



Design of an Intelligent Monitoring System Based on Internet of Things (IoT) with Random Forest Regression Algorithm for Height Detection in Cherry Tomato Plants

Perancangan Sistem *Monitoring* Cerdas Berbasis *Internet of Things* (IoT) dengan Algoritma *Random Forest Regression* untuk Deteksi Ketinggian pada Tanaman Tomat Cherry

Vito Hafizh Cahaya Putra¹, Muhammad Al-Husaini²,
Ari Purno Wahyu³, Agung Rachmat Raharja⁴

¹Program Studi Informatika, Fakultas Teknologi Kreatif, Universitas Satu, Indonesia

²Program Studi Teknik Informatika, Fakultas Teknik, Universitas Siliwangi, Indonesia

³Program Studi Teknik Informatika, Fakultas Teknik, Universitas Widyatama, Indonesia

⁴Program Studi Teknik Informatika, Fakultas Kesehatan dan Teknik, Universitas Bandung, Indonesia

E-mail: vito.putra@univ.satu.ac.id¹, alhusaini@unsil.ac.id²,
ari.purno@widyatama.ac.id³, agungmaat@bandunguniversity.ac.id⁴

Received Aug 10th 2024; Revised Sep 17th 2024; Accepted Oct 10th 2024; Available Online Nov 24th 2024

Corresponding Author: Vito Hafizh Cahaya Putra

Copyright © 2025 by Authors, Published by Institut Riset dan Publikasi Indonesia (IRPI)

Abstract

Cherry tomatoes are a valuable commodity in Indonesia, with demand increasing every year. This study develops an intelligent IoT-based monitoring system for cherry tomato plants using Random Forest Regression (RFR) algorithms. The system utilizes an ESP32 microcontroller and five sensors to monitor environmental parameters, along with actuators to regulate optimal conditions. Sensor data is processed and stored on the Thingspeak platform and integrated with Google Colab for plant height prediction. The prediction results are displayed on an LCD screen and sent as notifications via the Telegram application. This research fills the gap from previous studies by integrating various sensors, actuators, and cloud platforms into a comprehensive system. System evaluation shows a Mean Squared Error (MSE) of 0.8294 and an R^2 Score of 0.8939, and Black Box testing ensures optimal functionality in various scenarios. The results of this study can provide benefits in the application of IoT and machine learning technologies for the monitoring and management of cherry tomato plants, with the hope of increasing agricultural efficiency and productivity.

Keywords: Cherry Tomato, Intelligent Monitoring System, Internet of Things, Random Forest Regression

Abstrak

Tomat cherry merupakan komoditas bernilai di Indonesia dengan permintaan yang meningkat setiap tahunnya. Penelitian ini mengembangkan sistem pemantauan cerdas berbasis *Internet of Things* (IoT) untuk tanaman tomat cherry menggunakan algoritma *Random Forest Regression* (RFR). Sistem ini memanfaatkan mikrokontroler ESP32 dan lima sensor untuk memantau parameter lingkungan, serta aktuator untuk pengaturan kondisi optimal. Data sensor diproses dan disimpan di platform Thingspeak dan diintegrasikan dengan Google Colab untuk prediksi ketinggian tanaman. Hasil prediksi ditampilkan di layar LCD dan dikirimkan sebagai notifikasi melalui aplikasi Telegram. Penelitian ini mengisi kesenjangan dari studi sebelumnya dengan mengintegrasikan berbagai sensor, aktuator, dan platform *cloud* dalam satu sistem yang komprehensif. Evaluasi sistem menunjukkan nilai *Mean Squared Error* (MSE) sebesar 0.8294 dan R^2 Score sebesar 0.8939, serta hasil pengujian *Black Box Testing* memastikan fungsionalitas optimal dalam berbagai skenario. Hasil penelitian ini dapat memberikan manfaat dalam penerapan teknologi IoT dan *machine learning* untuk *monitoring* dan pengelolaan tanaman tomat cherry, harapannya meningkatkan efisiensi dan produktivitas pertanian.

Kata Kunci: *Internet of Things*, *Random Forest Regression*, *Sistem Monitoring Cerdas*, Tomat Cherry

1. PENDAHULUAN

Tomat cherry adalah komoditas pertanian yang bernilai di Indonesia dengan permintaan yang meningkat setiap tahunnya. Data Badan Pusat Statistik menunjukkan permintaan tomat cherry naik dari 1.020.333 ton pada 2019 menjadi 1.114.399 ton pada 2021, kenaikan sekitar 94.066 ton, dan diperkirakan terus meningkat. Tomat cherry memiliki keunggulan nutrisi, seperti meningkatkan imunitas, mengelola tekanan darah, dan memperbaiki kesehatan kulit [1], yang menarik minat banyak petani untuk menanamnya.

Namun, budidaya tomat cherry memiliki tantangan, seperti penggunaan pot yang terlalu kecil, penyiraman yang tidak tepat, kepadatan tanaman, kurang sinar matahari, suhu rendah, pemupukan yang salah, dan penopangan yang terlambat [2]. Petani tentunya tidak diberikan informasi prediksi terkait pertumbuhan tanamannya, berdasarkan parameter lingkungan yang terdapat pada tanaman tomat cherry. Pertumbuhan tanaman merupakan aspek krusial yang harus diperhatikan dalam budidaya tomat cherry, mengingat setiap bagian tanaman saling berinteraksi secara sinergis dalam proses pertumbuhannya. Evaluasi pertumbuhan tanaman umumnya dilakukan dengan mengamati parameter seperti jumlah daun, diameter batang, dan tinggi tanaman. Dalam budidaya tomat, pertumbuhan tanaman sangat penting, terutama karena setiap bagian dari tanaman saling berinteraksi secara sinergis selama proses pertumbuhan. Namun, menjaga kondisi pertumbuhan ideal untuk menanam tomat cherry menghadapi tantangan yang signifikan. Saat ini, petani harus mengukur secara manual jumlah daun, diameter batang, dan tinggi tanaman [3]. Adanya faktor internal dan eksternal mempengaruhi pertumbuhan tanaman tomat. Faktor internal seperti hormon dan gen menentukan potensi pertumbuhan dan produktivitas tanaman. Namun, petani sering kali kesulitan untuk memantau faktor eksternal yaitu cahaya, suhu, kelembapan, nutrisi, dan pH tanah secara efektif [4]. Metabolisme tanaman sering terganggu oleh cahaya yang tidak cukup, suhu yang tidak sesuai, atau kelembapan yang tidak konsisten, yang menghambat pertumbuhan. Nutrisi yang dikelola dengan buruk dan pH tanah yang rendah berdampak negatif pada kemampuan tanaman untuk menyerap nutrisi yang diperlukan untuk pertumbuhan. Tomat cherry memiliki permintaan dipasaran yang terus bertambah dan nilai jual yang tinggi, sehingga diperlukannya kebutuhan mendesak untuk solusi berbasis teknologi yang dapat membantu petani memantau dan mengelola faktor-faktor lingkungan tersebut secara lebih efektif dan efisien pada tanaman tomat cherry.

Di era Industri 4.0 [5], teknologi seperti kecerdasan buatan, komputasi awan, pencetakan 3D, robotika, big data, RFID, dan *Internet of Things* (IoT) semakin terintegrasi dan berkembang pesat. IoT, yang terdiri dari "*Internet*" sebagai jaringan komputer dan "*Things*" sebagai objek fisik yang terhubung melalui sensor, memiliki pengaruh signifikan di era ini. IoT [6] memungkinkan objek mengirimkan data melalui jaringan internet tanpa bantuan komputer dan manusia, menjadikan objek tampak "hidup" dengan data yang diproses menjadi informasi atau tindakan. Arsitektur IoT terdiri dari 7 lapisan: *Physical Devices & Controller, Connectivity, Edge Computing, Data Accumulation, Data Abstraction, Application, dan Collaboration & Processes* [7]. IoT telah dimanfaatkan di berbagai bidang, seperti kesehatan, transportasi, lingkungan hingga agrikultur, di mana IoT membawa perubahan signifikan dalam konsep smart agrikultur. Teknologi IoT dalam bidang agrikultur [8] memungkinkan pemantauan real-time terhadap kondisi tanaman, tanah, dan lingkungan melalui sensor yang terhubung, serta analisis data untuk pengambilan keputusan yang lebih baik. Pemanfaatan IoT dalam pertanian meningkatkan efisiensi penggunaan air, mengurangi penggunaan pupuk dan pestisida, serta meningkatkan hasil panen. Teknologi *Internet of Things* mendukung praktik pertanian yang lebih ramah lingkungan [9].

Berbagai peneliti telah melakukan penelitian yang mendalam tentang penerapan IoT dalam pemantauan tanaman. Sebagian besar penelitian di sektor pertanian mengenai IoT terutama berfokus pada pemantauan parameter lingkungan seperti suhu, kelembapan, kelembapan tanah, dan parameter terkait lainnya, seperti yang dijelaskan dalam tinjauan pustaka pada bagian studi literatur. Masih sedikit penelitian yang mengeksplorasi hubungan antara monitoring menggunakan IoT dan penggunaan algoritma machine learning, dalam konteks memprediksi pertumbuhan tanaman. Contohnya terdapat penelitian yang menggunakan machine learning menggunakan teknologi IoT, namun penelitian tersebut memiliki fokus untuk memprediksi tren suhu lingkungannya, bukan melakukan prediksi mengenai pertumbuhan tanaman tomat cherry.

