266* sebagai pengontrol utama yang terhubung ke berbagai sensor, termasuk sensor kelembapan tanah, suhu dan kelembapan udara. Sensor kelembapan tanah mengukur kadar air tanah untuk mengontrol pompa air otomatis melalui *relay*, sedangkan sensor *Temperature* mendeteksi suhu untuk mengatur *motor servo* yang membuka atau menutup atap secara otomatis. Data dari semua sensor ditampilkan secara lokal pada *LCD 12C/TWI* dan dikirimkan ke aplikasi *Blynk* melalui koneksi *Wi-Fi*, memungkinkan pengguna memantau dan mengontrol sistem dari jarak jauh. Jalur kabel merah menunjukkan koneksi elektronik, sedangkan selang biru menandakan jalur air dari pompa untuk menyiram tanaman. *Prototype* ini dirancang untuk memastikan kondisi lingkungan yang optimal bagi tanaman apel, seperti penyiraman otomatis saat tanah kering dan perlindungan dari intensitas cahaya berlebih, menjadikannya solusi pintar dan efektif.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil

4.1.1 Implementasi

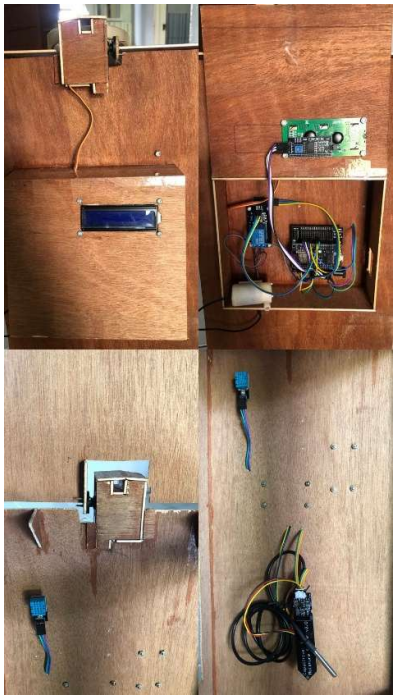
a. Cara Kerja IoT

Sistem *Internet of Things (IoT)* yang diterapkan pada *prototype* sistem penyiraman tanaman apel ini bekerja secara otomatis dengan menggabungkan fungsi sensor, mikrokontroler, aktuator, serta aplikasi pemantauan jarak jauh. IoT pada sistem ini diterapkan melalui konektivitas antara NodeMCU ESP8266 sebagai mikrokontroler utama, berbagai sensor, serta aplikasi *Blynk* yang terhubung ke internet. Pertama, sistem akan melakukan inisialisasi dan membaca data dari sensor kelembapan tanah (*soil moisture*), sensor suhu (DS18B20), dan sensor kelembapan udara (DHT11). Data dari sensor kelembapan tanah digunakan untuk menentukan kapan penyiraman perlu dilakukan, jika kadar kelembapan tanah berada di bawah 30%, maka NodeMCU akan secara otomatis mengaktifkan pompa DC mini untuk menyiram tanaman. Setelah tanah mencapai kelembapan ideal sekitar 45%, pompa akan berhenti bekerja. Sementara itu, sensor suhu digunakan sebagai acuan untuk mengontrol motor servo yang menggerakkan sistem buka-tutup atap. Jika suhu melebihi 27°C, atap akan tertutup secara otomatis untuk mengurangi paparan sinar matahari, dan akan kembali terbuka jika suhu turun ke bawah ambang batas tersebut, memastikan tanaman tetap mendapatkan cahaya yang cukup. Seluruh data dari sensor ditampilkan secara lokal melalui layar LCD 16x2 serta dikirim secara *real-time* ke aplikasi Blynk melalui koneksi Wi-Fi. Melalui Blynk, pengguna dapat

memantau kondisi lingkungan tanaman dari jarak jauh menggunakan *smartphone*, serta melakukan kontrol manual seperti menyalakan atau mematikan pompa dan mengatur buka-tutup atap sesuai kebutuhan. Dengan demikian, sistem ini memungkinkan proses penyiraman dan pengaturan pencahayaan dilakukan secara otomatis berbasis data lingkungan, sekaligus memberikan kemudahan dalam pemantauan dan pengendalian jarak jauh tanpa intervensi manual yang berlebihan.

b. Perangkat Keras

Pada tahap ini, penulis mengimplementasikan hasil dari desain sistem ke dalam bentuk perangkat keras yang telah dirancang sebelumnya. Proses implementasi ini dilakukan dengan merakit seluruh komponen perangkat keras ke dalam sebuah *prototype* sistem penyiraman tanaman apel berbasis *Internet of Things* (IoT). Komponen utama yang digunakan meliputi NodeMCU ESP8266 sebagai mikrokontroler, sensor soil moisture untuk mengukur kelembapan tanah, sensor DS18B20 untuk mengukur suhu, serta sensor DHT11 untuk mendeteksi kelembapan udara. Selain itu, pompa DC mini dan motor servo digunakan sebagai aktuator untuk menjalankan fungsi penyiraman dan sistem buka-tutup atap secara otomatis. Rangkaian komponendapat dilihat pada Gambar 4.2.



Gambar 4.2 Implementasi perangkat keras

Setiap komponen dirangkai dan dihubungkan sesuai dengan desain rangkaian perangkat keras yang telah direncanakan. Setelah perangkat keras selesai dirakit, *prototype* diuji secara langsung untuk memastikan bahwa seluruh komponen dapat berfungsi dengan baik dan sesuai dengan tujuan dari sistem yang telah dirancang. Pengujian dilakukan dengan memberikan variasi kondisi lingkungan untuk melihat bagaimana sistem merespons perubahan yang terdeteksi oleh sensor. Hasil dari implementasi ini menjadi dasar untuk mengevaluasi apakah sistem telah memenuhi tujuan perancangan, yaitu menciptakan sistem penyiraman otomatis dan kontrol atap berbasis IoT. Berikut adalah hasil akhir *prototype* sistem penyiraman tanaman apel berbasis *Internet of Things* (IoT) dapat dilihat pada Gambar 4.3.



Gambar 4. 3 Prototype Sistem Penyiraman Tanaman Apel Berbasis Internet of Things (IoT)

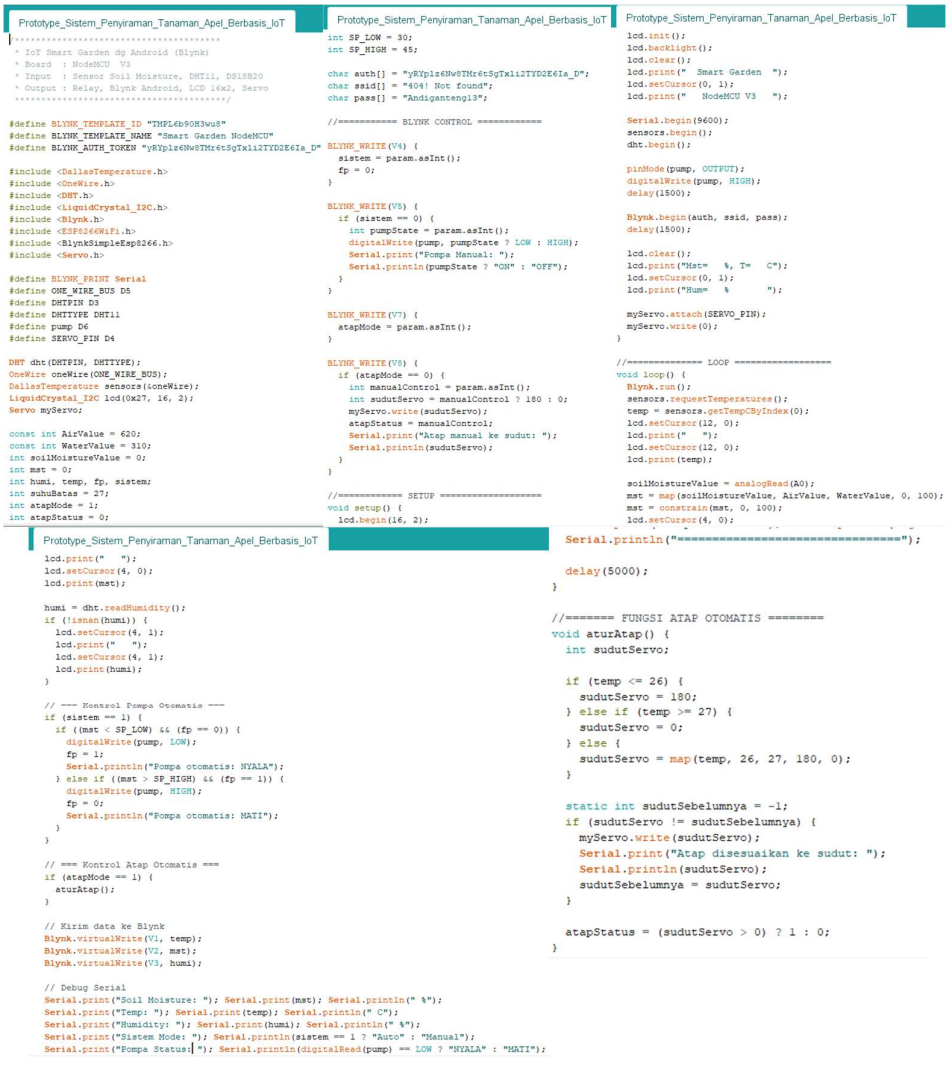
c. Implementasi Perangkat Lunak

Setelah melakukan perancangan dan perakitan komponen, selanjutnya adalah implementasi untuk perangkat lunak yaitu untuk pembuatan program menggunakan

software Arduino IDE dan *Blynk* yang akan digunakan untuk *prototype* sistem penyiraman tanaman apel berbasis *internet of thing (IoT).*

1) Implementasi Program pada Arduino IDE

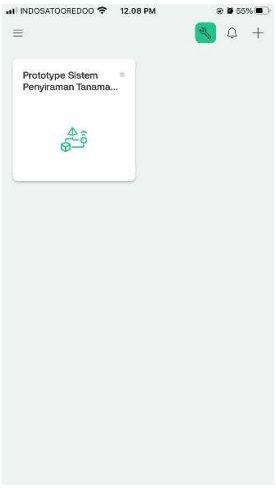
Source code program pada Arduino IDE yang akan digunakan untuk membangun *prototype* sistem penyiraman tanaman apel berbasis *internet of thing (IoT).* Berikut adalah proses programnya dapat dilihat pa da Gambar 4.4.



