

BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Penelitian pada kali ini akan menggunakan metode eksperimen dan deskriptif. Metode eksperimen akan berupa uji tanah pada saat memiliki kadar air 0% hingga 100% dengan bantuan pengapian, uji pembukaan dari valve berapa volume air yang keluar dari tangka air, serta pembuatan sistem pemantauan berupa data untuk memantau secara keseluruhan dari perancangan dan pembuatan hardware dan software, uji coba alat, troubleshooting komponen jika terjadi error, pengambilan data, analisis data, dan penggunaan metode machine learning. Sedangkan metode deskriptif akan meliputi tahap studi literatur untuk mempelajari konsep fisika yang terkandung dalam sensor.

3.2 Prosedur Penelitian



Gambar 3. 1 Diagram Prosedur Penelitian

3.2.1. Studi Literatur

Pada tahap ini dilakukan penelusuran literatur tentang sistem pemantauan kualitas udara serta tanah, dan mempelajari fungsi sensor yang terhubung dengannya diperiksa dari sudut pandang fisik. Selain itu, tahap ini juga mencari berbagai jenis bahan referensi dari berbagai jurnal dan publikasi yang diterbitkan oleh peneliti sebelumnya untuk membangun sistem kontrol air dalam tanah untuk tanaman tomat.

3.2.2. Perancangan Perangkat Keras dan Lunak

Tahap pengembangan alat digunakan untuk merancang alat berupa perangkat keras dan perangkat lunak yang akan digunakan dalam sistem monitoring kualitas udara dan tanah. Membuat diagram rangkaian dan diagram rangkaian sistem instrumen pada perancangan perangkat keras dapat dibantu oleh Fritzing untuk pembuatan skematik dan rangkaian pada breadboard virtual lalu berpindah kepada perancangan pada papan ESP32. Pada saat perancangan perangkat lunak membuat diagram alir program yang akan digunakan dan memilih bahasa pemrograman yang akan digunakan untuk membantu penelitian.

3.2.3. Pembuatan Perangkat Keras dan Lunak

Pada tahap pembuatan alat ini, elemen-elemen yang sudah dirancang digabungkan menjadi sebuah sistem. Sistem perangkat keras dihubungkan berbagai komponen dan menghubungkannya ke komponen pelindung dalam wadah atau lingkungan. Sistem perangkat lunak melakukan beberapa pengkodean untuk mendukung pemrograman komponen dan membuat struktur data serta penghubungan ke Firebase. Stepper motor juga di pasang kepada kran air sebagai alat pengatur kadar air dalam tanah. Semua yang langsung berhubungan dengan perangkat keras deprogram melalui ESP32 dengan aplikasi Arduino IDE yang akan disimpan dan diterapkan pada mikrokontroler ESP32.

3.2.4. Uji Sensor dan Aktuator

Tahap uji lab merupakan tahapan untuk mengetahui berapakah nilai yang dibutuhkan oleh alat untuk mencapai sesuatu. Pada tahap uji ini memberikan tes kepada stepper motor dan kran berapa banyak debit air yang

dikeluarkan alat dengan waktu tertentu. Pengujian pada sensor DHT22 terdapat sensor kelembaban dan temperatur dengan memberikan perubahan suhu pada ruangan isolasi untuk mendapatkan bagaimana responsifitas dari sensor serta pengujian pada sensor kadar air tanah dan kalibrasi terhadap sensor tersebut.

3.2.5. Pengujian Alat dan Sistem

Tahap pengujian alat merupakan tahap awal pengujian komponen gabungan sebelum pengujian sistem di lingkungan nyata. Perbaiki kesalahan atau kesalahan perangkat keras atau perangkat lunak sehingga dapat segera diperbaiki. Stepper motor di uji untuk berapa derajat stepper motor harus dibuka dalam rentang waktu yang sama, volume air yang keluar di data.

Tahap pengujian sistem merupakan tahap implementasi sistem yang dibuat diterapkan langsung pada lingkungan tanah dan udara sekitar tumbuhan. Fase ini memastikan bahwa semua sistem berjalan dengan lancar dan data dapat ditransfer dari awal hingga akhir dan jika ada kesalahan selama implementasi, dapat dilakukan uji coba ulang dan memperbaikinya.

3.2.6. Pengambilan Data dan Analisis Data

Tahap pengumpulan data merupakan tahap sistem berhasil diimplementasikan di lingkungan dan dapat berfungsi dengan baik. Data dari sensor keadaan tanah dan lingkungan sekitar tumbuhan dicatat oleh sensor node dan dikirim melalui sistem komunikasi untuk disimpan dalam database. Data yang disimpan dalam database dapat digunakan kemudian untuk analisis dan koneksi ulang ke sistem Internet.

Tahap analisis data merupakan tahapan data yang didapatkan diolah yang biasa disebut dengan proses *Data Science* data di siapkan, dibersihkan, *EDA (Exploratory Data Analysis)*, *Feature Engineering*, *Modeling*, *Evaluation*. EDA merupakan langkah pertama dalam memahami data, jadi sangat penting sebelum rekayasa fitur dan pemodelan, ini memungkinkan analisis memahami apa isi data yang digunakan Ini didasarkan pada distribusi, frekuensi, korelasi, dll. Pada kenyataannya, rasa ingin tahu sangat penting. Pemahaman konteks data juga dianggap dapat menjawab pertanyaan mendasar. Pada umumnya EDA dapat dilakukan dengan beberapa cara:

- Univariate Analysis: analisis deskriptif dengan satu variabel.