Maka diperlukannya penelitian yang fokus pada memprediksi pertumbuhan tanaman tomat cherry berdasarkan pengukuran tinggi tanaman tersebut. Tujuan dari penelitian ini adalah untuk memprediksi tinggi pertumbuhan tanaman tomat cherry menggunakan sistem cerdas yang berbasis IoT dan algoritma *Random Forest Regression*. Mikrokontroler ESP32 digunakan dalam sistem ini bersama dengan lima sensor (DHT11, kelembapan tanah, probe pH tanah, LDR, dan ultrasonik) serta aktuatur. Ini berkaitan dengan servo, lampu LED ungu, kipas angin, dan layar LCD. Data yang dikumpulkan oleh sensor-sensor ini dianalisis menggunakan algoritma *Random Forest Regression* yang ditulis dalam Python. Google Colab berfungsi sebagai platform untuk memprediksi pertumbuhan tanaman tomat cherry. Sistem ini telah dikonfigurasi untuk terhubung dengan platform cloud *Thingspeak* dan aplikasi Telegram untuk memberikan notifikasi terkait pH tanaman yang berada tidak diambang batas aman kepada pengguna. Diharapkannya, hasil dari

penelitian ini dapat memberikan solusi inovatif bagi petani tomat cherry dalam meningkatkan produktivitas dan kualitas hasil panen.

2. STUDI LITERATUR

Dibawah ini, terdapat berbagai penelitian yang telah dilakukan yang berkaitan dengan pengembangan sistem *monitoring* cerdas serta penggunaan algoritma prediksi pada tanaman tomat cherry:

Penelitian [10] ini bertujuan untuk merancang dan menerapkan sistem pemantauan berbasis *Internet of Things* (IoT) untuk pertanian terlindung seperti tanaman tomat cherry, yang biasanya digunakan dalam rumah kaca komersial. Digunakannya berbagai sensor untuk mengumpulkan data lingkungan seperti suhu, kelembaban, intensitas cahaya, dan konsentrasi CO₂. Adanya metode prediksi berbasis jaringan saraf tiruan (ANN) digunakan untuk memprediksi tren suhu dengan akurasi dalam 24 jam ke depan dengan kesalahan hanya ± 1 °C.

Terdapat penelitian yang dilakukan oleh [11] membandingkan produksi tomat yang ditanam dengan sistem IoT dan metode konvensional di sebuah rumah kaca. Hasil penelitian menunjukkan bahwa produktivitas tomat yang terintegrasi dengan sistem IoT meningkat sekitar 19% dibandingkan dengan tanaman konvensional. Selain itu, persentase tomat *Grade A-B* meningkat sebesar 9,8% sementara *off-grade* menurun sebesar 9,6%. Sensor yang digunakan yaitu Soil Moisture dan Soil pH untuk memantau lingkungan dari kelembaban dan pH tanah. Adanya ESP32 untuk melakukan pengiriman data melalui protokol MQTT. *Relay* digunakan untuk menyalakan aktuator.

Pengembangan sistem yang dilakukan oleh [12] bertujuan untuk membuat penyiraman otomatis berbasis IoT untuk tanaman tomat, yang dapat dikendalikan dari jarak jauh melalui koneksi internet. Sistem ini dirancang untuk memastikan kelembaban tanah dan suhu udara yang optimal bagi pertumbuhan tomat, yang mana kelembaban tanah harus berada di antara 60% hingga 80% dan suhu udara antara 24 hingga 28 derajat Celsius. Digunakannya DHT11 untuk deteksi suhu, *relay* untuk menyalakan *water pump*, serta ESP8266 sebagai pusat kendali atau mikrokontroler yang memiliki kemampuan pengiriman data menggunakan WiFi.

Penelitian yang dilakukan oleh [13] menggunakan berbagai teknik pembelajaran mesin menggunakan Algoritma *Random Forest Regression* untuk memprediksi tinggi tanaman dan menemukan bahwa algoritma dapat memprediksi tinggi tanaman dengan koefisien determinasi (*R-squared*) berkisar antara 0,31 hingga 0,87 untuk musim tanam penuh. Ketika difokuskan pada fase-fase fenologis utama, *R-squared* berkisar dari 0,2 hingga 0,83.

Penelitian ini berfokus untuk [14] mengevaluasi metode *Machine Learning* (ML) dan *Deep Learning* (DL) untuk memprediksi hari menuju kematangan, tinggi tanaman, dan hasil biji pada kedelai menggunakan data multispektral dari UAV. Eksperimen dilakukan pada 524 genotipe kedelai selama dua musim tanam, dengan model DL dan *Random Forest* (RF) menunjukkan prediksi tinggi tanaman yang kuat ($r > 0,77$). Model DL unggul dalam memprediksi DM dengan $r = 0,66$. Studi ini menunjukkan pendekatan efisien untuk prediksi variabel tanaman kedelai berbasis data penginderaan jauh.

Dari berbagai penelitian tersebut telah menggunakan berbagai sensor dan aktuator dalam membangun sistem untuk monitoring parameter-parameter seperti suhu dan kelembaban udara, kelembaban tanah, pH tanah, hingga sensor cahaya, serta penggunaan aplikasi telegram untuk pemberian notifikasi, baik untuk tanaman tomat cherry yang skala kecil maupun yang besar seperti kebun cerdas. Namun, penelitian sebelumnya menunjukkan kesenjangan atau digunakan terpisah dalam penggunaan berbagai sensor beserta aktuatornya dan *platform cloud* tersebut yang belum terintegrasi dalam satu sistem. Penelitian untuk memprediksi tinggi tanaman tersebut menggunakan algoritma *Random Forest Regression*, tidak menggunakan teknologi IoT beserta berbagai perangkat fisiknya seperti sensor, aktuator, modul komunikasi, dan komponen tambahan lainnya, serta tidak adanya platform untuk menyimpan data hasil deteksi sensor seperti *Thingspeak* yang dapat diintegrasikan dengan google colabs untuk dilakukan prediksi dari ketinggian tanamannya dengan memanfaatkan data dari *cloud platform* tersebut.

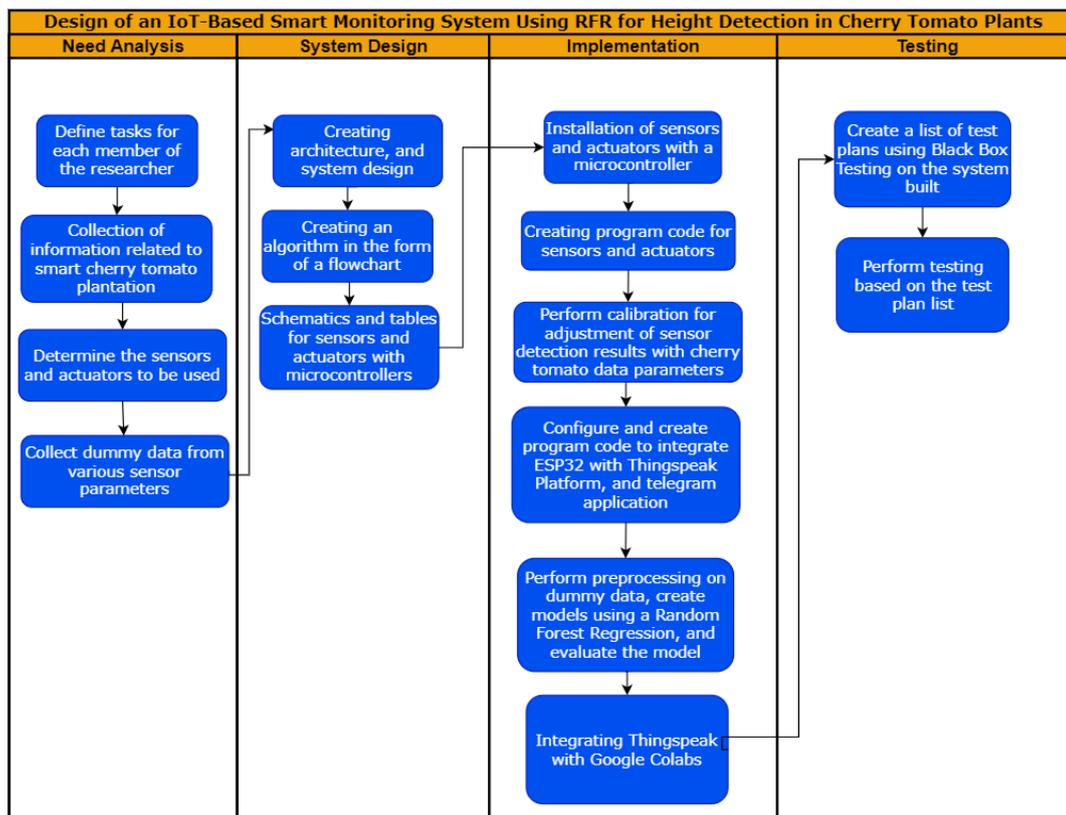
Berdasarkan gap dari berbagai penelitian tersebut, Penelitian ini bertujuan untuk mengembangkan sistem monitoring cerdas berbasis IoT untuk tanaman tomat cherry dengan menggunakan algoritma *Random Forest Regression* untuk mendeteksi ketinggian tanaman. Sistem ini mengintegrasikan mikrokontroler ESP32 sebagai unit pemrosesan utama yang juga berfungsi untuk pengendalian sistem dan pengiriman data melalui jaringan Wi-Fi menggunakan protokol HTTP. Beberapa sensor digunakan untuk mendeteksi parameter lingkungan yang berbeda, termasuk DHT22 untuk suhu dan kelembaban, *Soil Moisture* untuk kelembaban tanah, pH Probe untuk tingkat keasaman, dan *Light Dependent Resistor* (LDR) untuk intensitas cahaya. Aktuator yang diterapkan dalam sistem ini meliputi layar *Liquid Crystal Display* (LCD), pompa air, kipas angin, dan lampu *Ultraviolet*. Data yang diperoleh dari sensor ditampilkan dalam bentuk grafik *monitoring* melalui *platform Thingspeak*, serta notifikasi dikirimkan melalui aplikasi Telegram apabila tingkat pH tanah berada di luar batas yang diinginkan. Algoritma *Random Forest Regression* digunakan untuk memprediksi ketinggian tanaman dengan melakukan integrasi antara Google Colab dan *platform Thingspeak*. Hasil prediksi ini kemudian dapat ditampilkan pada layar LCD. Penelitian ini menghadapi keterbatasan terkait

