

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Spesifikasi Sistem

Dalam pengembangan perangkat IoT, penulis menggunakan perangkat keras (*hardware*) dan perangkat lunak (*software*) sebagai sarana pemrograman dan pengujian sistem perangkat IoT. Salah satu perangkat yang digunakan adalah laptop pribadi penulis dengan sistem operasi *windows* sebagai perangkat utama penulis untuk mengerjakan penelitian ini. *Software* utama yang digunakan penulis untuk mengerjakan penelitian ini yaitu *Arduino IDE* untuk membuat sistem pemrograman perangkat IoT penelitian ini.

4.1.1 Spesifikasi Hardware Laptop Penulis

Laptop yang digunakan penulis adalah Asus V550VX dengan spesifikasi sebagai berikut;

- CPU : *Intel® Core™ i7-6700HQ* CPU
- GPU : *NVIDIA GTX 950M*
- RAM : 16 GB
- Storage: 638 GB HDD & 292 GB SSD
- OS : *Windows 11 Home*

4.1.2 Spesifikasi Hardware Arduino IDE

Penulis menggunakan *Arduino IDE* untuk melakukan pemrograman sistem perangkat IoT yang ada di dalam ESP32. Penggunaan *Arduino IDE* memang dikhususkan untuk membuat program papan mikrokontroler arduino. Bahasa pemrograman yang biasa dipakai pada *Arduino IDE* adalah bahasa C atau C++ serta memiliki *library* yang dapat memudahkan dalam pengembangan proyek IoT.

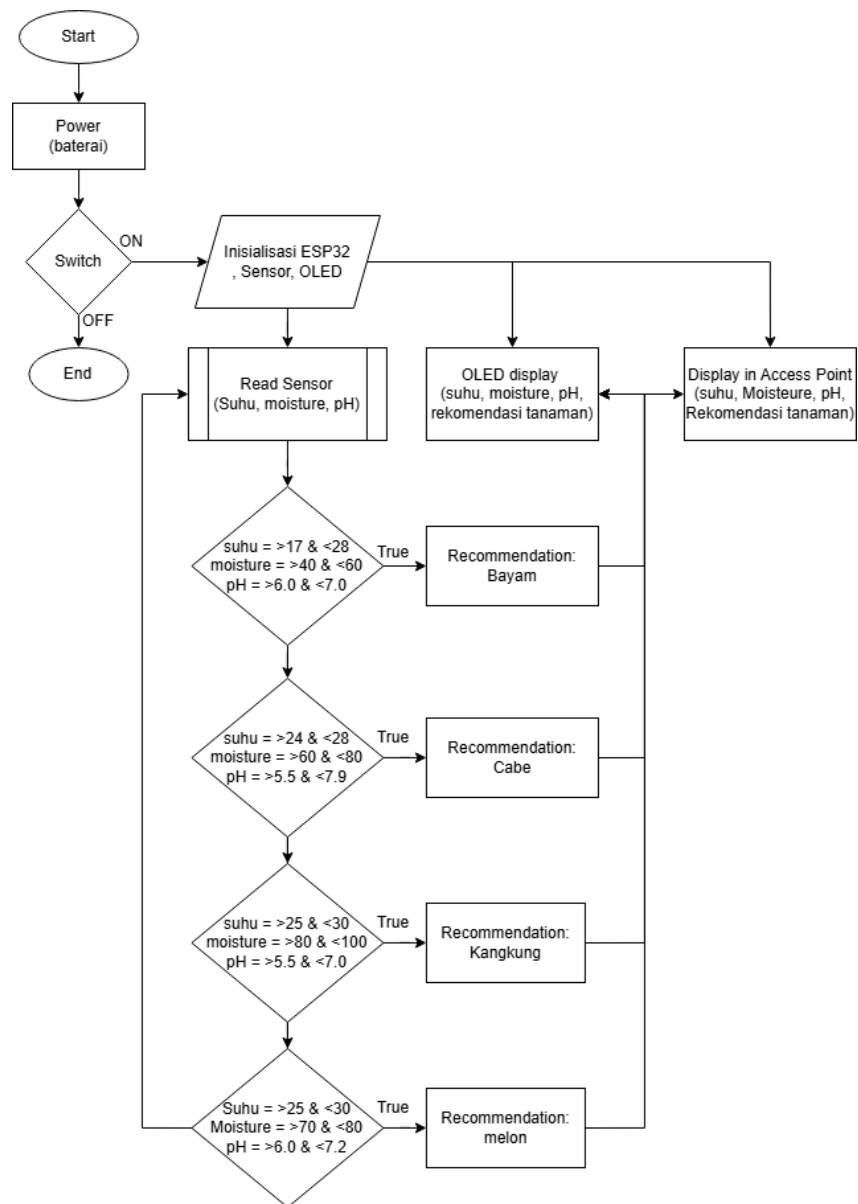
4.2 Implementasi Solusi

Pada bagian implementasi sistem ini menjelaskan tentang sistem yang ada di dalam perangkat IoT yang penulis buat, meliputi implementasi hardware, dan implementasi frontend.