- Bivariat Analysis: analisis relasi dengan dua variabel yang biasanya dengan target variabel.
- Multivariat Analysis: analisis yang menggunakan lebih dari atau sama dengan tiga variabel.

Perekayasa data sebelum dibuatnya model perlu melewati beberapa tahapan dari pembersihan data dari variabel yang tidak digunakan, menghapus nilai yang tidak sesuai, nilai yang tidak sesuai dengan format maupun data yang *corrupted*. Proses ini dibantu dengan Google Colab merupakan produk dari Google Research membantu untuk menulis dan mengeksekusi melalui kode Python. Melihat secara langsung data dari mean, std, min, max, 25%, 50%, 75%, hingga jumlah data yang dimiliki.

3.2.7. Uji coba algoritma dan Kesimpulan

Setelah menganalisis data yang didapatkan, data akan diuji kedalam beberapa algoritma dari *k-Near Neighbor(kNN)*, *Decisioin tree*, *Random forest*, dan *Deep neural network*. Kemudian menganalisis bagaimana kinerja dari sistem algoritma yang diterapkan pada data yang telah dianalisis dan memberikan kesimpulan atau penjelasan ilmiah yang mendukung penelitian ini.

3.3 Alat dan Bahan

Pada penelitian kali ini untuk kontrol kadar air didalam tanah, kendali jarak jauh serta algoritma yang akan digunakan alat dan bahan yang mendukung untuk pembuatan sistem tersebut dan komponennya terdapat pada tabel 3.1.

Tabel 3. 1 Komponen Penyusun Perangkat Keras Sistem Kendali Tumbuhan Tomat Ceri

Komponen Elektronika	Spesifikasi	Keterangan
ESP32	Tensilica Xtensa LX6 WiFi+Bluetooth Support	1 buah
DHT22	SKU-SN0137	1 buah
Capasitive Soil Moisture Sensor	-	2 buah

V.1.2

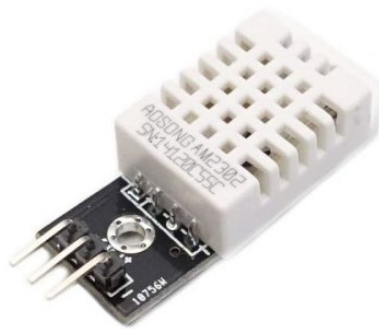
Stepper Motor	Nema 17 Stepper Motor 1,8°	1 buah
Selang Air	5mm	1 meter
Dripper Air	-	1 buah
Biji Tomat Ceri	-	5 buah

Pengujian kapasitansi sensor terhadap pengaruh jenis tanah sebagai sampel uji dilakukan pada empat jenis tanah. Adapun alat dan bahan yang digunakan dalam persiapan uji ditunjukkan oleh Tabel 3.2.

Tabel 3. 2 Alat dan bahan sampel uji

Alat dan Bahan	Keterangan
Oven	1 buah
Timbangan	Resolusi 0,1g
Mesh	Ukuran lubang 125mm
Pasir	360 gram
Medium Tanah	8 Sampel 252 gram

3.3.1. Sensor DHT22

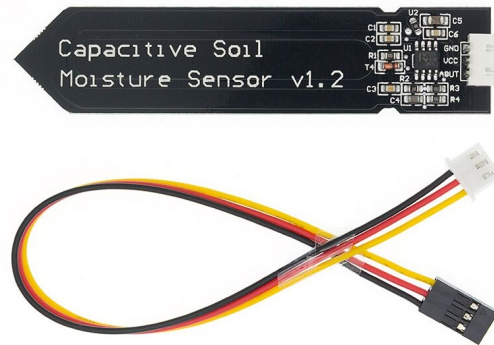


Gambar 3. 2 Sensor DHT-22

Pengukuran perubahan temperatur dan kelembaban udara disekitar tumbuhan tomat menggunakan sensor jenis kapasitif untuk mendapatkan nilai kelembabannya dan NTC sebagai sensor suhunya. Sensor ini dapat beroperasi pada lingkungan sekitar sampai cukup ekstrim bagi manusia dengan resolusi pengukuran $-40^{\circ}\text{C} - 80^{\circ}\text{C}$ dan akurasi pengukuran hingga $\pm 0.5^{\circ}\text{C}$ serta sensitifitas sebesar $\pm 0.1^{\circ}\text{C}$. Sedangkan sensor kelembabannya dapat bekerja

pada kelembaban 0 – 100%RH dan akurasi hingga $\pm 2\%RH$ dengan sensitifitas sebesar $\pm 0.1\%RH$. Sensor ini memiliki waktu respon dengan rata-rata 2 detik. (Aosong Electronics, 2021).

3.3.2. Sensor Kadar Air dalam Tanah berbasis Kapasitansi

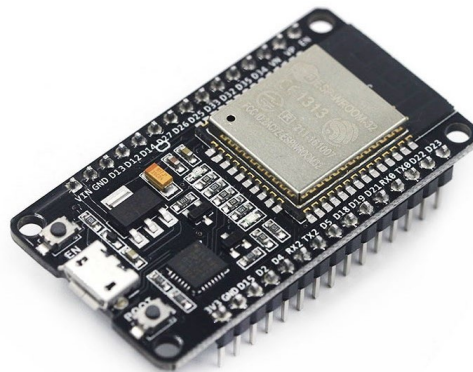


Gambar 3. 3 Capacitive Soil Moisture Sensor

Pada pengukuran perubahan kelembaban tanah disekitar tumbuhan tomat menggunakan sensor jenis kapasitif untuk mendapatkan nilai kelembabannya. Sensor ini beroperasi menggunakan tegangan 3.3-5 Volt dengan output tegangan 0-3 Volt. Keuntungan dari Capacitive Soil Moisture Sensor adalah terbuat dari bahan tahan korosi memberikan masa pakai yang lama.