Gambar 4. 4 Proses program

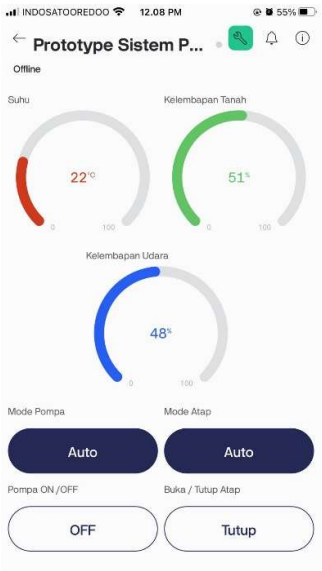
2) Implementasi Blynk

Pada aplikasi *Blynk* terdapat beberapa fitur yang dapat digunakan sesuai dengan fungsi dan program yang telah dibuat. Berikut adalah tampilan awal aplikasi *Blynk* pada *smartphone* pada Gambar 4.5.



Gambar 4. 5 Tampilan awal aplikasi blynk

Tampilan *user interface* aplikasi *Blynk* pada *smartphone* dapat dilihat pada Gambar 4.6.



Gambar 4. 6 Tampilan user interface

4.1.2 Pengujian

a. Pengujian Black Box

Pada tahapan ini pengujian alat secara menyeluruh dengan metode pengujian *black box.* Pengujian *black box* dilakukan untuk mengamati hasil *input* dan *output* dari program yang telah dibuat. Setelah pengujian dilakukan secara keseluruhan maka hasil yang dapat diperoleh, program yang dibuat dapat berjalan sesuai dengan yang diharapkan. Pengujian *black box* dapat dilihat pada Tabel 4.1.

Tabel 4. 1 Pengujian Black Box

No	Pengujian	Hasil yang diharapkan	Hasil Pengujian
1.	Tes koneksi ke bord NodeMCU ESP8266	Port terdeteksi	Berhasil
2.	Tes sensor Soil Moisture	Dapat berfungsi dan mendeteksi kelembapan tanah secara *real-time*	Berhasil
3.	Tes sensor Temperature	Dapat berfungsi dan mendeteksi suhu secara *real-time*	Berhasil
4.	Tes sensor Humidity	Dapat berfungsi dan mendeteksi kelembapan udara secara *real-time*	Berhasil
5.	Uji pengiriman data dari sensor ke aplikasi pemantauan.	Data real-time ditampilkan dengan akurat di aplikasi.	Berhasil
6.	Uji fungsi manual untuk menyalakan atau mematikan pompa melalui aplikasi.	Pompa dapat dikendalikan secara manual melalui aplikasi blynk	Berhasil
7.	Uji fungsi manual untuk sistem buka tutup atap melalui aplikasi.	Atap dapat dikendalikan secara manual melalui aplikasi blynk	Berhasil
8.	Uji fungsi otomatis saat kelembapan tanah di bawah ambang batas yang telah ditentukan.	Pompa aktif untuk melakukan penyiraman.	Berhasil
9.	Uji fungsi otomatis sistem buka tutup saat	Sistem dapat menutup atau membuka secara otomatis	Berhasil

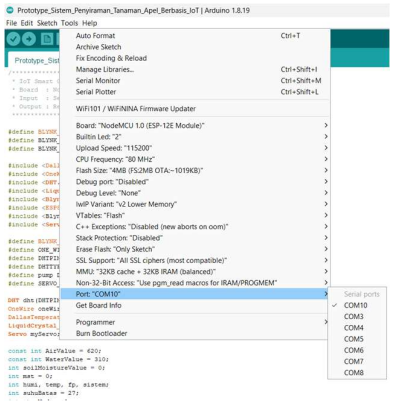
No	Pengujian	Hasil yang diharapkan	Hasil Pengujian
	suhu mencapai batas yang telah ditentukan	sesuai kebutuhan.	
10.	Uji pengiriman data ke LCD	Dapat menampilkan data suhu, kelembapan tanah dan kelembapan udara secara *real-time*	Berhasil

b. Hasil Pengujian

Berikut adalah penjelasan dari hasil pengujian *blackbox* berdasarkan setiap skenario yang diuji:

1. Tes koneksi board NodeMCU ESP8266

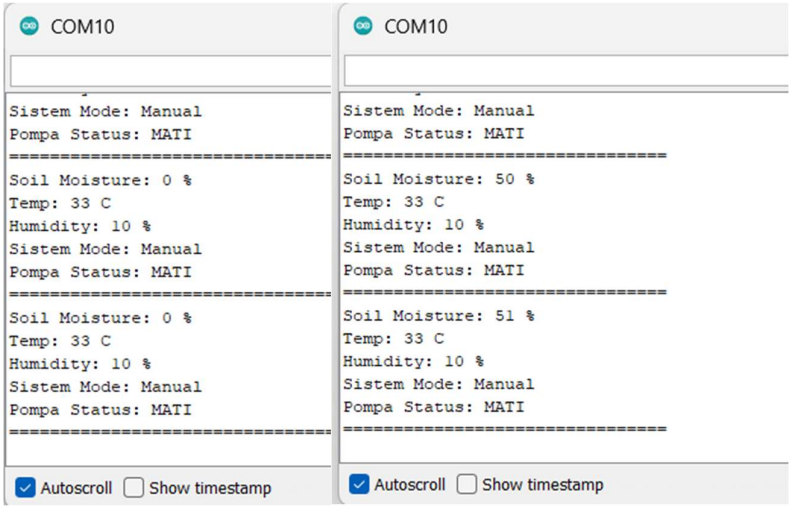
Pengujian koneksi ke board NodeMCU ESP8266 dengan menggunakan *software* Arduino IDE untuk mengetahui apakah *port* telah terkoneksi pada Arduino IDE. Jika *port* berhasil terkoneksi maka program dapat di *upload*. NodeMCU ESP8266 berhasil terkoneksi dapat dilihat pada Gambar 4.7.



Gambar 4. 7 NodeMCU ESP8266 Berhasil Terkoneksi

2. Tes sensor Soil Moisture

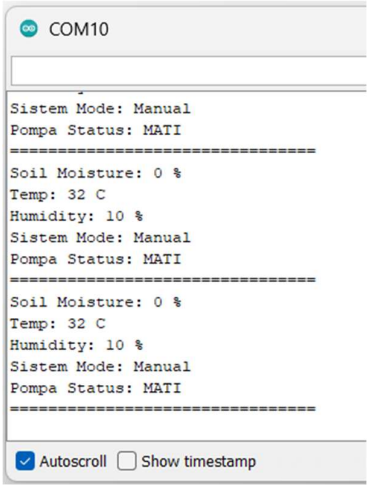
Pengujian ini dilakukan untuk memastikan sensor *soil moisture* berfungsi dengan baik dalam mendeteksi kadar kelembapan tanah. Sensor dihubungkan ke NodeMCU, lalu data dibaca melalui *serial monitor* di Arduino IDE. Tanah kering menghasilkan nilai analog yang lebih rendah, sedangkan tanah basah menghasilkan nilai yang lebih basah. Hasil pengujian menunjukkan bahwa sensor dapat mendeteksi perubahan kondisi tanah secara real-time dengan akurat sesuai perubahan lingkungan. Hasil pengujian dapat dilihat pada Gambar 4.8.



Gambar 4. 8 Sensor Kelembapan Tanah Berfungsi

3. Tes sensor Temperature

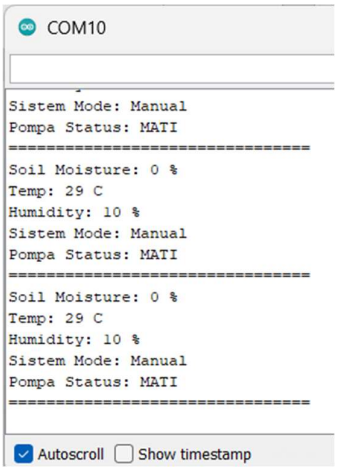
Sensor suhu DS18B20 diuji untuk memastikan kemampuannya dalam membaca suhu lingkungan sekitar tanaman. Sensor dihubungkan ke NodeMCU dan dibaca melalui *serial monitor*. Nilai suhu ditampilkan dalam satuan derajat Celsius. Hasil pengujian menunjukkan bahwa sensor mampu membaca suhu secara *real-time* dan responsif terhadap perubahan suhu lingkungan. Hasil pengujian dapat dilihat pada Gambar 4.9.



Gambar 4. 9 Sensor Suhu(Temp) Berfungsi

4. Tes sensor Humidity

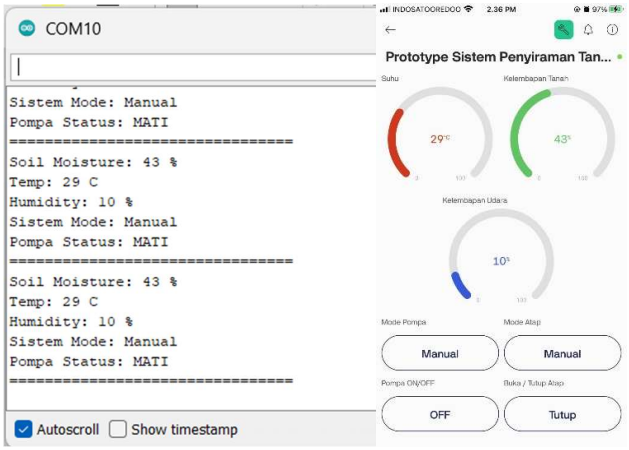
Sensor DHT11 digunakan untuk mendeteksi kelembapan udara di sekitar tanaman. Pengujian dilakukan dengan mengamati pembacaan kelembapan di *serial monitor*. Sensor merespons dengan baik terhadap perubahan kelembapan udara, dan nilai yang dihasilkan stabil dalam kondisi ruangan. Hasil pengujian dapat dilihat pada Gambar 4.10.



Gambar 4. 10 Sensor Kelebapan Udara(Humidity) Berfungsi

5. Uji pengiriman data dari sensor ke aplikasi pemantauan

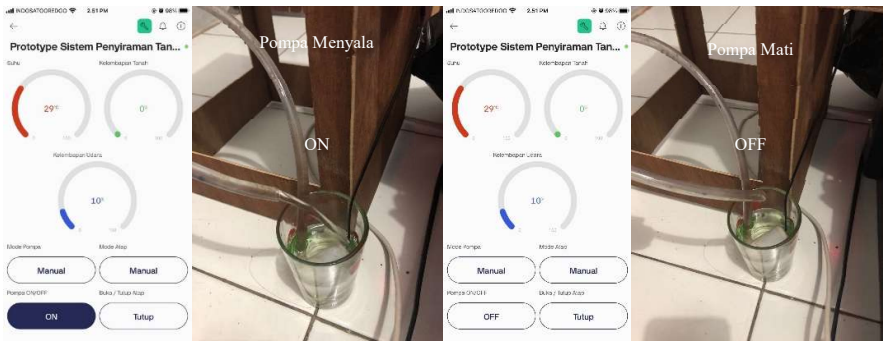
Uji ini bertujuan untuk memastikan bahwa data dari seluruh sensor dapat dikirim secara *real-time* ke aplikasi *Blynk* melalui koneksi Wi-Fi. Hasil menunjukkan bahwa semua nilai sensor kelembapan tanah, suhu, dan kelembapan udara ditampilkan secara akurat pada antarmuka aplikasi *Blynk* di smartphone. Hasil pengujian dapat dilihat pada Gambar 4.11.



