

Pemantauan Dan Analisis Klasifikasi Pertumbuhan Tanaman Cabai Berbasis *Greenbox*

Monitoring And Analysis Of Greenbox Based Chillies Classification

1st Nurul Azizah
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

azizahnurul@student.telkomuniversity.
ac.id

2nd Nyoman Bogi Aditya Karna
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

aditya@telkomuniversity.ac.id

2nd Sussi
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

sussiss@telkomuniversity.ac.id

Abstrak

Setiap tahunnya, hasil produksi cabai selalu meningkat dan petani belum mampu memenuhi kebutuhan permintaan pasar yang sangat tinggi dikarenakan sering kali mengalami gagal panen akibat tanaman cabai rawit yang rusak oleh hama dan juga penyakit. Permasalahan yang dihadapi adalah terkendala akan situasi untuk *monitoring* tanaman jarak jauh dan kurangnya pengetahuan tentang parameter pertumbuhan pada tanaman cabai. Berdasarkan masalah tersebut, pembuatan *website* dan klasifikasi pertumbuhan akan menjadi solusi untuk bisa menghasilkan tanaman cabai dengan pertumbuhan yang ideal. Data yang ditampilkan pada *website* adalah data yang telah diambil oleh perangkat IoT berdasarkan pada kondisi *greenbox* dan pertumbuhan cabai yang disimpan di dalam database *firebase*. Selanjutnya, klasifikasi pertumbuhan bertujuan untuk membuat penentuan parameter paling berpengaruh dalam pertumbuhan tanaman cabai. Algoritma yang digunakan untuk pengklasifikasian adalah *Decision Tree*. Berdasarkan hasil pengujian fungsionalitas pada *website*, seluruh fitur dapat berjalan dengan baik dan optimal. Pada pengujian QoS, rata-rata *delay* yang dihasilkan sebesar 48,11 ms, rata-rata *throughput* yang dihasilkan sebesar 701,78 Kbps, sedangkan rata-rata *packet loss* yang dihasilkan sebesar 1,26%. Sedangkan, untuk pengklasifikasian tanaman dengan algoritma *Decision Tree*, diperoleh akurasi sebesar 99,93% dan mampu mengkategorikan pertumbuhan tanaman cabai, yaitu kurang optimal dan optimal.

Kata Kunci : Cabai, *Dataset*, *Decision Tree*, *Greenbox*, *Python*, *Website*.

Abstract

Every year, the production of chili is always increasing and farmers have not been able to meet the needs of very high market demand because harvest failures are often caused by cayenne pepper plants damaged by pests and diseases. The problems faced were the situation constrained to monitor plants remotely and the lack of knowledge about growth parameters in chili plants. Based on these problems, website

creation and growth classification will be the solution to produce chili plants with ideal growth. The data displayed on the website is data taken by IoT devices in greenbox conditions and chili growth stored in the firebase database. Furthermore, the classification aims to determine the most influential parameters in chili plant growth. The algorithm used for classification is Decision Tree. Based on the results of functionality testing on the website, all features can run well and optimally. In QoS testing, the average delay generated is 48.11 ms, the average throughput is 701.78 Kbps, while the average packet loss is 1.26%. Meanwhile, for classifying plants using the Yahoo Decision Tree, an accuracy of 99.93% was obtained and was able to categorize the growth of chili plants, namely less than optimal and optimal.

Keywords: Chili, *Dataset*, *Decision Tree*, *Greenbox*, *Python*, *Website*.

I. PENDAHULUAN

Di Indonesia, hasil produksi tanaman cabai setiap tahunnya selalu meningkat. Hal ini sesuai dengan data dari Kementerian Pertanian Republik Indonesia, bahwa pada tahun 2015-2019 secara Nasional rata-rata kenaikan produksi tanaman cabai, khususnya cabai rawit adalah sekitar 5% - 25% tiap tahunnya [1]. Kenaikan produksi ini tergolong tinggi jika dibandingkan dengan tanaman hollikultur lainnya, seperti bawang merah, lengkuas dan kunyit. Walaupun, hasil produksi tiap tahunnya selalu meningkat, namun harga pasar selalu tinggi. Berdasarkan data yang diperoleh dari Kementerian Pertanian Republik Indonesia menyatakan bahwa harga tanaman cabai di Indonesia bisa mencapai kisaran Rp40.000.00/kg – Rp80.000.00/kg [2]. Banyak petani yang kesulitan menanam tanaman cabai dikarenakan tanaman ini tergolong tanaman yang sangat rentan dengan hama dan penyakit terlebih lagi ketika musim hujan, sehingga perlu diperhatikan faktor-faktor yang dapat memengaruhi pertumbuhan tanaman cabai, seperti intensitas cahaya,

kelembapan udara dan kelembapan tanah serta kita juga harus memperhatikan kadar pH pada tanah [3].

Internet of Things (IoT) adalah sebuah konsep yang memungkinkan objek atau benda tertentu mempunyai kemampuan untuk mengirimkan data menggunakan jaringan internet. IoT merupakan perkembangan teknologi yang menjanjikan yang dapat mengoptimalkan kehidupan dengan sensor cerdas dan benda yang memiliki jaringan untuk bekerjasama dalam internet [4]. Selain itu, IoT juga dapat digabung dengan teknologi lain, seperti *Machine Learning*. *Machine Learning (ML)* adalah salah satu cabang dari *Artificial Intelligence (AI)* yang mencakup perancangan dan pengembangan algoritma yang dapat memungkinkan komputer untuk menentukan suatu pola tertentu dengan memanfaatkan suatu data. Tujuan utama dari ML adalah untuk membuat suatu keputusan ataupun klasifikasi dari data-data yang akan diolah. Salah satu algoritma di dalam ML adalah *Decision Tree*, yaitu algoritma sederhana yang menggunakan pohon keputusan dengan menggunakan klasifikasi terhadap sekelompok objek.

Adapun dalam penelitian Tugas Akhir akan dibagi menjadi 3 tahapan perencanaan, yaitu perancangan purwarupa *greenbox* dan alat, purwarupa sistem otomasi pengapuran tanah, dan pembuatan website serta model klasifikasi. Penulis akan berfokus pada pembuatan *website* yang akan berfungsi sebagai *monitoring* data dan perancangan model klasifikasi menggunakan algoritma *Decision Tree* untuk dapat menentukan hasil pengklasifikasian pada pertumbuhan tanaman cabai.