3.3.3. ESP32

Mikrokontroler yang digunakan adalah ESP32 dengan chip Tensilica Xtensa LX6 yang memiliki clock 160/240MHz. ESP 32 memiliki banyak *library* pendukung yang berada di berbagai sumber dan dikarenakan *library* ini *open-sourced* menyebabkan banyaknya iterasi *library* yang sangat baik.

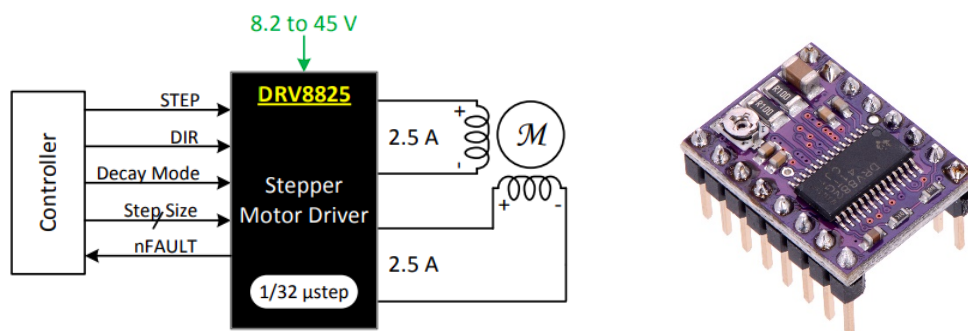


Gambar 3. 4 ESP32 (Espressif, 2020)

ESP32 memiliki total 39 GPIO (*General Purpose Input Output*), 18 pin mendukung *Analog to Digital Converter* (ADC), 3 pin mendukung SPI interfaces, 3 pin mendukung UART (*Universal Asynchronous Receiver-Transmitter*) interfaces, 2 pin mendukung I2C (*Inter Integrated Circuit*) interfaces, 16 pin PWM (*pulse width modulation*), 2 pin mendukung *Digital-to-Analog Converter* (DAC), 2 pin mendukung I2S (*Inter-IC Sound*) interfaces, dan 10 pin mendukung pembacaan berbasis kapasitif GPIO. Pada penelitian ini kita banyak memanfaatkan ADC dari ESP32 karena selain mendukung pembacaan sinyal analog, ADC pada ESP32 juga digunakan untuk memberikan koneksi dengan Wi-Fi menuju database. Dengan *flash memory* sebanyak 4MB dan *clock speed* sebesar 240MHz ESP32 sangat mendukung berbagai macam komputasi ringan (Espressif, 2021).

3.3.4. DRV8825

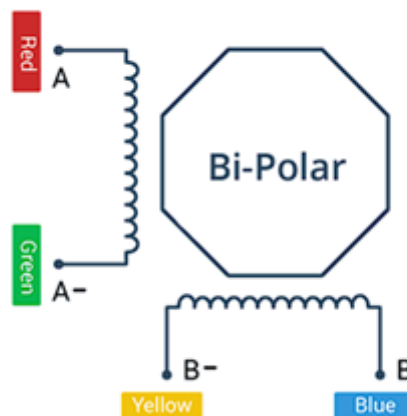
DRV8825 merupakan IC kontroler yang berfungsi dan khusus untuk stepper motor berjenis bipolar, dengan 2 H-Bridge driver dan *microstepping indexer*. DRV8825 dapat disuplai daya bertegangan 8,2V hingga 45V dengan output maksimal 2,5A. Interface STEP/DIR memudahkan untuk dihubungkan dengan mikrokontroler dengan konfigurasi full-step hingga 1/32-step dengan fitur *microstep indexer*, dengan hanya mengontrol arus yang diberikan pada masing-masing lilitan pada stepper motor, driver motor ditunjukkan pada gambar 3.5.



Gambar 3. 5 DRV8825 (Texas Instrument, 2022)