4.2.1 Implementasi Hardware

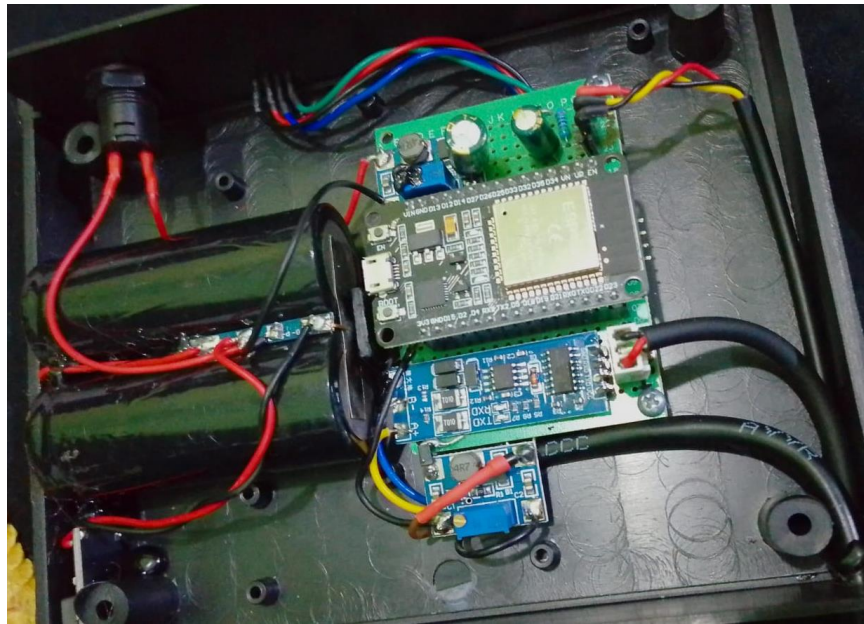
Implementasi hardware disini ialah penggunaan *software Arduino Integrated Development Environment (IDE)* untuk mengembangkan kode program yang akan dijalankan oleh mikrokontroler ESP32. Sebelum memulai pembuatan kode nya, penulis perlu mendeklarasikan library yang akan digunakan untuk memaksimalkan fungsionalitas hardware yang ada di perangkat IoT. *Library* pada *Arduino IDE* merupakan sekumpulan fungsi dan *variabel* yang sudah dibuat sebelumnya oleh pengembang maupun komunitas, sehingga memudahkan proses penulisan program. Dalam pengembangannya terdapat beberapa *library* yang digunakan oleh penulis di penelitian ini;

1. Adafruit GFX Library: menyediakan sintaks umum dan sekumpulan fungsi grafis untuk layer LCD
2. Adafruit BusIO: *library* ini punya fungsi untuk mengabstraksi interface UART, I2C, dan SPI.
3. Adafruit SSD1306: Mengatur komunikasi antara mikrokontroler (seperti Arduino) dan layar OLED.
4. ArduinoJson: Mengonversi struktur data C++ (seperti data sensor) menjadi string JSON yang dapat dikirim melalui jaringan atau port serial dan Memfasilitasi pertukaran data antara perangkat IoT, server web, API, dan layanan web.
5. DallasTemperature: *library* ini berfungsi untuk membaca nilai suhu yang diukur oleh sensor DS18B20 dalam satuan derajat celsius.
6. OneWire: *library* ini digunakan untuk mendapatkan fungsi yang lebih spesifik pada penggunaan sensor DS18B20 serta mengambil pembacaan suhu secara digital.



Gambar 4. 1 Flowchart Rancangan Sistem Perangkat

Pada gambar 4.1 flowchart rancangan sistem perangkat diatas menggambarkan alur program sistem yang dimulai dari *power supply* yang aktif dari *switch* atau saklar untuk menghidupkan perangkat IoT, lalu ESP32 akan memulai inisialisasi pin dari beberapa sensor seperti sensor suhu, kelembaban, dan sensor pH, dan perangkat lain seperti layar OLED dan *modul wifi*.



Gambar 4. 2 Rangkaian di dalam kotak Perangkat IoT

Setelah fungsi pin dari ESP32 menyala, dilanjutkan pada tiga sensor yang ada di gambar yaitu sensor suhu, kelembaban, dan pH yang mengirim data ke ESP32 yang dideteksi oleh sensor-sensor tersebut. Pada ESP32 perhitungan parameter untuk menentukan hasil analisa data yang diambil oleh sensor, penulis menggunakan teori logika IF ELSE yang sederhana yang nanti nya hasil perhitungan nya akan dikirim ke *webserver http* dan ke *output layer OLED*

4.2.2 Implementasi Software

Implementasi software di sini menjelaskan tentang kode pemrograman sistem yang ada pada perangkat IoT yang penulis rancang. Implementasi ini berfokus pada cara kerja sistem pemrograman pada perangkat IoT di *arduino IDE* dan pemrograman untuk *web server* nya.

```

3 // --- WiFi AP Hotspot ---
4 #include <WiFi.h>
5 const char* ap_ssid = "Plant Rekomendasi";
6 const char* ap_password = "";
7
8 // --- Web Server dan Web Html ---
9 #include <WebServer.h>
10 #include "index.h"
11 #include <ArduinoJson.h>
12
13 // --- Sensor Suhu dan Sensor Kelembaban ---
14 #include <OneWire.h>
15 #include <DallasTemperature.h>
16 const int oneWireBus = 32;
17 const int soilMoisturePin = 34;
18
19 // --- OLED Display ---
20 #include <Wire.h>
21 #include <Adafruit_GFX.h>
22 #include <Adafruit_SSD1306.h>
23 #define SCREEN_WIDTH 128
24 #define SCREEN_HEIGHT 64
25 #define OLED_RESET -1
26 Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
27
28 // --- Sensor Suhu & Port Web Server ---
29 OneWire oneWire(oneWireBus);
30 DallasTemperature sensors(&oneWire);
31 WebServer server(80);