II. KAJIAN TEORI

Bab ini berisi tentang definisi dan konsep dasar dari metode-metode yang penulis gunakan untuk merancang tugas akhir ini.

A. Greenbox

Greenhouse atau yang dikenal sebagai rumah kaca adalah suatu bangunan yang rancang untuk budidaya tanaman yang dapat disesuaikan lingkungan tumbuhnya sehingga kondisi pada tanaman dapat terjaga secara optimal [6]. *Greenhouse* berfungsi untuk merawat dan menghindari tanaman dari perubahan cuaca yang berlebihan yang dapat berdampak pada pertumbuhan tanaman. Misalnya, curah hujan yang tinggi akan membuat tanaman cepat tumbuh dan bersifat sukulen sehingga lebih mudah terserang penyakit, busuk, dan mati [7]. Namun, *greenhouse* yang digunakan lebih kecil dari ukuran biasanya, yaitu *greenbox*. *Greenbox* lebih diperuntukkan untuk masyarakat yang ingin budidaya tanaman yang tidak mempunyai lahan luas.

B. Tanaman Cabai Rawit

Tanaman cabai rawit (*Capsicum Frutescens*) adalah salah satu jenis tanaman dengan kandungan gizi yang tinggi, seperti kandungan senyawa yang disebut *capsaicin*, yang mampu merangsang keluarnya endorfin sehingga dapat menghilangkan rasa sakit dan menimbulkan perasaan lebih sehat, cabai rawit juga mengandung vitamin A, B6, C, E, K, zat besi serta beberapa mineral penting, seperti magnesium, mangan, fosfor, selenium, dan zinc [8]. Cabai rawit dapat tumbuh pada dataran rendah maupun dataran

tinggi dengan ketinggian 1-1500 mdpl dengan lahan subur yang mengandung banyak unsur hara, gembur, cukup air, dan mengandung humus, serta memiliki curah hujan dengan suhu udara berkisar 25-32° C. Namun, petani kesulitan menaman tanaman cabai rawit karena tidak bisa mengatur keasaman (pH) pada tanah [9], sehingga unsur hara pada tanah rendah dan tanaman tidak dapat tumbuh optimal sesuai dengan harapan. Umumnya, pH tanah yang baik untuk tanaman adalah sekitar 5,5-7,5, jika kondisi tanah terlalu asam atau terlalu basa dapat menyebabkan pertumbuhan tanaman terganggu [10]. Untuk itu, harus dilakukan pengapuran dengan tujuan untuk menetralkan tanah dan memperkuat kerja dinding sel tanaman sehingga tegak dan tahan terhadap serangan jamur.

C. Internet of Things

Internet Of Things (IoT) adalah era baru dalam bidang internet [11], yang mampu mengubah setiap aspek kehidupan manusia dan sudah banyak diaplikasikan dalam kehidupan manusia, seperti pada bidang komunikasi, bisnis, pendidikan, dan lain-lain [12]. Menurut *Coordinator and support action for global RFID-related activities and standardisation* menyatakan *internet of things (IoT)* sebagai sebuah infrastruktur koneksi jaringan global, yang mengkoneksikan benda fisik dan virtual melalui eksploitasi data *capture* dan teknologi komunikasi [13].

D. CherryPy Web Server

CherryPy adalah *web server* sederhana yang dapat melakukan kinerja dengan optimal untuk menjalankan serta memfungsikan suatu *website* dengan menggunakan protokol HTTP [14]. *Web server* ini juga dapat dioperasikan di Windows atau Linux.

E. Website

Website adalah salah satu aplikasi yang berisi dokumen multimedia, seperti teks, gambar, suara, animasi, dan video di dalam situs web yang menggunakan protokol *Hyper Text Transfer Protocol (HTTP)* [15], kemudian *Website* akan dikonversi menjadi *Hyper Text Markup Language (HTML)* yang dapat diakses dengan menggunakan perangkat lunak yang disebut dengan browser [15][16].

a. HTML

HTML adalah sebuah bahasa pemrograman yang digunakan untuk membuat sebuah halaman *web*, menampilkan berbagai informasi dan juga dapat digunakan sebagai *link-link* menuju halaman *web* lainnya dengan menggunakan kode tertentu [17].

b. Django Framework

Django adalah *web framework fullstack (front-end dan back-end)* [17], berbasis bahasa pemrograman python yang didesain untuk membuat aplikasi *web* yang dinamis, kaya akan fitur, dan aman [18]. *Framework* dapat membantu kita membuat *website* dengan lebih cepat dibandingkan harus menulis kode dari nol [19].

c. CSS

CSS atau *Cascading Style Sheet* digunakan untuk mengatur *style* elemen yang ada di dalam halaman *web* [17], seperti pewarnaan, *font* yang

akan digunakan, dan lainnya. Selain itu, CSS juga dapat mengatur tata letak dan juga *layout responsive*, artinya dapat menyesuaikan di mana letak *website* itu akan dibuka [20].

d. JavaScript

JavaScript adalah Bahasa pemrograman yang digunakan untuk HTML dan *web* untuk *server*, pc, laptop, dan lainnya yang dapat membuat konten yang ditampilkan pada *web browser* menjadi dinamis dan interaktif [17]. *JavaScript* sifatnya *case sensitive*, artinya dalam penggunaannya penulisan huruf kecil dan huruf besar akan memiliki arti yang berbeda.

e. Bootstrap

Bootstrap merupakan salah satu jenis *framework* untuk CSS, HTML dan *JavaScript* yang akan digunakan untuk perancangan *website*. Keunggulan dari menggunakan *bootstrap* adalah dapat menyesuaikan dengan kebutuhan *website*, yaitu dapat memilih fitur CSS dan *JavaScript* sesuai dengan kebutuhan [17].