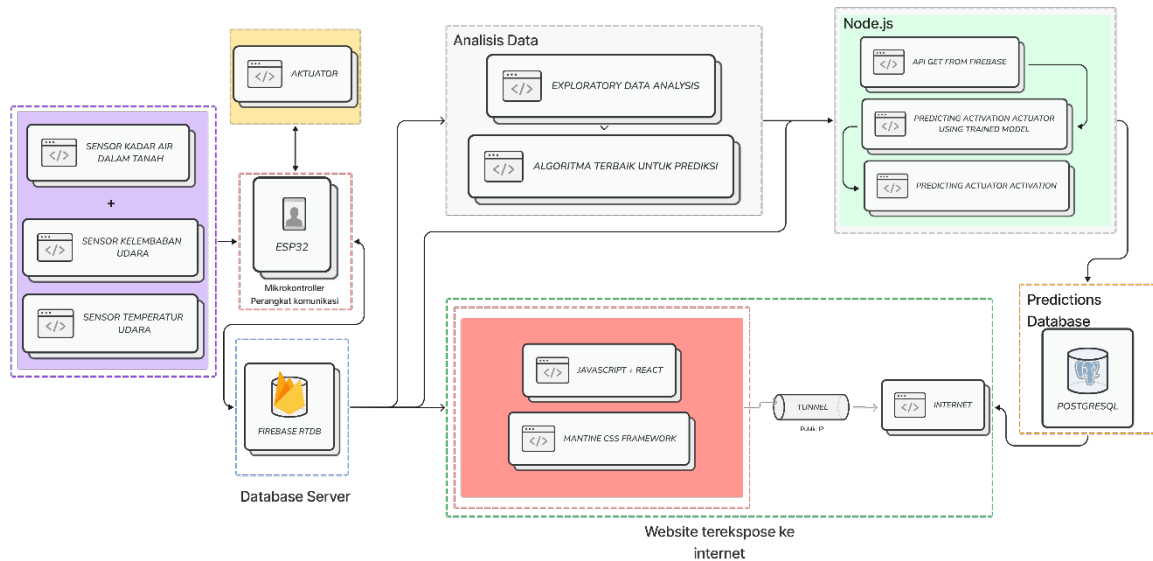
3.3.5. Nema 17 Stepper Motor 1,8

Nema 17 merupakan stepper motor yang memiliki derajat putar setiap stepnya sebesar 1,8 derajat, perputaran dari rotor diakibatkan oleh elektromagnetik diakibatkan oleh pulsa HIGH yang dikirimkan menuju koil sehingga menggerakkan menuju gigi terdekat dan menggerakkan motor sebanyak satu step. Setiap stepnya juga bergerak dipengaruhi oleh perilaku pulsa yang dikirimkan, dari frekuensinya untuk mengatur kecepatan motor, urutan pulsa untuk mengatur arah putar motor, ataupun jumlah pulsa mengatur berapa banyak putaran yang terjadi. Stepper motor membutuhkan driver untuk mengaktifkan fungsi dari stepper motor itu sendiri karena stepper motor membutuhkan kontrol dan sekuensi yang tepat. skematika stepper motor dapat terlihat pada gambar 3.6.



Gambar 3. 6 Rangkaian Stepper Motor Nema 17 1,8

3.4 Diagram Blok Sistem Data



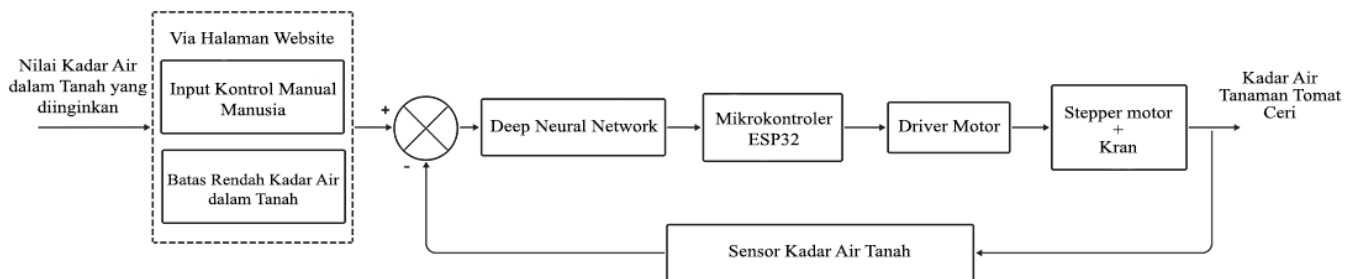
Gambar 3. 7 Diagram Blok Sistem Data Tanaman Tomat dan Metode EDA

Keseluruhan sistem kendali lingkungan di daerah tumbuhan serta tanah, ini terdiri beberapa komponen yang saling berhubungan. Diagram 3.2 ini menjelaskan bagaimana cara hubungan antar komponen satu sama lain. Semua sensor dan yang diperlukan dikoneksikan kepada ESP32, ESP32 memproses dan mengirimkan data menuju Firebase. Data dari Firebase lalu di berikan visualisasi di internet dengan bantuan JavaScript. Penggunaan JavaScript disini sangat diperlukan untuk melakukan pembacaan data berasal dari client-side yang menyimpan data khususnya Firebase yang kita gunakan sebagai database data akan dilakukan pengambilan data yang sudah di publish dari ESP32 menuju Firebase lalu pada saat web URL terbuka JavaScript akan mengaktifkan pengambilan data.

Pada proses terakhir kita dapatkan semua data dari Firebase database dan akan diolah menjadi EDA sehingga kita dapat menganalisis data apa yang memiliki korelasi antara satu sama lain, sehingga memudahkan kita dalam perancangan model Deep Neural Network.