```

Gambar 4. 3 Inisialisasi dan library program

Pemrograman pada *Arduino IDE* dimulai dengan menambahkan beberapa *library* untuk menghubungkan ESP32 dengan sensor, layar OLED, dan *server*. Seperti *library* *wifi.h* dipakai untuk membuat wifi access point, *webserver.h* dan *ArduinoJson.h* digunakan untuk menampilkan data sensor melalui web server sederhana, sedangkan *OneWire.h* dan *DallasTemperature.h* digunakan untuk membaca sensor suhu DS18B20, serta *library* *Adafruit_GFX.h* dan *Adafruit_SSD1306.h* digunakan untuk menampilkan data sensor ke layar OLED *display*.

Pada sensor kelembaban, sensor tersebut menghasilkan data analog yang perlu dikalibrasikan agar bisa ditampilkan kepada user dalam bentuk persentase. Karena itu diperlukan penilaian ADC (Analog-to-Digital Converter) dari sensor kelembaban yang dipetakan menjadi persen (%). Dengan mengikuti acuan tersebut, kondisi tanah kering (0%)

ditetapkan dengan nilai 2600 ADC, sedangkan untuk kondisi tanah basah(100%) ditetapkan dengan nilai 1300 ADC. Dengan pemetaan ini, sistem dapat memberikan informasi kelembaban tanah yang lebih representatif.

```

33 // --- Kalibrasi Sensor Kelembaban pakai nilai raw analog ---
34 int KondisiKering = 2600;
35 int KondisiBasah = 1300;
36
37 // --- RS485 PH Sensor ---
38 const byte ph_soil_reg[8] = {0x01, 0x03, 0x00, 0x00, 0x00, 0x01, 0x84, 0x0A};
39 #define RESPONSE_LENGTH 7
40
41 uint16_t calculateCRC(byte *array, uint8_t length) {
42     uint16_t crc = 0xFFFF;
43     for (uint8_t i = 0; i < length; i++) {
44         crc ^= array[i];
45         for (uint8_t j = 0; j < 8; j++) {
46             if ((crc & 0x0001) != 0) {
47                 crc >>= 1;
48                 crc ^= 0xA001;
49             } else {
50                 crc >>= 1;
51             }
52         }
53     }
54     return crc;
55 }

```

Gambar 4. 4 program kaliuresis sensor kelembaban dan sensor pH RS485

Penggunaan sensor pH diperlukan komunikasi melalui protokol RS485 menggunakan Modbus RTU. Data hasil pembacaan diverifikasikan dengan CRC16 agar komunikasi valid. Raw value yang diterima sensor akan diolah dan dibagi 10 untuk mendapatkan nilai pH yang aktual dengan kondisi tanah.

```

58 // --- Timer buat ambil data serta tampilkan ke OLED ---
59 unsigned long previousLogMillis = 0;
60 unsigned long previousDisplayMillis = 0;
61 const long displayInterval = 3000; // 3 detik
62
63 // --- Nilai Sensor saat ini ---
64 float currentTemperatureC = 0.0;
65 int currentMoisturePercent = 0;
66 float soil_ph = 0.0;

```

Gambar 4. 5 timer dan tampilan data sensor untuk OLED display

OLED digunakan sebagai media yang memvisualisasikan data langsung di perangkat kepada user. Data yang ditampilkan meliputi nama tanaman yang direkomendasikan, suhu tanah dalam derajat celcius, kelembaban tanah dalam persen, serta nilai pH tanah.



Gambar 4. 6 Output OLED

Data sensor yang dibaca akan diperbarui setiap 3 detik OLED. Output OLED ditunjukkan seperti pada gambar diatas.

```
// --- Wi-Fi Access Point ---
WiFi.softAP(ap_ssid, ap_password);
IPAddress myIP = WiFi.softAPIP();
Serial.print("Masuk mode AP Hotspot. IP Address: ");
Serial.println(myIP);

display.clearDisplay();
display.setTextColor(SSD1306_WHITE);
display.setCursor(0, 0);
display.setTextSize(2);
display.println("WiFi AP:");
display.setTextSize(1);
display.println(ap_ssid);
display.println(ap_password);
display.display();
delay(2000);

// --- Web Server ---
server.on("/", HTTP_GET, handleRoot);
server.on("/data", HTTP_GET, handleData);

server.begin();
Serial.println("Web Server dijalankan.");
```

Gambar 4. 7 program web server wifi access point

Pada pemrograman untuk web servernya, penulis membuat *wifi access point* yang menggunakan SSID “*Plant Recommendation*” yang dimana user dapat terhubung ke *web server plant recommendation* tanpa koneksi internet. Dalam pemrograman tersebut memiliki dua *endpoint* utama yaitu (“/”) untuk menampilkan halaman web dari (index.h), dan (“/data”) untuk memberikan data JSON yang berupa data sensor suhu, kelembaban, pH, dan rekomendasi tanaman dalam *format* JSON.