F. Machine Learning

Machine learning adalah bagian dari kecerdasan buatan, didefinisikan sebagai suatu algoritma atau program yang memungkinkan komputer membuat sistem menjadi cerdas dengan cara mempelajari data-data yang telah disediakan dengan algoritma atau program tersebut tidak didefinisikan secara eksplisit. Solusi dari machine learning bersifat soft computing, yaitu hasil dari algoritma yang digunakan tidak seratus persen benar, tetapi terdapat parameter akurasi [21]. Umumnya, algoritma-algoritma pada machine learning dibuat untuk dapat menyelesaikan tiga kebutuhan, yaitu prediksi, clustering, dan klasifikasi. Prediksi adalah kegiatan menebak nilai output berdasarkan nilai input yang digunakan, clustering adalah kegiatan mengelompokkan data berdasarkan tingkat kemiripan data-data tersebut, dan klasifikasi adalah kegiatan mengelompokkan data-data berdasarkan kategori tertentu yang selanjutnya hasil pengelompokkan tersebut digunakan lagi untuk mendeteksi kelompok suatu data [21].

a. Supervised Learning

Supervised Learning adalah teknik mengelompokkan suatu data yang memiliki label dan menghasilkan data yang diprediksi sesuai dengan parameter terkait dari kelas tertentu [22]. Klasifikasi ini melakukan pelatihan pada data yang digunakan untuk membentuk suatu model sesuai dengan fitur-fitur yang ada pada data [23].

b. Unsupervised Learning

Unsupervised Learning adalah salah satu tipe algoritma yang tujuannya untuk menarik kesimpulan dari dataset dengan mempelajari suatu data hanya dari kedekatannya saja atau yang biasa disebut *clustering* [24].

c. Semi Supervised Learning

Semi Supervised Learning adalah tipe algoritma yang berada diantara *Supervised* dan *Unsupervised Learning* [22]. Cara kerja algoritma ini adalah dengan menentukan dan mempelajari struktur variabel input dengan algoritma *unsupervised learning*. Selanjutnya, membuat

prediksi dari data yang tidak berlabel dan memasukkannya kembali sebagai data latihan dengan menggunakan algoritma *supervised learning*.

G. Decision Tree

Decision tree adalah salah satu teknik yang digunakan untuk klasifikasi terhadap sekumpulan objek atau *record* [25], yang dilakukan dengan pemisahan data berdasarkan nilai atribut dalam dataset sehingga setiap cabang yang terkait diberi label oleh satu regresi atau klasifikasi. Ada beberapa persyaratan yang harus dipenuhi jika ingin menggunakan algoritma *decision tree*, yaitu: algoritma ini mempresentasikan *supervised learning*, dataset harus banyak dan bervariasi, dan kelas atribut target harus diskrit [25].

Adapun proses dalam *decision tree*, yaitu:

- a. Mengubah data bentuk tabel menjadi model pohon. Pada tahap ini, kita akan menentukan atribut mulai dari akar, cabang hingga menuju keputusan [26].
- b. Mengubah model pohon menjadi *rule*. Formula yang digunakan untuk membangkitkan *rule*, yaitu:

IF premis THEN konklusi [26].

Premis dinyatakan sebagai simpul akar dan cabang, sedangkan konklusi dinyatakan sebagai simpul daun. Ketika premis yang digunakan terdapat dalam satu atribut maka akan dihubungkan dengan hubungan disjungsi, sebaliknya premis yang memiliki lanjutan dari premis pada cabang selanjutnya akan dihubungkan dengan konjungsi [26].

- c. Menyederhanakan *rule* (*Pruning*).

Pada tahap ini, kita perlu membuat tabel distribusi dengan menyatakan semua nilai kejadian pada setiap *rule*, lalu menghitung tingkat independensi antara atribut dengan target atribut, selanjutnya mengeliminasi kriteria yang tidak perlu, yaitu kriteria dengan tingkat independensi tinggi [26].

- a. C4.5

Algoritma C4.5 adalah algoritma untuk permasalahan klasifikasi yang diperkenalkan oleh Quinlan pada tahun 1996 pada *data mining* sebagai algoritma versi pembenahan pada ID3 yang mampu digunakan pada fitur kategorikal (nominal) maupun pada fitur numerik (*ratio*) dengan melakukan *pruning decision tree* dan penurunan *rule set*, menggunakan kriteria *gain*, dan menentukan fitur untuk pemecahan *node* pada pohon yang diinduksi [27].

H. Quality of Service

Quality of Service (QoS) adalah metode untuk mengukur seberapa baik jaringan untuk menyediakan layanan yang baik dan suatu usaha untuk mendefinisikan karakteristik dan sifat dari suatu layanan [28].

- a. Delay

Delay (*Latency*) adalah waktu yang dibutuhkan oleh data untuk menempuh jarak asal ke tujuan. Beberapa hal yang dapat mempengaruhi *delay*, yaitu jarak, media fisik, *congesti* atau lamanya waktu proses [28]. Untuk mengetahui seberapa bagus *delay*, maka nilai *delay* dapat dihitung dengan persamaan sebagai berikut.

$$Delay = \frac{Packet\ Length}{Link\ Bandwidth}$$

b. Throughput

Throughput yaitu kecepatan transfer data efektif yang diukur dalam bit per second (bps). Throughput adalah jumlah total kedatangan paket yang sukses diamati pada tujuan selama interval waktu tertentu dibagi interval waktu [28]. Throughput dapat dihitung dengan menggunakan persamaan sebagai berikut.

$$Throughput = \frac{Paket\ yang\ diterima}{Lama\ pengamatan}$$

c. Packet Loss

Packet Loss merupakan parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang yang terjadi karena collision dan congestion pada jaringan [28]. Nilai dari packet loss adalah berupa persentase (%). Semakin besar persentase packet loss, maka data yang diterima akan semakin jelek. Packet loss dapat dihitung dengan menggunakan persamaan sebagai berikut.

$$= \frac{Packet\ Loss}{(Paket\ data\ dikirim - Paket\ data\ diterima)} \times 100\%$$