Gambar 4. 11 Pengiriman Data Secara Real-Time Berhasil

6. Uji fungsi manual untuk menyalakan atau mematikan pompa melalui aplikasi

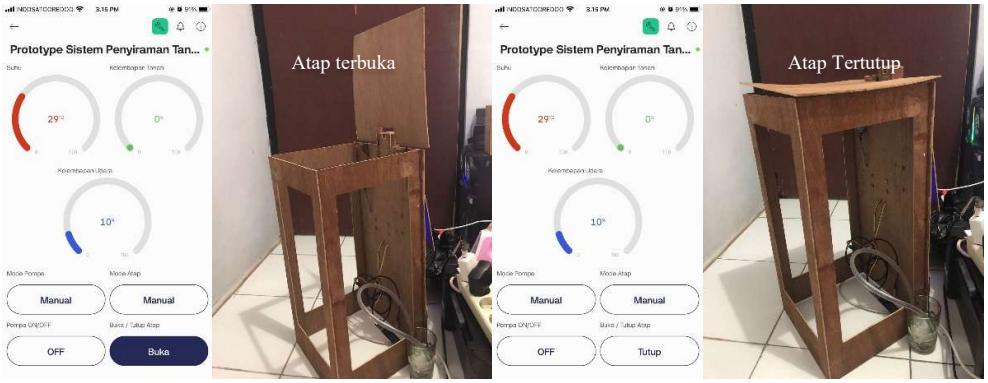
Pengujian ini dilakukan dengan meletakkan pompa air di dalam gelas yang berisi air untuk memastikan bahwa kontrol manual pompa dapat diaktifkan melalui aplikasi *Blynk*. Dalam mode manual, pengguna dapat menyalakan atau mematikan pompa air dengan menekan tombol ON/OFF pada aplikasi. Ketika tombol ON ditekan, pompa menyala. Saat tombol OFF ditekan, pompa berhenti bekerja. Respon pompa terhadap perintah dari aplikasi tergolong cepat dan stabil. Hasil pengujian dapat dilihat pada Gambar 4.12.



Gambar 4. 12 Kontrol Manual Pompa Berfungsi

7. Uji fungsi manual untuk sistem buka tutup atap melalui aplikasi

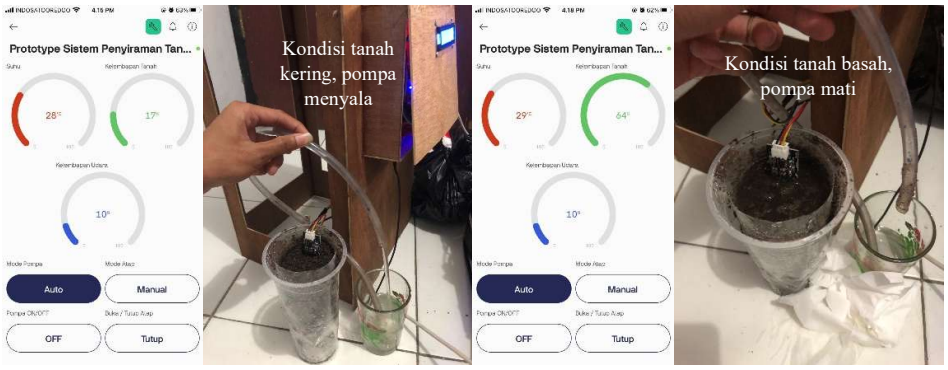
Dalam mode manual, pengguna juga dapat mengontrol sistem buka-tutup atap menggunakan tombol Buka dan Tutup yang tersedia di aplikasi *Blynk*. Ketika suhu belum mencapai ambang batas, namun pengguna ingin menutup atap secara langsung, tombol ini dapat digunakan. Pengujian dilakukan dengan menekan tombol TUTUP untuk menutup atap dan tombol BUKA untuk membukanya kembali. Hasilnya menunjukkan bahwa *motor servo* merespons dengan baik terhadap perintah manual dari aplikasi, dan proses buka-tutup berlangsung secara lancar. Hasil pengujian dapat dilihat pada Gambar 4.13.



Gambar 4. 13 Kontrol Manula Buka-Tutup Atap Berfungsi

8. Uji fungsi otomatis saat kelembapan tanah di bawah ambang batas

Sistem diuji dengan mengondisikan tanah dalam keadaan kering (nilai sensor < 30%). Sensor kelembapan tanah terlebih dahulu diletakkan pada tanah yang benar-benar kering untuk memastikan bahwa sistem dapat mendeteksi kondisi kekurangan air. Hasilnya, sistem secara otomatis mengaktifkan pompa air untuk melakukan penyiraman. Setelah itu, sensor dipindahkan ke tanah yang basah untuk melihat perubahan nilai kelembapan. Ketika nilai kelembapan tanah meningkat dan mencapai batas ideal (> 45%), pompa berhenti bekerja secara otomatis. Hasil pengujian dapat dilihat pada Gambar 4.14.



Gambar 4. 14 Mode Pompa Otomatis Berfungsi

9. Uji fungsi otomatis saat suhu mencapai batas tertentu

Pengujian dilakukan dengan menaikkan suhu lingkungan di atas 27°C. Untuk mensimulasikan peningkatan suhu, sensor suhu dipanaskan menggunakan korek api dari jarak aman agar tidak merusak komponen. Sistem secara otomatis mengaktifkan motor servo untuk menutup atap guna melindungi tanaman dari paparan sinar matahari berlebih. Selanjutnya, untuk menurunkan suhu, sensor dicelupkan ke dalam air bersuhu dingin yang ditempatkan di dalam gelas hingga suhu terbaca turun di bawah suhu 27°C. Ketika suhu kembali di bawah batas yang

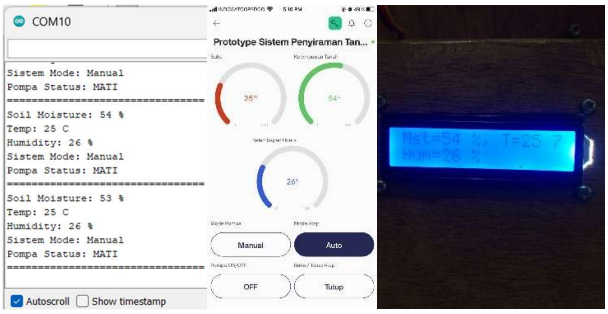
ditentukan, atap kembali terbuka secara otomatis. Hasil pengujian dapat dilihat pada Gambar 4.15.



Gambar 4. 15 Mode Otomatis Buka-Tutup Atap Berfungsi

10. Uji pengiriman data ke LCD

LCD 16x2 digunakan untuk menampilkan data sensor secara lokal. Pengujian dilakukan dengan memastikan bahwa setiap pembacaan sensor ditampilkan di layar LCD secara real-time. Selain itu, data sensor juga dipantau melalui Serial Monitor dan aplikasi Blynk untuk memastikan kesesuaian dan sinkronisasi data di semua tampilan. Hasil menunjukkan bahwa LCD menampilkan nilai kelembapan tanah, suhu, dan kelembapan udara secara akurat dan stabil. Hasil pengujian dapat dilihat pada Gambar 4.16.



Gambar 4. 16 LCD Berfungsi

4.2 Pembahasan

4.2.1 Source Code

Pada proses ini penulis akan membahas *source code* program pada Arduino IDE yang akan digunakan untuk membuat *prototype* sistem penyiraman tanaman apel berbasis *Internet of Things*. Implementasi program dapat dilihat pada Gambar 4.17.

Prototype_Sistem_Penyiraman_Tanaman_Apel_Berbasis_IoT

```
/*****************************************
 * IoT Smart Garden dg Android (Blynk)
 * Board  : NodeMCU  V3
 * Input  : Sensor Soil Moisture, DHT11, DS18B20
 * Output : Relay, Blynk Android, LCD 16x2, Servo
 *****************************************/

#define BLYNK_TEMPLATE_ID "TMPL6b90H3wu8"
#define BLYNK_TEMPLATE_NAME "Smart Garden NodeMCU"
#define BLYNK_AUTH_TOKEN "yRYplz6Nw8TMr6tSgTxli2TYD2E6Ia_D"

#include <DallasTemperature.h>
#include <OneWire.h>
#include <DHT.h>
#include <LiquidCrystal_I2C.h>
#include <Blynk.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Servo.h>

#define BLYNK_PRINT Serial
#define ONE_WIRE_BUS D5
#define DHTPIN D3
#define DHTTYPE DHT11
#define pump D6
#define SERVO_PIN D4

DHT dht(DHTPIN, DHTTYPE);
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
LiquidCrystal_I2C lcd(0x27, 16, 2);
Servo myServo;

const int AirValue = 620;
const int WaterValue = 310;
int soilMoistureValue = 0;
int mst = 0;
int humi, temp, fp, sistem;
int suhuBatas = 27;
int atapMode = 1;
int atapStatus = 0;
```

Prototype_Sistem_Penyiraman_Tanaman_Apel_Berbasis_IoT

```
int SP_LOW = 30;
int SP_HIGH = 45;

char auth[] = "yRYplz6Nw8TMr6tSgTxli2TYD2E6Ia_D";
char ssid[] = "404! Not found";
char pass[] = "Andigantengl3";

//========== BLYNK CONTROL ===========

BLYNK_WRITE(V4) {
  sistem = param.asInt();
  fp = 0;
}

BLYNK_WRITE(V5) {
  if (sistem == 0) {
    int pumpState = param.asInt();
    digitalWrite(pump, pumpState ? LOW : HIGH);
    Serial.print("Pompa Manual: ");
    Serial.println(pumpState ? "ON" : "OFF");
  }
}

BLYNK_WRITE(V7) {
  atapMode = param.asInt();
}

BLYNK_WRITE(V8) {
  if (atapMode == 0) {
    int manualControl = param.asInt();
    int sudutServo = manualControl ? 180 : 0;
    myServo.write(sudutServo);
    atapStatus = manualControl;
    Serial.print("Atap manual ke sudut: ");
    Serial.println(sudutServo);
  }
}

//========== SETUP ==================
void setup() {
  lcd.begin(16, 2);
```