Tabel 4. 1 Logika sederhana pemrograman rekomendasi tanaman

no	Rekomendasi Tanaman	Suhu (°C)	Kelembaban (%)	pH
1	Bayam	17 – 28	40 – 60	6.0 – 7.0
2	Cabe	24 – 28	60 – 80	5.5 – 7.0
3	Kangkung	25 – 30	80 – 100	5.5 – 7.0
4	Melon	25 - 30	70 – 80	6.0 – 7.2
5	XXX	-	-	-

Untuk logika pemrograman tanaman menggunakan fungsi if-else sederhana untuk menentukan jenis tanaman yang cocok berdasarkan perbandingan nilai *range ideal* parameter suhu, kelembaban dan pH tanah dari empat rekomendasi tanaman pilihan penulis. Dan apabila pengukuran sensor menunjukan tidak ada parameter yang sesuai dari empat tanaman tersebut maka pada jenis tanaman akan menampilkan keterangan “x x x” sebagai penanda bahwa lahan tidak memenuhi kriteria parameter empat tanaman yang diuji. Contoh *output* nya seperti yang ditunjukkan pada gambar 4.5.

Secara *flow* kerja program perangkat, sistem menginisialisasi OLED, sensor-sensor, *hotspot wifi*, dan *web server*. *Server* akan menangani request user pada (*handleclient*) sekaligus memperbarui data sensor setiap tiga detik *interval loop*.


```
// --- Buat Web Html ---
void handleRoot() {
  server.send(200, "text/html", WEB_HTML);
}

void handleData() {
  bacaSensor();
  String rekomendasiTanaman = cekTanaman(currentTemperatureC, currentMoisturePercent, soil_ph);
  StaticJsonDocument<200> doc; // Allocate a fixed-size JSON document (adjust size if needed)
  doc["temperature"] = currentTemperatureC;
  doc["moisture"] = currentMoisturePercent;
  doc["ph"] = soil_ph;
  doc["plant"] = rekomendasiTanaman;

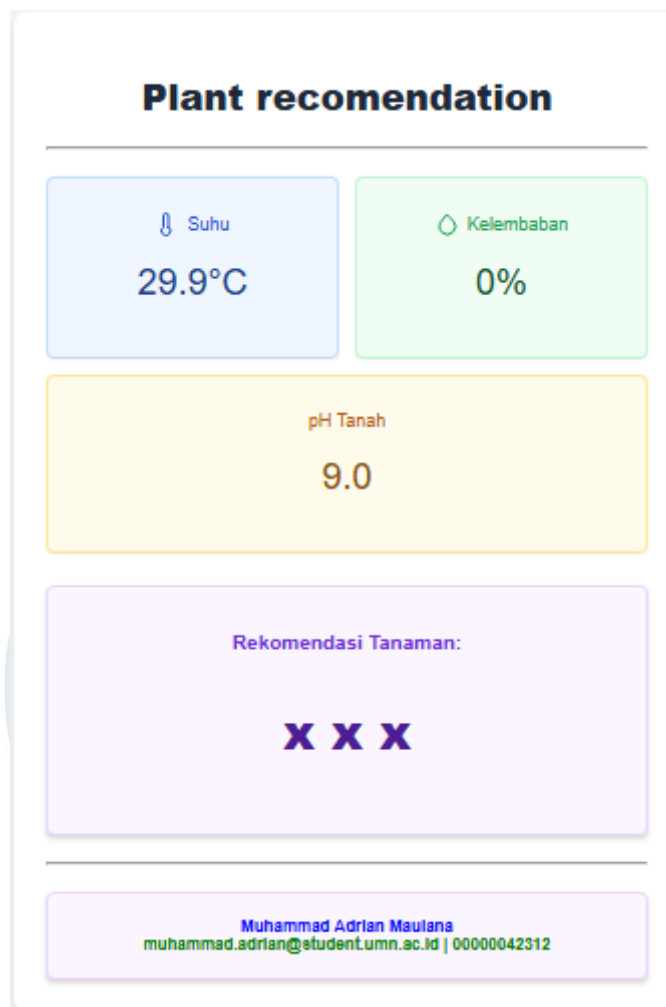
  String jsonString;
  serializeJson(doc, jsonString); // Convert JSON document to string

  server.send(200, "application/json", jsonString);
}
```