III. METODE

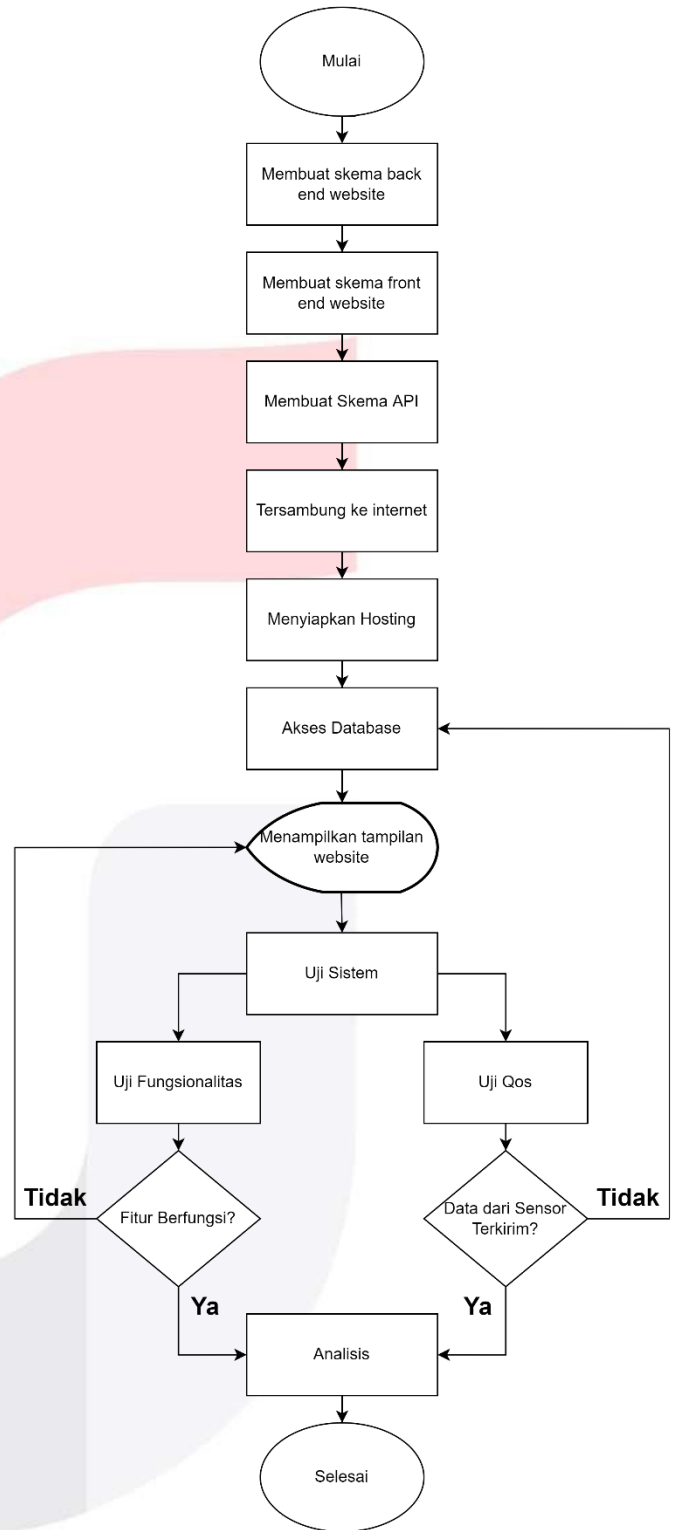
A. Desain Sistem

Secara umum, desain sistem ini pada penelitian ini adalah membuat perangkat IoT, membuat sistem otomatis pengapuran tanah, membuat website untuk kegiatan monitoring, dan analisa model klasifikasi pertumbuhan tanaman. Pada perangkat IoT tersebut akan dipasang beberapa sensor, yaitu:

- Sensor BH-1750 berfungsi sebagai sensor cahaya.
- Sensor DHT-22 berfungsi untuk mendeteksi kelembaban udara dan suhu ruangan.
- Sensor YL-69 merupakan sensor kelembaban yang mendeteksi tingkat kelembaban pada tanah.
- Sensor pH probe adalah sebuah sensor yang mengukur sebuah keasaman suatu objek.

Proses monitoring dilakukan dengan menggunakan website yang dikelola oleh admin terhadap tanaman cabai sehingga monitoring dapat dilakukan secara teliti dan lebih terfokus untuk menghasilkan tanaman dengan pertumbuhan yang optimal. Selanjutnya, data yang diperoleh dari perangkat IoT juga akan disimpan dalam database untuk keperluan analisa model klasifikasi pertumbuhan tanaman cabai. Data yang terdapat di dalam database sebelumnya akan diolah terlebih dahulu menjadi dataset dalam bentuk file Excel.

B. Diagram Alir Pengerjaan

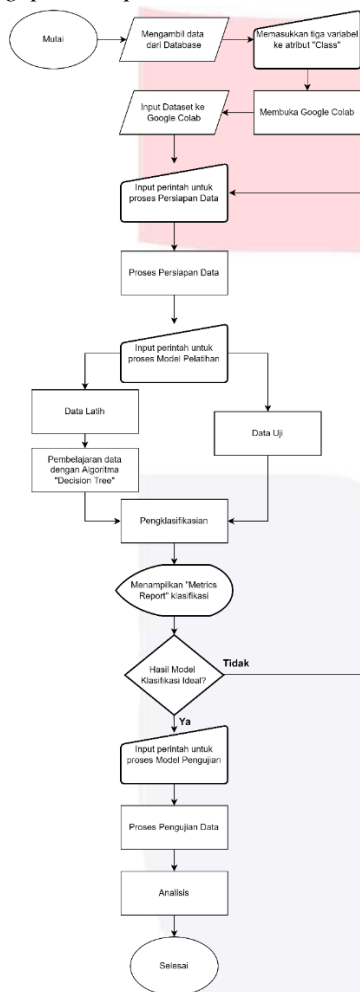


GAMBAR 3.1 Diagram Alir Pengerjaan Website

Pada gambar 3.1 menjelaskan alur kerja yang akan menjelaskan mengenai perancangan website sebagai media untuk monitoring tanaman. Alur pada diagram tersebut dimulai dengan menyambungkan perangkat ke internet kemudian mulai membuat tampilan pada website. Tampilan website akan dibuat sederhana dan menarik sehingga siapapun yang akan mengakses website akan merasa nyaman. Selanjutnya, dilakukan analisis terhadap website dengan menampilkan hasil data untuk kondisi

greenbox serta tanaman. Data tersebut merupakan hasil data yang diperoleh dari sensor-sensor pada perangkat Iot yang telah dibuat dan disimpan ke dalam *database*. Apabila tidak terdapat masalah, maka hasil data dari *database* akan berhasil terhubung dengan *website* dan dapat ditampilkan.