perangkat keras, seperti sensor, aktuator, dan modul komunikasi, serta berbagai komponen lainnya. Oleh karena itu, digunakan aplikasi simulasi WokWi yang menyediakan representasi simulasi perangkat keras dengan fungsionalitas yang sesuai. Temuan dari penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam bidang pertanian cerdas, khususnya dalam penerapan teknologi IoT dan *machine learning* untuk *monitoring* dan pengelolaan tanaman tomat cherry.

3. METODE DAN KAJIAN PUSTAKA

3.1. Metode Penelitian Pengembangan Sistem

Terdapat metode penelitian yang terdiri dari berbagai tahapan agar penelitian yang dilakukan memiliki hasil yang baik, sebagai gambar 1.

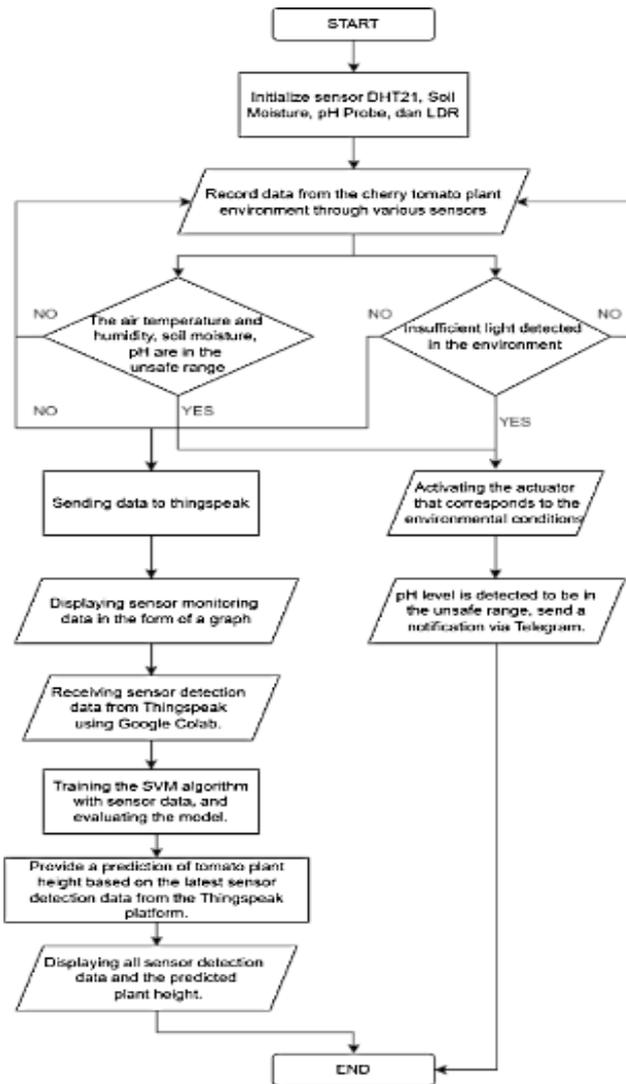


Gambar 1. Metode Penelitian Pengembangan Sistem

Gambar 1 menggambarkan alur kerja sistematis dalam perancangan sistem monitoring cerdas berbasis IoT dengan algoritma *Random Forest Regression* untuk deteksi ketinggian tomat cherry. Penelitian dibagi menjadi empat tahap utama: Analisis Kebutuhan, Desain Sistem, Implementasi, dan Pengujian. Pada tahap Analisis Kebutuhan, langkah-langkahnya mencakup penentuan tugas untuk setiap anggota tim, pengumpulan informasi terkait perkebunan tomat cherry cerdas, pemilihan sensor dan aktuator yang sesuai, serta pengumpulan data *dummy* untuk setiap parameter sensor. Tahap Desain Sistem melibatkan pembuatan arsitektur sistem, desain kebun, dan algoritma dalam bentuk *flowchart*, serta pembuatan skematik dan tabel untuk unit sensor dan aktuator yang terhubung dengan mikrokontroler. Implementasi mencakup pemasangan sensor dan aktuator pada mikrokontroler, kalibrasi hasil deteksi sensor, konfigurasi dan integrasi program dengan platform IoT seperti ESP32 dan *ThingSpeak*, serta aplikasi Telegram. Data yang diperoleh diproses menggunakan model *Random Forest Regression* untuk deteksi ketinggian tomat cherry. Tahap Pengujian mencakup pembuatan rencana pengujian menggunakan metode *Black Box Testing* dan pelaksanaan pengujian berdasarkan rencana tersebut untuk memastikan sistem berfungsi sesuai spesifikasi.

3.2. Alur Kerja Sistem

Sistem ini diawali dengan inisialisasi berbagai sensor, termasuk DHT21, Soil Moisture, pH Probe, dan LDR, untuk mengukur kondisi lingkungan tanaman. Data dari lingkungan tanaman tomat cherry kemudian direkam melalui sensor-sensor tersebut. Apabila suhu udara, kelembaban, kelembaban tanah, atau pH berada di luar rentang yang aman, data tersebut dikirimkan ke platform cloud *Thingspeak* untuk dipantau dan ditampilkan dalam bentuk grafik. Jika terdeteksi kekurangan cahaya, sistem akan mengaktifkan aktuator yang sesuai untuk menyesuaikan kondisi lingkungan.



Gambar 2. Alur Kerja Sistem

Jika pH terdeteksi berada di luar rentang yang aman, sistem akan mengirimkan notifikasi melalui Telegram. Data sensor kemudian diterima dari *Thingspeak* menggunakan Google Colab, di mana algoritma *Random Forest Regression* dilatih dengan data sensor ini untuk mengevaluasi model. Model *Random Forest Regression* ini kemudian digunakan untuk memprediksi ketinggian tanaman tomat cherry berdasarkan data sensor terbaru dari platform *Thingspeak*. Seluruh data deteksi sensor dan prediksi tinggi tanaman yang ditampilkan melalui layar LCD, memungkinkan petani untuk memantau kondisi dan pertumbuhan tanaman secara real-time, sehingga meningkatkan efisiensi dan produktivitas dalam budidaya tomat cherry. Alur kerja system dapat dilihat pada gambar 2.