Prototype_Sistem_Penyiraman_Tanaman_Apel_Berbasis_IoT

```
  lcd.init();
  lcd.backlight();
  lcd.clear();
  lcd.print("  Smart Garden  ");
  lcd.setCursor(0, 1);
  lcd.print("   NodeMCU V3   ");

  Serial.begin(9600);
  sensors.begin();
  dht.begin();

  pinMode(pump, OUTPUT);
  digitalWrite(pump, HIGH);
  delay(1500);

  Blynk.begin(auth, ssid, pass);
  delay(1500);

  lcd.clear();
  lcd.print("Mst=   %, T=   C");
  lcd.setCursor(0, 1);
  lcd.print("Hum=   %       ");

  myServo.attach(SERVO_PIN);
  myServo.write(0);
}

//============= LOOP ================
void loop() {
  Blynk.run();
  sensors.requestTemperatures();
  temp = sensors.getTempCByIndex(0);
  lcd.setCursor(12, 0);
  lcd.print("   ");
  lcd.setCursor(12, 0);
  lcd.print(temp);

  soilMoistureValue = analogRead(A0);
  mst = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
  mst = constrain(mst, 0, 100);
  lcd.setCursor(4, 0);
```

Prototype_Sistem_Penyiraman_Tanaman_Apel_Berbasis_IoT

```
  lcd.print("   ");
  lcd.setCursor(4, 0);
  lcd.print(mst);

  humi = dht.readHumidity();
  if (!isnan(humi)) {
    lcd.setCursor(4, 1);
    lcd.print("   ");
    lcd.setCursor(4, 1);
    lcd.print(humi);
  }

  // === Kontrol Pompa Otomatis ===
  if (sistem == 1) {
    if ((mst < SP_LOW) && (fp == 0)) {
      digitalWrite(pump, LOW);
      fp = 1;
      Serial.println("Pompa otomatis: NYALA");
    } else if ((mst > SP_HIGH) && (fp == 1)) {
      digitalWrite(pump, HIGH);
      fp = 0;
      Serial.println("Pompa otomatis: MATI");
    }
  }

  // === Kontrol Atap Otomatis ===
  if (atapMode == 1) {
    aturAtap();
  }

  // Kirim data ke Blynk
  Blynk.virtualWrite(V1, temp);
  Blynk.virtualWrite(V2, mst);
  Blynk.virtualWrite(V3, humi);

  // Debug Serial
  Serial.print("Soil Moisture: "); Serial.print(mst); Serial.println(" %");
  Serial.print("Temp: "); Serial.print(temp); Serial.println(" C");
  Serial.print("Humidity: "); Serial.print(humi); Serial.println(" %");
  Serial.print("Sistem Mode: "); Serial.println(sistem == 1 ? "Auto" : "Manual");
  Serial.print("Pompa Status: "); Serial.println(digitalRead(pump) == LOW ? "NYALA" : "MATI");
```

```
  Serial.println("==================================");

  delay(5000);
}

//======= FUNGSI ATAP OTOMATIS ========
void aturAtap() {
  int sudutServo;

  if (temp <= 26) {
    sudutServo = 180;
  } else if (temp >= 27) {
    sudutServo = 0;
  } else {
    sudutServo = map(temp, 26, 27, 180, 0);
  }

  static int sudutSebelumnya = -1;
  if (sudutServo != sudutSebelumnya) {
    myServo.write(sudutServo);
    Serial.print("Atap disesuaikan ke sudut: ");
    Serial.println(sudutServo);
    sudutSebelumnya = sudutServo;
  }

  atapStatus = (sudutServo > 0) ? 1 : 0;
}
```

Gambar 4. 17 Code Program

Baris pertama pada *source code* adalah untuk koneksi dari NodeMCU ESP8266 ke aplikasi blynk. Terdapat auth token, kode tersebut adalah token blynk yang sudah di copy-paste pada aplikasi blynk. Baris token dapat dilihat pada Gambar 4.18.

```
#define BLYNK_TEMPLATE_ID "TMPL6b90H3wu8"
#define BLYNK_TEMPLATE_NAME "Smart Garden NodeMCU"
#define BLYNK_AUTH_TOKEN "yRYplz6Nw8TMr6tSgTxli2TYD2E6Ia_D"
```

Gambar 4. 18 Konfigurasi Blynk

Baris kedua adalah perintah untuk memanggil library board NodeMCU ESP8266, *library* sensor kelembapan tanah, *library* sensor kelembapan udara, *library* sensor suhu, *library* motor servo, *library* LCD display dan *library* blynk. Fungsi dari library ini untuk memudahkan dalam penulisan program. Library dapat dilihat pada Gambar 4.19.

```
#include <DallasTemperature.h>  // Untuk sensor suhu DS18B20
#include <OneWire.h>            // Protokol komunikasi DS18B20
#include <DHT.h>                // Untuk sensor DHT11
#include <LiquidCrystal_I2C.h>  // Untuk LCD I2C 16x2
#include <Blynk.h>              // Core Blynk
#include <ESP8266WiFi.h>        // WiFi ESP8266
#include <BlynkSimpleEsp8266.h> // Blynk khusus untuk ESP8266
#include <Servo.h>              // Mengontrol motor servo
```

Gambar 4. 19 Library

Baris selanjutnya digunakan untuk memberikan nama atau label yang lebih mudah dipahami pada pin-pin yang digunakan di NodeMCU. Misalnya, #define ONE_WIRE_BUS D5 berarti pin D5 akan digunakan untuk sensor suhu DS18B20, sedangkan #define DHTPIN D3 menunjukkan bahwa pin D3 digunakan untuk sensor DHT11. #define DHTTYPE DHT11 memberi tahu bahwa jenis sensor DHT yang digunakan adalah DHT11. Kemudian, #define pump D6 menandai bahwa pin

D6 dipakai untuk mengontrol pompa air, dan #define SERVO_PIN D4 menunjukkan bahwa pin D4 digunakan untuk motor servo. Dengan menggunakan #define, kita dapat menulis kode program yang lebih rapi dan mudah dipahami, karena kita bisa menyebut nama fungsional seperti pump atau SERVO_PIN dari pada menyebut langsung nomor pinnya dapat dilihat pada Gambar 4.20.

```
#define BLYNK_PRINT Serial
#define ONE_WIRE_BUS D5
#define DHTPIN D3
#define DHTTYPE DHT11
#define pump D6
#define SERVO_PIN D4

DHT dht(DHTPIN, DHTTYPE);              // Objek sensor DHT11
OneWire oneWire(ONE_WIRE_BUS);         // Objek protokol OneWire
DallasTemperature sensors(&oneWire);   // Objek DS18B20
LiquidCrystal_I2C lcd(0x27, 16, 2);    // Objek LCD 16x2 alamat 0x27
Servo myServo;                         // Objek servo
```

Gambar 4. 20 Pin dan Objek

Baris selanjutnya baris variabel dan kontrol, kode ini mendefinisikan beberapa variabel penting untuk membaca sensor dan mengontrol logika *prototype*. Variabel AirValue dan WaterValue adalah nilai batas kalibrasi sensor kelembapan tanah, di mana 620 mewakili kondisi tanah kering (air), dan 310 mewakili kondisi tanah basah (water). Variabel soilMoistureValue digunakan untuk menyimpan nilai mentah dari sensor kelembapan tanah, sedangkan mst digunakan untuk menyimpan nilai kelembapan yang sudah dikonversi ke dalam bentuk persen.

Selanjutnya, humi dan temp masing-masing menyimpan nilai kelembapan udara dan suhu udara dari sensor, fp bisa digunakan untuk status pompa (misalnya ON/OFF), sistem untuk mode kerja otomatis atau manual, suhuBatas adalah batas suhu maksimum sebelum atap otomatis terbuka (dalam °C), atapMode menandakan

apakah mode atap otomatis aktif (1) atau manual (0), dan atapStatus menunjukkan status atap saat ini (terbuka/tutup).

Terakhir, SP_LOW dan SP_HIGH adalah nilai batas bawah dan atas untuk kontrol pompa dengan **histeresis**, yaitu teknik agar pompa tidak sering hidup-mati saat nilai kelembapan mendekati ambang batas. Pompa akan menyala jika kelembapan di bawah SP_LOW, dan mati jika sudah naik di atas SP_HIGH. Baris kode dapat dilihat pada Gambar 4.21.

```
const int AirValue = 620;
const int WaterValue = 310;
int soilMoistureValue = 0;
int mst = 0;
int humi, temp, fp, sistem;
int suhuBatas = 27;
int atapMode = 1;
int atapStatus = 0;

int SP_LOW = 30;
int SP_HIGH = 45;
```

Gambar 4. 21 Variabel Kalibrasi dan Kontrol

Baris selanjutnya baris Wi-Fi dan auth, kode ini digunakan untuk menyimpan data autentikasi dan informasi jaringan WiFi yang diperlukan agar NodeMCU dapat terhubung ke internet dan ke platform Blynk. Variabel auth[] menyimpan Blynk Auth Token, yaitu kunci unik yang menghubungkan perangkat fisik ke akun dan proyek di aplikasi Blynk. Variabel ssid[] berisi nama WiFi (SSID) yang akan digunakan oleh NodeMCU untuk terhubung ke jaringan, sedangkan pass[] adalah password WiFi tersebut. Ketiga informasi ini wajib dimasukkan agar sistem dapat mengakses internet dan mengirim data sensor atau menerima perintah dari aplikasi Blynk secara *real-time*. Baris kode dapat dilihat pada Gambar 4.22.

```
char auth[] = "yRYplz6Nw8TMr6tSgTxli2TYD2E6Ia_D";
char ssid[] = "404! Not found";
char pass[] = "Andigantengl3";
```

Gambar 4. 22 WiFi dan Auth

Baris selanjutnya berisi fungsi-fungsi BLYNK_WRITE() yang digunakan untuk membaca input dari aplikasi Blynk melalui pin virtual (V4, V5, V7, dan V8).

Fungsi BLYNK_WRITE(V4) digunakan untuk menentukan mode kerja sistem: otomatis (1) atau manual (0). Nilai ini disimpan dalam variabel sistem. Saat mode diubah, pompa otomatis dimatikan (fp = 0) untuk mencegah konflik antara mode otomatis dan manual.

Fungsi BLYNK_WRITE(V5) memungkinkan kontrol pompa air secara manual, tetapi hanya jika sistem dalam mode manual (sistem == 0). Nilai dari Blynk (1 atau 0) digunakan untuk mengatur pin pompa. Karena relay aktif-low, nilai LOW berarti pompa menyala (ON), dan HIGH berarti pompa mati (OFF). Status pompa juga ditampilkan di Serial Monitor.