Setelah data terolah dan mendapatkan persentase yang cukup baik, implementasi algoritma dapat berlangsung tanpa menghilangkan fitur input manual manusia seperti batas minimal kadar air dalam tanah ataupun input secara

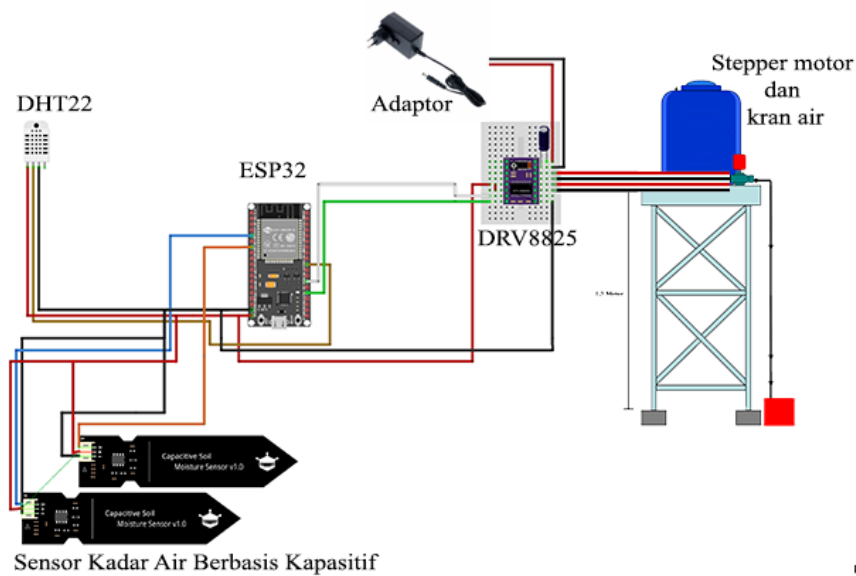
manual untuk menyalakan sistem kendali, sehingga dapat dijelaskan dengan gambar 3.8.



Gambar 3. 8 Diagram sistem kendali air dalam tanah

Sistem kendali memiliki alur pengendalian sesuai pada gambar 3.8 yang berawal dari nilai kadar air dalam tanah yang diinginkan dengan bantuan interface dari halaman website pengguna dapat mengendalikan batas rendah kadar air dalam tanah sehingga bila nilai kadar air tanah sama atau kurang dari nilai tersebut akan mengaktifkan driver motor serta pengguna bisa mengaktifkan driver motor secara manual, dari input tersebut mikrokontroler akan memproses nilai kendali dan untuk memerintahkan driver motor untuk menyala sehingga stepper motor memutar kran untuk menambahkan kadar air tanah pada tomat ceri. Setelah proses tersebut selesai sensor kadar air tanah akan memberikan feedback berupa nilai kadar air tanah terbaru pada tanah dari tanaman tomat ceri. Dari feedback yang diterima algoritma DNN akan memprediksi apakah nilai kadar air tanah sudah cukup atau tidak berasal dari data-data latihan yang didapatkan selama beberapa minggu.

Implementasi perangkat keras pada penelitian ini merupakan satu sistem besar yang mencakup pembacaan sensor, pengambilan data, pengiriman data, dan pengaktifan motor stepper untuk mengalirkan air menuju tumbuhan jika diperlukan. ESP32 dihubungkan oleh sensor *Capacitive Soil Moisture Sensor*, dan DHT22, dengan bantuan DRV8825 untuk menggerakkan stepper motor. Sehingga seluruh data tersimpan.



Gambar 3. 9 Rangkaian Sistem Perangkat Keras Sistem Kontrol Kadar Air dalam Tanah

Skematik hubungan antara mikrokontroler dengan sensor, driver, dan stepper motor dapat terlihat pada gambar 3.9 memperlihatkan hubungan I/O yang terdapat pada ESP32 sehingga hasil dari proses pembacaan dapat diproses dan tersimpan secara baik di cloud dengan hanya satu mikrokontroler. Konfigurasi pin setiap sensor dan aktuator saling terkoneksi, koneksi tersebut dapat terlihat pada table 3.3.

Tabel 3. 3. Konfigurasi Pin Komponen terhadap ESP32 dan DEV8825

Komponen	Pin Komponen	Pin ESP32
<i>Capacitive Soil Moisture Sensor</i>	VCC GND OUTPUT	5V GND GPIO 32 & GPIO 34
DHT22	VCC GND OUTPUT	5V GND GPIO 5
DRV8825	VMOT GNDMOT GND RST SLP STEP DIR	12V GND GND 5V 5V GPIO4 GPIO2
Komponen	Pin Komponen	Pin DRV8825

Nema 17 Stepper Motor	A	B2
	A-	B1
	B-	A1
	B	A2

Tabel 3.3. memberikan gambaran konfigurasi pin yang digunakan sensor, driver, dan motor stepper menuju ESP32 memiliki 3 tegangan kerja yang berbeda yaitu 12V, 5V dan 3.3V. Perbedaan tegangan tersebut dibutuhkan agar setiap I/O bekerja dengan baik. Tegangan kerja 12V digunakan untuk menggerakkan Nema 17 Stepper Motor melalui DRV8825, tegangan 5V digunakan sebagai pemberi daya *Capacitive Soil Moisture Sensor* dan DHT22. Penggunaan 5V pada DRV8825 untuk RST dan SLP sehingga tidak aktif atau motor berjalan dengan normal, jika RST dan SLP tidak diberikan tegangan secara otomatis teraktivasi. Sedangkan pin komponen pada Nema 17 Stepper Motor memiliki dua coil yaitu A dan B dan setiap coil memiliki arah dari + ke - untuk mengatur arah gerak sehingga saat input dari mikrokontroler untuk memberikan arah searah dengan jarum jam sesuai dengan pergerakan motor.