3.2. Kajian

3.3.1 Internet of Things (IoT)

Internet of Things adalah teknologi yang memungkinkan perangkat elektronik dan sensor untuk berkomunikasi melalui internet guna meningkatkan kenyamanan hidup [15]. Teknologi ini menyediakan solusi cerdas untuk berbagai masalah di dunia bisnis, pemerintahan, serta sektor publik dan swasta di seluruh dunia. IoT semakin penting dan dapat dirasakan dalam kehidupan sehari-hari. Pada dasarnya, *Internet of Things* (IoT) adalah sebuah inovasi yang menggabungkan berbagai perangkat pintar dan sensor ke dalam sebuah sistem yang cerdas. Selain itu, IoT memanfaatkan teknologi kuantum dan nanoteknologi untuk mencapai tingkat penyimpanan, pemantauan, dan kecepatan pemrosesan yang sebelumnya tidak mungkin dicapai. Banyak studi dan laporan ilmiah, baik yang *online* maupun cetak, menjelaskan potensi transformasi *Internet of Things*. Penelitian ini menjadi dasar untuk mengembangkan rencana bisnis yang inovatif. Cisco mendefinisikan arsitektur *Internet of Things* yang terdiri dari tujuh lapisan [16][17], yaitu:

1. *Physical Devices & Controller*
Perangkat fisik dan kontrol mencakup sensor dan aktuator yang secara langsung menangkap informasi dari lingkungan. Pada lapisan ini, data pertama-tama diproses oleh pengendali sebelum diteruskan ke lapisan berikutnya.
2. *Connectivity*
Konektivitas mengacu pada hubungan perangkat IoT yang menggunakan protokol komunikasi seperti Wi-Fi, Bluetooth, Zigbee, atau LoRa untuk mentransmisikan informasi baik ke cloud maupun ke server lainnya.
3. *Edge Computing*
Dalam komputasi tepi, pemrosesan data dilakukan dekat dengan sumber data, yang mengarah pada pengurangan latensi dan memungkinkan pengambilan keputusan secara real-time.
4. *Data Accumulation*
Selama akumulasi data, informasi yang dikumpulkan disusun dan disimpan sementara pada tingkat ini untuk diproses kemudian.
5. *Data Abstraction*
Tingkat abstraksi data berkaitan dengan pemrosesan, pembersihan, dan penyaringan data yang dikumpulkan. Ini memastikan bahwa hanya data yang sesuai dan berkualitas tinggi yang digunakan dalam aplikasi atau analisis di masa depan.
6. *Application*
Kota pintar, rumah pintar, dan banyak solusi IoT seperti kesehatan digital diimplementasikan dengan data yang diproses oleh lapisan aplikasi.
7. *Collaboration & Processes*
Tingkat kolaborasi dan proses memfasilitasi interaksi antara perangkat, data, dan aplikasi sambil juga mendorong otomatisasi alur kerja dan integrasi sistem IoT.

3.5.1 *Random Forest Regression*

Random Forest Regression adalah [18] algoritma ensemble yang digunakan untuk prediksi nilai kontinu dengan cara membangun beberapa pohon keputusan (*Decision Trees*) dan menggabungkan hasil prediksinya. Algoritma ini bekerja dengan teknik *Bootstrap Aggregating* atau *Bagging*, yang melibatkan pembuatan banyak pohon keputusan independen dari berbagai subset data yang diambil secara acak (*sampling* dengan penggantian). Prediksi akhir adalah rata-rata dari semua prediksi tree secara individu. Algoritma ini dirumuskan sebagai [19] berikut:

1. **Pembentukan Tree**
Random Forest membangun beberapa pohon keputusan dengan bootstrap sampling (pengambilan sampel dengan penggantian) dari data latih. Setiap pohon menggunakan subset acak dari fitur, yang membantu mengurangi overfitting dan meningkatkan generalisasi.
2. **Prediksi**
Setelah membangun banyak pohon keputusan, *Random Forest* menggabungkan hasil prediksi dari semua pohon untuk mendapatkan prediksi akhir. Untuk regresi, hasil prediksi rata-rata dari semua pohon digunakan sebagai hasil final. Prediksi akhir y adalah rata-rata dari prediksi semua pohon keputusan [33].

$$\hat{y} = \frac{1}{B} \sum_{i=1}^B T_i(x) \quad (1)$$

Penjelasan notasi: B adalah jumlah total pohon hutan; $T_i(x)$ adalah prediksi pohon ke- i ; x adalah input data; dan \hat{y} adalah prediksi rata-rata atau agregasi dari seluruh pohon.

4. **HASIL DAN PEMBAHASAN**

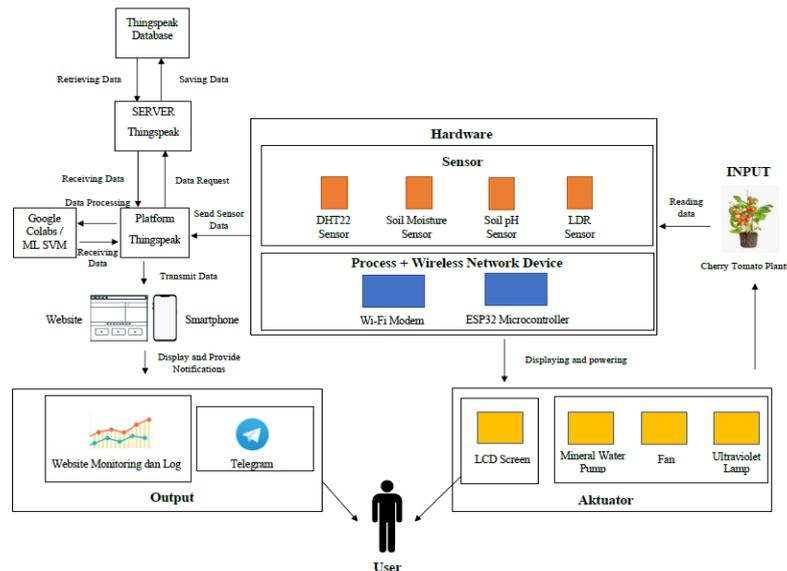
4.1 **Analisis Kebutuhan**

Analisis kebutuhan merupakan langkah esensial dalam pengembangan sistem yang bertujuan untuk merinci berbagai informasi yang dibutuhkan. Pada tahap ini, ditetapkan tugas setiap anggota tim peneliti, mencakup pengembangan perangkat keras seperti mikrokontroler, sensor dan aktuator, integrasi *platform cloud ThingSpeak* dan aplikasi Telegram dengan mikrokontroler, serta pengembangan model prediktif

menggunakan algoritma *Random Forest Regression* untuk estimasi tinggi tanaman tomat cherry juga dilakukan pada tahap ini.

Informasi terkait monitoring cerdas untuk tanaman tomat cherri cerdas berbasis IoT dan prediksi tinggi tanaman dikumpulkan melalui sumber akademis seperti jurnal, buku, dan internet. Informasi ini meliputi pemilihan mikrokontroler, sensor, aktuator, modul komunikasi, serta perangkat keras lainnya yang diperlukan, termasuk platform cloud dan aplikasi pendukung lainnya. Data *dummy* yang diperoleh dari hasil deteksi sensor untuk parameter seperti suhu, kelembaban udara, kelembaban tanah, pH tanah, intensitas cahaya, dan tinggi tanaman dikumpulkan menggunakan teknologi AI generatif. Data ini kemudian digunakan sebagai data training dalam proses machine learning menggunakan algoritma RFR, yang bertujuan untuk menghasilkan model prediktif yang akurat.

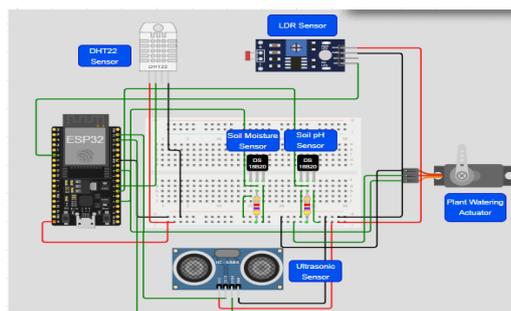
4.2. Arsitektur Sistem



Gambar 3. Arsitektur Sistem

Arsitektur sistem pada gambar 3 menggambarkan sebuah sistem monitoring cerdas pada tanaman tomat cherry yang menggunakan algoritma *Random Forest Regression* untuk memprediksi ketinggian tanaman. Sistem ini terdiri dari beberapa komponen utama: sensor, perangkat pemrosesan dan jaringan, server, platform analitik, serta antarmuka pengguna. Sensor-sensor seperti DHT21, sensor kelembaban tanah, sensor pH tanah, dan LDR bertugas membaca berbagai parameter lingkungan tanaman dan mengirimkan data tersebut ke mikrokontroler ESP32 melalui jaringan Wi-Fi. Data yang dikumpulkan kemudian dikirim ke platform *Thingspeak* [20] untuk penyimpanan dan analisis lebih lanjut. Pada tahap ini, algoritma *Random Forest Regression* yang dijalankan di Google Colabs memproses data untuk memprediksi ketinggian tanaman. Hasil analisis dapat diakses oleh pengguna melalui situs web monitoring dan aplikasi Telegram, yang juga menyediakan notifikasi. Selain itu, sistem ini mampu mengaktifkan aktuator seperti layar LCD, pompa air mineral, kipas angin, dan lampu ultraviolet berdasarkan kondisi tanaman untuk memastikan pertumbuhan optimal.