Fungsi BLYNK_WRITE(V7) digunakan untuk menyimpan status mode kontrol atap: otomatis (1) atau manual (0) ke dalam variabel atapMode.

Fungsi BLYNK_WRITE(V8) hanya dijalankan jika atap dalam mode manual (atapMode == 0). Fungsi ini mengubah posisi servo motor berdasarkan input dari Blynk: jika pengguna menyalakan kontrol (input 1), maka servo bergerak ke sudut 180° (atap terbuka); jika dimatikan (input 0), servo kembali ke 0° (atap tertutup). Baris kode dapat dilihat pada Gambar 4.23.

```
//========== BLYNK CONTROL ============

BLYNK_WRITE(V4) {
  sistem = param.asInt();
  fp = 0;
}

BLYNK_WRITE(V5) {
  if (sistem == 0) {
    int pumpState = param.asInt();
    digitalWrite(pump, pumpState ? LOW : HIGH);
    Serial.print("Pompa Manual: ");
    Serial.println(pumpState ? "ON" : "OFF");
  }
}

BLYNK_WRITE(V7) {
  atapMode = param.asInt();
}

BLYNK_WRITE(V8) {
  if (atapMode == 0) {
    int manualControl = param.asInt();
    int sudutServo = manualControl ? 180 : 0;
    myServo.write(sudutServo);
    atapStatus = manualControl;
    Serial.print("Atap manual ke sudut: ");
    Serial.println(sudutServo);
  }
}
```

Gambar 4. 23 Fungsi Blynk

Bris selanjutnya berisi inisialisasi awal sistem saat NodeMCU pertama kali dinyalakan. Pertama, LCD 16x2 diaktifkan dengan perintah lcd.begin(), lcd.init(), dan lcd.backlight(), lalu menampilkan tulisan "Smart Garden" dan "NodeMCU V3" sebagai salam pembuka. Kemudian Serial.begin(9600) digunakan untuk memulai komunikasi serial agar data bisa ditampilkan di Serial Monitor. Sensor suhu DS18B20 dan DHT11 juga diinisialisasi dengan sensors.begin() dan dht.begin().

Pin pompa diatur sebagai output menggunakan pinMode(pump, OUTPUT) dan langsung dimatikan (karena aktif LOW) dengan digitalWrite(pump, HIGH). Setelah itu, koneksi ke aplikasi Blynk dilakukan melalui Blynk.begin(auth, ssid,

pass), yang akan menghubungkan NodeMCU ke internet dan platform Blynk menggunakan token dan WiFi yang telah ditentukan.

Selanjutnya, LCD diperbarui untuk menampilkan format kosong yang nantinya diisi nilai-nilai sensor secara berkala. Terakhir, motor servo diinisialisasi dengan myServo.attach(SERVO_PIN) dan diposisikan ke sudut 0 derajat sebagai posisi awal (atap tertutup). Fungsi setup() ini memastikan bahwa semua perangkat keras siap dan sistem terhubung ke Blynk sebelum masuk ke loop utama. Baris kode dapat dilihat pada Gambar 4.24.

```
//============ SETUP ==================
void setup() {
  lcd.begin(16, 2);
  lcd.init();
  lcd.backlight();
  lcd.clear();
  lcd.print("  Smart Garden  ");
  lcd.setCursor(0, 1);
  lcd.print("   NodeMCU V3   ");

  Serial.begin(9600);
  sensors.begin();
  dht.begin();

  pinMode(pump, OUTPUT);
  digitalWrite(pump, HIGH); // Pompa mati dulu (aktif LOW)
  delay(1500);

  Blynk.begin(auth, ssid, pass); // Koneksi ke Blynk
  delay(1500);

  lcd.clear();
  lcd.print("Mst=   %, T=   C");
  lcd.setCursor(0, 1);
  lcd.print("Hum=   %       ");

  myServo.attach(SERVO_PIN);
  myServo.write(0); // Servo awal 0 derajat
}
```

Gambar 4. 24 Setup()

Fungsi loop() ini merupakan inti dari program, yang berjalan terus-menerus selama perangkat aktif. Fungsi pertama yang dipanggil adalah Blynk.run(), yang menjaga koneksi dan sinkronisasi dengan aplikasi Blynk.

Selanjutnya, membaca suhu dari sensor DS18B20 menggunakan sensors.requestTemperatures() dan menyimpan nilai suhu dalam variabel temp.

Nilai ini ditampilkan di LCD pada kolom kanan baris pertama. Selanjutnya, membaca kelembapan tanah dari pin analog A0. Nilai mentah ini diubah menjadi persen menggunakan fungsi map() dan dibatasi dari 0–100 dengan constrain(). Hasil akhirnya disimpan di variabel mst dan juga ditampilkan di LCD.

Selanjutnya, membaca kelembapan udara dari sensor DHT11. Nilainya disimpan di variabel humi dan ditampilkan di LCD pada baris kedua, jika pembacaan berhasil (bukan NaN).

Selanjutnya, logika pompa otomatis, yang hanya aktif saat sistem berada dalam mode otomatis (sistem == 1). Jika kelembapan tanah turun di bawah SP_LOW, pompa menyala (LOW). Sebaliknya, jika kelembapan naik di atas SP_HIGH, pompa dimatikan (HIGH). Status pompa dicetak ke Serial Monitor.

Selanjutnya, mengecek apakah mode atap otomatis sedang aktif (atapMode == 1). Jika ya, maka fungsi aturAtap() dipanggil untuk mengatur sudut servo berdasarkan suhu.

Selanjutnya, mengirimkan data suhu (temp), kelembapan tanah (mst), dan kelembapan udara (humi) ke aplikasi Blynk menggunakan pin virtual V1, V2, dan V3 agar bisa dimonitor secara *real-time*.

Terakhir, menampilkan debug ke Serial Monitor, termasuk nilai kelembapan tanah, suhu, kelembapan udara, mode sistem (otomatis/manual), dan status pompa, lalu diakhiri dengan delay(5000) agar loop berjalan setiap 5 detik. Dengan urutan ini, sistem dapat membaca data sensor, mengontrol perangkat, dan mengirimkan informasi ke aplikasi dengan lancar. Baris kode dapat dilihat pada Gambar 4.25.

```
//============== LOOP ==================
void loop() {
  Blynk.run();
  sensors.requestTemperatures();
  temp = sensors.getTempCByIndex(0);
  lcd.setCursor(12, 0);
  lcd.print("   ");
  lcd.setCursor(12, 0);
  lcd.print(temp);

  soilMoistureValue = analogRead(A0);
  mst = map(soilMoistureValue, AirValue, WaterValue, 0, 100);
  mst = constrain(mst, 0, 100);
  lcd.setCursor(4, 0);
  lcd.print("   ");
  lcd.setCursor(4, 0);
  lcd.print(mst);

  humi = dht.readHumidity();
  if (!isnan(humi)) {
    lcd.setCursor(4, 1);
    lcd.print("   ");
    lcd.setCursor(4, 1);
    lcd.print(humi);
  }
// === Kontrol Pompa Otomatis ===
 if (sistem == 1) {
   if ((mst < SP_LOW) && (fp == 0)) {
     digitalWrite(pump, LOW);
     fp = 1;
     Serial.println("Pompa otomatis: NYALA");
   } else if ((mst > SP_HIGH) && (fp == 1)) {
     digitalWrite(pump, HIGH);
     fp = 0;
     Serial.println("Pompa otomatis: MATI");
   }
 }

 // === Kontrol Atap Otomatis ===
 if (atapMode == 1) {
   aturAtap();
 }

 // Kirim data ke Blynk
 Blynk.virtualWrite(V1, temp);
 Blynk.virtualWrite(V2, mst);
 Blynk.virtualWrite(V3, humi);

 // Debug Serial
 Serial.print("Soil Moisture: "); Serial.print(mst); Serial.println(" %");
 Serial.print("Temp: "); Serial.print(temp); Serial.println(" C");
 Serial.print("Humidity: "); Serial.print(humi); Serial.println(" %");
 Serial.print("Sistem Mode: "); Serial.println(sistem == 1 ? "Auto" : "Manual");
 Serial.print("Pompa Status: "); Serial.println(digitalRead(pump) == LOW ? "NYALA" : "MATI");
 Serial.println("================================");

 delay(5000);
}
```

Gambar 4. 25 Loop()

Kemudian baris yang terakhir digunakan untuk mengatur posisi atap otomatis berdasarkan suhu yang terbaca dari sensor DS18B20. Jika suhu kurang dari atau sama dengan 26°C, atap akan dibuka sepenuhnya dengan posisi servo pada sudut 180°. Sebaliknya, jika suhu lebih dari atau sama dengan 27°C, atap akan ditutup sepenuhnya dengan posisi servo pada sudut 0°. Jika suhu berada di antara 26°C dan 27°C, sudut servo dihitung secara proporsional menggunakan fungsi map() untuk menghasilkan sudut yang sesuai dengan suhu tersebut. Fungsi ini juga memastikan bahwa servo hanya bergerak jika ada perubahan sudut yang signifikan, dengan membandingkan sudut baru dengan sudut sebelumnya yang disimpan dalam variabel sudutSebelumnya. Jika ada perubahan, posisi servo diubah, dan perubahan tersebut dicetak ke Serial Monitor. Selain itu, status atap (atapStatus) ditentukan berdasarkan posisi servo, di mana jika sudut lebih besar dari 0, atap dianggap terbuka (atapStatus = 1), dan jika sudut 0, atap dianggap tertutup (atapStatus = 0). Dengan demikian, fungsi ini memungkinkan atap untuk menyesuaikan secara otomatis dan efisien berdasarkan suhu yang terukur. Baris kode dapat dilihat pada Gambar 4.26.

```
//======= FUNGSI ATAP OTOMATIS ========
void aturAtap() {
  int sudutServo;

  if (temp <= 26) {
    sudutServo = 180;
  } else if (temp >= 27) {
    sudutServo = 0;
  } else {
    sudutServo = map(temp, 26, 27, 180, 0);
  }

  static int sudutSebelumnya = -1;
  if (sudutServo != sudutSebelumnya) {
    myServo.write(sudutServo);
    Serial.print("Atap disesuaikan ke sudut: ");
    Serial.println(sudutServo);
    sudutSebelumnya = sudutServo;
  }

  atapStatus = (sudutServo > 0) ? 1 : 0;
}
```

Gambar 4. 26 Fungsi aturAtap()