Gambar 4. 8 program implementasi web server

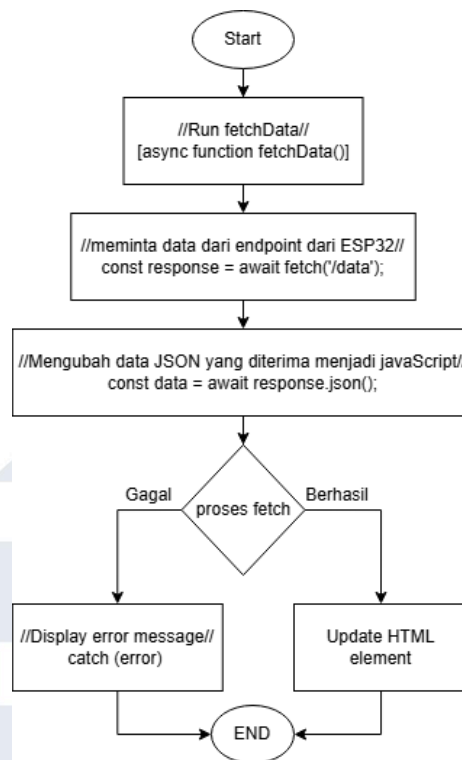
Bagian program ini merepresentasikan komunikasi antara perangkat IoT ESP32 dengan user melalui *web server* berbasis HTTP. Dalam gambar diatas, terdapat fungsi `handleRoot()` yang berguna untuk menampilkan *interface user* berbasis HTML, sedangkan fungsi `handleData()` untuk menyajikan data sensor dalam *format* JSON yang dapat digunakan untuk integrasi atau analisa lebih lanjut.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 4. 9 interface website

Untuk tampilan websitenya sendiri terkesan sederhana yang memang didesain sedemikian rupa agar user utama pada penelitian ini yaitu petani berfokus pada parameter yang ditunjukkan oleh perangkat IoT. Terdapat empat parameter utama yaitu suhu, kelembaban, pH tanah dan rekomendasi tanaman.



Gambar 4. 10 Diagram flow website local Access point

Pada program HTML disini menunjukkan integrasi komunikasi dengan fungsi fetchData() yang digunakan untuk mengambil data terbaru dari alamat /data yang disediakan oleh web server di ESP32. Data tersebut dikirim dalam format JSON berisi HTML element berupa parameter value sensor dan rekomendasi tanaman. Setelah data diterima maka website akan melakukan refresh data secara otomatis berupa HTML element parameter masing-masing. Mekanisme ini looping secara berkala dalam interval setiap 5 detik.

4.3 Pengujian dan Analisis Solusi

Pengujian yang dilakukan penulis untuk menguji perangkat IoT terbagi menjadi tiga tahap, *testing* awal atau pengujian fungsionalitas, pengujian akurasi, dan *User Acceptance Test* (UAT).

4.3.1 Pengujian Fungsionalitas

Untuk pengujian dan analisis pertama dari perangkat IoT yang dibuat, penulis melakukan pengujian awal di beberapa titik lahan tanah

di sekitar tempat observasi pertanian di perumahan dinas korem 052 yang belum ditanam banyak tanaman. Percobaan awal di test dengan memasukan ketiga sensor kedalam lahan tanah dan mengaktifkan perangkat IoT dengan menekan *switch* yang ada pada perangkat. Setelah perangkat diaktifkan, penulis menganalisa parameter yang dihasilkan oleh sensor pada layar OLED dan mengecek apakah website dapat bekerja dengan optimal.



Gambar 4. 11 percobaan pertama memasukan sensor ke tanah

Pada percobaan pertama di lahan yang ada di gambar di diatas, parameter perangkat menganalisa bahwa tanah di sekitar lahan perumahan dinas korem 052 memiliki suhu tanah rata – rata sekitar 27,9 celcius, kelembaban 78%, dan pH tanah 6.0. Dengan parameter itu, rekomendasi tanaman yang disarankan untuk ditanam pada tanah di sekitar lahan perumahan dinas korem menurut perangkat IoT adalah tanaman cabai yang ditunjukan pada gambar di bawah ini.



Gambar 4. 12 hasil percobaan pertama perangkat IoT pada lahan tanah perumahan dinas korem 052

Dari percobaan ini dapat disimpulkan bahwa secara fungsional dari segi sensor hingga informasi yang diberikan alat ini dapat bekerja dengan cukup baik.

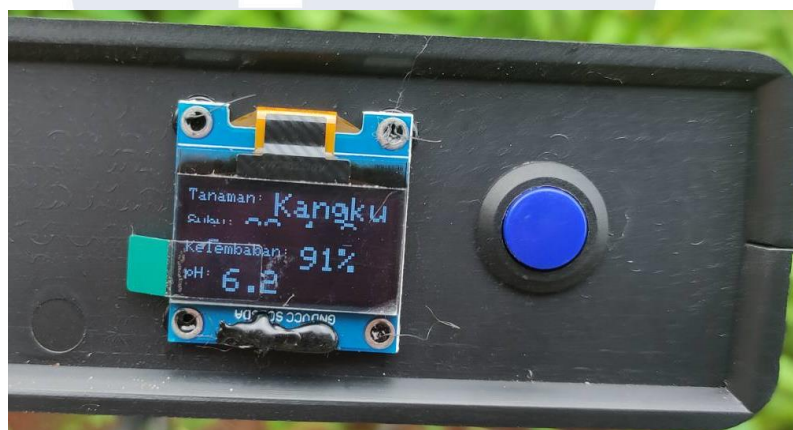
4.3.2 Pengujian Akurasi

Pada percobaan kedua, yaitu pengujian akurasi, penulis mencoba melakukan pengujian perangkat IoT di BRMP (Badan Perakitan dan Modernisasi Pertanian) untuk menganalisa keakuratan perangkat IoT dengan membandingkan keakuratan parameter perangkat IoT yang di buat oleh penulis dengan parameter di alat yang dimiliki BRMP.