3.5 Persiapan Uji Sampel

Sampel uji dipersiapkan dengan adanya penimbangan sampel tanah sebelum dan sesudah pengeringan, proses ini dilakukan dengan bantuan oven pada suhu 105 – 110 °C sampai kadar air pada tanah menghilang. Kadar air menghilang dapat ditunjukkan dari hasil penimbangan sampai tanah tidak terjadi perubahan pada saat pengeringan berlangsung. Setelah sampel tanah dikeringkan dan ditimbang, pemberian air dengan konsentrasi berbeda sehingga sampel tanah yang memiliki kadar air tanah sesuai keinginan. Berdasarkan persamaan (2.1) pada Bab II, penentuan massa air yang ditambahkan setiap sampel tanah kering dapat ditentukan dengan persamaan

$$\text{Kadar air \%} = \frac{M_{air}}{M_{tanah}} \times 100\% \quad (2.1)$$

Persamaan (3.1) didasarkan pada perhitungan kadar air tanah menggunakan metode ASTM D4643-00 “*Standard Test Method for Determination of Water (Moisture) Content of Soil by the Microwave Oven Heating*”. Adapun prosedur penyiapan sampel uji dilakukan sebagai berikut :

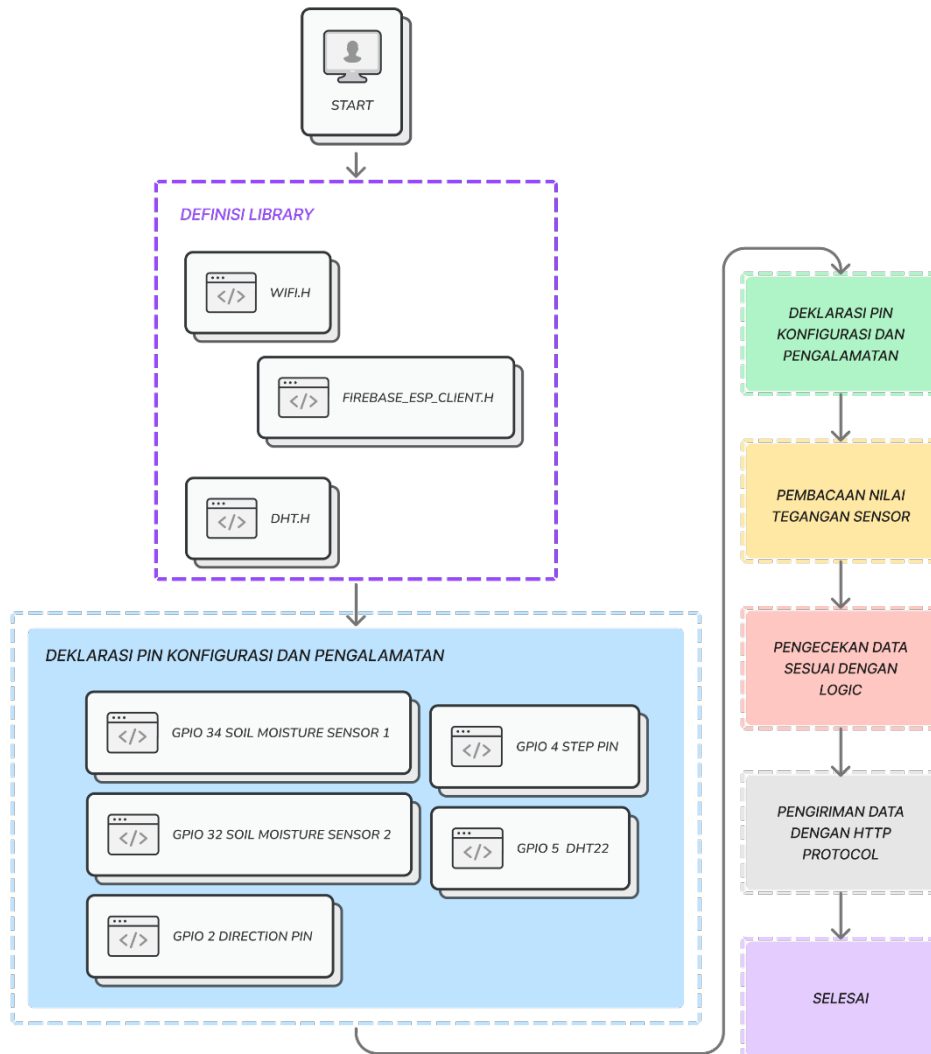
- a. Menimbang massa sampel tanah basah sebesar 252 gram.
- b. Mengeringkan sampel tanah basah pada *oven* dengan suhu 105-110°C sampai kandungan airnya hilang (± 12 jam).
- c. Mengayak sampel tanah yang sudah dikeringkan menggunakan mesh ukuran 125mm
- d. Menimbang massa sampel tanah kering yang telah diayak.
- e. Mencatat data massa sampel tanah kering.
- f. Mengulang dengan sampel pasir dengan massa awal sebesar 360 gram.
- g. Menambahkan air pada sampel tanah kering dengan persentasi kadar air tertentu.

3.6 Uji Stepper Motor Kran Air

Uji stepper motor dilakukan untuk memberikan karakterisasi untuk setiap mengetahui pergerakan dari stepper motor inilah yang mengatur berapa banyak jumlah volume air yang diinginkan pada stepper motor bipolar dengan spesifikasi step dari motor bergerak sebanyak 1,8 derajat sehingga untuk mencapai 90 derajat diperlukan sebesar 50 step. Uji stepper juga memberikan informasi sudut berapa berapakah stepper motor memiliki volume air yang baik hingga tidak memunculkan perbedaan antara satu sudut dengan lainnya, juga pada sudut berapa derajat kran tidak terbuka. Metode yang digunakan untuk mengukur uji ini menggunakan variabel waktu dan derajat putar kran oleh stepper motor.