4.3. Skematik Sensor dengan Mikrokontroler



Gambar 4. Skematik Sensor dengan Mikrokontroler

Berdasarkan skematik pada gambar 4 menunjukkan konfigurasi sistem monitoring berbasis *Internet of Things* (IoT) untuk budidaya tanaman tomat cherry, yang memanfaatkan mikrokontroler ESP32 sebagai unit pengendali utama. Sistem ini mencakup berbagai sensor, yaitu DHT22 untuk mengukur suhu dan kelembaban udara, LDR untuk mengukur intensitas cahaya, sensor kelembaban tanah untuk mengukur kadar air tanah, sensor pH tanah untuk memantau tingkat keasaman tanah, serta sensor ultrasonic untuk mendeteksi tinggi dari tanaman tomat cherrynya. Dikarenakan sensor yang terdapat pada simulator Wokwi jenisnya terbatas, maka akan dilakukan kalibrasi pada sensor Soil Moisture dan Soil pH. Berikut ini merupakan proses penulisan kode program agar berbagai sensor tersebut dapat melakukan pendeteksian lingkungannya:

1. Sensor DHT22 untuk mendeteksi Suhu dan Kelembaban Udara

Sensor ini akan mendeteksi suhu dan kelembaban udara dilingkungan sekitar tanaman tomat cherrynya. Kelembaban udara yang baik harus berada di 60 - 85 % [21], karena kelembaban yang tinggi menjadi faktor penghambat pertumbuhan tanaman karena banyak hama dan penyakit. Namun, jika kelembapan rendah, maka proses berbuah menjadi lebih sulit [22]. Kode program sensor DHT22 yang digunakan dalam mendeteksi suhu sebagai berikut:

```
float h = dht.readHumidity();
float t = dht.readTemperature();

if (isnan(h) || isnan(t)) {
  Serial.println("Failed to read from DHT sensor!");
  bot.sendMessage(CHAT_ID, "Failed to read from DHT sensor!", "");
  return;
}
Serial.print("Humidity: ");
Serial.print(h);
Serial.print(" %\t");
Serial.print("Temperature: ");
Serial.print(t);
Serial.println(" *C");
```

2. Sensor LDR untuk mendeteksi Cahaya

Sensor LDR akan mendeteksi kondisi cahaya dilingkungan sekitarnya, dalam hal ini untuk mendeteksi apakah ada cahaya matahari atau tidak. Apabila kondisi cahaya telah redup, dalam artian sudah mendekati ke malam hari, maka akan dinyalakan lampu LED. Berikut kode program sensor LDR yang digunakan:

```
int ldrValue = analogRead(LDR_PIN);
Serial.print("LDR value: ");
Serial.println(ldrValue);
```

3. Sensor Kelembaban Tanah (*Soil Moisture*)

Sensor ini bertujuan untuk mendeteksi kelembaban tanah, berdasarkan kondisi air didalam tanah pada tanaman tomat cherry tersebut. Dikarenakan tidak adanya sensor *Soil Moisture* di platform Wokwi, maka dilakukan penyesuaian agar sensor yang bernama DS18B20 menjadi sensor *Soil Moisture* dengan dilakukan kalibrasi, agar memiliki hasil pendeteksian sensor *Soil Moisture*. Kode program yang digunakan pada sensor *Soil Moisture* sebagai berikut:

```
sensors1.requestTemperatures();
float rawTemp1 = sensors1.getTempCByIndex(0);
float soilMoisture = map(rawTemp1, -10, 85, 0, 100);
Serial.print("Soil moisture: ");
Serial.print(soilMoisture);
Serial.println(" %");
```

4. Sensor pH Tanah (*Soil pH*)

Sensor ini digunakan untuk mendeteksi pH dari tanah, dengan range pendeteksian 0 – 14. Hal yang sama dilakukan pada sensor pH Tanah karena tidak adanya sensor ini di platform Wokwi, yang memerlukan proses kalibrasi menggunakan sensor yang bernama DS18B20. Kode program yang digunakan pada sensor *Soil pH* sebagai berikut:

```
sensors2.requestTemperatures();
float rawTemp2 = sensors2.getTempCByIndex(0);
float soilPH = map(rawTemp2, -10, 85, 3, 10);
Serial.print("Soil pH: ");
Serial.println(soilPH);
```

5. Sensor Ultrasonic

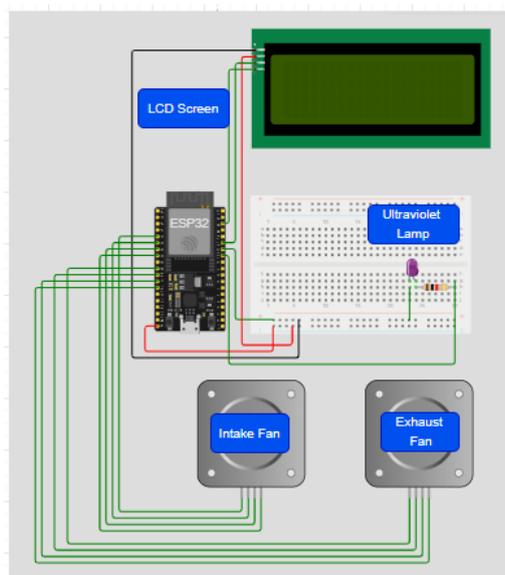
Sensor digunakan untuk mendeteksi tinggi dari tanaman tomat cherry sebagai parameter untuk melakukan monitoring pertumbuhan dari tanaman tomat cherry. Kode program yang digunakan pada sensor ini yaitu:

```
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);

long duration = pulseIn(ECHO_PIN, HIGH);
float distance = duration * 0.034 / 2;
Serial.print("Tomato plant height: ");
Serial.print(distance);
Serial.println(" cm");
```

Data yang dikumpulkan dari sensor-sensor ini diproses oleh mikrokontroler ESP32, yang kemudian mengatur aktuator penyiraman tanaman secara otomatis. Data hasil deteksi sensor yang telah terkumpul dikirimkan ke platform cloud Thingspeak menggunakan jaringan Wi-Fi (*Wireless Fidelity*) dengan protokol HTTP (*Hypertext Transfer Protocol*) [23] untuk dianalisis lebih lanjut.

4.4. Skematik Aktuator dengan Mikrokontroler



Gambar 5. Skematik Aktuator dengan Mikrokontroler

Skematik pada gambar 5, menjelaskan mengenai sistem aktuator yang terintegrasi dengan mikrokontroler ESP32 untuk pengaturan kondisi lingkungan tanaman tomat cherry. Mikrokontroler ESP32 diintegrasikan dengan cloud platform thingspeak untuk mengambil data hasil deteksi sensor dan ditampilkan pada layar LCD. Mikrokontroler ESP32 mengendalikan berbagai aktuator, seperti lampu *ultraviolet* yang menyediakan pencahayaan tambahan pada saat cahaya diruangan telah redup terutama di malam hari. Adanya cahaya matahari [24] sangat dibutuhkan oleh tanaman tomat cherry, setidaknya tanaman tomat cherry membutuhkan cahaya matahari selama 6 - 8 jam perhari, yang berguna untuk melakukan fotosintesis. Terdapat dua kipas angin (masuk dan keluar) yang mengatur sirkulasi udara. Suhu ideal [25] memiliki pengaruh yang baik pada warna buah Tomat yaitu berwarna merah merata, dengan suhu 21-27 °C. Semua komponen ini beroperasi secara otomatis berdasarkan data sensor yang diterima dan diproses oleh ESP32, memastikan bahwa kondisi lingkungan tanaman selalu berada dalam kisaran optimal.

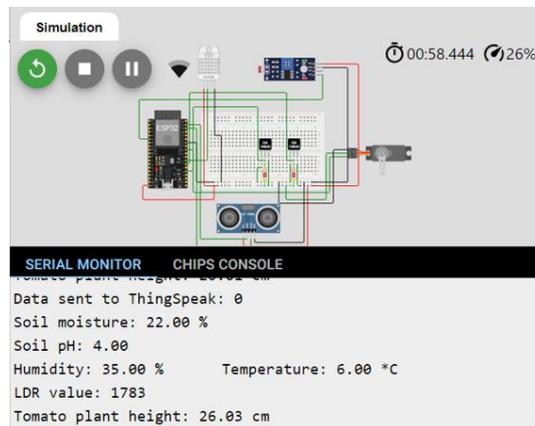
4.5. Implementasi

Dibagian ini akan difokuskan untuk melakukan pemrograman dan berbagai konfigurasi pada sistem yang akan dibangun, dengan tahapan sebagai berikut:

4.5.1 Mengirimkan Data dari ESP32-1 ke Platform Thingspeak

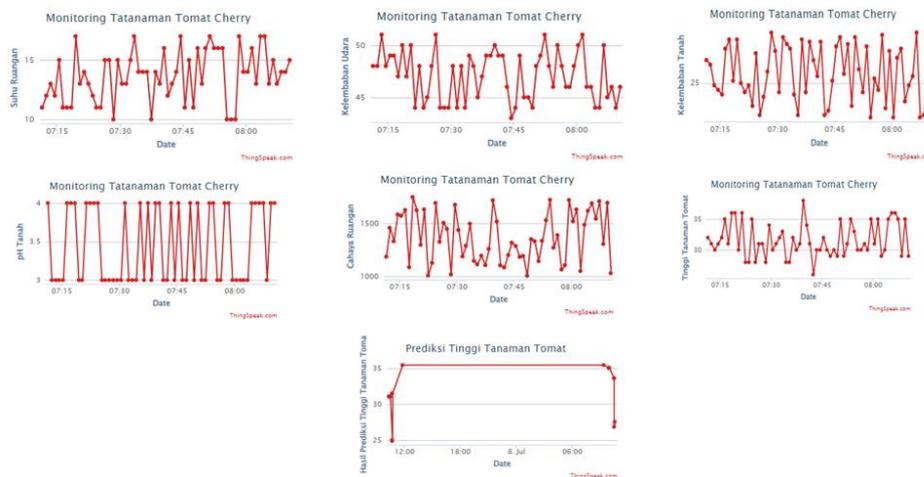
Data yang telah diproses oleh mikrokontroler ESP32-1 akan dikirimkan pada platform Thingspeak melalui protokol HTTP (*Hypertext Transfer Protocol*) menggunakan API (*Application Programming Interface*) yang telah disediakan oleh *platform Thingspeak* tersebut. Kelembaban tanah harus berada pada rentang 60 – 80 %, seandainya kelembaban tanah berada diangka kurang dari 60%, maka akan dinyalakan pompa air menggunakan servo untuk menjaga kelembaban tanahnya. Adanya API yang digunakan untuk mengirimkan data hasil seluruh deteksi sensor menggunakan *Write API Key*.