Tahap selanjutnya adalah menyiapkan *hosting* untuk *website* agar admin dapat dengan mudah menggunakan melakukan *monitoring greenhouse* dan tanaman di manapun dan kapanpun. Setelah semua selesai, maka akan dilakukan analisis pengaruh QoS dengan menguji nilai *delay*, *throughput*, dan *packet loss*.



GAMBAR 3.2 Diagram Alir Pengerjaan Model Klasifikasi Pertumbuhan Tanaman.

Pada gambar 3.2 menjelaskan alur kerja yang akan menjelaskan mengenai perencanaan model Analisa pertumbuhan tanaman cabai sebagai tahap akhir dari penelitian ini. Akan dibuat dataset dengan format *file Excel* yang diperoleh dari data yang tersimpan di dalam *database* sekaligus membuat atribut. Atribut akan terdiri dari 5, yaitu intensitas cahaya, kelembapan udara, suhu ruangan, kelembapan tanah, dan kadar pH tanah. *Dataset* yang telah dibuat kemudian di *upload* ke dalam *google colab* yang dioperasikan dengan bahasa pemrograman *python*. Selanjutnya, data akan diolah dengan algoritma *Decision Tree*. Hasil dari pengolahan data tersebut akan didapatkan hasil pengklasifikasian berupa optimal, kurang optimal, dan tidak optimal dari atribut yang digunakan untuk

mendapatkan model klasifikasi pertumbuhan tanaman cabai yang ideal.

IV. HASIL DAN PEMBAHASAN

A. Pengujian Fungsionalitas Website

Pada pengujian ini diimplementasikan pada sistem untuk mengetahui kondisi setiap fitur yang ada pada situs *web* yang telah dirancang dapat berfungsi dengan baik dan sebagaimana mestinya. Pengujian pada tahap ini akan melakukan uji sistem pada situs *web* dari proses di menu login sampai *logout*.

a. Pengujian Halaman Home

Pengujian pada halaman *home* ini bertujuan untuk mengetahui seluruh fitur yang ada pada *web*, mulai dari fitur *register*, *login*, *informasi*, dan *logout*.

TABEL 4.1 Pengujian Halaman Home

| Pengujian | Tahapan Uji | Keterangan | Hasil |
|---------------------------------|--|--|----------|
| Menampilkan halaman <i>home</i> | Mengakses halaman <i>home</i> | Jika akun telah login, akan muncul notifikasi "ANDA TELAH BERHASIL LOGIN" | Berhasil |
| | | Jika akun belum login, akan muncul notifikasi "ANDA BELUM LOGIN. SILAHKAN LOGIN TERLEBIH DAHULU" | Berhasil |
| | Admin menekan <i>register</i> pada navbar | Berhasil mengakses halaman <i>register</i> | Berhasil |
| | Admin menekan <i>login</i> pada navbar | Berhasil mengakses halaman <i>login</i> | Berhasil |
| | Ketika berhasil login, admin menekan informasi pada navbar | Berhasil mengakses halaman informasi | Berhasil |
| | Ketika berhasil | Berhasil keluar dari | Berhasil |

| | | | |
|--|--|--|--|
| | login, admin menekan <i>logout</i> pada navbar | halaman <i>home</i> dan menampilkan kembali halaman <i>login</i> | |
|--|--|--|--|

Berdasarkan tabel diatas, dapat disimpulkan bahwa fitur-fitur yang ada pada halaman tersebut sudah sesuai dan berjalan dengan baik khususnya untuk navbar agar memudahkan admin untuk berpindah halaman.

b. Pengujian Halaman *Register*

Pengujian pada halaman *register* bertujuan untuk mengetahui apakah pengelola cabai dapat membuat sebuah akun baru untuk bisa mengakses situs *web*.

TABEL 4.2 Pengujian Halaman *Register*

| Pengujian | Tahapan Uji | Keterangan | Hasil |
|-------------------------------------|---|---|----------|
| Menampilkan halaman <i>register</i> | Admin mengisi kolom yang dibutuhkan, seperti <i>username, email, password,</i> dan konfirmasi <i>password</i> | Admin masuk ke halaman <i>login</i> | Berhasil |
| | Admin mengosongkan kolom pada halaman <i>register</i> | Muncul notifikasi " <i>Please fill out this field</i> " | |

Berdasarkan tabel di atas, hasil pengujian ini dapat digunakan untuk mengetahui apakah admin dapat membuat akun baru untuk mengakses situs web dan masuk ke halaman login atau tidak.

c. Pengujian Halaman Login

Pengujian pada halaman login bertujuan untuk mengetahui apakah pengelola cabai dapat berhasil untuk melakukan akses masuk ke dalam situs web.

TABEL 4.3 Pengujian Halaman *Login*

| Pengujian | Tahapan Uji | Keterangan | Hasil |
|----------------------------------|--|------------------------------------|----------|
| Menampilkan halaman <i>login</i> | Admin memasukkan <i>username</i> dan <i>password</i> | Admin masuk ke halaman <i>home</i> | Berhasil |

| | | | |
|--|--|--|----------|
| | yang telah dibuat pada halaman <i>register</i> | | |
| | Admin mengosongkan kolom <i>username</i> atau <i>password</i> | Muncul notifikasi " <i>Please fill out this field</i> " | Berhasil |
| | <i>Username</i> atau <i>password</i> yang dimasukkan admin salah | Muncul notifikasi " <i>Please enter a correct username and password . Note that both fields may be case-sensitive.</i> " | Berhasil |

Berdasarkan tabel di atas, hasil pengujian ini dapat digunakan untuk mengetahui apakah admin dapat masuk ke halaman login atau tidak.

d. Pengujian Halaman Informasi

Pengujian pada halaman informasi ini dimaksudkan untuk mengetahui mengenai seluruh informasi, yaitu informasi tanaman dan kondisi *greenbox* sudah memenuhi untuk dapat digunakan atau tidak.