Gambar 4. 13 percobaan perbandingan akurasi perangkat pada lahan kangkung di BRMP.

Percobaan dilakukan pada lahan kangkung yang ada di BRMP dengan kondisi tanaman yang belum mencapai masa panen. Pada pengujian tersebut, perangkat menangkap analisa tanah dengan rata-rata parameter suhu tanah 29.1 C, kelembaban 91 %, dan pH tanah antara 6.0 hingga 6.2 dengan rekomendasi tanaman yang ditampilkan adalah tanaman kangkung. Di sisi lain perangkat milik BRMP berfokus pada parameter pH, suhu, dan NPK di dalam tanah. oleh karena itu titik acuan yang dapat di jadikan point objek perbandingan dengan perangkat IoT milik penulis adalah pada parameter pH nya. Karena unsur pH merupakan unsur yang paling krusial dalam menentukan tanah yang bagus bagi tanaman. Pada perangkat miliki BRMP menunjukkan pH tanah pada rata-rata di 5.8 hingga 6.4 dengan suhu yang serupa.



Gambar 4. 14 hasil percobaan akurasi perangkat milik penulis di lahan kangkung BRMP

Hasil perbandingan pengukuran parameter pH tanah masing-masing perangkat menunjukkan tingkat akurasi yang tinggi. Perangkat yang dikembangkan penulis mendeteksi pH pada rentang 6.0 – 6.2 (rata-rata 6.1), sedangkan alat pembanding (BRMP) menunjukkan rentang 5.8 – 6.4 (rata-rata 6.1). Karena kedua alat memiliki nilai rata-rata yang identik, dan rentang pembacaan alat penulis berada sepenuhnya di dalam rentang pembacaan alat BRMP yang termasuk dalam toleransi alat standar, maka dapat disimpulkan bahwa perangkat IoT milik penulis

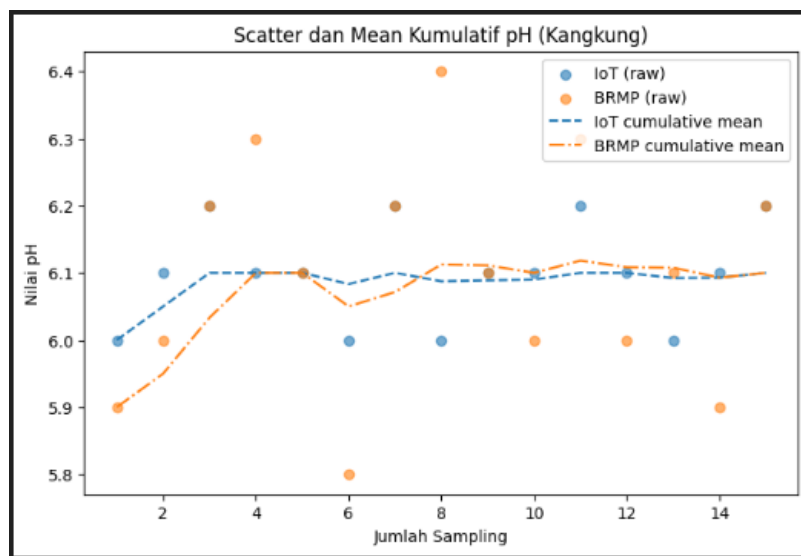
valid dan akurat dalam mendeteksi pH tanah dan merekomendasi jenis tanaman dengan tepat di lokasi pengujian.

Tabel 4. 2 Perbandingan rata-rata pH pada lahan BRMP

Lahan	Jumlah Sampling	Mean pH Device	Mean pH BRMP Device
Kangkung	5	6.1	6.1
	10	6.9	6.1
	15	6.1	6.1
Cabai	5	6.3	6.3
	10	6.3	6.2
	15	6.4	6.2
Melon	5	5.7	6.5
	10	6.1	6.5
	15	6.3	6.5

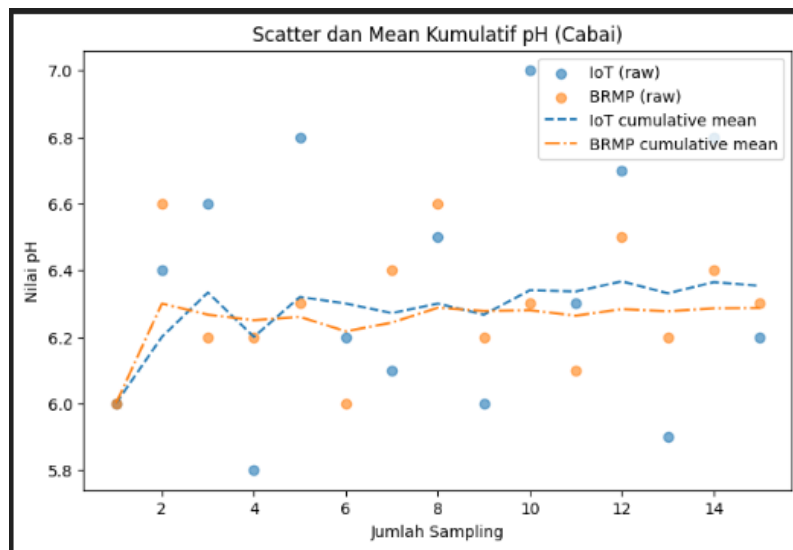
untuk mengevaluasi konsistensi dan keandalan pembacaan, penulis melakukan perbandingan lanjutan dengan melakukan multi sampling pengukuran parameter rata-rata pH tanah pada tiga lahan tanaman yang ada di BRMP. Jenis tanaman yang dipilih mengambil 3 dari 4 rekomendasi tanaman yang ada di dalam perangkat IoT milik penulis. Pengujian dilakukan dengan *multi sampling*, yaitu mengambil nilai rata-rata dari tiga titik jumlah sampling berbeda, yaitu 5 *sampling*, 10 *sampling*, dan 15 *sampling* di lahan yang sama. Walaupun terdapat batasan dalam pengujian sampling di lahan kangkung dan cabai. Dimana dari pihak BRMP hanya memperbolehkan melakukan *sampling* pada sisi samping lahan dan tidak memperbolehkan melakukan *sampling* di bagian tengah lahan, dikarenakan pihak BRMP tidak ingin lahan yang sudah dikelola oleh petani BRMP menjadi rusak.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