4.3 Penelitian

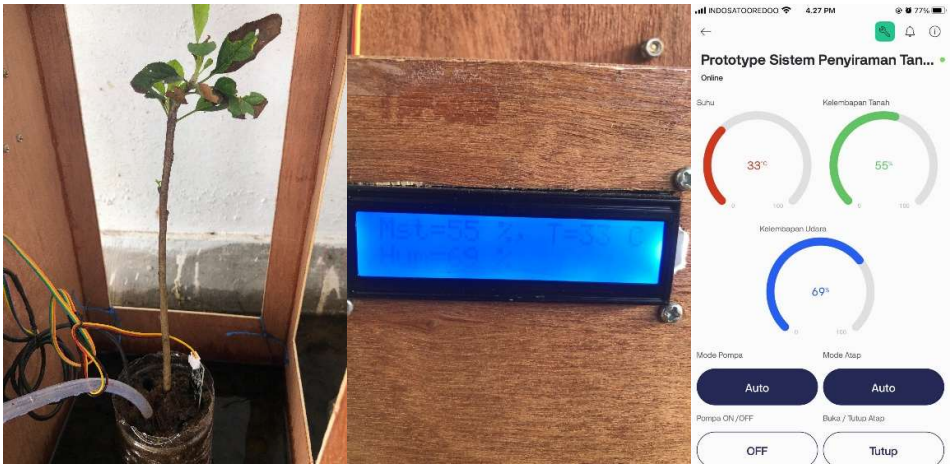
Pengamatan ini dilakukan dengan mengamati tanaman apel selama tiga minggu dan setiap minggunya hanya satu kali pengamatan. Tujuannya adalah untuk melihat bagaimana tanaman tersebut beradaptasi terhadap kondisi di dataran rendah serta mengamati perubahan fisik yang terjadi, seperti pada daun dan batang. Melalui observasi ini, diharapkan dapat diketahui apakah tahan tanaman apel dapat tumbuh dengan baik.

a. Pengamatan Minggu Pertama

Pada minggu pertama pengamatan, tanaman apel berada dalam kondisi yang baik dengan tingkat kelembapan tanah 55%, kelembapan udara 69% dan suhu 33°C sebagaimana ditampilkan pada LCD 16x2 dan aplikasi Blynk. Setiap sensor yang terhubung dengan NodeMCU ESP8266 berhasil mengirimkan data secara *real-time*

ke sistem, yang kemudian ditampilkan di LCD dan juga dapat diakses diaplikasi blynk melalui perangkat seluler.

Secara visual, tanaman apel tampak kokoh, dengan batang tegak dan daun yang masih segar, mulai tumbuh cabang serta pucuk yang baru tanpa tanda-tanda layu. Tampak terlihat juga ada beberapa daun yang rusak dan menghitam akibat pengiriman bibit apel melalui kurir sebelumnya. *Prototype* sistem penyiraman tanaman apel berbasis *internet of things* bekerja dengan baik, terbukti dari data sensor yang terus diperbarui secara otomatis. Berdasarkan pengamatan minggu pertama, dapat disimpulkan bahwa tanaman apel belum mengalami stres akibat kekurangan air dan paparan sinar matahari secara berlebih serta masih mendukung pertumbuhan. Berikut dapat dilihat pada Gambar 4.27.



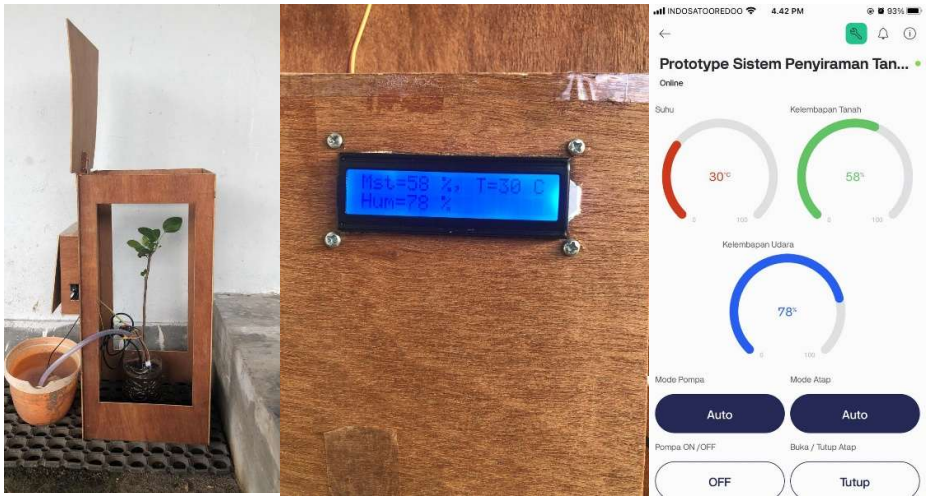
Gambar 4. 27 Pengamatan Minggu Pertama

b. Pengamatan Minggu Kedua

Pada minggu kedua pengamatan, tanaman apel masih dalam kondisi yang baik dengan tingkat kelembapan tanah 58%, kelembapat udara 78% dan suhu 30°C sebagaimana ditampilkan pada LCD 16x2 dan aplikasi Blynk. Setiap sensor yang

terhubung dengan NodeMCU ESP8266 berhasil mengirimkan data secara *real-time* ke sistem, yang kemudian ditampilkan di LCD dan juga dapat diakses diaplikasi blynk melalui perangkat seluler.

Secara visual, tanaman apel tampak masih kokoh, dengan batang tegak dan daun yang masih segar. Tampak terlihat daun yang rusak dan menghitam sebelumnya beberapa sudah gugur dan mulai gugur serta mulai penghijau kembali. Hal ini mengindikasikan adanya proses pemulihan dan regenerasi pada tanaman. *Prototype* sistem penyiraman tanaman apel berbasis *internet of things* masih bekerja dengan baik, terbukti sistem berjalan dengan baik. Berdasarkan pengamatan minggu kedua, dapat disimpulkan bahwa tanaman apel sudah tumbuh lebih baik tanpa ada mengalami stres akibat kekurangan air dan paparan sinar matahari secara berlebih. Berikut dapat dilihat pada Gambar 4.28.



Gambar 4. 28 Penelitian Minggu Kedua

c. Pengamatan Minggu Ketiga

Pada minggu ketiga pengamatan atau yang terahir, tanaman apel sudah dalam kondisi sangat baik dengan tingkat kelembapan tanah 58%, kelembapat udara 73%

dan suhu 27℃ sebagaimana ditampilkan pada LCD 16x2 dan aplikasi Blynk. Setiap sensor yang terhubung dengan NodeMCU ESP8266 berhasil mengirimkan data secara *real-time* ke sistem, yang kemudian ditampilkan di LCD dan juga dapat diakses diaplikasi blynk melalui perangkat seluler.

Secara visual, tanaman apel tampak masih kokoh dan daun yang segar. Tampak terlihat pucuk dan daun baru mulai tumbuh menggantikan daun yang rusak sebelumnya. Prototipe sistem penyiraman otomatis berbasis Internet of Things (IoT) juga terbukti berjalan dengan baik selama pengamatan. Sistem mampu mempertahankan kondisi lingkungan yang mendukung pertumbuhan tanaman, serta merespons data sensor secara akurat. Berdasarkan pengamatan minggu ketiga, dapat disimpulkan bahwa sistem IoT yang diterapkan efektif dalam memantau dan mengatur kebutuhan air tanaman apel, serta memberikan dampak positif terhadap pertumbuhan tanaman secara keseluruhan. Berikut dapat dilihat pada Gambar 4.29.



Gambar 4. 29 Penelitian Minggu Ketiga

Tabel 4. 2 Tabel Pengamatan

Minggu	Kelembapan Tanah (%)	Kelembapan Udara (%)	Suhu (°C)	Kondisi Tanaman Apel Secara Visual
1	55%	69%	33 °C	Batang tegak, daun segar, mulai tumbuh cabang dan pucuk baru. Beberapa daun rusak/menghitam akibat pengiriman.
2	58%	78%	30 °C	Tanaman tetap kokoh, daun segar. Daun rusak mulai gugur dan tergantikan dengan daun baru.
3	58%	73%	27 °C	Tanaman dalam kondisi sangat baik, tumbuh pucuk dan daun baru menggantikan yang rusak.

Hasil penelitian menunjukkan bahwa sistem penyiraman otomatis dan kontrol atap berbasis *Internet of Things (IoT)* yang dirancang dan diimplementasikan berhasil memenuhi seluruh tujuan yang telah ditetapkan. Sistem mampu bekerja secara otomatis dengan membaca data dari sensor kelembapan tanah, sensor suhu, dan sensor kelembapan udara. Ketika kelembapan tanah terdeteksi di bawah ambang batas 30%, sistem secara otomatis mengaktifkan pompa penyiraman, dan akan berhenti saat kelembapan tanah kembali di atas 45%. Hal ini membuktikan bahwa sistem efektif dalam menjaga kadar air tanah yang ideal tanpa pemborosan air. Selain itu, sistem juga berhasil mengontrol paparan sinar matahari terhadap tanaman menggunakan *motor servo* yang digerakkan secara otomatis berdasarkan data suhu. Ketika suhu lingkungan melebihi 27°C, atap akan tertutup untuk melindungi tanaman dari panas berlebih, dan akan terbuka kembali saat suhu menurun ke rentang ideal antara 16°C hingga 27°C. Semua data dari sensor

ditampilkan secara *real-time* melalui LCD dan aplikasi *Blynk*, yang juga memungkinkan pengguna melakukan kontrol manual dari jarak jauh, baik untuk menyalakan pompa maupun mengatur buka tutup atap.