TABEL 4.4 Pengujian Halaman Informasi

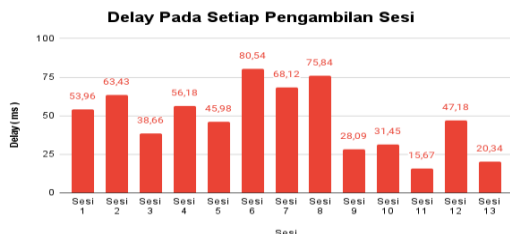
| Pengujian | Tahapan Uji | Keterangan | Hasil |
|---|-----------------------------|--|----------|
| Menampilkan halaman informasi | Mengakses halaman informasi | Berhasil menampilkan halaman informasi | Berhasil |
| Menampilkan informasi pertumbuhan tanaman cabai | | Berhasil menampilkan informasi pH dan kelembapan tanah | |
| Menampilkan informasi kondisi <i>greenbox</i> | | Berhasil menampilkan informasi kelembapan, suhu, dan intensitas cahaya | |

Berdasarkan hasil uji padah halaman informasi, menunjukkan bahwa halaman tersebut dapat menampilkan semua informasi yang dibutuhkan.

B. Pengukuran *Quality of Service* (QoS)

Pada tahap ini akan dilakukan pengujian untuk menentukan kualitas jaringan dari *website* menuju database yang telah dibuat berdasarkan parameter QoS, yaitu *delay*, *throughput*, dan *packet loss*.

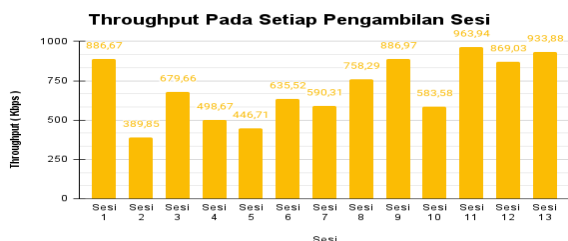
a. Pengujian *Delay*



GAMBAR 4.1 Hasil Pengujian *Delay*

Berdasarkan gambar 4.1, hasil pengujian yang didapatkan dengan menggunakan *software wireshark*, diperoleh rata-rata *delay* sebesar 48,11 ms dengan nilai tertinggi diperoleh pada sesi ke 6, yaitu sebesar 80,54 ms dan nilai terendah diperoleh pada sesi ke 11, yaitu sebesar 15,67 ms. Dari hasil pengujian ini, *delay* dikategorikan sangat baik karena memiliki nilai kurang dari 150 ms.

b. Pengujian *Throughput*



GAMBAR 4.2 Hasil Pengujian *Throughput*

Berdasarkan gambar 4.2, hasil pengujian yang didapatkan dengan menggunakan *software wireshark*, diperoleh rata-rata *throughput* sebesar 701,78 Kbps dengan nilai tertinggi diperoleh pada sesi ke 11, yaitu sebesar 963,94 Kbps dan nilai terendah diperoleh pada sesi ke 2, yaitu sebesar 389,85 Kbps. Dari hasil pengujian ini, *throughput* dikategorikan cukup karena nilai rata-rata berada pada rentang 700-1200 kbps.

c. Pengujian *Packet Loss*



GAMBAR 4.3 Hasil Pengujian *Packet Loss*

Berdasarkan gambar 4.3 hasil pengujian yang didapatkan dengan menggunakan *software wireshark*, diperoleh rata-rata *packet loss* sebesar 1,26% dengan nilai tertinggi diperoleh pada sesi ke 3, yaitu sebesar 2,4% dan nilai terendah diperoleh pada sesi ke 8, yaitu sebesar 0,35%. Dari hasil pengujian ini, *packet loss* dikategorikan sangat baik karena memiliki nilai rata-rata berada pada rentang 0 – 2%.

C. Pengukuran Fungsionalitas Dataset

a. Analisis Dataset dari Database

Data yang telah diambil oleh database *Firebase* akan disimpan di dalam beberapa file *Google Spreadsheets* dengan ekstensi (.xlsx) berdasarkan dari sensor perangkat IoT. Selanjutnya data-data tersebut akan digabungkan ke dalam satu *file* sebagai awal untuk membuat dataset.

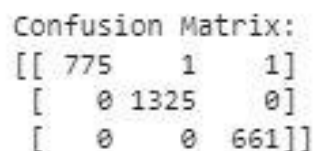
b. Analisis Tahap Persiapan Data

Pada tahap ini, data yang tersedia akan ditampilkan sebelum masuk ke tahap model pelatihan di fase selanjutnya. Dataset yang ditampilkan diambil dari *google drive*. Bagian ini hanya mencakup seputar analisis pada *dataframe* dengan perintah yang dijalankan pada *notebook*.

c. Analisis Fase Model Pelatihan

Tahapan awal yang dilakukan untuk analisis ini adalah dengan normalisasi data dengan menggunakan Label Encoder dengan library *Scikit-learn*. Dengan menggunakan *label encoder*, *categorical data* dapat diubah menjadi bentuk numerik.

D. Confusion Matrix



GAMBAR 4.4 *Confusion Matrix*

a. Accuracy Score

Analisis dari metrik ini adalah menginformasikan skor akurasi dari model klasifikasi. Perhitungan nilai akurasi juga dapat diperoleh dari *confusion matrix* yang ditampilkan pada tabel 4.6.

TABEL 4.5 *Confusion Matrix*

| | Kurang Optimal | Optimal | Tidak Optimal |
|--------------|----------------|---------|---------------|
| True Positif | 775 | 1325 | 661 |

Dari tabel 4.5 diperoleh jumlah data True Positive sebanyak 2763, sedangkan banyaknya

data adalah 2763. Sehingga, nilai akurasi yang diperoleh berdasarkan data tersebut adalah $2761/2763 \times 100\% = 99,927\% \approx 99,93\%$.

b. Precision

Berdasarkan hasil visualisasi dari *Confusion Matrix* perhitungan nilai presisi dapat dilihat sebagai berikut.