3.7 Perancangan dan Desain Alur Pemrograman

Penggunaan alur data dimulai dari pengambilan data pada tanaman pada pot oleh *Capacitive Soil Humidity*, serta suhu dan kelembaban udara dikirimkan menuju ESP32 berupa tegangan dan diubah menjadi informasi data yang sesuai dengan alur diprogram menggunakan bahasa C++ pada Arduino IDE.



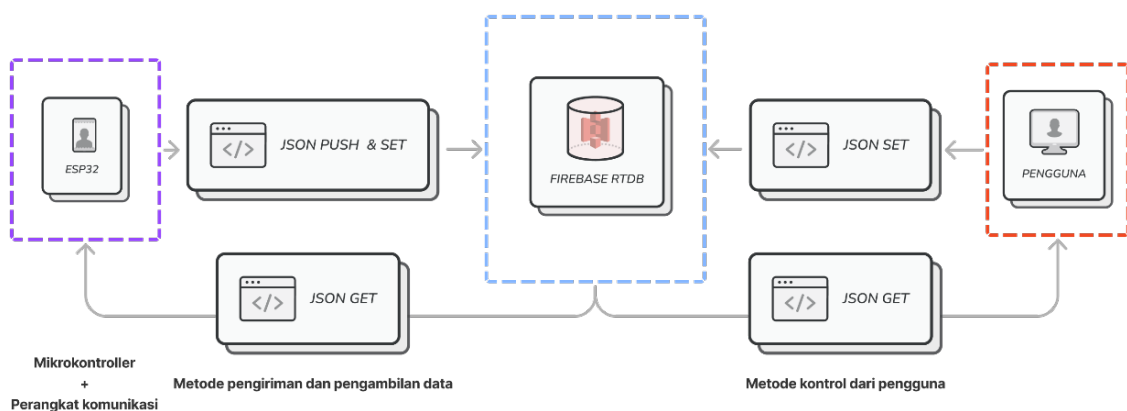
Gambar 3. 10. Diagram alur pemrograman sistem kendali pada ESP32

Alur sistem pemrograman ESP32 pada gambar 3.10 berawal dari pendefinisian library yang digunakan pada pembacaan dari sensor suhu ruangan dan kelembaban ruangan serta pengiriman data menggunakan WiFi dan database Firebase Google, data yang dikirimkan berupa *JavaScript Object Notation (JSON)*. Deklarasi pin yang digunakan merupakan sinyal analog untuk membaca

kelembaban tanah GPIO 34, 32, sedangkan GPIO 5 menerima sinyal digital untuk membaca kelembaban ruangan dan suhu ruangan tumbuhan. Penggerak dari stepper motor untuk membuka kran juga membutuhkan sinyal digital dari GPIO 2 untuk mengatur arah gerak dari motor dan GPIO 4 untuk mengatur kecepatan dari *microstep* yang dilakukan oleh motor untuk berputar.

Pengkonfigurasi dan pengalamanan sensor dari fungsi library yang digunakan untuk membaca voltase yang diterima dari sensor. Mikrokontroler ESP32 membutuhkan kecepatan pembacaan data serial pada 115200 bit/s, sehingga mikrokontroler bekerja untuk menerima input dan memberikan output dari nilai voltase dari sensor dan aktuator senilai 115200 bit untuk setiap detiknya. Lalu hasil dari input sensor tersimpan dengan variabel yang sudah dideklarasikan untuk proses logika dan operasi matematik yang dilakukan.

Sesudah pembacaan data dari sensor lalu tersimpan pada variabel yang sudah ditentukan, serta pemrosesan matematik untuk memunculkan data yang mudah dimengerti oleh manusia, data tersebut dikirimkan ke database google dengan format *JavaScript Object Notation* (JSON), yaitu format berbentuk object yang biasa digunakan untuk pengiriman “key” atau definisi dari data yang dikirim dan “value” berupa nilai yang dikirimkan, keduanya bisa berbentuk *integer* ataupun *string*. Dari data yang dikirimkan langsung menuju database menggunakan jalur komunikasi secara *Wireless-Fidelity* (Wi-Fi) yang sudah dideklarasikan dan fitur yang sudah terdapat pada ESP32 tanpa harus menggunakan Raspberry-Pi.