Gambar 6 mengenai aktifitas pengiriman data ke *platform Thingspeak*:



Gambar 6. Mengirimkan Data dari ESP32-1 ke *Platform Thingspeak*

Data yang tersimpan pada *platform Thingspeak* dapat ditampilkan dalam bentuk grafik, dengan keterangan data hasil deteksi dan waktunya, seperti gambar 7.

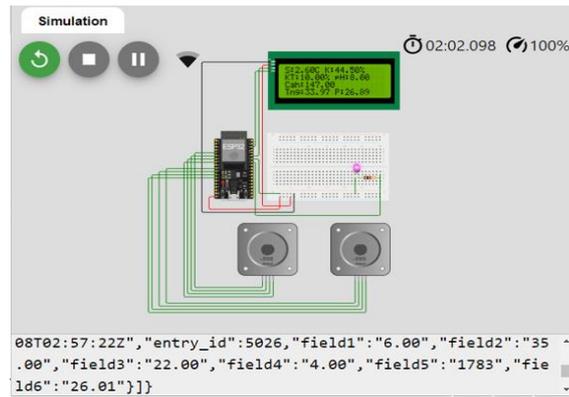


Gambar 7. Grafik Monitoring *Platform Thingspeak*

4.5.2 Menerima Data dari *Platform Thingspeak* ke ESP32-2

Data yang telah diproses baik melalui mikrokontroler ESP32-1 ataupun *machine learning* untuk melakukan prediksi, data tersebut perlu untuk ditarik atau fetch dari *platform Thingspeak* ke ESP32-2 untuk menyalakan aktuator yang sesuai dengan kondisinya. Suhu ideal memiliki pengaruh yang baik pada warna buah Tomat yaitu berwarna merah merata, dengan suhu 24-28 °C. Kelembapan udara yang baik harus berada di 80 %, karena kelembapan yang tinggi menjadi faktor penghambat pertumbuhan tanaman karena banyak hama dan penyakit. Namun, jika kelembapan rendah, maka proses berbuah menjadi lebih sulit [21][22]. Parameter Ph pada tanah yang baik harus selalu berada di rentang 5 – 7, pH derajat keasaman atau pH tanah sangat berpengaruh terhadap pertumbuhan dan perkembangan suatu tanaman [26]. Adanya cahaya matahari sangat dibutuhkan oleh tanaman tomat cherry, setidaknya tanaman tomat cherry membutuhkan cahaya matahari selama 6 - 8 jam perhari, yang berguna untuk melakukan fotosintesis [24]. Terkait *API Keys* yang digunakan itu berbeda dengan *API Keys* untuk proses pengiriman data ke platform Thingspeak, *API Keys* yang digunakan yaitu *Read API Keys* untuk menerima data dari *platform Thingspeak*.

Gambar 8 merupakan gambar sistem dalam menerima data dari *platform Thingspeak* ke ESP32-2:



Gambar 8. Menerima Data dari Platform Thingspeak ke ESP32-2

4.5.3 Integrasi Aplikasi Telegram Dengan ESP32-1

Aplikasi telegram [27] digunakan untuk memberikan notifikasi apabila terdeteksi pH pada tanaman tomat cherry yang tidak berada diambang batas aman. Parameter Ph pada tanah yang baik harus selalu berada di rentang 5 - 7, pH derajat keasaman atau pH tanah sangat berpengaruh terhadap pertumbuhan dan perkembangan suatu tanaman. Mikrokontroler yang digunakan untuk terhubung dengan aplikasi telegram yaitu ESP32-1, dalam menghubungkan antara aplikasi telegram dengan ESP32-1 menggunakan telegram bot yang bernama *BotFather*.

Pengiriman notifikasi melalui aplikasi telegram, seperti pada gambar 9.



Gambar 9. Notifikasi Melalui Aplikasi Telegram

4.5.4 Penjelasan Proses *Machine Learning*

Pada bagian ini akan dijelaskan mengenai tahapan-tahapan dalam proses pembelajaran mesin [28] yang meliputi pengumpulan data, persiapan data, seleksi fitur, pembagian data, prapemrosesan, pelatihan model, evaluasi model, serta implementasi *real-time*. Tahapan-tahapan ini dirancang untuk memastikan bahwa model yang digunakan mampu memberikan prediksi yang baik.

4.5.4.1 Pengumpulan Data

Data diperoleh dari sistem *Generative Artificial Intelligence* (AI) sebanyak 5000 sampel, di mana setiap sampel terdiri dari beberapa parameter yang disesuaikan dengan kondisi lingkungan sebenarnya berdasarkan deteksi sensor. Parameter yang dikumpulkan mencakup suhu (DHT11), kelembaban udara, kelembaban tanah, pH tanah, dan intensitas cahaya (LDR) sebagai fitur, serta tinggi tanaman sebagai target. Pada tabel 1 terdapat beberapa data sampel dari berbagai parameternya:

Tabel 1. Data *Dummy* Sensor

| Field1 (Suhu Udara) | Field2 (Kelembaban Udara) | Field3 (Kelembaban Tanah) | Field4 (pH Tanah) | Field5 (Intensitas Cahaya) | Field6 (Tinggi Tanaman) |
|------------------------|---------------------------------|---------------------------------|----------------------|----------------------------------|-------------------------------|
| 17 | 50 | 14 | 4 | 1145 | 30 |
| 13 | 43 | 24 | 3 | 1712 | 34 |
| 11 | 50 | 30 | 3 | 1499 | 32 |
| 12 | 47 | 17 | 3 | 1459 | 29 |
| 11 | 48 | 22 | 4 | 1255 | 31 |

Data ini perlu disesuaikan dengan format yang terdapat pada *platform ThingSpeak*, karena akan dilakukan proses training dan testing menggunakan data terbaru yang terdapat pada platform tersebut. Selain itu, data hasil prediksi juga akan dikirimkan kembali ke *platform ThingSpeak* menggunakan format yang sama.

4.5.4.2 Persiapan Data, Seleksi Fitur, Pembagian Data, dan Prapemrosesan

Data yang telah diperoleh akan melewati berbagai tahapan, yaitu:

1. Persiapan Data: Data disimpan dalam file CSV bernama *Final_Dummy_Data_Tomat.csv* dan dimuat ke dalam program menggunakan pustaka *pandas*.
2. Seleksi Fitur dan *Target*: Fitur yang dipilih meliputi *field1* hingga *field5*, yang mencerminkan suhu, kelembaban udara, kelembaban tanah, pH tanah, dan intensitas cahaya. *Target* yang diprediksi adalah *field6*, yang mencerminkan ketinggian tanaman tomat.
3. Pembagian Data: Data dibagi menjadi dua subset: set pelatihan dan set pengujian dengan rasio 80:20 untuk memastikan model dapat diuji dengan data yang tidak terlihat sebelumnya.
4. Prapemrosesan: proses dilakukan dengan menstandarisasi fitur-fitur yang dipilih menggunakan *StandardScaler* dalam *pipeline* pembelajaran mesin untuk memastikan konsistensi penerapan prapemrosesan pada data pelatihan dan pengujian.

4.5.4.3 Cross Validation

Model dievaluasi menggunakan *cross-validation* [29] untuk mengukur performa model secara lebih akurat dengan membagi data pelatihan ke dalam beberapa subset. *Cross-validation* menggunakan formula R^2 sebagai berikut:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (2)$$

4.5.4.4 Pelatihan Model

Pelatihan Model: Model dilatih menggunakan seluruh data pelatihan setelah *cross-validation* selesai.