Gambar 4. 30 Before & After

Efektivitas sistem ini turut ditunjukkan melalui dokumentasi visual pertumbuhan tanaman apel dari minggu ke minggu. Pada Gambar 4.29 dokumentasi minggu pertama, tanaman terlihat layu dengan warna daun yang pucat, serta beberapa batang tampak kering—menandakan kondisi lingkungan yang belum optimal. Namun setelah sistem diterapkan dan berjalan secara otomatis selama tiga minggu, terlihat perubahan yang signifikan pada tanaman. Pada dokumentasi minggu ketiga, tanaman apel menunjukkan pertumbuhan yang lebih sehat, dengan daun yang lebih hijau, segar, dan mulai munculnya tunas-tunas baru. Perubahan ini menunjukkan bahwa kondisi lingkungan yang dijaga secara otomatis oleh sistem IoT mampu menciptakan lingkungan tumbuh yang lebih stabil dan

sesuai untuk tanaman apel, meskipun berada di dataran rendah dengan tantangan suhu tinggi dan kelembapan yang rendah. Dengan demikian, sistem ini terbukti mampu mendukung pertumbuhan tanaman secara optimal dan efisien dalam penggunaan waktu, tenaga, dan sumber daya.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengamatan dan implementasi *prototype* sistem penyiraman tanaman apel berbasis *Internet of Things (IoT)*, maka dapat disimpulkan bahwa:

a. Sistem berhasil dirancang dan diimplementasikan dengan memanfaatkan NodeMCU ESP8266 sebagai pengendali utama, sensor kelembapan tanah, sensor suhu, dan sensor kelembapan udara untuk memantau kondisi lingkungan tanaman apel secara *real-time*.
b. Penyiraman tanaman dapat dilakukan secara otomatis ketika kelembapan tanah berada di bawah ambang batas tertentu, serta pengaturan atap juga dilakukan secara otomatis berdasarkan suhu lingkungan. Hal ini menunjukkan bahwa sistem mampu bekerja secara mandiri tanpa intervensi manual yang intensif.
c. Aplikasi *Blynk* berfungsi dengan baik untuk monitoring dan kontrol jarak jauh, memungkinkan pengguna memantau dan mengontrol sistem kapan saja dan di mana saja.
d. Hasil pengujian *black box* membuktikan bahwa semua komponen perangkat keras dan perangkat lunak berfungsi sesuai dengan tujuan perancangan, baik dalam mode otomatis maupun manual.
e. Dengan *prototype* ini, proses perawatan tanaman apel menjadi lebih efisien, hemat waktu dan tenaga, serta dapat mendukung pertumbuhan tanaman secara optimal di daerah dataran rendah.

5.2 Saran

Adapun saran yang dapat diberikan untuk pengembangan lebih lanjut dari pengamatan ini adalah sebagai berikut:

a. Disarankan untuk menggunakan NodeMCU ESP32 pada pengembangan sistem berikutnya. ESP32 memiliki lebih banyak pin analog (termasuk dua pin A0 atau ADC) dan kapasitas pemrosesan yang lebih tinggi dibandingkan ESP8266. Hal ini memungkinkan integrasi lebih banyak sensor atau aktuator sekaligus, sehingga sistem dapat dikembangkan menjadi lebih kompleks dan fleksibel.
b. Sistem dapat ditingkatkan dengan menambahkan fitur notifikasi otomatis melalui email atau pesan instan (WhatsApp/Telegram) saat kondisi kritis terjadi, seperti kelembapan sangat rendah atau gangguan koneksi.
c. Perlu dilakukan pengujian langsung di lahan pertanian apel yang sebenarnya untuk menilai ketahanan perangkat terhadap kondisi cuaca dan lingkungan yang sesungguhnya, serta mengevaluasi performa sistem dalam jangka panjang.
d. Disarankan untuk mengintegrasikan sistem dengan sumber energi terbarukan seperti panel surya agar sistem dapat digunakan di lokasi pertanian yang jauh dari sumber listrik utama.
e. Untuk pengembangan ke depan, sistem dapat ditambahkan algoritma pembelajaran mesin (*machine learning*) guna memprediksi kebutuhan penyiraman berdasarkan pola cuaca dan pertumbuhan tanaman.

DAFTAR PUSTAKA

adar BakhshBaloch, Q. (2017). *Dymas Arrio Prihutomo*. *11*(1), 92–105. https://repository.ub.ac.id/id/eprint/4595/1/DYMAS ARRIO PRIHUTOMO.pdf

Ahmad fatoni,Dhany Dwi Nugroho, A. I. (2015). Rancang Bangun Alat Pembelajaran Microcontroller Berbasis ATmega 328 di Universitas Serang Raya. *JurnalJurnal PROSISKO Vol. 2 No. 1 Maret 2015*, *2*(1), 10–18.

Anggara, D. S., Suryanto, A., & Ainurrasjid. (2017). Kendala produksi apel (Malus sylvestris Mill) Var. Manalagi di Desa Poncokusumo Kabupaten Malang. *Jurnal Produksi Tanaman*, *5*(2), 198–207.

Ardiansah, R., Susanto, R., & Pradana, A. I. (2023). JUPITER (Jurnal Pendidikan Teknik Elektro) Sistem Penyiraman Otomatis Pada Tanaman dengan Monitoring Berbasis IoT (Internet of Things). *Jurnal Pendidikan Teknik Electro*, *08*(01), 31–38.

Chandra, Subandi, F. (2023). Prototype Sistem Alat Penyiraman Tanaman Cabai Otomatis Berbasis Web Menggunakan Mikrokontroler Nodemcu Esp8266. *Teknologi Informasi (SENAFTI)*, *2*(1), 500–507.

Effendi, N., Ramadhani, W., & Farida, F. (2022). Perancangan Sistem Penyiraman Tanaman Otomatis Menggunakan Sensor Kelembapan Tanah Berbasis IoT. *Jurnal CoSciTech (Computer Science and Information Technology)*, *3*(2), 91–98. https://doi.org/10.37859/coscitech.v3i2.3923

Hariono, T., Mahdalena, A., & Ashoumi, H. (2021). Automatic Water Temperature Control System In Hydroponic Plants With Peltier Tec1 12706 And Temperature Sensors DS18B20. *MULTIDISCIPLINE - International Conference 2021*, *1*(1), 438–445.

Isnawati, I., & Ali, H. (2024). Pengaruh Pendidikan, Informasi dan Komunikasi terhadap Internet of Things. *Jurnal Manajemen Pendidikan Dan Ilmu Sosial*, *5*(3), 312–319. https://doi.org/10.38035/jmpis.v5i3.1953

Khaledi, M. T. Al, Nasri, & Hanafi. (2022). RANCANG BANGUN SISTEM RUMAH PINTAR MENGGUNAKAN PLATFORM GOOGLE FIREBASE BERBASIS IoT (INTERNET of THINGS). *Jurnal Tektro*, *06*(02), 194–202.

Meisak, D., Hendri, & Agustini, S. R. (2022). Penerapan Metode Prototype Pada Perancangan Sistem Informasi Penjualan Mediatama Solusindo Jambi. *STORAGE: Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *1*(4), 1–11. https://doi.org/10.55123/storage.v1i4.1066

Mindasari, S., As'ad, M., & Meilantika, D. (2022). Sistem Keamanan Kotak Amal di Musala Sabilul Khasanah Berbasis Arduino UNO. *Jurnal Teknik Informatika Mahakarya (JTIM)*, *5*(2), 7–13.

Nur Alfan, A., & Ramadhan, V. (2022). Prototype Detektor Gas Dan Monitoring

Suhu Berbasis Arduino Uno. *PROSISKO: Jurnal Pengembangan Riset Dan Observasi Sistem Komputer*, *9*(2), 61–69. https://doi.org/10.30656/prosisko.v9i2.5380

Prasetyo & Rahaju. (2012). *(Malus sylvestris Mill) DENGAN MENGGUNAKAN pertumbuhan baru dalam sektor pertanian , karena memiliki potensi pasar yang buah apel , sehingga impor merupakan satu-satunya jalan yang ditempuh untuk Besarnya produk hortikultura khususnya buah segar menunju*. 38–52.

Rahaju, J., Pertanian, F., Wisnuwardhana, U., Pertanian, F., & Wisnuwardhana, U. (2014). Dampak Perubahan Iklim Terhadap Usaha Apel Di Kecamatan Poncokusumo Kabupaten Malang. *Agromix*, *5*(1), 1–9. https://doi.org/10.35891/agx.v5i1.697

Rinaldy, R., Christianti, R. F., & Supriyadi, D. (2013). Pengendalian Motor Servo Yang Terintegrasi Dengan Webcam Berbasis Internet Dan Arduino. *JURNAL INFOTEL - Informatika Telekomunikasi Elektronika*, *5*(2), 17. https://doi.org/10.20895/infotel.v5i2.4

Rozaq, I. A., & DS, Y. N. (2017). Uji Karakterisasi Sensor Suhu Ds18B20 Waterproof Berbasis Arduino Uno Sebagai Salah Satu Parameter Kualitas Air. *Prosiding SNATIF Ke-4*, 303–309.

Satriyo Cahya Rachmanda, & Aprilianto, T. (2023). Rancang Bangun Alat Pemberi Pakan Ikan Konsumsi Otomatis Berbasis Mikrokontroler. *Jurnal Sistem Komputer Asia*, *1*(01), 82–94. https://doi.org/10.32815/jiskomsia.v1i01.33

Septian, O. :, & Aji, P. (2018). *Alat Monitoring Tetesan Infus…(Septian Prastyo Aji) 78 INFUSING MONITORING TOOLS USING WEB ONLINE BASED ESP8266 WITH ARDUINO IDE PROGRAMMING*. *7*(1), 78–86.

Setyaningsih, E., Calvinus, Y., & Arifandi, L. (2023). Perancangan Sistem Database untuk Pengontrol Sensor Suhu dan Kelembapan Melalui Web untuk Ruangan Laboratorium LED. *Seminar Nasional Teknik Elektro (SNTE)*, *1*, 273–282.

Sumiati, E., & Santoso, B. (2023). Perancangan Alat Penyiraman Tanaman Krisan Otomatis Menggunakan Nodemcu Berbasis Internet of Things (IOT). *Jurnal Ilmu Komputer Dan Science*, *2*(2), 513–519.

Surapati, U., & Anwar, P. (2022). Implementasi Sistem Pemeliharaan Tanaman Hias Berbasis Internet of Things di Dira Plants Bumi Flora Semanan Jakarta Barat. *Jurnal Pendidikan Dan Konseling*, *4*(5), 834–851.

Suska, U. (2016). Sensor Soil. *UIN Suska*, *5*(1), 1689–1699.

Syukhron, I. (2021). Penggunaan Aplikasi Blynk untuk Sistem Monitoring dan Kontrol Jarak Jauh pada Sistem Kompos Pintar berbasis IoT. *Electrician*, *15*(1), 1–11. https://doi.org/10.23960/elc.v15n1.2158

Tullah, R., Sutarman, S., & Setyawan, A. H. (2019). Sistem Penyiraman Tanaman Otomatis Berbasis Mikrokontroler Arduino Uno Pada Toko Tanaman Hias Yopi. *Jurnal Sisfotek Global*, *9*(1). https://doi.org/10.38101/sisfotek.v9i1.219

Ummah, M. S. (2019). NodeMCU V3. *Sustainability (Switzerland)*, *11*(1), 1–14. http://scioteca.caf.com/bitstream/handle/123456789/1091/RED2017-Eng-8ene.pdf?sequence=12&isAllowed=y%0Ahttp://dx.doi.org/10.1016/j.regsciurbeco.2008.06.005%0Ahttps://www.researchgate.net/publication/305320484_SISTEM_PEMBETUNGAN_TERPUSAT_STRATEGI_MELESTARI

Wijaya, & Ridwan, (2019). Klasifikasi Jenis Buah Apel Dengan Metode K-Nearest Neighbors. *Jurnal SISFOKOM*, *8*(1), 74–78.