TABEL 4.6 Perhitungan Nilai Presisi

| | Kurang Optimal | Optimal | Tidak Optimal |
|---------------|----------------------|-----------------|---------------|
| True Positif | 775 | 1325 | 661 |
| False Positif | 2 | 0 | 0 |
| Precision | $775/(775+2) = 0,99$ | $1325/1325 = 1$ | $661/661 = 1$ |

Dari tabel 4.6 diperoleh nilai presisi dari masing-masing label pengklasifikasian. Sehingga, total nilai presisinya adalah $(0,99 + 1 + 1)/3 = 0,99 \approx 1$.

c. Recall

Berdasarkan hasil visualisasi dari *Confusion Matrix* perhitungan nilai *recall* dapat dilihat sebagai berikut.

TABEL 4.7 Perhitungan Nilai Recall

| | Kurang Optimal | Optimal | Tidak Optimal |
|----------------|----------------|--------------------|------------------------|
| True Positif | 775 | 1325 | 661 |
| False Negative | 0 | 1 | 1 |
| Recall | $775/775 = 1$ | $1325/1325 = 0,99$ | $661/(661 + 1) = 0,99$ |

Dari tabel 4.7 diperoleh nilai *recall* dari masing-masing label pengklasifikasian. Sehingga, total nilai presisinya adalah $(1 + 0,99 + 0,99)/3 = 0,99 \approx 1$.

d. f1 Score

f1-score adalah perbandingan rata-rata nilai *presisi* dan *recall*. Adapun perhitungannya sebagai berikut.

$$f1 - score = \frac{2 \times (presisi \times recall)}{presisi + recall}$$

Sehingga, berdasarkan rumus di atas diperoleh f1 score sebesar $\frac{2 \times (1 \times 1)}{1 + 1} = 1$.

E. Classification Report

Di tahap ini adalah analisis terakhir yang diturunkan dari *Confusion Matrix*.

| Classification Report: | | | | |
|------------------------|-----------|--------|----------|---------|
| | precision | recall | f1-score | support |
| KURANG OPTIMAL | 1.00 | 1.00 | 1.00 | 777 |
| OPTIMAL | 1.00 | 1.00 | 1.00 | 1325 |
| TIDAK OPTIMAL | 1.00 | 1.00 | 1.00 | 661 |
| accuracy | | | 1.00 | 2763 |
| macro avg | 1.00 | 1.00 | 1.00 | 2763 |
| weighted avg | 1.00 | 1.00 | 1.00 | 2763 |

GAMBAR 4.5 Classification Report

Pada gambar 4.5 menunjukkan nilai dari parameter performansi yang menjadi pertimbangan dalam menentukan model klasifikasi. Terdapat 3 label pengklasifikasian yang terbaca sesuai dengan dataset awal yang digunakan, yaitu kurang optimal, optimal, dan tidak optimal.

V. KESIMPULAN

Keseluruhan sistem terkait dengan monitoring data telah berfungsi dengan baik dan optimal. *Website* berhasil membaca data yang tersimpan di dalam database *Firestore* dan mampu memperbaharui data sesuai dengan pengambilan data yang dilakukan perangkat IoT selama 24 jam penuh. Hasil pengujian fungsionalitas pada website berjalan dengan baik dan fitur yang ada di dalamnya dapat berjalan. Terdapat perbedaan tampilan dari *Website* yang berada di *localhost* dengan *Website* yang telah dihosting, yaitu navbar dan gambar yang telah didesain tidak dapat terbaca. Pada pengujian QoS, *delay* yang dihasilkan berada dalam kategori sangat baik dengan nilai rata-rata yang diperoleh sebesar 48,11 ms. Selanjutnya, *throughput* dikategorikan cukup dengan nilai rata-rata yang diperoleh sebesar 701,78 Kbps. Terakhir, *packet loss* dikategorikan sangat baik dengan nilai rata-rata sebesar 1,26%. Dataset yang digunakan dalam ML menggunakan klasifikasi *Supervised Learning*. Terdapat tiga label yang terlihat pada model klasifikasi, yaitu kurang optimal, optimal, dan tidak optimal. Hasil parameter akurasi adalah 99,93%. Model klasifikasi mampu memenuhi label data optimal dan kurang optimal.

REFERENSI

- [1] "BDSP." <https://aplikasi2.pertanian.go.id/bdsp/id/komoditas> (accessed May 07, 2021).
- [2] "- SIMHARGA | Kementan." https://aplikasi2.pertanian.go.id/simharga2017/konsumen_kota/hkp4 (accessed May 07, 2021).
- [3] S. A. Andayani, "FAKTOR-FAKTOR YANG MEMPENGARUHI PRODUKSI CABAI MERAH," vol. 1, p. 262, 2016.
- [4] Wilianto and A. Kurniawan, "View of Vol 8 No 2 (2018)," *MATRIX*, vol. 8, p. 36, 2018, Accessed: May 07, 2021. [Online]. Available: <http://ojs.pnb.ac.id/index.php/matrix/issue/view/87/16>.
- [5] F. J. Kaunang, R. Rotikan, and G. S. Tulung, "Pemodelan Sistem Prediksi Tanaman Pangan Menggunakan Algoritma Decision Tree," *CogITO Smart J.*, vol. 4, no. 1, p. 213, Jun. 2018, doi: 10.31154/cogito.v4i1.115.213-218.