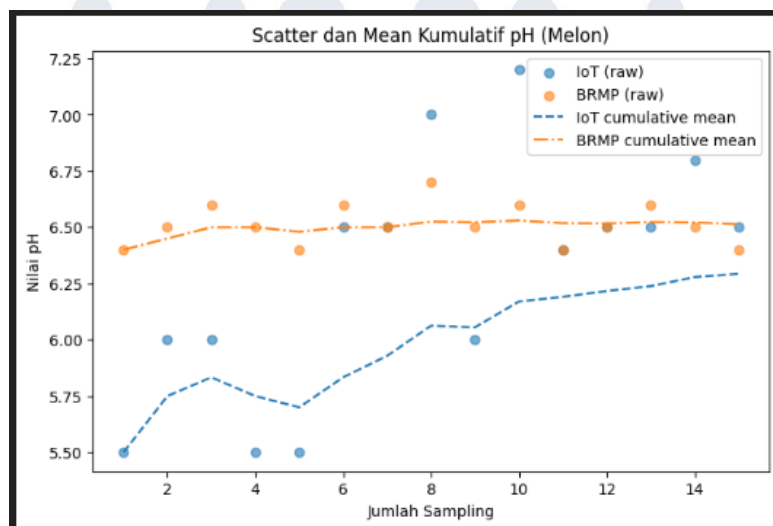
Gambar 4. 15 Grafik sebaran sampel pH di lahan kangkung

Pada pengujian tersebut didapatkan sebuah hasil analisis yang tertera pada tabel 4.2 yang menunjukkan perhitungan rata-rata tiga jumlah sampling dengan perbandingan nilai rata-rata alat milik BRMP. Pada lahan tanaman kangkung, perangkat IoT menunjukkan nilai rata-rata pH yang cukup konsisten dengan nilai pH yang stabil walaupun terdapat deviasi di pengujian *sampling* ke sepuluh. Secara rata-rata keseluruhan menunjukkan bahwa perangkat IoT mampu mendeteksi pH tanah di lahan tanaman kangkung secara konsisten mendekati nilai yang diperoleh alat BRMP, yang stabil di 6.1 untuk semua jumlah *sampling*.



Gambar 4. 16 Grafik sebaran sampel pH di lahan cabai

Pada lahan cabai, kedua perangkat menunjukkan pola yang konsisten serta akurat. *Mean* pH IoT berada pada 6.3–6.4, sedangkan alat BRMP berada pada 6.2–6.3. Perbedaan antar perangkat sangat kecil (± 0.1), yang berada dalam batas toleransi pengukuran pH tanah. Hal ini menunjukkan bahwa perangkat IoT bekerja sangat baik dalam mendeteksi pH lahan tanah cabai, dengan memiliki fluktuasi pH kecil antar titik sampling.



Gambar 4. 17 Grafik sebaran sampel pH di lahan melon

Pada lahan melon, variasi pH tanah lebih bervariasi dibandingkan kangkung dan cabai. Pada sampling 5 titik tanah, perangkat IoT menghasilkan rata-rata 5.7, lebih rendah dibanding rata-rata alat BRMP di 6.5, mencerminkan sensitivitas terhadap nilai-nilai ekstrem 5.5 yang dominan di data sampling awal. Namun, ketika jumlah sampling ditingkatkan menjadi 10 dan 15 titik, nilai rata-rata IoT naik menuju nilai stabil di 6.1–6.3, mendekati rata-rata alat BRMP yaitu 6.5. Ini menunjukkan bahwa semakin besar sampling, semakin akurat dan representatif pembacaan IoT. Perbedaan kecil yang tersisa disebabkan oleh rentang variasi pH melon yang lebih luas (5.5 – 7.2), sehingga memerlukan lebih banyak titik sampling untuk mencapai kestabilan.