4.5.4.5 Prediksi dan Evaluasi

Model yang telah dilatih digunakan untuk memprediksi ketinggian tanaman pada data pengujian. Hasil prediksi kemudian dievaluasi menggunakan metrik *Mean Squared Error* (MSE) dan R^2 Score [30]. Rumusnya yaitu:

1. *Mean Squared Error* (MSE)

$$MSE = \frac{1}{2} - \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (3)$$

2. R^2 Score

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (4)$$

Terdapat output dari kode program tersebut terlihat pada gambar 10.

```
Mean Squared Error: 0.8293746923333332
R^2 Score: 0.8939160260588597
Cross-Validated R^2 Scores: [0.88327349 0.90348099 0.90080124 0.89200691 0.90221045]
Mean Cross-Validated R^2 Score: 0.8963546176884721
```

Gambar 10. Output Evaluasi

4.5.4.6 Interpretasi Hasil

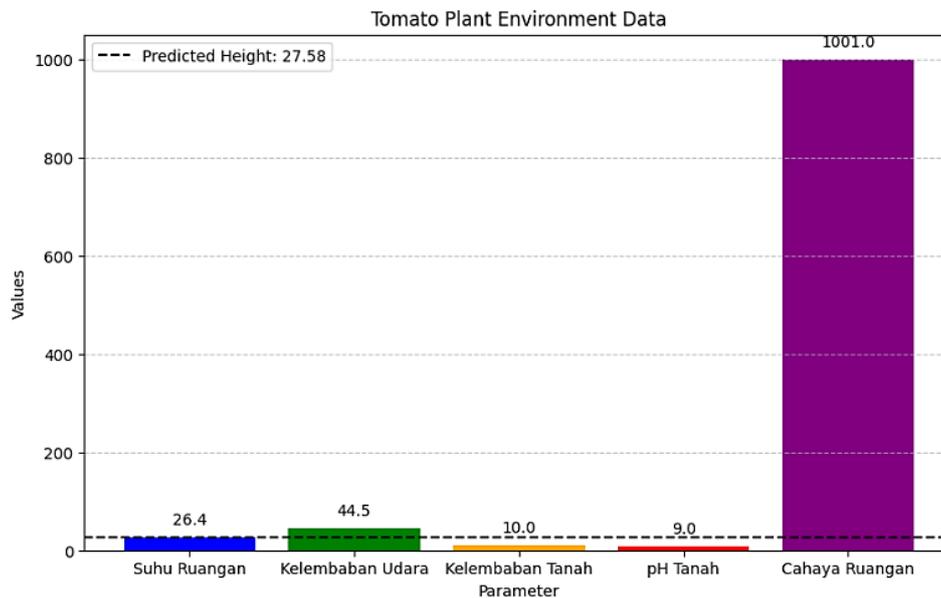
Interpretasi hasil atau *feature importances* [31] digunakan untuk mengetahui hasil evaluasi yang ditampilkan pada layar untuk memahami seberapa baik model bekerja dan fitur mana yang paling berpengaruh. Interpretasi hasil yang didapatkan/ *output* dapat ditunjukkan pada tabel 2:

Tabel 2. Interpretasi Hasil

| Feature | Importances |
|---------|-------------|
| Field1 | 0.035337 |
| Field2 | 0.023859 |
| Field3 | 0.173470 |
| Field4 | 0.123934 |
| Field5 | 0.643399 |

4.5.4.7 Implementasi *Real Time*

Data terbaru dari sensor dibaca secara *real-time* dari *Thingspeak*, diproses untuk memprediksi ketinggian tanaman, dan hasilnya dikirim kembali ke *Thingspeak*. *Output Implementasi Real Time* ditunjukkan pada gambar 12.



Gambar 12. *Output Implementasi Real Time*

4.5.5 *Black Box Testing*

Pengujian atau *testing* sistem yang telah dibangun sangat diperlukan untuk memastikan bahwa seluruh fungsi sistem dapat beroperasi dengan baik. Pada penelitian ini, metode *Black Box Testing* [32] akan digunakan untuk melakukan pengujian tersebut. Tabel 3 menyajikan daftar pengujian yang akan dilakukan:

Tabel 3. *Black Box Testing*

| No. | Hal yang akan diuji | Skenario yang dilakukan | Hasil yang diharapkan | Hasil dari pengujian |
|-----|--|---|---|----------------------|
| 1 | Menampilkan hasil deteksi seluruh sensor dalam bentuk grafik | Mengirimkan data seluruh sensor melalui ESP32 menggunakan protokol HTTP | Seluruh data hasil deteksi sensor ditampilkan dalam bentuk grafik <i>monitoring</i> | Berhasil |
| | | Tidak berhasil mengirimkan seluruh data sensor melalui ESP32 menggunakan protokol HTTP | Seluruh data hasil deteksi sensor tidak berhasil ditampilkan dalam bentuk grafik <i>monitoring</i> | Berhasil |
| 2 | Menyalakan aktuator servo untuk menyalakan air | Sensor <i>soil moisture</i> mendeteksi kelembaban tanah tidak berada di ambang batas aman | Servo Air dapat menyala sesuai dengan kondisi parameter kelembaban tanah hasil deteksi sensor <i>soil moisture</i> yang berada di ambang batas tidak aman | Berhasil |
| | | Sensor <i>soil moisture</i> mendeteksi kelembaban tanah berada di ambang batas aman | Servo Air tidak menyala dikarenakan kondisi parameter kelembaban tanah hasil deteksi sensor <i>soil moisture</i> berada di ambang batas aman | Berhasil |
| 3 | Pengiriman notifikasi telegram | Sensor pH mendeteksi pH tanah tidak berada di ambang batas aman | Sistem mengirimkan notifikasi melalui telegram terkait pH tanah | Berhasil |
| | | Sensor pH mendeteksi pH tanah berada di ambang batas aman | Sistem tidak mengirimkan notifikasi melalui telegram terkait pH tanah | Berhasil |
| 4 | Menyalakan kipas angin masuk dan keluar | Sensor DHT11 mendeteksi suhu ruangan terlalu panas | Kipas angin masuk dan keluar dapat menyala apabila suhunya terdeteksi oleh sensor terlalu panas | Berhasil |

| No. | Hal yang akan diuji | Skenario yang dilakukan | Hasil yang diharapkan | Hasil dari pengujian |
|-----|--|--|---|----------------------|
| 5 | Menyalakan lampu UV | Sensor DHT11 mendeteksi suhu ruangan tidak panas atau dingin | Kipas angin masuk tidak dinyalakan apabila suhunya terdeteksi oleh sensor tidak panas | Berhasil |
| | | Sensor LDR mendeteksi cahaya diruangan dalam kondisi gelap atau redup | Lampu UV dapat dinyalakan apabila cahayanya gelap atau redup atau malam hari | Berhasil |
| | | Sensor LDR mendeteksi cahaya diruangan dalam kondisi tidak gelap atau tidak redup | Lampu UV tidak dinyalakan apabila cahayanya tidak gelap atau tidak redup atau siang hari | Berhasil |
| 6 | Hasil seluruh deteksi sensor dan prediksi tinggi tanaman dapat ditampilkan pada layar LCD. | Menerima data hasil deteksi sensor dari thingspeak menggunakan <i>Write API Keys</i> | Layar LCD dapat menampilkan seluruh hasil deteksi sensor dan prediksi tinggi tanaman | Berhasil |
| | | Tidak berhasil menerima data hasil deteksi sensor dari thingspeak menggunakan <i>Write API Keys</i> | Layar LCD tidak menampilkan hasil deteksi sensor atau prediksi tinggi tanaman | Berhasil |
| 7 | Menampilkan hasil prediksi tinggi tanaman tomat cherry | Melakukan parsing data menggunakan <i>Write API Keys</i> di google colabs untuk data deteksi sensor terakhir yang tersimpan di <i>Thingspeak</i> , data tersebut digunakan untuk sebagai input pada model <i>machine learning</i> | Nilai prediksi tinggi tanaman tomat dapat ditampilkan dalam satuan cm | Berhasil |
| 8 | | Tidak berhasil melakukan parsing data menggunakan <i>Write API Keys</i> di google colabs untuk data deteksi sensor terakhir yang tersimpan di <i>Thingspeak</i> , data tersebut digunakan untuk sebagai input pada model <i>machine learning</i> | Nilai prediksi tinggi tanaman tomat dapat ditampilkan dengan nilai 0, yang memiliki arti gagal dalam parsing data deteksi sensor dari <i>Thingspeak</i> | Berhasil |

5. KESIMPULAN

Penelitian ini mengembangkan sistem pemantauan tanaman tomat cherry berbasis *Internet of Things* (IoT) dengan fitur deteksi tinggi tanaman menggunakan algoritma *Random Forest Regression*. Sistem ini memonitor parameter lingkungan menggunakan sensor DHT11, *Soil Moisture*, pH Probe, dan LDR. Aktuator seperti Servo, Kipas Angin, Lampu Ultraviolet, dan Layar LCD diaktifkan sesuai kondisi yang ditetapkan sistem. Data sensor dapat dilihat melalui grafik di platform Thingspeak atau layar LCD, serta notifikasi pH yang tidak sesuai dikirim melalui aplikasi Telegram. Prediksi tinggi tanaman tomat cherry menggunakan *Random Forest Regression* dilakukan berdasarkan data sensor terakhir dari Thingspeak, dengan evaluasi menggunakan *Mean Squared Error* (MSE) sebesar 0.8294 dan R^2 Score sebesar 0.8939. Pengujian *Black Box Testing* memastikan seluruh fungsi sistem bekerja dengan baik dalam berbagai skenario. Hasil pengujian *Black Box Testing* tersebut menunjukkan semua skenario berhasil, menunjukkan bahwa sistem ini berfungsi optimal dalam memantau dan memprediksi pertumbuhan tanaman tomat cherry secara *real-time*.