- [6] J. T. Pertanian, F. T. Pertanian, and U. G. Mada, "Rancangbangun Aktuator Pengendali Iklim Mikro di dalam Greenhouse untuk Pertumbuhan Tanaman Sawi (*Brassica rapa* var. *parachinensis* L.)," *Agritech J. Fak. Teknol. Pertan. UGM*, vol. 34, no. 2, pp. 213–222, 2014, doi: 10.22146/agritech.9512.
- [7] N. Rostini, "9 Strategi Bertanam Cabai Bebas Hama & Penyakit - Dr. Neni Rostini, Ir. MS. - Google Buku." https://books.google.co.id/books?hl=id&lr=&id=uPuDAgAAQBAJ&oi=fnd&pg=PA8&dq=dapat+mengganggu+pembuahan+serta+lebih+rentan+terkena+hama&ots=AD55JKQDjg&sig=-USePPZe7xLSWPApVH-G2mlqFzE&redir_esc=y#v=onepage&q&f=false (accessed Apr. 23, 2021).
- [8] A. MS., "Kiat Sukses Budidaya Cabai Rawit - Alif S.M. - Google Buku." https://books.google.co.id/books?hl=id&lr=&id=2iaeDgAAQBAJ&oi=fnd&pg=PA20&dq=didaya+tanaman+cabai+rawit&ots=lkVfAnCyGo&sig=PtHCyAsqoNWO8R4XzC8ft_2NMiE&redir_esc=y#v=onepage&q=didaya+tanaman+cabai+rawit&f=true (accessed Apr. 23, 2021).
- [9] M. Jamilah and U. Dwiputranto, "Pertumbuhan Cabai Merah (*Capsicum annum* L.) pada Tanah Masam yang Diinokulasi Mikoriza Vesikula Arbuskula (MVA) Campuran dan Pupuk Fosfat," doi: 10.20884/1.mib.2016.33.1.347.
- [10] "Mengatasi Tanah yang Terlalu Masam." http://kaltim.litbang.pertanian.go.id/ind/index.php?option=com_content&view=article&id=693&Itemid=59 (accessed Apr. 23, 2021).
- [11] E. Adriantantri and J. Dedy Irawan, "IMPLEMENTASI IoT PADA REMOTE MONITORING DAN CONTROLLING GREEN HOUSE," 2018.
- [12] M. Ameer, "(PDF) Analysis on Internet of Things, Application, Challenges and Related Future Technologies (A Literature Review)," 2018. https://www.researchgate.net/publication/341205441_Analysis_on_Internet_of_Things_Application_Challenges_and_Related_Future_Technologies_A_Literature_Review (accessed Apr. 23, 2021).
- [13] D. Setiadi, M. Nurdin, and A. Muhaemin, "PENERAPAN INTERNET OF THINGS (IoT) PADA SISTEM MONITORING IRIGASI (SMART IRIGASI)," *J. Infotronik*, vol. 3, no. 2, pp. 95–102, Dec. 2018, Accessed: Apr. 23, 2021. [Online]. Available: <http://183.91.79.105/index.php/infotronik/article/view/108>.
- [14] M. F. Ismail, *PERANCANGAN DAN IMPLEMENTASI IOT PLATFORM UNTUK PEMANTAUAN TANAMAN PAKCOY PADA TAHAP PENYEMAIAN.*
- [15] P. S. Hasugian, "Perancangan Website Sebagai Media Promosi Dan Informasi," *J. Inform. Pelita Nusant.*, vol. 3, no. 1, pp. 82–86, 2018.
- [16] W. Komputer, *Membangun Website Tanpa Modal.* 2010.
- [17] Z. N. Majesty *et al.*, "Website Organisasi Immg Menggunakan DJANGO FRAMEWORK," 2020.
- [18] D. Saputra and R. F. Aji, "View of Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django Dan Ruby On Rails Untuk Akses Data Dengan," *Bangkit Indones.*, vol. 2, no. VII, Oct. 2018, Accessed: May 07, 2021. [Online]. Available: <https://journal.sttindonesia.ac.id/index.php/bangkitindonesia/article/view/90/63>.
- [19] M. Nasir and N. Natasya, "SISTEM MONITORING AKUARIUM BERBASIS MIKROKONTROLER DAN DJANGO WEB FRAMEWORK," *Elektr. Borneo*, vol. 6, no. 1, pp. 25–28, Apr. 2020, doi: 10.35334/jeb.v6i1.1356.
- [20] R. Rachmanto, *9 Langkah Praktis Membuat Website Gratis.* 2017.
- [21] P. D. Kusuma, *Machine Learning Teori, Program, Dan Studi Kasus.* 2020.
- [22] A. S. Susilo, *BOK CHOY GROWTH PREDICTION MODEL ANALYSIS BASED ON SMART FARM USING MACHINE LEARNING.* 2021.
- [23] A. I. Riaddy, Y. Sibaroni, S. Si, A. Aditsania, and M. Si, "Ekstraksi Informasi Pada Makalah Ilmiah Dengan Pendekatan Supervised Learning," *eProceedings Eng.*, vol. 3, no. 1, Apr. 2016, Accessed: Feb. 03, 2022. [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/3755>.
- [24] D. Yolanda, M. H. Hersyah, and E. Marozi, "Implementasi Metode Unsupervised Learning Pada Sistem Keamanan Dengan Optimalisasi Penyimpanan Kamera IP," vol. 5, 2021.
- [25] Y. Yusuf, "Perbandingan Performansi Algoritma Decision Tree C5.0, Cart," *Seminar*, vol. 2007, no. Snati, pp. 0–3, 2007.
- [26] D. Sartika and D. I. Sensuse, "View of Perbandingan Algoritma Klasifikasi Naive Bayes, Nearest Neighbour, dan Decision Tree pada Studi Kasus Pengambilan Keputusan Pemilihan Pola Pakaian," *Jatiji*, vol. 1, no. 2, Mar. 2017, Accessed: May 07, 2021. [Online]. Available: <http://jurnal.mdp.ac.id/index.php/jatiji/article/view/78/50>.
- [27] A. Muhidin and A. B. Saputra, "View of Penerapan Data Mining Untuk Klasifikasi Kualitas Pipa Pvc Menggunakan Metode Algoritma C4.5 Studi Kasus Pt Cipta Aneka Agung," *SIGMA- J. Teknol. Pelita Bangsa*, vol. 9, no. 4, Jun. 2019, Accessed: May 07, 2021. [Online]. Available: <https://jurnal.pelitabangsa.ac.id/index.php/sigma/article/view/435/264>.
- [28] R. Wulandari, "View of ANALISIS QoS (QUALITY OF SERVICE) PADA JARINGAN INTERNET (STUDI KASUS : UPT LOKA UJI TEKNIK PENAMBANGAN JAMPANG KULON - LIPI)," *J. Tek. Inform. dan Sist. Inf.*, vol. 2, no. 2, Aug. 2016, Accessed: May 07, 2021. [Online]. Available: <https://journal.marannatha.edu/index.php/jutisi/article/view/620/616>.