Gambar 3. 11 Alur pengiriman dan pengambilan data

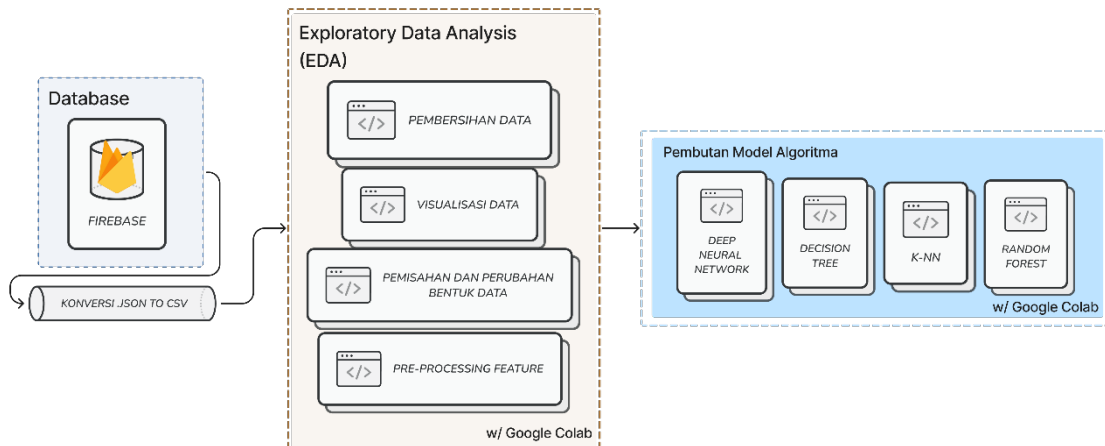
Alur program pada ESP32 sesuai dengan gambar 3.11 merupakan alur pengiriman data, data yang dikirimkan melalui *Wi-Fi* dengan fungsi *JSON Push Method* semua data dikirimkan secara *asynchronous* menambahkan data kedalam database sehingga kita dapat menyimpan seluruh data selama database ada, berbeda dengan *JSON Set Method* merupakan perubahan data secara spesifik pada data yang hanya memiliki perubahan data. Pengguna juga dapat memberikan input kepada mikrokontroler melalui *user-interface* yang dapat mengaktifkan motor untuk mengairi tumbuhan dengan *Method Set JSON* untuk mengirimkan data spesifik yang ingin diganti, yaitu pengaktifan tersebut. ESP32 menggunakan *Method Get* untuk mengambil data spesifik yang sedang dipantau, jika nilai yang terpantau berubah ESP32 melakukan validasi dan menjalankan operasi logika untuk mengaktifkan stepper motor, jika stepper motor sudah mati data juga dikirimkan menuju database menggunakan *Method Push* dan *Set*. Semua data yang dikirimkan dan didapatkan merupakan sistem *NoSQL* server sehingga memudahkan perubahan database jika ingin adanya penambahan fitur secara *realtime*.

Data yang berada pada database dapat langsung menjadi output file berupa *.json* untuk pengolahan data memerlukan bantuan dari *converter* untuk mengubah data menjadi *csv* memudahkan untuk menganalisa data dari perubahannya, apakah adanya anomali ataupun eksploratori data yang saling berkorelasi satu sama lain dengan bantuan statistika.

3.8 Pembersihan data, Analisis data dan Pembuatan Model

Setelah implementasi pada *greenhouse*, penerapan prediksi *structured-data* menggunakan menggunakan *Deep Neural Network*. Algoritma ini mengekstrak nilai persentase kadar air dalam tanah dengan sedikit *weight* dari temperatur udara dan kelembaban udara dalam *greenhouse*. Proses ini dibantu dengan Google Colab merupakan produk dari Google Research membantu untuk menulis dan mengeksekusi melalui kode Python melalui browser dan cocok untuk data analisis dan machine learning, selain itu bahasa Python untuk memudahkan pengolahan data untuk machine learning seperti: Matplotlib (sebagai visualisasi dan plotting data), Pandas (pengolahan dan manipulasi serta analisis data), Numpy

(mengubah data menjadi matriks dan vektor), dan TensorFlow (membantu dalam pembuatan neural network dengan Keras).



Gambar 3. 12 Diagram alur proses pembuatan model algoritma

Data yang tersimpan pada database Firebase dapat diakses sesuai dengan alur pada gambar 3.12 dan konversikan dari json menjadi csv untuk memudahkan pengolahan data. Pembersihan data dilakukan untuk menghapus beberapa variabel yang tidak dibutuhkan pada saat mencari korelasi antara satu variabel dengan lainnya seperti timestamp, nilai bit, key untuk membedakan satu data dengan data lainnya. Variabel yang digunakan sesuai dengan seluruh rangkaian percobaan yaitu kadar air dalam tanah, kelembaban udara, dan temperatur suhu ruang *greenhouse*. Bila ada nilai yang tidak sesuai, tidak terpakai, serta nilai yang kosong, format yang salah ataupun nilai yang *corrupted* dilakukan pada saat pembersihan data. Dari data yang telah dibersihkan, data tersebut dapat divisualisasikan untuk melihat adakah korelasi antara satu nilai dengan nilai lainnya.

Pemisahan data dan perubahan bentuk data dari dataframe menjadi dataset tensorflow merupakan langkah dalam pembuatan model, memisahkan data sebanyak 80% dan 20% secara acak menciptakan model tidak terpaku pada nilai yang spesifik berbeda pada saat data didapatkan sehingga terdapat diversifikasi antara data satu dan lainnya tanpa harus terpengaruh oleh waktu, urutan data secara kronologikal tidak dibutuhkan sehingga pemisahan ini bisa dilaksanakan serta hasil dari validasi data lebih realistis pada data yang dikumpulkan dan telah

dilatih modelnya. Perubahan menjadi dataset tensorflow memudahkan proses transformasi juga mengubah format data masuk sesuai dengan format Tensorflow.

Pre-processing feature merupakan salah satu langkah untuk menyamakan variabel satu dengan variabel lainnya diakibatkan perbedaan jangkauan data yang didapatkan pada saat pengambilan data, contohnya perbedaan jarak data antara temperatur ruang dengan kelembaban ruang *greenhouse*. Temperatur maksimal dari ruangan *greenhouse* tidak mempunyai nilai dengan jarak yang sama, sehingga diperlukan normalisasi semua variabel terukur dan kendali. Normalisasi tersebut memudahkan untuk pengolahan data karena setiap datanya berubah menjadi nilai dengan range yang sama.