Penelitian ini memiliki kelemahan, yakni perangkat keras yang digunakan, seperti mikrokontroler, sensor, dan aktuator, masih terbatas pada perangkat yang disediakan di lingkungan simulasi Wokwi. Hal ini menyebabkan keterbatasan dalam variasi jenis perangkat yang dapat digunakan. Selain itu, pengiriman data melalui ESP32 menggunakan protokol *Hypertext Transfer Protocol* (HTTP) terkadang mengalami kegagalan, terutama karena bergantung pada koneksi Wi-Fi yang disediakan oleh simulasi Wokwi, yang kualitasnya dapat terpengaruh oleh jumlah pengguna yang mengakses simulasi secara bersamaan dari berbagai lokasi di seluruh dunia. Harapannya, penelitian ini di masa depan dapat menggunakan perangkat fisik secara langsung, memanfaatkan protokol yang lebih andal seperti *Message Queue Telemetry Protocol* (MQTTP) untuk pengiriman data ke cloud, serta membandingkan performa algoritma *Random Forest Regression* dengan algoritma lain dalam memprediksi tinggi tanaman tomat cherry.

REFERENSI

- [1] Badan Pusat Statistik, “Produksi Tanaman Sayuran, 2021-2023,” <https://www.bps.go.id/id/statistics-table/2/NjEjMg==/produksi-tanaman-sayuran.html>.
- [2] Liputan6, “8 Kesalahan Menanam Tomat di Pot, Hambat Pertumbuhan,” <https://www.liputan6.com/hot/read/4398175/8-kesalahan-menanam-tomat-di-pot-hambat-pertumbuhan>.
- [3] A. Mzibra *et al.*, “Improving growth, yield, and quality of tomato plants (*Solanum lycopersicum* L.) by the application of moroccan seaweed-based biostimulants under greenhouse conditions,” *Agronomy*, vol. 11, no. 7, Jul. 2021, doi: 10.3390/agronomy11071373.
- [4] Camilo Chiang, Daniel Bånkestad, and Günter Hoch, “Reaching Natural Growth: The Significance of Light and Temperature Fluctuations in Plant Performance in Indoor Growth Facilities,” *Multidisciplinary Digital Publishing Institute*, pp. 13–12, 2020.
- [5] KOMINFO, “Revolusi Industri 4.0,” <https://aptika.kominfo.go.id/2020/01/revolusi-industri-4-0/>.
- [6] Ashok J, Monika Bhatnagar, N. Rajendran, R. Priyadarshini, and Dinesh Mavaluru, *INTERNET OF THINGS (IOT): SYSTEMS AND APPLICATIONS*, 1st ed. NTL Publication, 2023.
- [7] Altexsoft, “IoT Architecture: the Pathway from Physical Signals to Business Decisions.” Accessed: May 30, 2023. [Online]. Available: <https://www.altexsoft.com/blog/iot-architecture-layers-components/>
- [8] Uniconverge Technologies, “IoT Based Smart Agriculture Monitoring System: Revolutionizing Modern Farming,” <https://www.uniconvergetech.in/blog/iot-based-smart-agriculture-monitoring-system>.
- [9] Abdennabi Morchid, Rachid El Alami, Aeshah A. Raezah, and Yassine Sabbar, “Applications of Internet of Things (IoT) and Sensors Technology to Increase Food Security and Agricultural Sustainability: Benefits and Challenges,” *Ain Shams Engineering Journal*, Sep. 2024.
- [10] Carlos A. Hernández-Morales, J.M. Luna-River, and Rafael Perez-Jimenez, “Design and Deployment of a Practical IoT-Based Monitoring System for Protected Cultivations,” *ELSEVIER*, vol. 186, pp. 51–64, Mar. 2022.
- [11] Dodi Yudo Setyawan, Rahmalia Syahputri, Nurfiana, and Nurjoko, “Internet of Things (IoT) Application in Smart Farming to Optimize Tomato Growth,” *DARMAJAYA*, 2022.
- [12] R. Nurhasanah, L. Savina, Z. M. Nata, and I. Zulkhair, “Design and Implementation of IoT based Automated Tomato Watering System Using ESP8266,” in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jun. 2021. doi: 10.1088/1742-6596/1898/1/012041.
- [13] Maria Carolina da Silva Andrea, João Pedro F. de Oliveira Nascimento, Fabrícia Conceição Menez Mota, and Rodrigo de Souza Oliveira, “Predictive Framework of Plant Height in Commercial Cotton Fields Using a Remote Sensing and Machine Learning Approach,” *ELSEVIER*, vol. 4, Aug. 2023.
- [14] Paulo Eduardo Teodoro *et al.*, “Predicting Days to Maturity, Plant Height, and Grain Yield in Soybean: A Machine and Deep Learning Approach Using Multispectral Data,” *Multidisciplinary Digital Publishing Institute*, Oct. 2021.
- [15] S. Kumar, P. Tiwari, and M. Zymbler, “Internet of Things is a revolutionary approach for future technology enhancement: a review,” *J Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0268-2.
- [16] Darine Ameyed, Fehmi Jaafar, Fabio Petrillo, and Mohamed Cheriet, “Quality and Security Frameworks for IoT-Architecture Models Evaluation,” *SpringerLink: SN Computer Science*, vol. 4, May 2023.
- [17] Omer Ali, Mohamad Khairi Ishak, Muhammad Kamran Liaquat Bhatti, Imran Khan, and Ki-II Kim, “A Comprehensive Review of Internet of Things: Technology Stack, Middlewares, and Fog/Edge Computing Interface,” *MDPI: Sensors*, Jan. 2022.
- [18] Ryan G. McClarren, “Decision Trees and Random Forests for Regression and Classification,” *SpringerLink: Machine Learning for Engineers*, May 2021.
- [19] Nima Beheshti, “Random Forest Regression,” <https://towardsdatascience.com/random-forest-regression-5f605132d19d>.
- [20] M. Artiyasa, I. Himawan Kusumah, A. Suryana, A. De Wibowo Muhammad Sidik, and A. Pradiftha Junfithrana, “Comparative Study of Internet of Things (IoT) Platform for Smart Home Lighting Control Using NodeMCU with Thingspeak and Blynk Web Applications,” *Jurnal Teknik Elektro*, vol. 2, no. 1, pp. 1–6, 2020.
- [21] Greg, “How Much Humidity Does My Cherry Tomato Need?,” <https://greg.app/cherry-tomato-humidity/>.
- [22] Vegetables Bayer, “Temperature, Humidity, and Water in Protected Culture Tomatoes,” <https://www.vegetables.bayer.com/au/en-au/resources/growing-tips/cultivation-insights/temperature--humidity--and-water-in-protected-culture-tomatoes.html#:~:text=Tomato%20plants%20can%20tolerate%20a,anywhere%20from%2060%20to%2085%25>.

- [23] MDN Web Docs, “An overview of HTTP,” <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>.
- [24] Patricia S York, “How To Grow And Care For Cherry Tomatoes,” <https://www.southernliving.com/garden/edible/how-to-grow-cherry-tomatoes#:~:text=Light,six%20to%20eight%20hours%20daily>.
- [25] Kiersten Rankel, “What Temperature Does My Sweet Valentine Cherry Tomato Need?,” <https://greg.app/sweet-valentine-cherry-tomato-temperature/>.
- [26] Amber Noyes, “Creating The Perfect Soil pH For Acid Loving Tomatoes,” <https://www.gardeningchores.com/soil-ph-for-tomato/>.
- [27] Claudia Roca, “What is Telegram and Why is It so Special?,” <https://global.thepower.education/blog/what-is-telegram-and-why-is-it-so-special>.
- [28] Sara Brown, “Machine learning, explained,” <https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained>.
- [29] Jason Brownlee, “A Gentle Introduction to k-fold Cross-Validation,” <https://machinelearningmastery.com/k-fold-cross-validation/>.
- [30] Walker Rowe, “Mean Square Error & R2 Score Clearly Explained,” <https://www.bmc.com/blogs/mean-squared-error-r2-and-variance-in-regression-analysis/>, May 2018.
- [31] Terence Shin, “Understanding Feature Importance in Machine Learning,” <https://builtin.com/data-science/feature-importance#:~:text=Feature%20importance%20involves%20calculating%20a,ones%20are%20the%20most%20important>.
- [32] Sourojit Das, “What is Black Box Testing: Types, Tools & Examples,” <https://www.browserstack.com/guide/black-box-testing>.
- [33] A. Rahmah, N. Sepriyanti, M. H. Zikri, I. Ambarani, and M. Y. bin Shahar, “Implementation of Support Vector Machine and Random Forest for Heart Failure Disease Classification,” *Public Research Journal of Engineering, Data Technology and Computer Science*, vol. 1, no. 1, pp. 34–40, Jul. 2023, doi: 10.57152/predatecs.v1i1.816.