L A M P I R A N

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
Jl. G. Obos No.114 Telp.0536-3224593, 3225515 Fax.0536-3225515 Palangka Raya
email : humas@stmikplk.ac.id - website : www.stmikplk.ac.id

SURAT TUGAS

No.255/STMIK-3.C.2/KP/IX/2024

Ketua Program Studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya menugaskan nama- nama tersebut di bawah ini :

1. Nama : Elia Zakharia, M.T.
   NIK : 199205262016104
   Sebagai Pembimbing I Dalam Pembuatan Program
2. Nama : Rommi Kaestria, M.Kom.
   NIK : 198605242011103
   Sebagai Pembimbing II Dalam Penulisan Tugas Akhir

Untuk membimbing Tugas Akhir mahasiswa :

Nama	:	Yoga Anditia
NIM	:	C2155201003
Program Studi	:	TEKNIK INFORMATIKA (55201)
Tanggal Daftar	:	30 Agustus 2024
Judul Tugas Akhir	:	Prototype Sistem Penyiraman Tanaman Apel Berbasis Internet of Things (IoT)

Demikian surat ini dibuat agar dapat dipergunakan sebagaimana mestinya dan dilaksanakan dengan penuh tanggung jawab.

Palangka Raya, 10 September 2024
Ketua Program Studi Teknik Informatika,

Lili Rusdiana, M.Kom.
NIK. 198707282011007

Tembusan :
1. Pembimbing I dan II
2. Mahasiswa yang bersangkutan
3. Arsip

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
Jl. G. Obos No.114 Telp.0536-3224593, 3225515 Fax.0536-3225515 Palangkaraya
email : humas@stmikplk.ac.id - website : www.stmikplk.ac.id

KARTU KEGIATAN KONSULTASI
TUGAS AKHIR

Nama Mahasiswa : YOGA ANDITIA
NIM : C2155201003
Tanggal Persetujuan Judul : ...
Judul Tugas Akhir : Prototype Sistem Penyiraman Tanaman Aloe Berbasis Internet of Thing (IoT)

No.	Tanggal Konsultasi		Uraian	Tanda Tangan
	Terima	Kembali		
1	14/10/24	14/10/24	-Konsultasi Bab 1 Perbaikan Pada bagian latar belakang	
			-Perbaikan perumusan masayah	
2	07/11/24	07/10/24	- Konsultasi Bab 2 Melengkapi isi	
3	04/12/24	04/12/24	- konsultasi Bab 3 membuat desain prototype	
4	13/12/24	13/12/24	- Konsultasi Bab 3 Perbaikan desain prototype	
5	14/12/24	14/12/24	- Perbaikan desain Prototype - Acc Seminar proposal	
6	05/12/24	05/12/24	-perbaikan penulisan Bab I Tujuan. perbaiki Sesuai Catatan yang diberikan, Perbaikan manfaat Sesuai Catatan yang diberikan	
7	09/12/24	09/12/24	- perbaikan Penulisan Bab II Pada dasar teori Sensor Soil untuk kata kata yang tidak Sesuai dengan KBBI	
8	16/12/24	16/12/24	- perbaikan penulisan Bab III Sesuai Catatan perbaikan yang telah diberikan	
9.	17/12/24	17/12/24	Lengkapi Seluruh Naskah Lampiran.	
	17/12/24	17/12/24	ACC Seminar Proposal	

No.	Tanggal Konsultasi		Uraian	Tanda Tangan
	Terima	Kembali		
	02/05/25	02/05/25	Konsultasi Bab 4 perbaikan pada bagian Implementasi	
	09/05/25	09/05/25	Penambahan gambar ~~pemetaan~~ ~~pada~~ prototype	
	14/05/25	14/05/25	perbaikan pembahasan pada bagian source code	
	19/05/25	19/05/25	perbaikan penelitian dan perbaikan font halaman	
	24/05/25	24/05/25	ACC sidang	
		23/05/25	perbaikan seruai tujuan penulisan	
		24/05/25	Lengkapi semua tuliran + naskah	
		24/05	acc sidang	

Menyetujui :

Dosen Pembimbing I,

Elia Zakharia, M.T.

Dosen Pembimbing II,

Rommi Kaestria, M.KOM.

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
Jl. G. Obos No.114 Telp.0536-3224593, 3225515 Fax.0536-3225515 Palangka Raya
email : humas@stmikplk.ac.id - website : www.stmikplk.ac.id

SURAT TUGAS PENGUJI TUGAS AKHIR

No.117/STMIK-3.C.2/AU/V/2025

Ketua Program Studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya menugaskan kepada nama- nama berikut :

1. Nama : Sam'ani, S.T., M.Kom.
   NIK : 197703252005105
   Sebagai Ketua
2. Nama : Elok Faiqotul Himmah, S.Si., M.Sc
   NIK : 198503092009003
   Sebagai Sekretaris
3. Nama : Ir. Abdul Hadi, S.T., M.Kom
   NIK : 198505102021101
   Sebagai Anggota
4. Nama : Elia Zakharia, M.T
   NIK : 199205262016104
   Sebagai Anggota
5. Nama : Rommi Kaestria, M.Kom
   NIK : 198605242011103
   Sebagai Anggota

Untuk menguji Tugas Akhir mahasiswa :

Nama Mahasiswa	:	Yoga Anditia
N I M	:	C2155201003
Hari / Tanggal Ujian	:	Rabu, 28 Mei 2025
Waktu	:	11.00 WIB
Judul TA	:	Prototype Sistem Penyiraman Tanaman Apel Berbasis Internet of Things (IoT)
Ruang	:	Sidang 1

Demikian surat ini dibuat agar dapat dipergunakan sebagaimana mestinya dan dilaksanakan dengan penuh tanggung jawab.

Palangka Raya, 26 Mei 2025

Ketua Program Studi Teknik Informatika,



Ir. ABDUL HADI, S.T., M.Kom
NIK. 198505102021101

Tembusan :
1. Dosen Penguji
2. Mahasiswa yang Bersangkutan
3. Arsip

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKARAYA
Jl. G. Obos No.114 Telp.0536-3224593, 3225515 Fax.0536-3225515 Palangka Raya
email : humas@stmikplk.ac.id - website : www.stmikplk.ac.id

BERITA ACARA
UJIAN TUGAS AKHIR

Periode : Mei 2025

Pada hari ini Rabu, 28 Mei 2025 Pukul 11.00 WIB sampai dengan ..... WIB telah dilaksanakan Ujian Tugas Akhir mahasiswa atas nama :

Nama Mahasiswa	:	Yoga Anditia
N I M	:	C2155201003
Program Studi	:	Teknik Informatika
Tahun Angkatan	:	2021
Judul TA	:	Prototype Sistem Penyiraman Tanaman Apel Berbasis Internet of Things (IoT)

Dosen Penguji :

	Nama		Nilai	Tanda Tangan
1.	Sam'ani, S.T., M.Kom	=	.....	(.....)
2.	Elok Faiqotul Himmah, S.Si., M.Sc	=	.....	(.....)
3.	Ir. Abdul Hadi, S.T., M.Kom	=	.....	(.....)
4.	Elia Zakharia, M.T	=	.....	(.....)
5.	Rommi Kaestria, M.Kom	=	.....	(.....)

Hasil Ujian : **LULUS** / ~~**TIDAK LULUS**~~ *) NILAI = 85,92.

Revisi (~~Tidak Ada~~ / Minor / ~~Mayor~~ *)

Catatan Penting :
1. Lama Perbaikan : 14 hari
2. Jika lebih dari 1 (satu) bulan dikenakan sanksi berupa denda sebesar Rp. 600.000,- (Enam ratus ribu rupiah) per bulan dari tanggal ujian.
3. Jika lebih dari 3 (tiga) bulan dari tanggal ujian maka hasil ujian dibatalkan dan wajib mengajukan judul dan pembimbing baru.

Demikian Berita Acara ini dibuat, agar dapat digunakan sebagaimana semestinya.

Palangka Raya, 28 Mei 2025

Ketua Penguji,

Sam'ani, S.T., M.Kom
NIK. 197703252005105

Mahasiswa,

Yoga Anditia
NIM. C2155201003

Mengetahui :
Ketua Program Studi Teknik Informatika,

Ir. Abdul Hadi, S.T., M.Kom
NIK. 198505102021101

Tembusan:
1. Arsip Prodi Teknik Informatika
2. Mahasiswa yang bersangkutan

Pengujian Sistem Black Box pada Penelitian

Nama : Yoga Anditia

NIM : C2155201003

Judul : *Prototype* Sistem Penyiraman Tanaman Apel Berbasis *Internet Of Things (IoT)*

No	Pengujian	Hasil yang diharapkan	Hasil Pengujian
1.	Tes koneksi ke bord NodeMCU ESP8266	Port terdeteksi	Berhasil
2.	Tes sensor Soil Moisture	Dapat berfungsi dan mendeteksi kelembapan tanah secara *real-time*	Berhasil
3.	Tes sensor Temperature	Dapat berfungsi dan mendeteksi suhu secara *real-time*	Berhasil
4.	Tes sensor Humidity	Dapat berfungsi dan mendeteksi kelembapan udara secara *real-time*	Berhasil
5.	Uji pengiriman data dari sensor ke aplikasi pemantauan.	Data real-time ditampilkan dengan akurat di aplikasi.	Berhasil
6.	Uji fungsi manual untuk menyalakan atau mematikan pompa melalui aplikasi.	Pompa dapat dikendalikan secara manual melalui aplikasi blynk	Berhasil
7.	Uji fungsi manual untuk sistem buka tutup atap melalui aplikasi.	Atap dapat dikendalikan secara manual melalui aplikasi blynk	Berhasil
8.	Uji fungsi otomatis saat kelembapan tanah di bawah ambang batas yang telah ditentukan.	Pompa aktif untuk melakukan penyiraman.	Berhasil
9.	Uji fungsi otomatis sistem buka tutup saat suhu mencapai batas	Sistem dapat menutup atau membuka secara otomatis sesuai kebutuhan.	Berhasil

No	Pengujian	Hasil yang diharapkan	Hasil Pengujian
	yang telah ditentukan		
10.	Uji pengiriman data ke LCD	Dapat menampilkan data suhu, kelembapan tanah dan kelembapan udara secara *real-time*	Berhasil

Palangka Raya, 06 Mei 2025

Ir. Abdul Hadi, S.T., M.Kom.