Kesimpulan dari pengujian *multi sampling* ini mendapat hasil yaitu pengujian pada lahan kangkung dan cabai memiliki distribusi pH yang stabil antar titik sampling, pembacaan perangkat IoT yang diberikan cukup mendekati dengan pengukuran alat BRMP dengan perbedaan rata-rata pH tanah hanya 0.1. Perbedaan rata-rata pH tanah yang terlihat jelas pada lahan melon, yang memiliki variasi pH yang lebih bervariasi. Namun setelah jumlah sampling diperluas menjadi 10-15 titik sampling, hasil pH dari perangkat IoT mendekati parameter pH yang dihasilkan alat BRMP dan menunjukkan stabilitas yang lebih baik.

Pengujian ini memperlihatkan bahwa pengujian menggunakan perangkat IoT di lahan pertanian lebih ideal dilakukan dengan 10-15 titik *sampling* yang menunjukkan semakin banyak titik sampling semakin baik akurasi parameter pengukuran nya. Dengan selisih rata-rata yang sangat kecil antara perangkat IoT dan alat milik BRMP di hampir semua kondisi, perangkat IoT terbukti valid, dapat diandalkan, dan mampu menjadi alternatif alat uji tanah konvensional.

4.3.3 User Acceptance Testing (UAT)

User Acceptance Testing digunakan sebagai tahap terakhir pengembangan perangkat IoT yang berguna menguji dan memastikan

bahwa sistem perangkat IoT yang dibuat penulis sesuai dengan kebutuhan user. Proses UAT yang dilakukan oleh penulis melibatkan user yang berupa para petani yang berada di BPP (Balai Penyuluhan Pertanian) di caringin kecamatan Legok, kabupaten Tangerang. Para petani akan menguji perangkat IoT baik dari segi fungsionalitas, kegunaan, kinerja, dan kemudahan penggunaan perangkat IoT.



Gambar 4. 18 percobaan UAT ke petani di daerah pertanian BPP caringin bersama petani

Di lapangan, penulis memberikan pengertian singkat ke petani akan maksud, tujuan dan manfaat yang ingin dicapai dengan penggunaan perangkat IoT pada pertanian. Setelah mencoba menggunakan perangkat IoT, penulis akan memberikan questioner berisikan 9 pertanyaan singkat tentang kepuasan para petani dalam menggunakan perangkat IoT untuk mengetahui parameter unsur hara tanah dan rekomendasi tanaman yang diberikan oleh perangkat IoT.

Tabel 4. 3 Tabel data kuesioner UAT petani di BPP caringin.

no	Pertanyaan	Nilai		
		Setuju	Netral	Tidak Setuju
F1	Perangkat ini berhasil mendeteksi dan menampilkan data, suhu, kelembaban, dan pH dengan jelas.	20		
F2	Informasi rekomendasi tanaman yang diberikan oleh perangkat ini cukup detail dan relevan dalam membuat keputusan penanaman.	15	4	1

U1	Perangkat ini mudah digunakan dan informasi yang ditampilkan dari perangkat ini jelas dan mudah dipahami.	17	3	
U2	Saya tidak perlu menghabiskan banyak waktu untuk memahami cara menggunakan perangkat ini.	20		
R1	Data suhu, kelembaban dan pH tanah yang ditampilkan konsisten dalam jangka waktu tertentu.	20		
R2	Perangkat ini berfungsi Dengan baik dan rekomendasi tanaman yang diberikan dapat dipercaya keakuratannya.	18	2	
E1	Penggunaan perangkat ini dapat menghemat waktu saya dan membantu dalam pengambilan keputusan penanaman menjadi lebih cepat.	20		
E2	Saya merasa bisa mendapat manfaat dari penggunaan perangkat ini	19	1	
A1	Saya berniat mempertimbangkan untuk menggunakan alat ini dalam aktivitas pertanian saya	14	5	1

Berdasarkan hasil UAT di lapangan terhadap 20 responden petani yang dari umur 28 hingga 60 tahun yang ada di sekitar BPP caringin menunjukkan antusiasme terhadap perangkat IoT yang di coba pada di lahan mereka. Dari total 180 jawaban dari Sembilan pertanyaan, secara total sebanyak 90,56% respon menyatakan setuju, 8,33% bersikap netral, dan hanya 1,11% yang tidak setuju terhadap implementasi IoT pada pertanian dalam menentukan parameter unsur hara tanah dan rekomendasi tanaman. Sebagian besar indikator seperti fungsionalitas perangkat mendeteksi data tanah (F1, U2, R1, E1) memperoleh tingkat persetujuan 100%, menunjukkan bahwa petani merasa alat mudah digunakan, hasilnya mudah dipahami, serta dapat bermanfaat dalam mendukung keputusan pertanian.

Walaupun begitu, terdapat dua aspek yang mencolok pada pertanyaan (F2) dan (A1) yang menunjukkan ada respons tingkat persetujuan petani lebih rendah (75% dan 70%), yang dikarenakan para petani tersebut lebih memilih cara tradisional untuk menentukan relevansi penanaman yang tepat dibandingkan menggunakan pendekatan yang lebih modern. Meskipun begitu, keduanya tetap menunjukkan kecenderungan positif yang kuat. Secara keseluruhan, data ini

membuktikan bahwa teknologi IoT pada lahan pertanian diterima dengan baik oleh petani, dan memiliki potensi untuk diadopsi sebagai alat pendukung penentuan kualitas tanah dan rekomendasi tanaman yang lebih tepat.



UMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA