

# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Kelapa sawit (*elaeis guineensis*) adalah tanaman industri sebagai bahan baku penghasil minyak masak, minyak industri, maupun bahan bakar. Beberapa permasalahan yang dihadapi pengusaha dan petani dalam pengembangan budidaya kelapa sawit adalah pengadaan bibit tanaman.

Pembibitan kelapa sawit adalah Langkah paling berpengaruh dalam proses pertumbuhan bibit dalam menentukan bagaimana kelapa sawit tumbuh sebelum menjadi tanaman produktif lapangan. Dalam kegiatan industri kelapa sawit khususnya pada tahap pembibitan dan penyiraman masih banyak menggunakan proses pembibitan atau pembiakan manual. Dimulai dengan pembibitan kelapa sawit dengan pengairan yang dilakukan secara manual menggunakan sambungan selang dengan penyiraman pagi dan sore, menyinari benih yang ditanam menggunakan sinar matahari langsung, yang menyebabkan Cahaya masuk yang berlebihan akan menyebabkan daun-daun bibit layu dan mati. Serta hawa yang kering disekitar media pembibitan yang terlampau kering dapat menyebabkan penguapan yang leih besar dan jika dalam waktu yang lama mengakibatkan Tanaman layu, sedangkan jika terlalu basah menyebabkan daun menjadi cenderung basah yang mengakibatkan pembusukan.

Berdasarkan Hasil penelitian, Temperatur optimal untuk pertumbuhan bibit sawit adalah 29°C - 30°C dengan tingkat kelembapan 80-90%, sedangkan kebutuhan penyinaran yang optimal pada rentang 5-7 jam/hari dengan intensitas 50-75% yang dimana akan berpengaruh pada pertumbuhan tinggi pohon dan pertumbuhan daun.

### **B. Rumusan Masalah**

Berdasarkan latar belakang yang telah diuraikan di atas, maka dibuatlah sistem budidaya kelapa sawit dengan pemantauan jarak jauh. Membuat perancangan suatu alat yang dapat memonitoring dan kontroling pada perangkat keras atau alat yang dibangun serta *interface* antar *user* dengan menggunakan aplikasi *Android* dan akses internet.

### **C. Batasan Masalah**

Penelitian ini memiliki beberapa batasan masalah yang bertujuan agar penulis terfokus pada aspek yang ada. Adapun batasan masalah dalam penelitian ini sebagai berikut:

1. Sistem ini dirancang untuk mengidentifikasi kelembapan udara menggunakan DHT22, kandungan air dalam kelembapan tanah menggunakan sensor *soil moisture*, dan cahaya menggunakan sensor *light dependent resistor* (LDR) terhadap tumbuh bibit kelapa sawit di fase pembibitan *pre-nusery*.
2. Sistem ini dibuat untuk melakukan kontrol terhadap *input* yang diberikan oleh ketiga sensor yang ada seperti sensor LDR yang mengontrol *grow up light* terhadap pencahayaan sebagai salah satu kebutuhan utama tumbuhan,

DHT22 yang melakukan kontrol terhadap *mist maker* terhadap kelembapan udara, dan *water pump* untuk penyiraman respon dari sensor *soil*.

3. Sistem ini diuji coba pada 2 *polybag* berisi bibit kelapa sawit berusia < 3 bulan.
4. Penelitian dilakukan dilahan pembibitan terbatas

#### **D. Tujuan Penelitian**

Adapun tujuan dari penelitian ini adalah untuk mewujudkan smartfarm yang dapat mengurangi kerugian waktu dan tenaga pada pembudidayaan kelapa sawit diakibatkan oleh bibit yang mati yang mana disebabkan tidak terkontrolnya pasokan cahaya, air, dan kelembapan udara yang tidak menentu.

#### **E. Manfaat Penelitian**

1. Manfaat bagi penulis

Menambah pengetahuan mengenai manfaat *Internet Of Things* (IOT) terhadap kehidupan terkhususnya sektor pertanian.

2. Manfaat bagi petani

Dapat membantu petani sawit dalam pemeliharaan pembudidayaan tanaman kelapa sawit.

3. Manfaat bagi Universitas

Penelitian rancangan ini diharapkan dapat digunakan sebagai referensi terhadap penelitian lanjut yang lebih mendalam pada masa yang akan datang.

## **F. Sistematika Penulisan**

Adapun sistematika penulisan pada Tugas Akhir ini yaitu:

### **BAB I PENDAHULUAN**

Dalam bab ini akan diuraikan alasan pemilihan judul

### **BAB II LANDASAN TEORI**

Dalam bab ini berisikan beberapa teori-teori yang mendukung dalam pembahasan penyusunan tugas akhir ini serta bahasa pemrograman yang digunakan sehingga memudahkan penulis dalam menyelesaikan masalah.

### **BAB III METODE PENELITIAN**

Pada bab ini menjelaskan mengenai tahapan - tahapan yang dilalui dalam penyelesaian tugas akhir ini, yaitu tempat penelitian dan waktu penelitian, metode pengumpulan data, alat dan bahan penelitian, tahap penelitian, metode pengujian serta gambaran desain sistem yang akan dirancang atau dibuat.

### **BAB IV ANALISIS DAN PERANCANGAN SISTEM**

Pada bab ini menjelaskan mengenai analisis sistem yang berjalan, analisis sistem yang diusulkan, rancangan sistem, rancangan input dan rancangan output.

### **BAB V PENGUJIAN SISTEM**

Pada bab ini menjelaskan mengenai metode pengujian dan teknik pengujian.

### **BAB VI PENUTUP**

Pada bab ini menjelaskan mengenai kesimpulan dan saran.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **A. Kajian Teori**

##### **1. Kelapa Sawit**

Sebagai negara yang dilalui garis khatulistiwa membuat Indonesia termasuk kedalam kategori beriklim tropis, wilayah negara dengan iklim tropis biasanya memperoleh sinar matahari sepanjang waktu. Ini yang membuat Indonesia Hanya memiliki dua musim, yaitu musim hujan dan kemarau. Hal inilah yang membuat negara Indonesia didominasi oleh petani karena curah hujan dan panas cenderung seimbang sehingga membuat tanaman tumbuh dengan baik salah satunya kelapa sawit yang memang dikenal sebagai tanaman tropis.

Kelapa sawit merupakan tanaman tropis yang dimanfaatkan sebagai tanaman industry penghasil minyak masak, minyak industry, maupun bahan bakar. Indonesia adalah penghasil minyak kelapa sawit terbesar di dunia. Di Indonesia penyebarannya berada di daerah Aceh, pantai timur Sumatera, Jawa, Kalimantan, dan Sulawesi. Terdapat beberapa spesies kelapa sawit yaitu *E. guineensis*, *E. oleifera*, dan *E. odora*. Varietas atau tipe kelapa sawit digolongkan berdasarkan dua karakteristik yaitu ketebalan endokarp dan warna buah. Kelapa sawit berbentuk pohon tingginya dapat mencapai 24 meter dari permukaan tanah dengan akar serabut mengarah ke bawah dan samping seperti terlihat pada Gambar 2.1. Seperti jenis tanaman palma lainnya daunnya tersusun majemuk

menyirip hanya saja dengan duri yang tidak terlalu tajam dan keras. Buah sawit mempunyai warna bervariasi dari hitam, ungu, hingga merah tergantung bibit yang digunakan.



**Gambar 2. 1** Kelapa Sawit

Sumber: (<https://bbibsingosari.ditjenpkh.pertanian.go.id/wp-content/uploads/2022/02/Perkebunan-Sawit.jpg>)

Proses pembudidayaan kelapa sawit dilakukan pembibitan sebelum dengan menggunakan biji kelapa sawit matang yang dapat menghasilkan kecambah kemudian tunas dan bakal akar pada plastic *polybag*. Menurut Sunarko(2009) pembibitan ada dua tahap yaitu pembibitan awal (*pre-nusery*) dan pembibitan utama (*main nusery*) Dalam proses *pre-nusery* dari biji hingga usia bibit 3 bulan yang dilakukan dilahan terbatas dan tertutup menggunakan rumah naungan yang dapat berupa paranet ataupun daun kelapa, fase ini juga termasuk yang paling penting karena meliputi masa awal pembiakan kelapa sawit sehingga segala hal terkecil harus diperhatikan mulai dari pasokan nutrisi, air, cahaya, dan hawa sekitar bibit. Adapun syarat tumbuh bibit pada fase *pre-nusery* yaitu

kebutuhan cahaya berkisar antara 50-70% dengan durasi 5-7 jam perhari yang berpengaruh dalam meningkatkan pertumbuhan perakaran dan tinggi pohon dimulai dari tunas, kemudian untuk kelembapan udara yang optimal berkisar antara 29-30°C.



**Gambar 2. 2** Manajemen pembibitan kelapa sawit *prenursery*

Sumber: (<https://docplayer.info/docs-images/108/174486007/images/38-0.jpg>)

kemudian tahap *main-Nursery* yaitu tahap lanjutan pembibitan terhadap bibit yang telah tumbuh diatas usia 3 bulan hingga 8 bulan pada lahan terbuka sebelum ditanam di lahan pertanian.

## 2. Pembibitan

Pembibitan (*nursery*) adalah salah satu upaya untuk memperbanyak tanaman dan merupakan kegiatan awal di lapangan yang bertujuan mempersiapkan bibit siap tanam. Dalam prakteknya pembibitan memiliki dua Teknik yaitu secara Generatif dan secara vegetative (Cahyono 2008), pembibitan generative merupakan pembibitan atau perbanyak tanaman dari hasil

perkawinan (seksual), bibit generative ini biasa dikenal dengan bibit dan biji, karena bibit ini diperoleh dengan cara menanam biji. Kemudian secara vegetative adalah pembibitan atau perbanyakkan tanaman dengan cara tertentu selain dari hasil perkawinan misalkan cangkok atau stek yang dimana diperoleh dari dahan atau ranting tanaman.

### 3. *Internet of Thing (IOT)*

*Internet of Thing (IoT)* merupakan sebuah konsep dimana suatu objek yang mempunyai kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan hubungan manusia ke manusia atau manusia ke komputer. Perkembangan IoT bisa dilihat mulai berdasarkan taraf konvergensi teknologi nirkabel, *microelectromechanical (MEMS)*, internet, & QR (*Quick Responses Code*). IoT juga tak jarang diidentifikasi menggunakan RFID (*Radio Frequency Identification*) menjadi metode komunikasi.

“A Things” dalam *Internet of Things* bisa didefinisikan sebagai subjek misalkan orang menggunakan monitor implant jantung, hewan peternakan menggunakan *transponder biochip*, sebuah mobil yang sudah dilengkapi built-in sensor buat memperingatkan pengemudi waktu tekanan ban rendah atau yang dikenal sebagai *tyre pressure monitoing*. Sejauh ini, IoT paling erat hubungannya dengan komunikasi *machine-to-machine (M2M)* pada bidang manufaktur & listrik, perminyakan, & gas. Produk dibangun menggunakan kemampuan komunikasi M2M yang tak jarang disebut sebagai sistem cerdas atau “smart”. Sebagai contoh yaitu *smart* kabel, *smart* meter, *smart grid sensor*.

Selain itu, juga meliputi teknologi berbasis sensor, seperti teknologi nirkabel *QR code* yang seringkali kita jumpai. Kemampuan IOT (*Internet of Things*) sendiri tidak perlu diragukan lagi. Berbagai teknologi yang sudah menerapkan system IOT, sebagai contoh sensor cahaya pada layar *smartphone*, sensor bunyi dari teknologi Google AI (*artificial intelegant*) dan *Amazon Alexa*.

Dan yang terbaru saat ini, penerapan *smart city* yang sudah dilakukan di kota-kota besar di berbagai negara maju. Sehingga, segala bentuk aktivitas masyarakat kota atau negara tersebut dapat termonitoring dengan baik oleh system dengan jaringan basis data berskala besar.

Penelitian pada IoT masih dalam tahap perkembangan. Oleh karena itu, tidak ada definisi dari Internet of Things. Menurut Ashton (2009) definisi awal IoT adalah *Internet of Things* memiliki potensi untuk mengubah dunia seperti pernah dilakukan oleh Internet.

Seperti namanya, *Internet of Things* bergantung pada internet untuk menjadi konektivitas antara sensor atau perangkat yang akan saling terkoneksi pada *cloud*. Data dari sensor yang akan dikirim ke *cloud* akan diproses oleh software yang akan menentukan aksi selanjutnya. *Action* ini bisa berbentuk pengiriman *alert*, penyesuaian jadwal, penutupan akses di alat, atau lainnya.

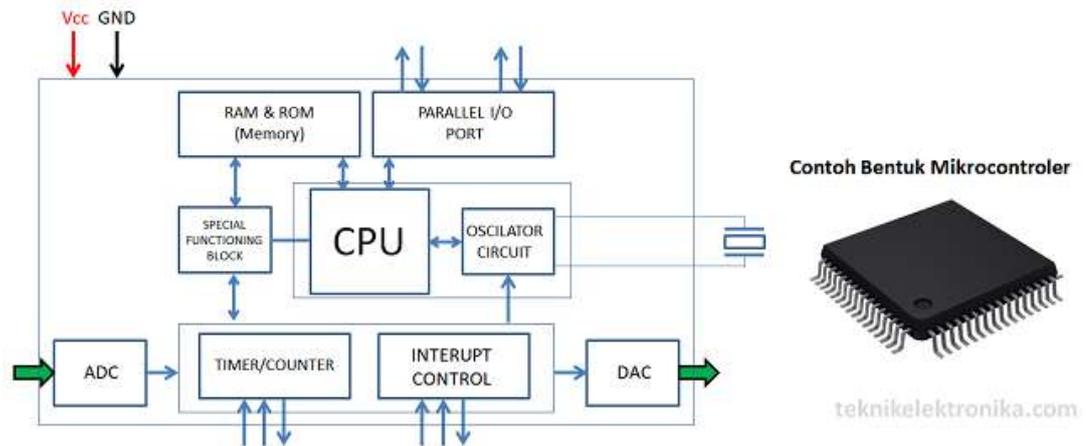
Cara kerja *internet of things* adalah memanfaatkan sebuah argumentasi dari algoritma bahasa pemrograman yang telah tersusun. Dimana, setiap argumen yang terbentuk akan menghasilkan sebuah interaksi yang akan membantu perangkat keras atau mesin dalam melakukan fungsi atau kerja. Sehingga, mesin

tersebut tidak memerlukan bantuan dari manusia lagi dan dapat dikendalikan secara otomatis. Faktor terpenting dari jalannya program tersebut terletak pada jaringan internet yang menjadi penghubung antar sistem dan perangkat keras. Tugas utama dari manusia adalah menjadi pengawas untuk memonitoring setiap tindakan dan perilaku dari mesin saat bekerja.

Kendala terbesar dari pengembangan *Internet of things* adalah dari sisi sumber daya yang cukup mahal, serta penyusunan jaringan yang sangat kompleks. Biaya pengembangan juga masih terlampau mahal dan tidak semua kota atau negara telah menggunakan IoT sebagai kebutuhan primer mereka.

#### **4. Mikrokontroler**

*Mikrokontroler* adalah computer *mikro* dalam satu *chip* tunggal. *Mikrokontroler* memadukan CPU, ROM, RWM, I/O parallel, I/O seri, *counter timer*, dan rangkaian *clock* dalam satu *chip* seperti pada gambar 2.3. Dengan kata lain, *mikrokontroler* adalah suatu alat elektronika digital yang mempunyai masukan serta keluaran dan kendali dengan program yang bias ditulis dan dihapus dengan cara khusus. Cara kerja *mikrokontroler* sebenarnya membaca dan menulis data (immerse 2014).



**Gambar 2. 3** Blok Diagram Mikrokontroler  
 Sumber: (<https://teknikelektronika.com>)

Sistem yang menggunakan *mikrokontroler* disebut sebagai *embedded system* atau *dedicated system*. *Embedded system* adalah sistem pengendali yang tertanam pada suatu produk, sedangkan *dedicated system* adalah sistem pengendali yang hanya dimaksudkan untuk suatu fungsi tertentu. Sebagai contoh printer adalah suatu *embedded system* karena didalamnya terdapat *mikrokontroler* sebagai pengendali dan juga *dedicated system* karena fungsi pengendali tersebut berfungsi hanya untuk menerima data mencetaknya. Hal ini berbeda dengan suatu PC yang dapat digunakan untuk berbagai macam keperluan, sehingga *mikroprosesor* pada PC sering disebut sebagai *general purpose microprocessor* (mikroprosesor serbaguna). Pada PC berbagai macam *software* yang disimpan pada media penyimpanan dapat dijalankan, tidak seperti mikrokontroler hanya terdapat satu *software* aplikasi (Mechatronicscrew 2011).

## 5. ESP32



**Gambar 2. 4** *Nodemcu ESP32*

*ESP32* adalah salah satu papan pengembangan yang dibuat oleh *Nodemcu* untuk evaluasi modul *ESP-WROOM-32*. Ini didasarkan pada mikrokontroler *ESP32* yang menawarkan *wifi*, *Bluetooth*, *Ethernet* dan daya rendah semua dalam satu chip.

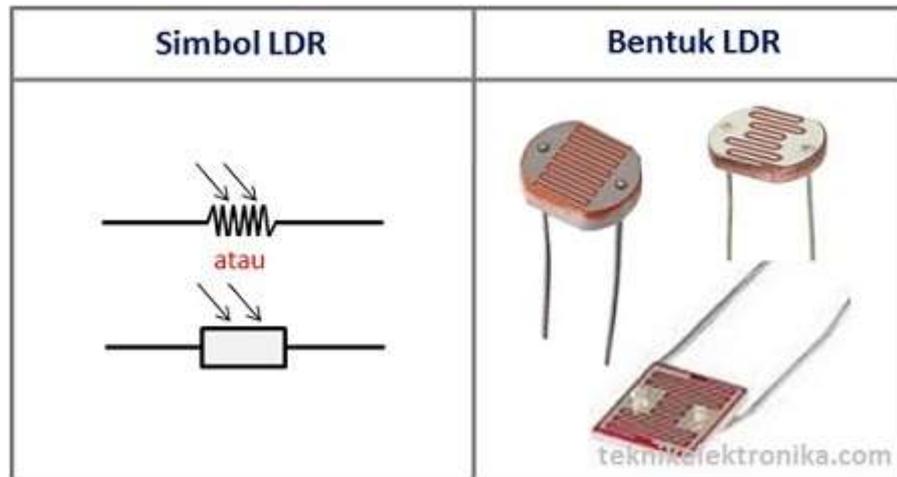
Daya ke *nodemcu ESP32* disuplai melalui konektor *USB Micro B On-board* atau langsung melalui pin *VIN*. Sumber Listrik dipilih secara otomatis.

Berikut fitur dan spesifikasi dari *Nodemcu ESP32*:

**Tabel 2. 1** Spesifikasi *nodemcu ESP32*

No.	Nama	Spesifikasi
1	Module Model	<i>ESP32</i>
2	Package	DIP-16
3	Size	27*40,5*4,5
4	SPI Flash	Default 32MBIT
5	RAM	Internal 320 kb + External 4M PSRAM

## 6. LDR (Light Dependent Resistor)



**Gambar 2. 5** Bentuk dan symbol dari *LDR*

Sumber (<https://teknikelektronika.com/pengertian-ldr-light-dependent-resistor-cara-mengukur-ldr/>)

*Light Dependent Resistor* atau disingkat dengan *LDR* adalah jenis Resistor yang nilai hambatan atau nilai resistansinya tergantung pada intensitas cahaya yang diterimanya. Nilai Hambatan *LDR* akan menurun pada saat cahaya terang dan nilai Hambatannya akan menjadi tinggi jika dalam kondisi gelap. Dengan kata lain, fungsi *LDR (Light Dependent Resistor)* adalah untuk menghantarkan arus listrik jika menerima sejumlah intensitas cahaya (Kondisi Terang) dan menghambat arus listrik dalam kondisi gelap.

Berikut merupakan tabel spesifikasi dari sensor *LDR*

**Tabel 2. 2** Spesifikasi Sensor *DHT22*

No.	Nama	Keterangan
1	Tegangan	3.3V – 5V
2	Output	Digital 0 dan 1
3	ukuran	3cm * 1.6cm

## 7. *Grow Light*



**Gambar 2. 6** Lampu *Growlight*

Lampu pertumbuhan *LED* dirancang untuk meniru sinar matahari yang memengaruhi proses fotosintesis tanaman. Mereka digunakan untuk penanaman dalam ruangan yang menyediakan pencahayaan yang cocok untuk budidaya. Karena lampu ini dikenal sebagai lampu tumbuh (*growlight*).

Lampu ini biasanya digunakan untuk budidaya Tingkat industri. Anda juga dapat menggunakannya untuk hortikultura, perbanyakan tanaman, berkebun dalam ruangan, produksi makanan, dan penggunaan rumah tangga. Namun, pilihan lain seperti lampu pijar, lampu pelepasan intensitas tinggi (*HID*), dan lampu neon juga digunakan selain *LED*. Namun teknologi *Light-Emitting Diodes* atau *LED* adalah yang paling populer karena menghasilkan yang paling tinggi *Photosynthetically Active Radiation (PAR)* dari setiap Cahaya. Ini menawarkan suhu warna yang bervariasi, spektrum Cahaya, intensitas, dan lain

sebagainya yang memfsilitasi pencahayaan ideal untuk berbagai jenis pertumbuhan tanaman.

Berikut Adalah spesikasi dari lampu tumbuh

**Tabel 2. 3** Spesifikasi Lampu Tumbuh

No	Nama	Spesifikasi
1	Model	Growlight E27
2	Daya	6 Watt
3	Jumlah LED	44 LED Merah + 16 LED Biru
4	Voltase	AC 85-265 Volt
5	Panjang Gelombang	625-630nm(Merah)+445-470nm(Biru)
6	Housing	Plastik

## 8. *DHT22*



**Gambar 2. 7** Sensor DHT22  
Sumber: (TokoTeknologi)

DHT22 adalah sensor digital kelembaban dan suhu relatif. Sensor DHT22 menggunakan kapasitor dan termistor untuk mengukur udara disekitarnya dan keluar sinyal pada pin data. DHT22 diklaim memiliki kualitas pembacaan yang baik, dinilai dari respon proses akuisisi data yang cepat dan ukurannya yang minimalis, serta dengan harga relatif murah jika dibandingkan dengan alat thermohyrometer.

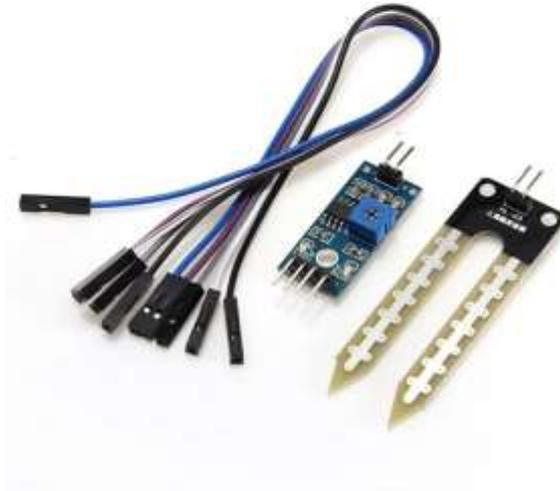
Sensor *DHT22* sangat mudah diaplikasikan pada *mikrokontroller* tipe Arduino karena memiliki tingkat stabilitas yang dapat dipercaya dan fitur kalibrasi yang memiliki hasil sangat akurat.

Berikut adalah spesifikasi dari sensor *DHT22*:

**Tabel 2. 4** Spesifikasi Sensor *DHT22*

Tegangan Kerja	3.3V – 5V
Arus Maksimun	2.5mA
Range Pengukuran Kelembapan	0% - 100%
Akurasi pengukuran kelembapan	2-5%
Range pengukuran suhu	40 °C – 80 °C
Akurasi Pengukuran Suhu	0.5 °C
Kecepatan Pengambilan Sampel	0.5 Hz
Ukuran	15.1 mm x 25 mm x 7.7 mm
Jumlah Pin	4 pin dengan jarak 0,1

## 9. Sensor *Soil Moisture*



**Gambar 2. 8** Sensor *soil moisture*

Sumber:

(<http://sc04.alicdn.com/kf/H9f1c550150e1456fa4af39144dc595b2n.jpg>)

Sensor *soil moisture* adalah sesuatu yang digunakan untuk mendeteksi adanya perubahan lingkungan fisik atau kimia. Variabel keluaran dari sensor yang diubah menjadi besaran listrik disebut transduser. Pada saat ini, sensor tersebut telah dibuat dengan ukuran sangat kecil dengan orde nano meter. Ukuran yang sangat kecil ini sangat memudahkan pemakaian dan menghemat energi. *Soil moisture sensor* YL-69 adalah sensor kelembaban yang dapat mendeteksi kelembaban dalam tanah. Sensor ini sangat sederhana, tetapi ideal untuk memantau taman kota, atau tingkat air pada tanaman pekarangan. Sensor ini terdiri dua probe untuk melewatkan arus melalui tanah, kemudian membaca resistansinya untuk mendapatkan nilai tingkat kelembaban. Semakin banyak air membuat tanah lebih mudah menghantarkan listrik (resistansi kecil), sedangkan tanah yang kering sangat sulit menghantarkan listrik (resistansi besar). Sensor ini

sangat membantu untuk mengingatkan tingkat kelembaban pada tanaman atau memantau kelembaban tanah.

**Tabel 2. 5** Spesifikasi Sensor Soil Moisture

Tegangan Input	3.3V atau 5V
Tegangan Output	0-4,2V
Arus	35mA
Value Range	1024 bit

#### 10. *Ultrasonic mist maker*



**Gambar 2. 9** *Ultrasonic Mist Maker*

*Ultrasonic mist maker* merupakan modul yang berfungsi untuk menghasilkan uap air atau kabut. *Mist Maker* memanfaatkan osilasi listrik pada frekuensi ultrasonik. keramik perubahan osilasi listrik menjadi osilasi mekanik, yang menciptakan kabut dan semprotan air. mister akan dimatikan secara otomatis jika air terlalu rendah. 1-1 / 2 "sampai 2" air adalah kedalaman operasi yang ideal, meskipun akan bekerja pada sampai dengan 4 "air. Fogger ultrasonically akan menguap sekitar 1 liter air di 4 sampai 5 jam. Untuk penggunaan di dalam ruangan. Tingkat air: approx. 80ml per jam.

Dan berikut adalah spesifikasi dari mist maker

**Tabel 2. 6** Spesifikasi *Ultrasonic Mist Maker*

NO.	Nama	Ketereangan
1	Nama Modul	<i>Ultrasonic mist maker</i>
2	Tingkat aliran Udara	25ml/jam
3	Tingkat kebisingan	<35 db
4	Kapasitas air	380ml/jam
5	Voltase	24 volt, 1 Ampere
6	Dimensi	3.5cm*32mm

#### 11. *Mini Submersible Water Pump DC 3V - 5V*

Merupakan pompa air mini yang ditenagai oleh motor DC. Pompa air mini ini mampu memompa air sebanyak 240 liter per jam (240 L/h) dengan rentang tegangan kerja 3 - 5 VDC. Pompa air mini ini sangat cocok untuk akuarium, sistem pengairan tanaman hidroponik, robotika, perangkat output mikrokontroler, dan lain sebagainya.



**Gambar 2. 10** *Water pump*

Dan berikut adalah table spesifikasi dari *Mini Submersible Water Pump DC 3V - 5V*

**Tabel 2.4** Spesifikasi *Mini Submersible Waterpump*

Tegangan kerja	3-5V DC
Konsumsi arus	120-330mA
Konsumsi daya	0,4-1,5w
Diameter, Panjang, tinggi pompa	24mm, 45mm, 33,mm

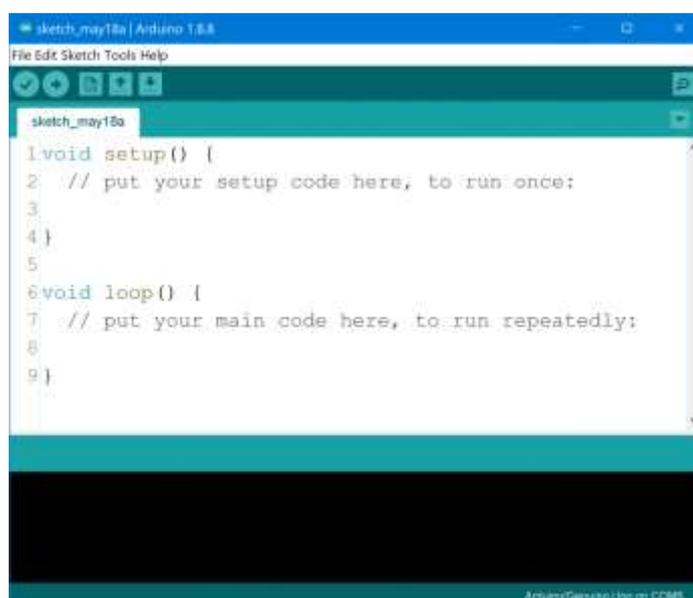
## 12. *Arduino Ide*

Menurut Djuandi. (2015). *Arduino Integreted Development Environment (IDE)* adalah “Software yang sangat canggih ditulis menggunakan Java”.

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

*Arduino IDE (Integreted Development Environment)* merupakan sebuah perangkat lunak yang berfungsi sebagai wadah untuk melakukan pemrograman dalam bentuk Bahasa . *Arduino IDE* ini dibuat dari Bahasa pemrograman Java

yang sudah dilengkapi dengan library C/C++ yang biasa disebut wiring yang membuat operasi input dan output menjadi lebih mudah. *Arduino IDE* juga dapat disebut sebagai software untuk mendesain sebuah fungsi-fungsi yang akan dituangkan kedalam perangkat keras. *Arduino IDE* ini sendiri sangat populer dan banyak digunakan oleh pengembang untuk melakukan perancangan sederhana hingga kompleks sekalipun. Seperti pada gambar 2.8 dibawah ini berikut adalah tampilan dari *Arduino IDE*.

The image shows a screenshot of the Arduino IDE software interface. The window title is "sketch\_may18a | Arduino 1.8.8". The menu bar includes "File", "Edit Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for file operations and execution. The main editor area shows a sketch named "sketch\_may18a" with the following code:

```
1 void setup() {  
2   // put your setup code here, to run once:  
3  
4 }  
5  
6 void loop() {  
7   // put your main code here, to run repeatedly:  
8  
9 }
```

The bottom of the IDE is a dark panel, likely for the serial monitor or console output.

**Gambar 2. 11** Tampilan *Arduino IDE*

Pada gambar 2.9 diatas terlihat tampilan *Arduino IDE* yang isinya terdapat 2 buah fungsi yang sudah tersedia yaitu, *void setup*, dan *void loop*. *Void setup* merupakan sebuah fungsi untuk meng-inisialisasi node-node yang akan digunakan. Sedangkan *void loop* merupakan sebuah paragraph untuk mengatur program agar melakukan aksi terhadap node-node yang digunakan atau memberi perintah untuk melakukan fungsi tertentu.

### 13. *Android*

*Android* adalah sistem operasi yang memang khusus dirancang untuk smartphome dan tablet. Sistem *Android* ini memiliki basis Linux yang mana dijadikan sebagai pondasi dasar dari sistem operasi Android. Linux sendiri merupakan sistem operasi yang memang khusus dirancang untuk komputer.

*Android* memang dirancang untuk dipasang pada perangkat-perangkat mobile touchscreen ( *smartphone* dan tablet). Sehingga sistem operasi yang berada di dalam *smartphone* saat ini memang menyesuaikan dari spesifikasi kelas low-end hingga *high-end*. Sehingga perkembangan sistem android memang cukup meningkat tajam.

*Android* merupakan sistem operasi yang terbuka (*open source*) yang mana berarti jika pihak Google memperbolehkan dan membebaskan bagi pihak manapun untuk dapat mengembangkan sistem operasi tersebut. Bahkan anda sendiri pun juga dapat mengembangkan sistem android yang memang sesuai dengan keinginan anda.

Uniknya, *Android* menggunakan nama-nama makanan untuk membedakan versi sistem android yang diluncurkannya. *Android* menggunakan huruf depan dari nama makanan tersebut sebagai penanda peningkatan versi sistemnya. Mulai dari *Cupcake Android* 1.5 (C), *Donuts Android* 1.6 (D), *Éclair Android* 2.0-2.1 (E) atau *Marshmallow Android* 6.0 (M).

#### 14. *Flutter*

*Flutter* adalah platform yang digunakan para developer untuk membuat aplikasi multiplatform hanya dengan satu basis coding (*codebase*). Artinya, aplikasi yang dihasilkan dapat dipakai di berbagai platform, baik mobile Android, iOS, web, maupun desktop. *Flutter* memiliki dua komponen penting, yaitu, *Software Development Kit (SDK)* dan juga *framework user interface*.

- a. *Software Development Kit (SDK)* merupakan sekumpulan tools yang berfungsi untuk membuat aplikasi supaya bisa dijalankan di berbagai platform.
- b. *Framework UI* merupakan komponen UI, seperti teks, tombol, navigasi, dan lainnya, yang dapat Anda kustomisasi sesuai kebutuhan.

*Flutter* juga merupakan platform yang gratis dan open source. Jika Anda ingin menggunakan *Flutter*, Anda perlu mempelajari bahasa pemrograman Dart. Berbeda dengan framework front-end pada umumnya yang menggunakan JavaScript sebagai bahasa pemrogramannya.

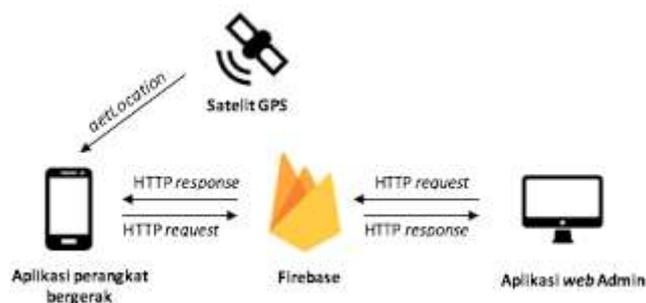
Sebelum resmi diluncurkan pada Desember 2018, *Flutter* sebenarnya telah dikembangkan oleh Google sejak 2015. Lalu, di tahun 2019, popularitas *Flutter* mulai meroket dan banyak developer berbondong-bondong menggunakan *Flutter*.

## 15. *Firebase*



**Gambar 2. 12** Logo *Firebase*

*Firebase* adalah penyedia layanan *Realtime database* dan *backend* sebagai layanan. Suatu aplikasi yang memungkinkan pengembang membuat API untuk disinkronisasikan untuk client yang berbeda-beda dan disimpan pada *cloud*-nya. *Firebase* memiliki banyak librari yang memungkinkan untuk mengintegrasikan layanan ini dengan *Android*, *IOS*, *JavaScript*, *Java*, *Objective-C* dan *Node.JS*. Database *Firebase* juga bisa diakses lewat *REST API*. *REST API* tersebut menggunakan *protocol Server-Sent Event* dengan membuat koneksi *HTTP* untuk menerima push *Notification* dari server. Kemampuan lain dari *firebase Realtime Database* adalah tetap progresif bahkan saat *offline* karena *SDK Firebase Realtime Database* Menyimpan data langsung ke *Disk Device* atau memori lokal. Setelah perangkat terhubung kembali dengan internet, perangkat pengguna atau user akan menerima berbagai perubahan yang terjadi.



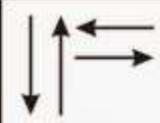
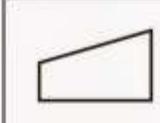
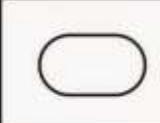
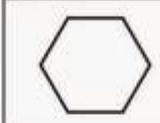
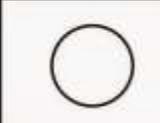
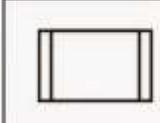
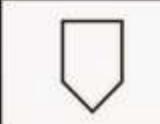
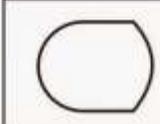
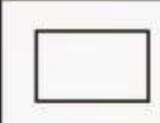
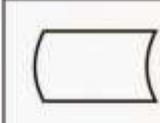
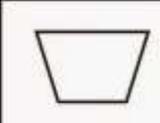
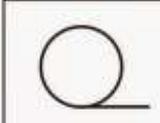
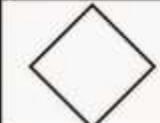
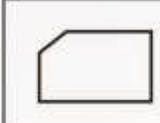
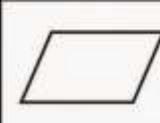
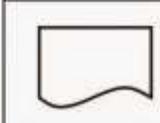
**Gambar 2. 13** Sistem kerja *Firebase*

## 16. *Flowchart*

*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* membantu analisis dan *programmer* untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternative-alternatif lain dalam pengoperasian. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut. *Flowchart* adalah bentuk gambar atau diagram yang mempunyai aliran satu atau dua arah secara sekuensial. *Flowchart* harus bias merepresentasikan maupun mendesain program. Oleh karena itu *flowchart* harus bias mempresentasikan komponen-komponen dalam Bahasa pemrograman.

*Flowchart* adalah suatu bagan dengan symbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses atau intruksi dengan proses lainnya dalam suatu program.

Berikut ini adaah symbol yang digunakan dalam menggambar suatu *flowchart*:

	<b>Flow Direction symbol</b> Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga connecting line.		<b>Simbol Manual Input</b> Simbol untuk pemasukan data secara manual on-line keyboard.
	<b>Terminator Symbol</b> Yaitu simbol untuk permulaan (start) atau akhir (stop) dari suatu kegiatan		<b>Simbol Preparation</b> Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam storage.
	<b>Connector Symbol</b> Yaitu simbol untuk keluar - masuk atau penyambungan proses dalam lembar / halaman yang sama.		<b>Simbol Predefine Proses</b> Simbol untuk pelaksanaan suatu bagian (sub-program)/prosedure
	<b>Connector Symbol</b> Yaitu simbol untuk keluar - masuk atau penyambungan proses pada lembar / halaman yang berbeda.		<b>Simbol Display</b> Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.
	<b>Processing Symbol</b> Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer		<b>Simbol disk and On-line Storage</b> Simbol yang menyatakan input yang berasal dari disk atau disimpan ke disk.
	<b>Simbol Manual Operation</b> Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh computer		<b>Simbol magnetik tape Unit</b> Simbol yang menyatakan input berasal dari pita magnetik atau output disimpan ke pita magnetik.
	<b>Simbol Decision</b> Simbol pemilihan proses berdasarkan kondisi yang ada.		<b>Simbol Punch Card</b> Simbol yang menyatakan bahwa input berasal dari kartu atau output ditulis ke kartu
	<b>Simbol Input-Output</b> Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya		<b>Simbol Dokumen</b> Simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau output dicetak ke kertas.

**Gambar 2. 14** Simbol *Flowchart*

### 17. UML (*Unified Modeling Language*)

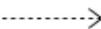
*Unified Modeling Language* merupakan alat perancangan sistem yang berorientasi pada objek. Secara filosofi kemunculan *UML* diilhami oleh konsep yang telah ada yaitu konsep permodelan *Object Oriented* (OO), sebab konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh obyek serta digambarkan atau dinotasikan dalam simbol-simbol yang cukup spesifik maka

*Object Oriented* memiliki proses standar dan bersifat independen (Haviluddin 2011).

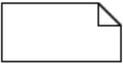
*Unified Modelling Language* (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual. Juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek.

Adapun daftar symbol *UML* yaitu :

**Tabel 2.5** *Simbol Use Case Diagram*

NO.	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

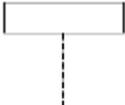
Lanjutan Tabel 2.5 Simbol Use Case Diagram

8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Tabel 2.6 Simbol Class Diagram

NO.	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Tabel 2.7 Simbol Sequence Diagram

NO.	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Tabel 2.8 Simbol State Chart Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali
3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2.9 Simbol Actifity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actifity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain

Lanjutan Tabel 2.9 Simbol Activity Diagram

2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

## 18. Kerangka Berpikir Ilmiah

Untuk lebih memperjelas kerangka berpikir yang akan ditulis dalam penelitian akhir ini. Maka berikut ini digambarkan kerangka pikir tersebut

Hingga saat ini pembibitan kelapa sawit prenuseri seringkali mengalami gagal tumbuh atau tumbuh dengan baik disebabkan factor cuaca yang tidak menentu semisal panas yang berlebih, kualitas udara yang terlalu kering, media tanam berupa tanah terlalu basah. Yang dimana akan mengakibatkan kerugian materi tenaga bahkan waktu.



Tingkat kelembapan tanah, intensitas cahaya yang stabil, dan suhu yang normal sangat berpengaruh terhadap tumbuh dari bibit sawit. Sebab jika salah satunya kurang akan menyebabkan pertumbuhan bibit kelapa sawit yang tidak normal bahkan gagal.



Untuk itu dibutuhkan sebuah sistem yang bias merekayasa cahaya, suhu, dan kandungan air yang dapat dikontrol dan dimonitor. Maka dari itulah dilakukan pembuatan **“RANCANG BANGUN KONTROL DAN MONITORING PEMBIBITAN KELAPA SAWIT DIMASA TANAM PRENUSERY”** untuk meminimalisir problem yang ada.



Diharapkan sistem ini bisa membantu petani dalam mengontrol dan memonitor darimana saja lahan pembibitan kelapa sawit prenusery yang dimiliki guna meminimalisir yang ada.

### **B. Kajian Hasil Penelitian Terdahulu**

1. Ahmad Ramadhan A, Mahasiswa Fakultas Teknik Prodi Teknik Informatika Universitas Muhammadiyah Parepare (2015) berhasil membuat dan menyelesaikan penelitian dengan judul “Alat Pengukur Kelembapan Tanah Berbasis Mikrokontroller”.
2. Gerry Dwi Utomo, Dedi Triyanto, Uray Ristian Mahasiswa Jurusan Rekayasa Sistem Komputer Fakultas MIPA Universitas Tanjungpura (2021) berhasil melakukan penelitian dan menerbitkan jurnal dengan judul “Sistem Monitoring dan Kontrol Pembibitan Kelapa Sawit Berbasis *Internet Of Things*” menggunakan *Capacitive Soil Moisture Sensor* untuk mendeteksi kadar air dalam tanah dan kelembapan tanah dan Sensor pH tanah untuk mendeteksi tingkat keasaman tanah.
3. Anis Tatik Maryani Dosen Fakultas Pertanian Universitas Jambi (2012) berhasil menerbitkan artikel dengan judul “Pengaruh Volume Pemberian Air Terhadap Pertumbuhan Bibit Kelapa Sawit Di Pembibitan Utama”.

Berdasarkan penelitian yang diuraikan di atas, maka dari itu peneliti bermaksud membuat penelitian membuat sistem pemeliharaan pembibitan Kelapa

sawit dengan memonitoring dan mengontrol Kelembapan udara dan tanah serta mengontrol kebutuhan air bibit kelapa sawit.

### C. Metode Pengujian

Beberapa *test-case* harus dilaksanakan dengan beberapa perbedaan strategi, *query*, atau jalur navigasi yang mewakili penggunaan sistem yang *typical*, kritis atau abnormal. Isu kunci pada pengembangan sistem adalah pemilihan sekelompok *test-case* yang cocok, sekecil dan secepat mungkin, untuk meyakinkan perilaku sistem secara detail. Pengujian harus mencakup *unit testing*, yang mengecek validasi dari prosedur dan fungsi-fungsi secara independen dari komponen sistem yang lain. Kemudian modul testing harus menyusul dilakukan untuk mengetahui apakah penggabungan beberapa unit dalam satu modul sudah berjalan dengan baik, termasuk eksekusi dari beberapa modul yang saling berelasi, apakah sudah berjalan sesuai karakteristik sistem yang diinginkan.

Jika struktur kendali antar modul sudah terbukti bagus, maka pengujian yang tak kalah pentingnya adalah pengujian unit. Pengujian unit digunakan untuk menguji setiap modul untuk menjamin setiap modul menjalankan fungsinya dengan baik. Ada 2 metode untuk melakukan unit testing, yaitu:

#### 1. White Box Testing

Coba *white box testing* merupakan metode perancangan *testcase* yang menggunakan struktural untuk mendapatkan *testcase*, test ini digunakan untuk meramal cara kerja perangkat lunak secara rinci kepada *logic path* (jalur logika), perangkat lunak di tes dengan kondisi dan perulangan secara fisik.

Contoh pengujian *white box testing* ini merupakan peringatan ketika user menginputkan password user yang salah, untuk kesalahan semacam ini akan memberikan suatu informasi kepada user mengenai kesalahan yang di lakukan.

## **2. Black Box**

Berdasarkan hasil uji coba yang dilakukan, seluruh navigasi dan tombol fasilitas program lainnya serta proses yang di jalankan tidak terjadi kesalahan, tetapi aplikasi mempunyai aturan-aturan yang sudah di tetapkan dan harus di ikuti karena apabila di hiraukan maka sistem akan menolak perintah yang tidak sesuai seperti kesalahan ketika user belum menginput data yang harusnya di input sesuai ketentuan sistem yang di jalankan dan sistem memberikan informasi kepada user karena data yang ingin diproses belum lengkap atau tidak memenuhi ketentuan untuk proses selanjut

## **BAB III**

### **METODE PENELITIAN**

#### **A. Jenis Penelitian**

Dalam pembuatan skripsi ini digunakan metode deskriptif yang menggambarkan fakta-fakta dan informasi secara sistematis, faktual dan akurat. Penelitian ini dilakukan melalui internet yang dapat memberikan sumber data dan pengetahuan mengenai system yang diteliti, kemudian mencocokkan dengan kemungkinan yang terjadi dalam usaha penyelesaian masalah.

#### **B. Lokasi dan Waktu Penelitian**

Lokasi dan waktu penelitian dalam penyusunan ini dilaksanakan di Kecamatan Sampaga Kabupaten Mamuju dalam waktu  $\pm 1$  bulan terhitung setelah sistem dibuat dan masa pembibitan prenusery.

#### **C. Alat dan Bahan**

Dalam pelaksanaan penelitian ini digunakan beberapa alat dan bahan yang terdiri dari Perangkat Keras (*Hardware*) dan Perangkat Lunak (*Software*).

### 1. Perangkat keras

Perangkat keras yang akan digunakan pada penelitian ini dapat dilihat pada table dibawah ini:

**Tabel 3.1** Spesifikasi Perangkat Keras

NO.	Spesifikasi	
1	Merk Laptop	Lenovo Ideapad 330
2	<i>Processor</i> Laptop	7 <sup>th</sup> Generation AMD A9-9425 APU
3	<i>Memory</i> Laptop	8GB
4	<i>Mikrokontroller</i>	ESP32-Cam
5	Jenis Sensor	<i>LDR, Soil Moisture, dht22</i>

### 2. Perangkat Lunak

Perangkat lunak yang digunakan untuk membuat aplikasi Kontrol dan Monitoring Pembibitan Kelapa Sawit berbasis *IoT* dapat dilihat pada table dibawah ini:

**Tabel 3.2** Spesifikasi Perangkat Lunak

No.	Spesifikasi	
1	Sistem Operasi	<i>Windows 10</i>
2	Tool Pemrograman	Arduiono IDE,Flutter
3	Bahasa Pemrograman	
4	<i>Database</i>	<i>Firebase</i>

### 3. Perangkat *Mobile*

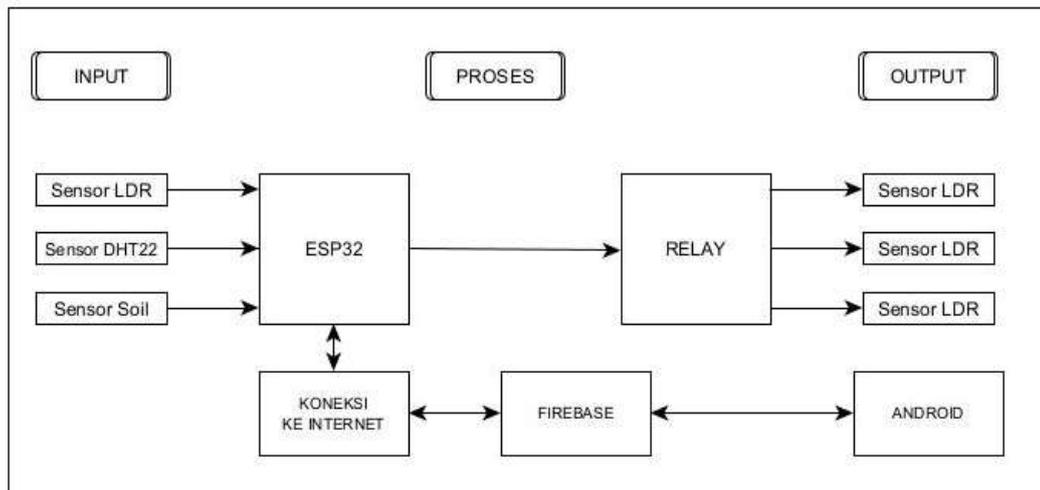
Perangkat *Android* mobile yang digunakan untuk menginstal dan menjalankan Aplikasi-aplikasi spesipikasinya dapat dilihat pada table berikut ini:

**Tabel 3.3** Spesifikasi Perangkat Mobile

No.	Spesifikasi	
1	Model	Samsung A23
2	OS	Android 12
3	RAM	6GB
4	Resolusi	90Hz
5	CPU	Snapdragon 680
6	LCD	6.6 inch 1080 x 2408 (FHD+)

#### D. Rancangan Sistem

Perancangan *Monitoring* dan Kontrol Pembibitan Kelapa Sawit Dimasa Tanam Prenusery ini meliputi perancangan perangkat keras dan perancangan perangkat lunak. Adapun blok diagram dari sistem yang akan dibuat dapat dilihat pada gambar dibawah.

**Gambar 3. 1** Diagram blok

##### a. Modul Sensor *LDR*

Berfungsi untuk membaca intensitas cahaya yang mengarah pada lahan pembibitan tertutup.

##### b. Modul Sensor *DHT22*

Berfungsi untuk membaca suhu dan kelembaban pada sekitar pohon bibit kelapa sawit.

**c. Modul Sensor Soil**

Berfungsi untuk membaca nilai kelembaban tanah pada *polybag* bibit kelapa sawit.

**d. Nodemcu ESP32**

Berfungsi sebagai *Mikrokontroller* untuk memproses input dari sensor yang kemudian menghasilkan output seperti yang kita inginkan.

**e. Modul Relay**

Berfungsi Sebagai pemutus dan penyambung aliran listrik.

**f. Growlight**

Berfungsi untuk menyinari lahan pembibitan.

**g. Mist Maker**

Berfungsi untuk menaikkan kelembaban udara dan menurunkan suhu lahan pembibitan.

**h. Waterpump**

Berfungsi untuk menyiram tanah guna menaikkan kelembaban tanah.

**i. Google Firebase**

Sebagai database untuk menyimpan data nilai sensor dan status *Relay*.

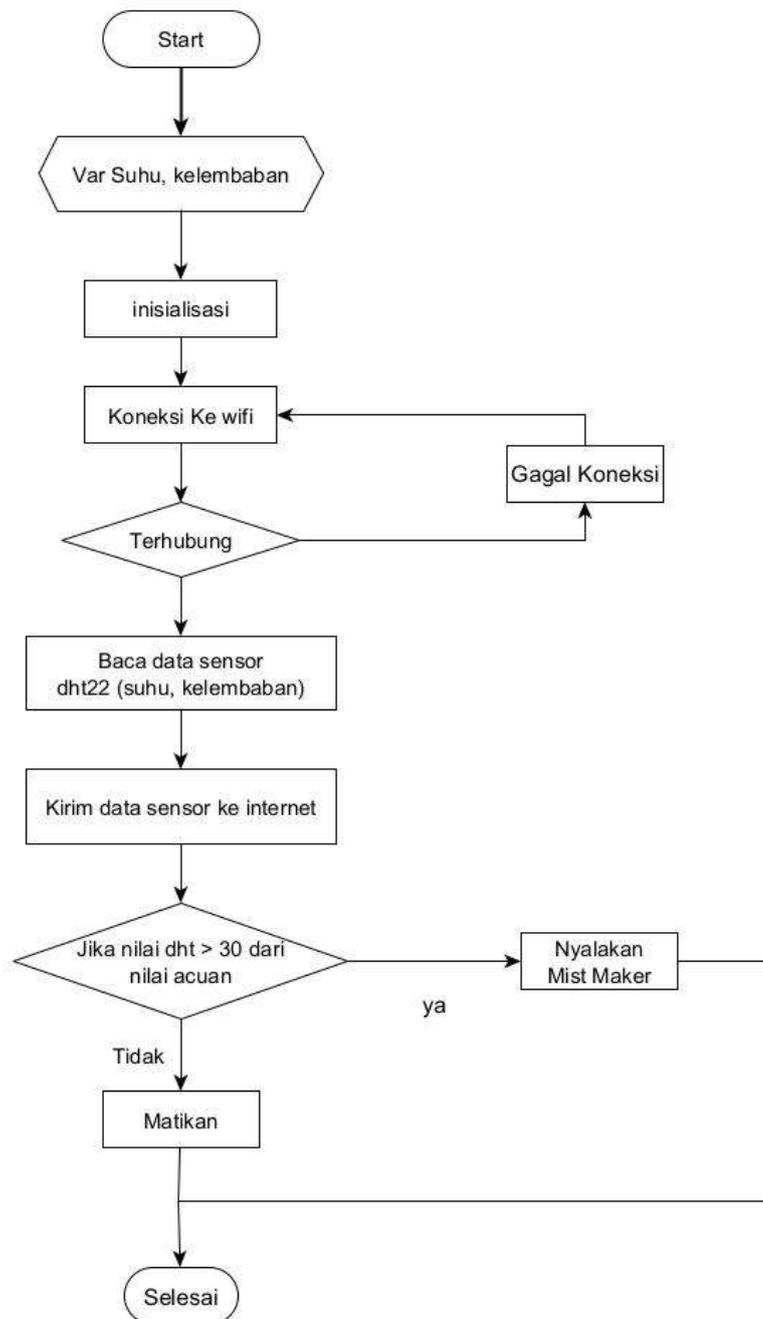
**j. Perangkat Android**

Berfungsi sebagai media untuk *memonitoring* sistem ini.

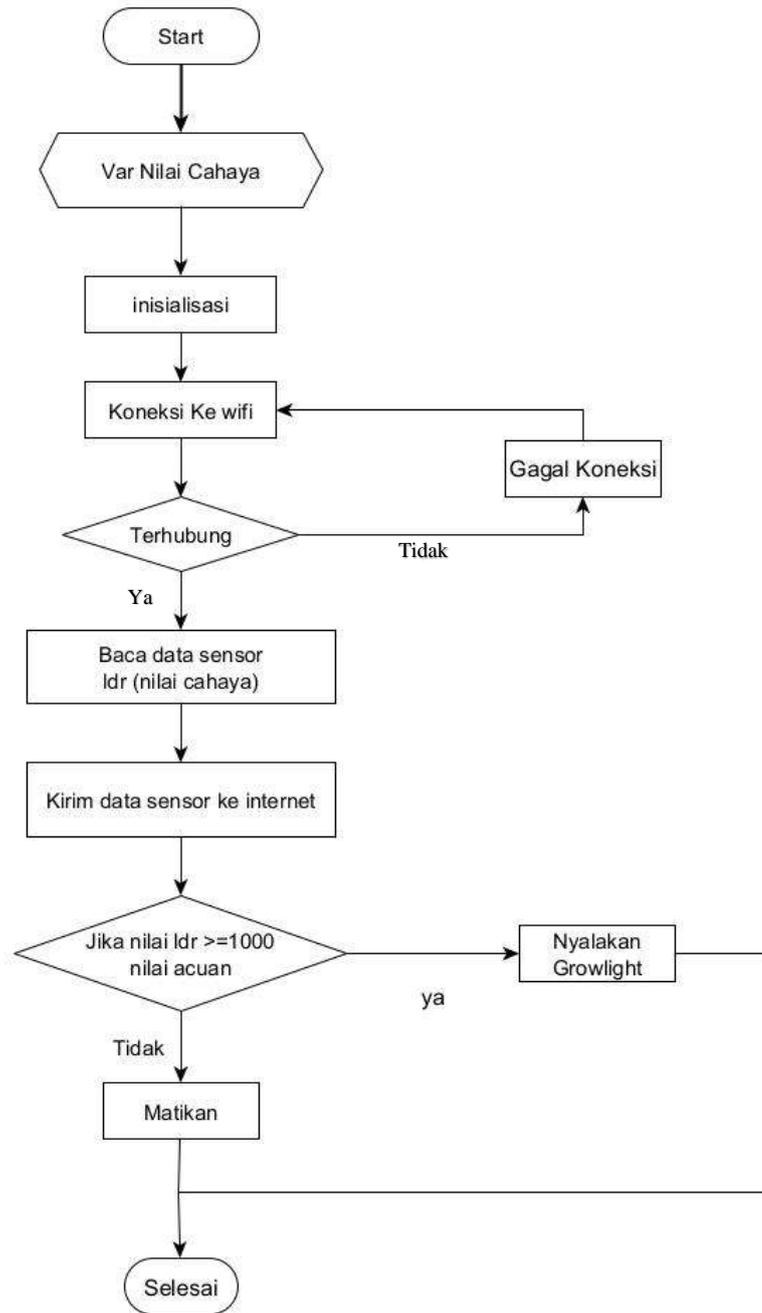
**1. Perancangan Perangkat Keras**

Perancangan perangkat keras sistem Rancang Bangun Kontrol dan Monitoring Pembibitan Kelapa Saeit ini menjadi bagian yang sangat penting dilakukan dalam suatu pembuatan alat karena dengan merancang terlebih dahulu dengan komponen yang tepat akan meminimalisir

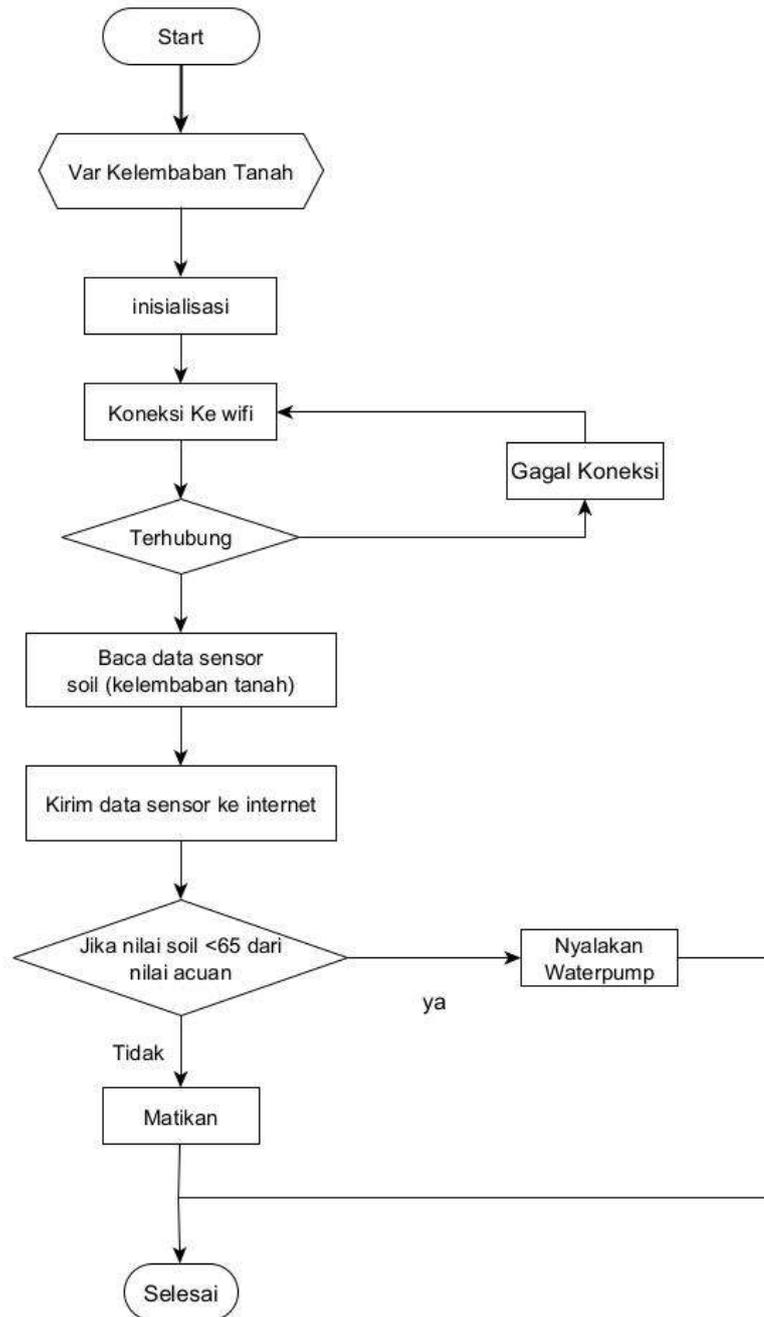
pembelian komponen yang berlebihan dan kerja alat dapat sesuai dengan apa yang diinginkan.



**Gambar 3. 2** Flowchart DHT22



**Gambar 3.3** *Flowchart* LDR sensor



**Gambar 3. 4** *Flowchart soil sensor*

Penjelasan Pada *flowchart* menjelaskan sistem bekerja pada sensor, dimulai dari sistem mengecek settingan waktu yang diatur oleh *user*, lalu sensor kelembaban yang mendeteksi tingkat kelembaban tanah yang akan di deteksi dan sensor *LDR* untuk mendeteksi tingkat intensitas cahaya terang dan gelap yang diterima dan sensor *dht22* untuk mendeteksi suhu dan kelembaban udara. Setelah semua data yang diperoleh dari kedua sensor akan tersimpan ke *database* kemudian hasil dari data yang diperoleh dari sensor dapat dimonitoring melalui aplikasi dengan bantuan *esp-32* agar data yang diperoleh bisa langsung ditampilkan ke aplikasi pada handphone. Jika hasil deteksi monitoring kelembaban tanah tidak normal atau kurangnya air maka pompa air secara otomatis akan aktif dan mengalirkan air pada bibit tanaman kelapa sawit tersebut. Sensor *LDR* akan memantau tingkat intensitas cahaya terang dan gelap yang diterima oleh sensor, jika intensitas cahaya yang di baca oleh sensor rendah atau gelap maka akan mengendalikan lampu *grow light* yakni dengan mengaktifkan lampu melalui aplikasi, dan jika cahaya yang diterima terang maka akan menonaktifkan lampu serta jika hasil pembacaan nilai dari suhu dan kelembaban udara pada sensor *dht22* kurang dari minimum suhu yang ditentukan maka akan mengaktifkan *mist maker* untuk menstabilkan suhu dan kelembaban bibit.

## **2. Rancangan Perangkat Lunak**

Perancangan perangkat lunak dimulai dari pembuatan *Flowchart* untuk program pada *ESP32* serta program monitor sistem.

## **E. Metode Pengumpulan Data**

Metode Pengumpulan data yang tepat yaitu dengan mempertimbangkan penggunaannya berdasarkan jenis data dan sumbernya. Data yang *objektifi* dan relevan dengan pokok permasalahan penelitian merupakan indikator keberhasilan suatu penelitian. Pengumpulan data penelitian ini dilakukan dengan cara sebagai berikut:

### **1. Observasi**

Merupakan metode pengumpulan data dengan cara mengadakan pengamatan langsung kepada objek penelitian yaitu dengan mengunjungi dan mengamati secara langsung lahan pembibitan kelapa sawit pase *pre-nusery*.

### **2. Wawancara**

Merupakan teknik pengumpulan data dengan cara mengadakan tanya jawab atau wawancara langsung kepada narasumber. Dalam penelitian ini, peneliti melakukan pengumpulan data dengan mewawancarai langsung pemilik lahan pertanian dan pemilik lahan pembibitan kelapa sawit.

### **3. Studi Pustaka**

Mengumpulkan data dengan mempelajari masalah yang berhubungan dengan objek yang diteliti, bersumber dari buku-buku pedoman, literatur yang disusun oleh para ahli untuk melengkapi data yang diperlukan dalam penelitian baik secara offline maupun online.

## F. Metode Pengujian

Setelah tahap perancangan *protoype* dibuat serta pemasangan perangkat yang dibutuhkan telah terpasang, selanjutnya dilakukan metode pengujian *white box* dan *black box*.

### a) White Box Testing

Coba *white box testing* merupakan metode perancangan *testcase* yang menggunakan struktural untuk mendapatkan *testcase*, test ini digunakan untuk meramal cara kerja perangkat lunak secara rinci kepada *logic fuzzy*, perangkat lunak di tes dengan kondisi dan perulangan secara fisik.

### b) Black Box

Berdasarkan hasil uji coba yang dilakukan, seluruh navigasi dan tombol fasilitas program lainnya serta proses yang di jalankan tidak terjadi kesalahan, tetapi aplikasi mempunyai aturan-aturan yang sudah di tetapkan dan harus di ikuti karena apabila di hiraukan maka sistem akan menolak perintah yang tidak sesuai seperti kesalahan ketika user belum menginput data yang harusnya di input sesuai ketentuan sistem yang di jalankan dan sistem memberikan informasi kepada user karena data yang ingin diproses belum lengkap atau tidak memenuhi ketentuan untuk proses selanjutnya.

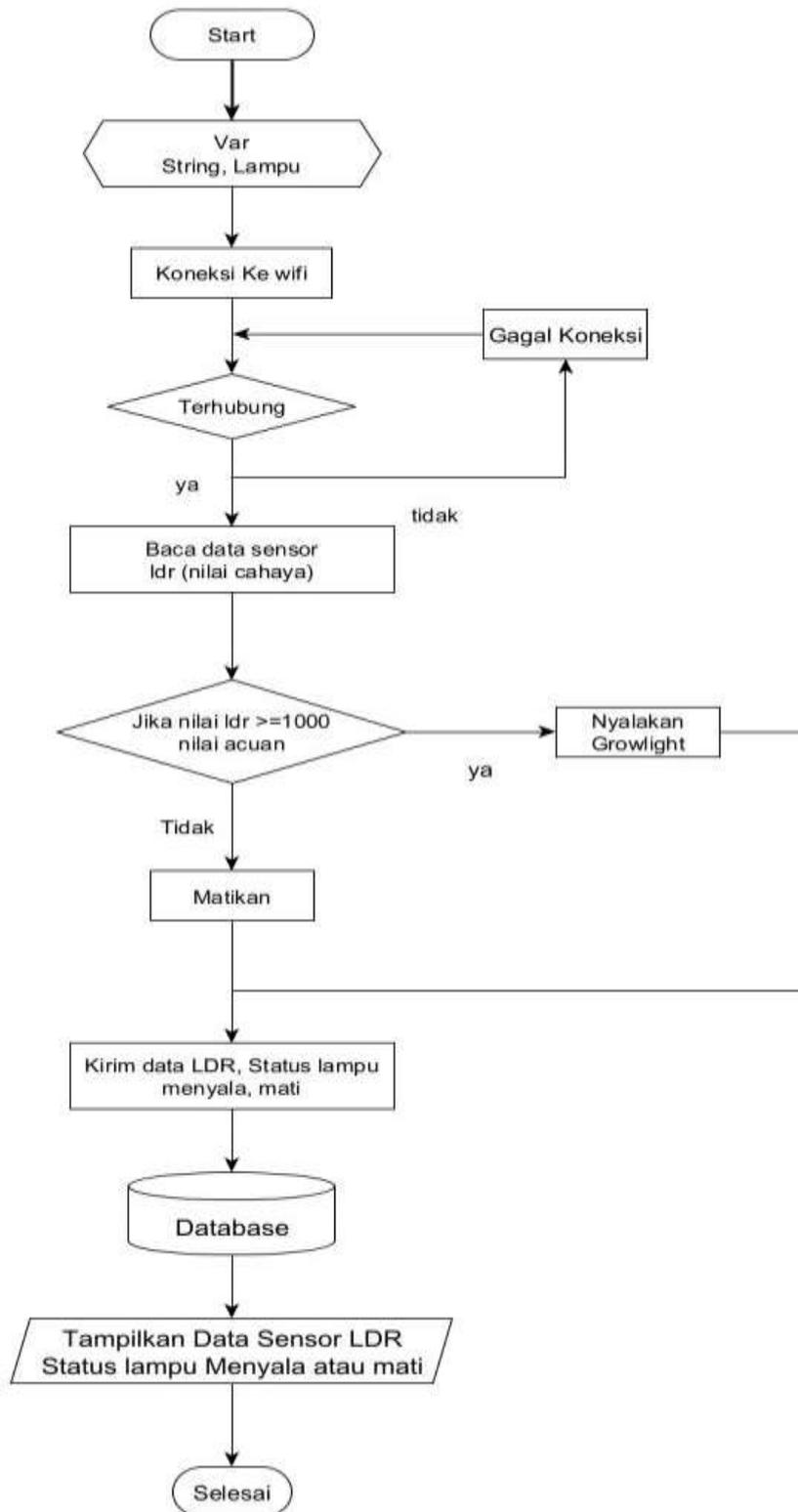
## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **A. Hasil**

##### **1. *Flowchart***

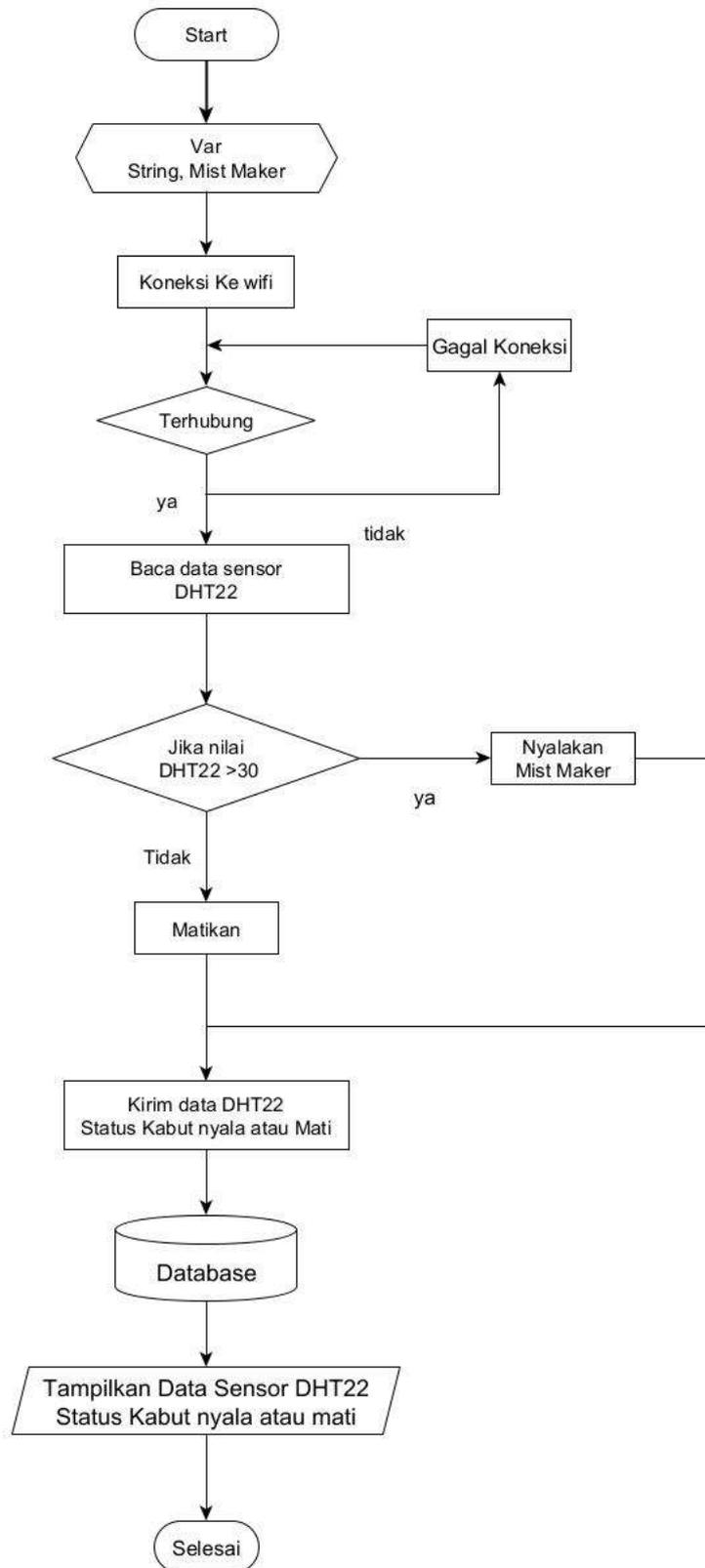
*Flowchart* merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Berikut adalah *flowchart* dan penjelasan masing-masing cara kerja dari sensor.



**Gambar 4. 1** *Flowchart* Monitoring LDR

Penjelasan Pada Gambar 4.1 diatas menjelaskan Ketika sistem dimulai dengan melakukan inisialisasi terlebih dahulu kemudian alat akan mencoba terkoneksi ke jaringan wifi yang telah dikalibrasikan. Ketika alat telah terkoneksi maka alat akan lanjut membaca nilai data yang diperoleh dari sensor *LDR* kemudian nilai tersebut akan disimpan dalam variable *lux* selanjutnya alat akan mengirim nilai tersebut kedalam database *firebase* yang akan ditampilkan nantinya pada perangkat *monitoring* pada *Android*. Setelah itu, apabila nilai Cahaya yang masuk kurang dari nilai acuan maka alat akan menyalakan Lampu *growlight*, dan akan mematikan lampu jika nilai Cahaya yang masuk lebih besar dari nilai acuan.

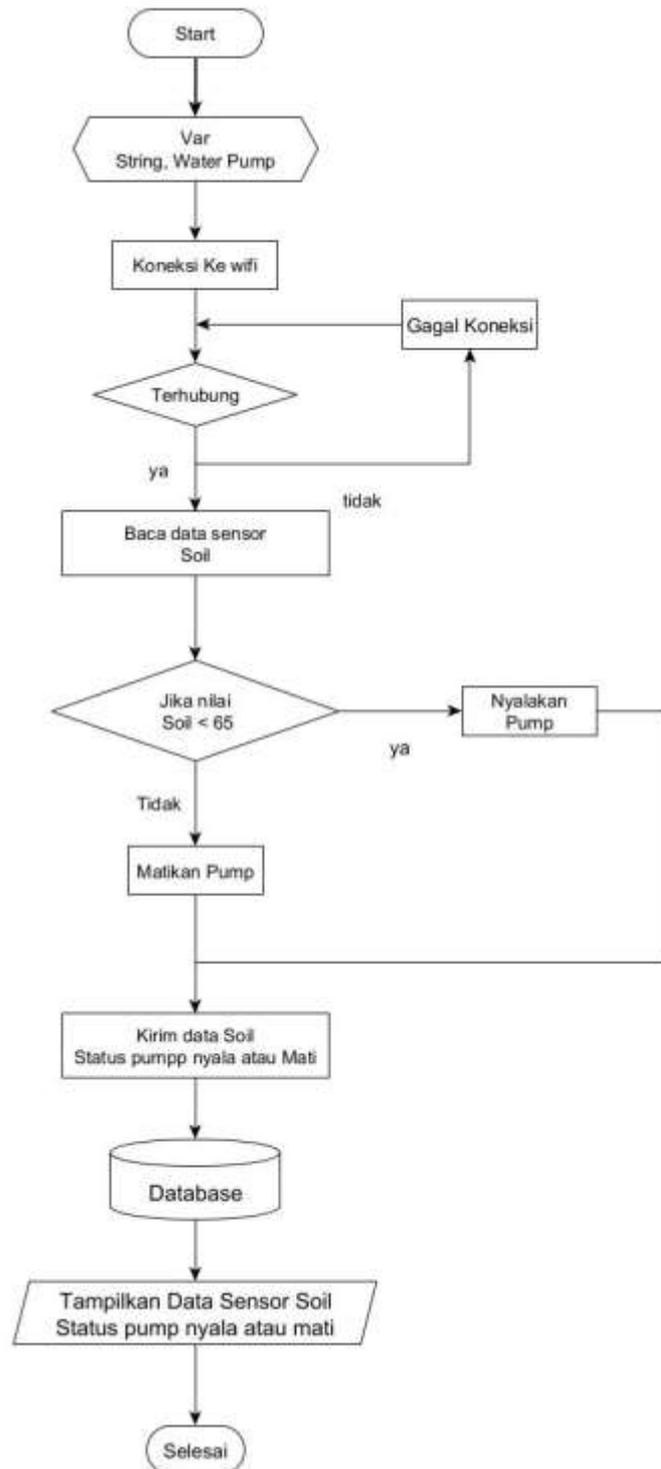
Selanjutnya adalah *flowchart* dari DHT22.



**Gambar 4. 2** Flowchart sensor DHT22

Pada Gambar 4.2 diatas menjelaskan flowchart DHT22, Ketika sistem dimulai dengan melakukan inisialisasi terlebih dahulu kemudian alat akan mencoba terkoneksi ke jaringan wifi yang telah dikalibrasikan. Ketika alat telah terkoneksi maka alat akan lanjut membaca nilai data yang diperoleh dari sensor DHT22 kemudian nilai tersebut akan disimpan dalam variable suhu, dan kelembaban udara selanjutnya alat akan mengirim nilai tersebut kedalam database *firebase* yang akan ditampilkan nantinya pada perangkat *monitoring* pada *Android*. Setelah itu, apabila nilai suhu, dan kelembaban yang masuk kurang dari nilai acuan maka alat akan menyalakan *mist Maker*, dan akan mematikan *mst maker* jika nilai Suhu, dan kelembaban udara yang masuk lebih besar dari nilai acuan.

Terakhir adalah *flowchart* dari *soil*



**Gambar 4.3** *Flowchart soil*

Seperti yang terlihat pada Gambar 4.3 diatas menjelaskan Ketika sistem dimulai dengan melakukan inisialisasi terlebih dahulu kemudian

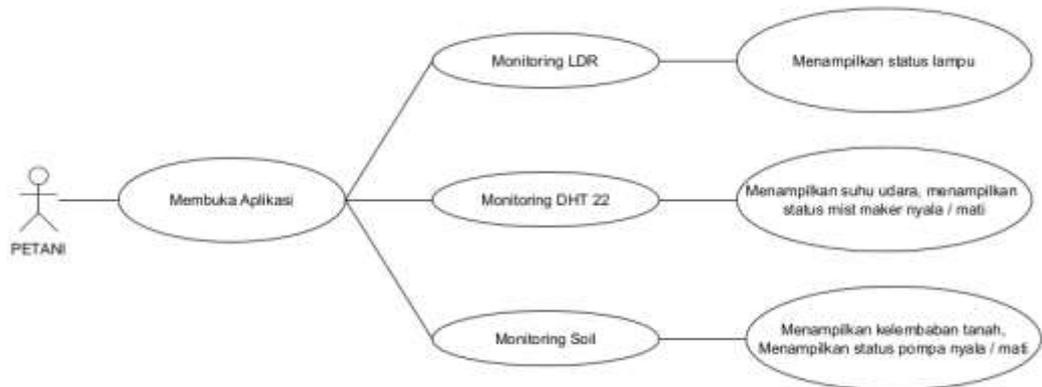
alat akan mencoba terkoneksi ke jaringan wifi yang telah dikalibrasikan. Ketika alat telah terkoneksi maka alat akan lanjut membaca nilai data yang diperoleh dari sensor *soil* kemudian nilai tersebut akan disimpan dalam variable kelembaban tanah selanjutnya alat akan mengirim nilai tersebut kedalam database *firebase* yang akan ditampilkan nantinya pada perangkat *monitoring* pada *Android*. Setelah itu, apabila nilai kelembaban tanah yang masuk kurang dari nilai acuan maka alat akan menyalakan *water pump mini*, dan akan mematikan *waterpump mini* jika nilai kelembaban yang terbaca lebih besar dari nilai acuan.

## **2. Sistem yang berjalan**

Sistem yang berjalan saat ini pada pertanian ialah petani hanya membiarkan bibit kelapa sawit mendapatkan pasokan Cahaya matahari yang tidak terkontrol yang biasa menyebabkan pohon kering dan layu disebabkan pasokan Cahaya matahari yang berlebih dan pohon yang berjamur dikarenakan pohon yang terlalu teduh.

## **3. Sistem Yang Diusulkan**

*Use case* diagram sistem yang diusulkan



**Gambar 4. 4** Use Case Diagram sistem yang diusulkan

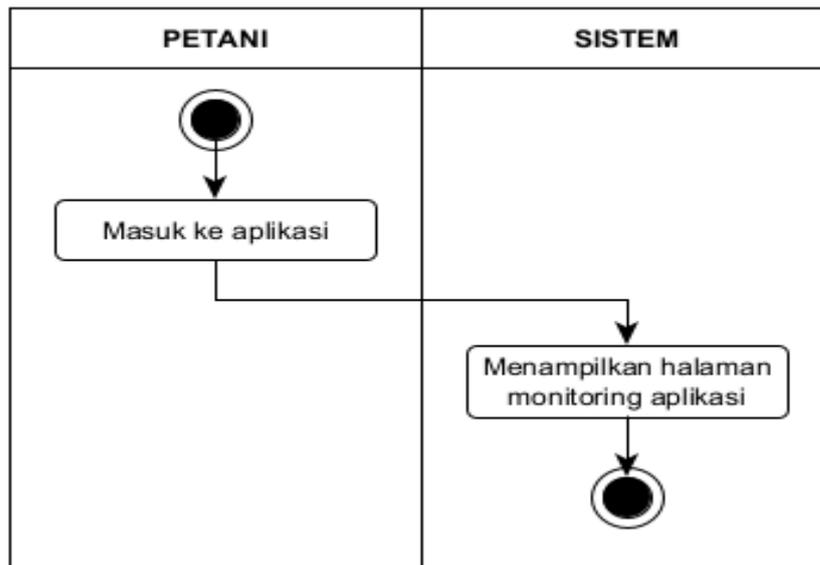
Penjelasan *use case* diagram aplikasi:

Tabel 4. 1 Usecase Diagram Aplikasi

Nama use case	Deskripsi use case
Halaman depan	Use case ini menampilkan halaman keadaan sensor dan status alat

#### 4. Activity diagram

Diagram ini menjelaskan proses urutan yang dilalui petani dalam menampilkan halaman monitoring.



**Gambar 4. 5** *Activity diagram* system yang diusulkan

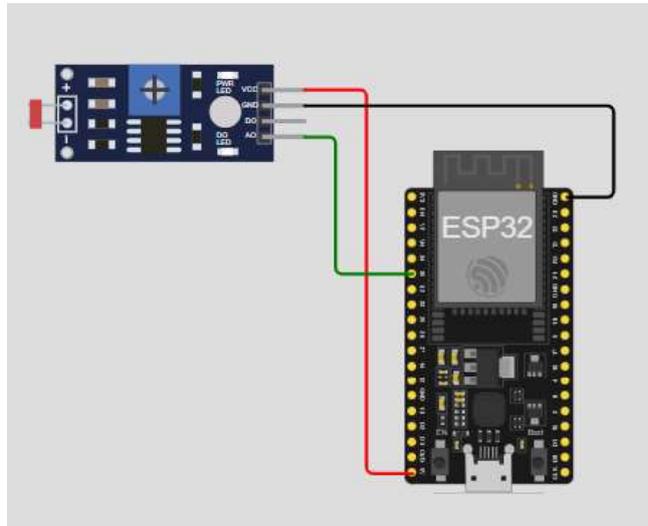
## B. Pembahasan

Pada bagian ini akan dibahas mengenai hasil rancangan perangkat lunak dan perangkat keras serta pengujian perangkat.

### 1. Hasil perancangan perangkat keras (hardware)

Pada perancangan perangkat keras (*Hardware*) menjelaskan beberapa alat yang digunakan pada penelitian ini. Adapun beberapa rangkaian perangkat keras yang digunakan sebagai berikut:

a) Rangkaian sensor ldr



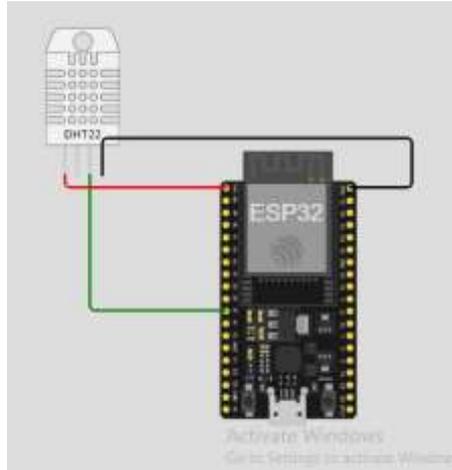
**Gambar 4. 6** Rangkaian Sensor *LDR*

Pada rangkaian ini penulis menggunakan sensor *ldr* yang digunakan untuk mengukur kondisi dari cahaya yang diterima resistor. Pada rangkaian ini dapat dilihat pada gambar menggunakan 3(tiga) buah pin pada sensor yang terhubung ke ESP32, pin Analog *output* pada sensor dihubungkan ke pin 35 pada esp-32, kemudian pin VCC pada sensor dihubungkan ke pin 5v esp32, selanjutnya pin GND pada sensor juga disambungkan ke pin GND yang ada pada Esp-32. Berikut table rangkaian tersebut

**Tabel 4. 2** Rangkaian *Nodemcu esp32* dengan Sensor *LDR*

NO	<i>ESP32</i>	<i>LDR sensor</i>
1	5v	VCC
2	GND	GND
No	<i>ESP32</i>	<i>LDR sensor</i>
3	35	Analog Output

**b) Rangkaian sensor dht22**



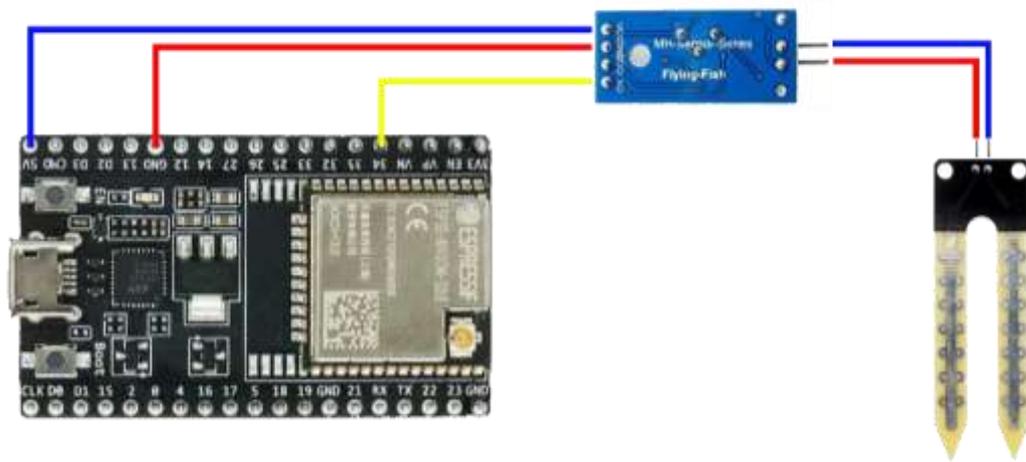
**Gambar 4. 7** Rangkaian sensor DHT22

Pada rangkaian diatas, penulis menggunakan sensor *dht22* yang digunakan untuk mengukur Cahaya disekitar pembibitan. Pada rangkaian ini dihubungkan masing-masing 3 pin, yang pertama pada pin vcc *dht22* dihubungkan dengan pin 3v dari esp32, sealnjutnya pin GND dengan GND, dan terakhir pin data dari *dht22* dihubungkan dengan pin 27 esp32. Berikut adalah tabel rangkaian dari dht22

**Tabel 4. 3** Rangkaian ESP32 dengan Sensor dht22

No.	DHT22	ESP32
1	VCC	3V3
2	GND	GND
3	Output	PIN 27

c) Rangkaian Sensor Soil



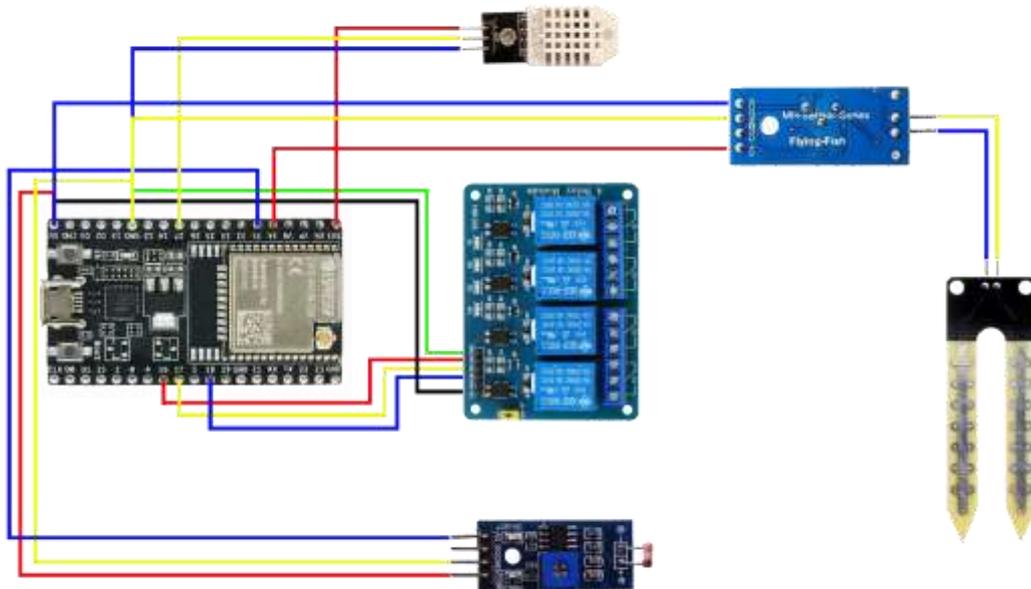
**Gambar 4. 8** Rangkaian Sensor Soil

Pada rangkaian gambar 4.8 diatas penulis menghubungkan soil dengan esp32 dengan menggunakan pin 34 dari esp32 ke pin analog *soil moisture*, selanjutnya GND ke GND, terakhir pin VCC dari *soil* ke pin 5v esp32. Berikut tabel dari rangkaian tersebut:

**Tabel 4. 4** Rangkaian *ESP32* dengan *Sensor Soil*

No.	<i>SOIL</i>	<b>ESP32</b>
1	Analog	Pin 34
2	GND	GND
3	VCC	5V

#### d) Rangkaian Keseluruhan Hardware



**Gambar 4. 9** Rangkaian Keseluruhan

Seperti yang terlihat pada gambar 4.9 Terlihat rangkaian keseluruhan *hardware* dari rancang bangun control dan monitoring pembibitan kelapa sawit.

## 2. Hasil Rancangan Perangkat Lunak (*Software*)

Penulis membuat perangkat lunak yang digunakan untuk memonitoring kerja perangkat keras yang dapat menampilkan nilai dari sensor dan status alat.



**Gambar 4. 10** Tampilan Aplikasi Pemilik

### 3. Rancangan prototype

Berikut ini merupakan rancangan *prototype* keseluruhan dari rancang bangun monitoring pembibitan kelapa sawit pase prenuserary



**Gambar 4. 11** Rancangan Prototype

Dari gambar 4.11 Diatas terlihat bentuk fisik rancangan dari sebuah sistem yang menggunakan Aquarium sebagai wadah pembibitan yang disimulasikan dan ditempatkan beberapa sensor didalamnya.



**Gambar 4. 12** Rangkaian Keseluruhan Alat

Pada gambar 4.12 Diatas menampilkan rancangan dari esp-32 ke sensor, dengan terdapat beberapa komponen didalamnya. Diantaranya 1

(satu) buah Mikrokontroler ESP-32, 1 (Satu) Buah sensor *soil*, kemudian 1 (Satu) buah sensor DHT22, dan 1 (Satu) buah sensor LDR, dan terakhir satu buah Modul Relay 5v 4 Chanel. Pada system ini ESP-32 dihubungkan menggunakan kabel jumper dengan sensor *soil*, *Ldr*, *DHT22*.

#### 4. Pengujian alat

##### a) Pengujian Nilai sensor LDR

Pada pengujian sensor *LDR* dilakukan dengan membandingkan nilai yang didapatkan oleh sensor *LDR* dengan nilai yang didapatkan dari LUX meter untuk mengetahui persentase perbedaan nilai dari kedua alat tersebut. Berikut ini merupakan hasil pengujian yang dilakukan. Perubahan nilai pada sensor *ldr* dilakukan dengan diberikan Cahaya dari senter.

**Tabel 4. 5** Pengujian sensor LDR dengan *Light Meter*

No	Light Meter ( <i>Lux</i> )	Sensor LDR ( <i>Lux</i> )	Error (%)
1	50	13	4%
2	40	20	2%
3	40	19	2,1%
<b>Rata-rata</b>			2,7%

Data tabel diatas merupakan hasil pengujian sensor *LDR*. Untuk mendapatkan nilai *Error* seperti yang didapat pada tabel diatas dilakukan perhitungan dengan rumus sebagai berikut:

$$\%error = \frac{Nilai\ Sensor - Nilai\ Acuan}{Nilai\ Acuan} \times 100\% \dots\dots\dots(1)$$

## b) Pengujian Nilai Sensor DHT22

Pengujian ini menggunakan sensor *dht22*, sensor diujikan dengan tujuan untuk mengetahui Tingkat *error* pada sensor *dht22* dengan membandingkan menggunakan *humidity temperature* merk HTC-1 pada pukul 14:22. Hasil uji sensor dapat dilihat pada tabel 4.6 dibawah.

**Tabel 4. 6** Pengujian Sensor DHT22 dengan *Hygrometer*

No	Humidity Sensor (C)	Sensor DHT22 (C)	Error (%)
1	28.5	29.7	4.21%
2	28.6	29.9	4.55%
3	28.7	29.8	3.83%
4	28.8	29.8	3.47%
5	29.0	29.9	3.10%
<b>Rata-rata</b>			3.83%

Data tabel diatas merupakan hasil pengujian sensor *DHT22* dengan persentase error 3.83%

c) Pengujian Nilai Sensor *Soil*

Dilakukan percobaan uji kelembaban tanah dengan sensor *Soil Moisture* pada bibit kelapa sawit dengan alat pembanding *Three-way meter* yang bisa terlihat pada tabel 4.7 dibawah.

**Tabel 4. 7** Pengujian kelembaban tanah

No	Jenis Tanah	<i>Soil Moisture</i>	<i>Three-way meter</i>	Error %
1	Tanah Kering	1,37	1,40	1,1
2	Tanah Lembab	40	44	2

3	Tanah Basah	72	75	2,2
<b>Rata-rata</b>				2,1

Pada tabel 4.7 merupakan tabel hasil percobaan pengujian sensor *soil moisture* dengan membandingkan produk *three-way* meter. Berdasarkan hasil yang diperoleh setelah membandingkan kedua alat tersebut didapatkan rata-rata tingkat *error* sebesar 2,1% dan perbandingan didapat karena ukuran panjang dari kutub sensor *soil* dan *three-way* berbeda.

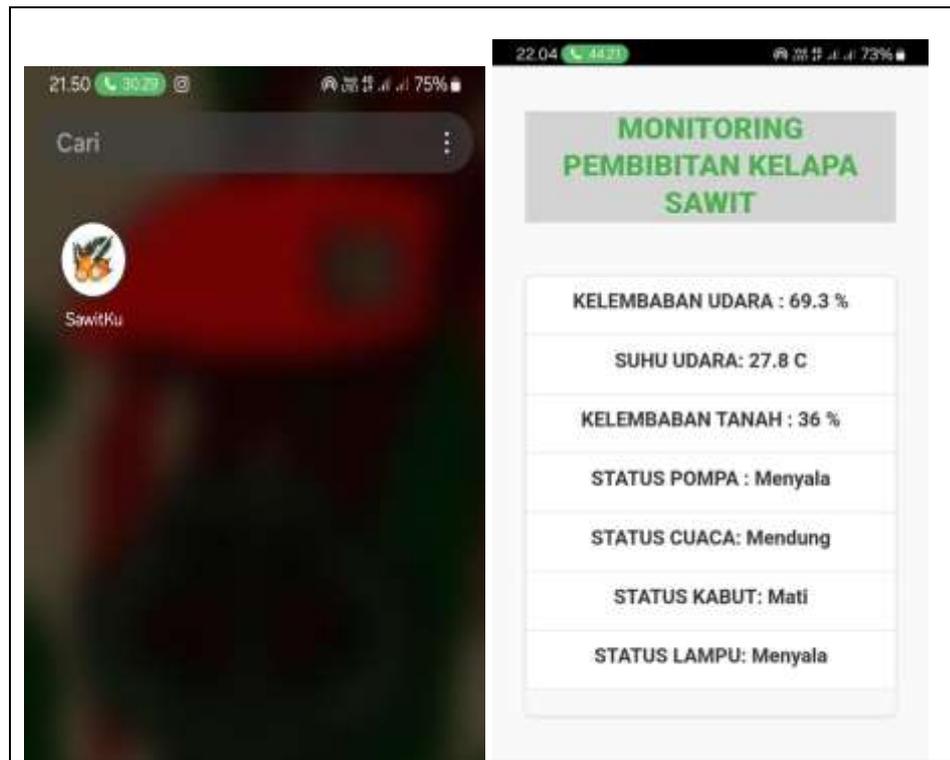
## 5. Pengujian Black Box

Berikut ini merupakan pengujian *Black Box* dari aplikasi Rancang bangun Kontrol dan monitoring pembibitan kelapa sawit dimasa tanam prenusery.

a) Pengujian membuka halaman tampilan aplikasi

**Tabel 4. 8** Pengujian Black Box membuka aplikasi

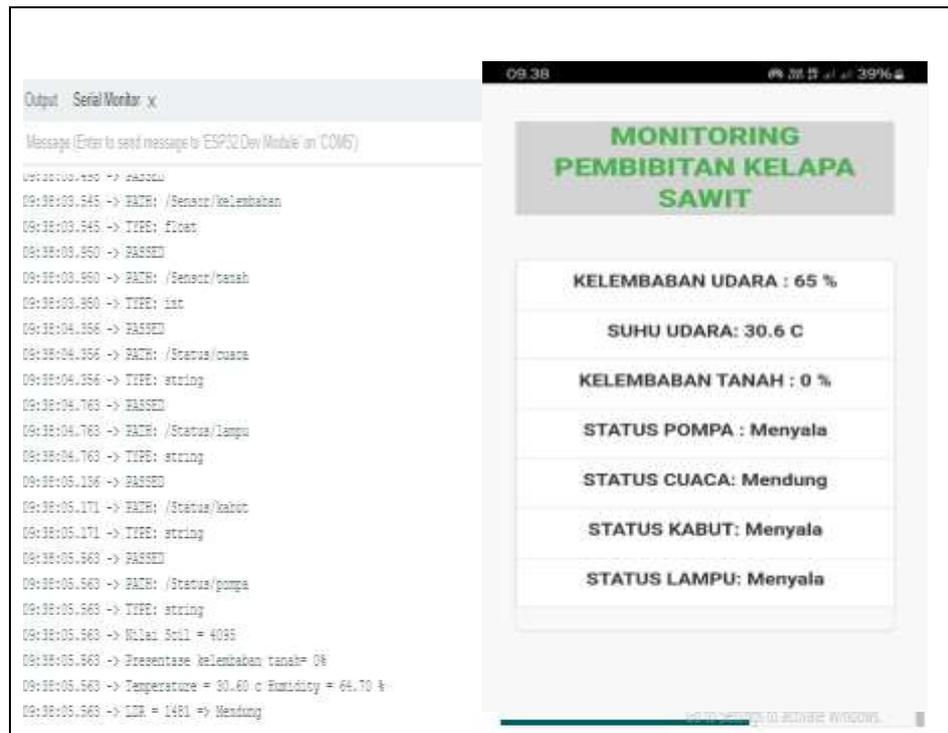
Uji Coba	Hasil yang diharapkan
Membuka Aplikasi	Tampil Nilai sensor dan status alat
Foto/ <i>screenshot</i>	



b) Pengujian monitoring sensor pada aplikasi

**Tabel 4. 9** Pengujian halaman *monitoring* sensor pada aplikasi

Uji Coba	Hasil yang diharapkan
Membuka halaman <i>monitoring</i> sensor	Nilai sensor pada halaman <i>monitoring</i> sesuai dengan nilai sensor yang didapatkan <i>serial monitor</i>
Foto/ <i>screenshot</i>	



c) Pengujian *monitoring* alat pada aplikasi

**Tabel 4. 10** Pengujian halaman *monitoring* sensor pada aplikasi

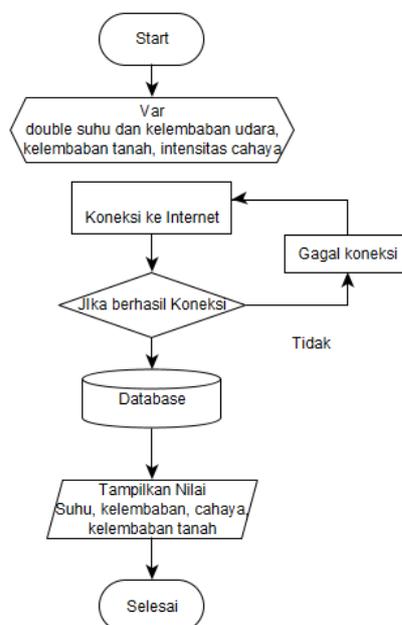
Uji Coba	Hasil yang diharapkan
Membuka halaman <i>monitoring</i> alat	Status alat pada halaman <i>monitoring</i> pada aplikasi tampil sesuai dengan status alat
Foto/ <i>screenshot</i>	



## 6. Pengujian White Box

a) Halaman *Monitoring* sensor

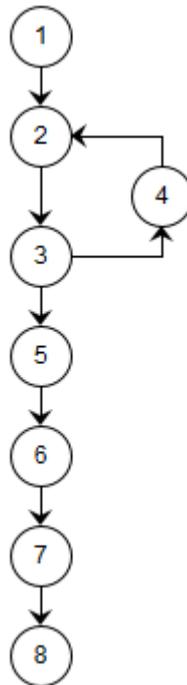
1) *Flowchart*



**Gambar 4. 13** *Flowchart monitoring*

Berdasarkan **Gambar 4.13** Diatas monitoring sensor dimulai dengan system yang akan melakukan inisiasi kemudian system akan mencoba terkoneksi ke internet. Apabila system telah terkoneksi dengan internet maka system akan mengambil nilai sensor dari database yaitu nilai Suhu, kelembaban, kelembaban tanah, intensitas Cahaya. Setelah nilai telah didapatkan maka system akan menampilkan ke dalam halaman monitoring sensor.

1) *Flowgraph*



**Gambar 4. 14** *Flowgraph*

Berdasarkan **Gambar 4.14** Yang disajikan diatas maka dapat dilakukan proses perhitungan sebagai berikut:



## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **A. Kesimpulan**

Berdasarkan hasil pembahasan dan pengujian alat yang telah dilakukan maka dapat ditarik kesimpulan sebagai berikut.

Pada penelitian yang telah dilakukan penulis dapat membuat *prototype* alat yang dapat memonitor suhu dan kelembaban udara, kelembaban tanah, dan intensitas cahaya sesuai dengan parameter tumbuh bibit kelapa sawit dengan menyalakan Mist maker apabila suhu dan kelembaban dibawah nilai daripada yang dibutuhkan bibit, menyalakan lampu jika kondisi matahari kurang Terik dalam hal membantu proses fotosintesis, serta menyalakan waterpump untuk mengairi bibit jika nilai kelembaban tanah turun serta dapat dipantau statusnya melalui aplikasi *Android*.

Sensor DHT22 yang digunakan pada alat tersebut telah diuji dengan melakukan perbandingan dengan *hygrometer* dengan nilai perbedaan sekitar 3,83%, sedangkan dengan sensor LDR yang dimana nilai yang didapatkan tidak berbeda jauh dengan nilai yang didapat oleh Lux meter yaitu persentase error 2,7% dan pada sensor *Soil* yang diuji dengan *three-way* meter mendapatkan persentase error sebesar 2,1%.

#### **B. Saran**

Pada penelitian ini penulis menyadari bahwa masih terdapat banyak kekurangan yang sangat diperlukan dan dikembangkan pada penelitian selanjutnya. Oleh karena itu, penulis memiliki beberapa saran untuk pengembangan selanjutnya sebagai berikut:

1. Pada penelitian selanjutnya dapat ditambahkan pompa untuk menyiram dan di set dengan Waktu penyiraman yang real time.
2. Kedepannya penulis berharap agar system yang dihasilkan selanjutnya dapat dikembangkan dengan menambahkan system pemberi pupuk otomatis agar pemberian pupuk dapat dikontrol.
3. Pada system yang diusulkan selanjutnya bisa digunakan pada tumbuhan yang berbeda dengan berdasarkan literatur yang ada.

## DAFTAR PUSTAKA

- A Ahmad Ramadhan. (2015). *Alat Pengukur Kelembapan Tanah Berbasis Mikrokontroller*. Skripsi Tidak Dipublikasikan. Parepare: Program Studi Teknik Informatika Fakultas Teknik Universitas Muhammadiyah Parepare.
- Adam CS. (2011). ‘*Pembuat Alat Ukur Kelembapan Tanah Berbasis Mikrokontroller Atmega8535*’, (<http://eprints.undip.ac.id/44582/1/untitled.pdf>) (Diakses pada 9 mei 2022).
- Amelia, A., Alam, S., & Selao, A. (2017). SISTEM INFORMASI PEMBIBITAN KELAPA SAWIT DI PT.PERKEBUNAN NUSANTARA XIV (PERSERO) BERBASIS WEB. *Digilib UMPAR*.
- Anis Tatik M. (2012). *Pengaruh Volume Pemberian Air Terhadap Pertumbuhan Bibit Kelapa Sawit di Pembibitan Utama*. Dosen Fakultas Pertanian Univeritas Jambi.
- Ardutech (2020). “*Mengenal ESP32 Development Kit untuk IOT (Intenet of Things)*” Diperoleh dari: <https://www.ardutech.com/mengenal-esp-32-development-kit-untuk-iot-internet-of-things>. (diakses 31 April 2022)
- Awang Dampo. (2015). “*Macam-Macam Alat Ukur*” (<http://www.sainsilmu.com/2015/05/macam-macam-alat-ukur-dalam-fsika-dan.html>) (Diakses pada 1 mei 2022).
- Darwis, Deviana (2018). “*Memonitoring Suhu dan Kelembaban Ruangan Berbasis WEB*”. Parepare: Universitas Muhammadiyah Parepare.
- Erik Nur Prasetyo. (2015). “*Prototype Penyiram Tanaman Persemaian Dengan Senosr Kelembaban Tanah Berbasis Arduino*” (<http://eprints.ums.ac.id/36628/23/NASKAH>). (Diakses pada 20 mei 2022).
- Effendi, N., Handoko, D., Azim, F., & Farida, F. (2024). *Perancangan sistem pemantauan kelembapan tanah pembibitan kelapa sawit berbasis internet of things*. 5(2), 358–366. <https://doi.org/10.37859/coscitech.v5i2.7572>.
- Gera Cerdas. (2014). “*Sensor Moisture*”,(<http://www.geraicerdas.com/produk-eliminasi/moisture-sensor-detail>) (Diakses pada 1 mei 2022).
- Irmayani Pawelloi, A. (2023). *RANCANG BANGUN SISTEM MONITORING SUHU DAN SALINITAS AIR PADA LAHAN RUMPUT BERBASIS*

*INTERNET of THINGS (IoT) Informasi Artikel* (Vol. 3, Issue 1).  
<http://jurnal.umpar.ac.id/indeks/jmosfet#5>

- Pratama, R. P., Mas' ud, A., Niswatin, C., & Rafiq, A. A. (2020). Implementasi DFPlayer untuk Al-Qur'an Digital berbasis Mikrokontroler ESP32. *INVOTEK: Jurnal Inovasi Vokasional Dan Teknologi*, 20(2), 51–58.
- Rahmawati, Y., Simanjuntak, I. U. V., & Simorangkir, R. B. (2022). Rancang Bangun Purwarupa Sistem Peringatan Pengendara Pelanggar Zebra Cross Berbasis Mikrokontroler ESP-32 CAM. *Jambura Journal of Electrical and Electronics Engineering*, 4(2), 189–195.
- Saptadi, A. H. (2014). “Perbandingan Akurasi pengukuran Suhu Dan Kelembaban Antara sensor DHT11 Dan dht22”. *JURNAL INFOTEL - Informatika Telekomunikasi Elektronika*, 6(2), 49. doi:10.20895/infotel.v6i2.16.
- Satya, T., & Puspasari, F. (2019). Kajian Ketidakpastian Pengukuran Suhu dan Kelembaban Udara pada Sensor DHT22 Berbasis Arduino Uno. *JURNAL ILMU FISIKA / UNIVERSITAS ANDALAS*, 11(2), 102-110. doi: 10.25077/jif.11.2.102-110.2019.
- Sukmawan, Y., Riniarti, D., Utoyo, B., & Rifai, A. (2019). EFISIENSI AIR PADA PEMBIBITAN UTAMA KELAPA SAWIT MELALUI APLIKASI MULSA ORGANIK DAN PENGATURAN VOLUME PENYIRAMAN. *Jurnal Pertanian Presisi (Journal of Precision Agriculture)*, 3(2). <https://doi.org/10.35760/jpp.2019.v3i2.2331>
- Utomo, G. D., Triyanto, D., & Ristian, U. (2021). “Sistem Monitoring dan Kontrol Pembibitan Kelapa Sawit Berbasis Internet Of Things”. *Jurnal Komputer Dan Aplikasi*. 9(2): 176-185.
- Wan Martin, (2023). “*Apa itu LED growlight Dan bagaimana cara kerjanya?*”, (<https://www.ledyilighting.com/id/led-grow-light-guide/>) (Diakses pada 23 Januari 2024)