

**DESAIN DAN IMPLEMENTASI SISTEM OTOMATISASI PENYIRAMAN
PUPUK CAIR AB MIX PADA TANAMAN MENGGUNAKAN APLIKASI
MOBILE BERBASIS FLUTTER**

*Design And Implementation Of Automated Watering And Liquid Fertilizer Ab Mix System
On Plants Using Flutter-Based Mobile Application*

PROYEK AKHIR

**Disusun sebagai salah satu syarat untuk memperoleh Gelar Ahli Madya pada
Program Studi Diploma-3 Teknologi Telekomunikasi Fakultas Ilmu Terapan
Universitas Telkom**

oleh:

PUJI ASTUTI

670521304



**DIPLOMA 3 TEKNOLOGI TELEKOMUNIKASI
FAKULTAS ILMU TERAPAN
UNIVERSITAS TELKOM
2024**

LEMBAR PENGESAHAN

Proyek Akhir dengan judul:

DESAIN DAN IMPLEMENTASI SISTEM OTOMATISASI PENYIRAMAN PUPUK CAIR AB MIX PADA TANAMAN MENGUNAKAN APLIKASI MOBILE BERBASIS FLUTTER

*Design And Implementation Of Automated Watering And Liquid Fertilizer Ab Mix System
On Plants Using Flutter-Based Mobile Application*

oleh:

PUJI ASTUTI

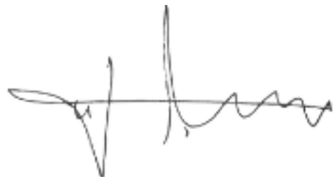
6705213048

Telah diterima dan disahkan sebagai salah satu syarat untuk memperoleh gelar Ahli Madya
pada Program Studi D3 Teknologi Telekomunikasi Universitas Telkom

Bandung, 22 Juli 2024

Menyetujui,

Pembimbing I



Unang Sunarya, S.T., M.T., Ph.D.

NIP. : 10840119-1

Pembimbing II



Denny Darlis S.Si., M.T

NIP. : 13770026-1

LEMBAR PERNYATAAN ORISINALITAS

Dengan ini, Saya :

Nama : PUJI ASTUTI
NIM : 6705213048
Alamat : Komplek Bojongsoang Asri Blok E no 11
No. Tlp/HP : 085778214033
Email : pujiastutixyz@gmail.com

Menyatakan bahwa Proyek Akhir dengan judul:

DESAIN DAN IMPLEMENTASI SISTEM OTOMATISASI PENYIRAMAN PUPUK CAIR AB MIX PADA TANAMAN MENGUNAKAN APLIKASI MOBILE BERBASIS FLUTTER

*Design And Implementation Of Automated Watering And Liquid Fertilizer Ab Mix System
On Plants Using Flutter-Based Mobile Application*

merupakan karya orisinil saya sendiri dan atas pernyataan ini, saya siap menanggung resiko/sanksi yang dijatuhkan kepada saya apabila kemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidakeaslian karya ini.



Bandung, 22 Juli 2019


PUJI ASTUTI
6705213048

IDENTITAS BUKU

Nama Penulis	:	PUJI ASTUTI
Tahun Pengesahan	:	2024
Pembimbing 1	:	Unang Sunarya, ST., MT., Ph.D.
Afiliasi Pembimbing 1	:	D3 Teknologi Telekomunikasi Universitas Telkom
Pembimbing 2	:	Denny Darlis S.Si., M.T
Afiliasi Pembimbing 2	:	D3 Teknologi Telekomunikasi Universitas Telkom
Program Studi	:	D3 Teknologi Telekomunikasi
Fakultas	:	Fakultas Ilmu Terapan
Jenis Buku	:	Laporan Proyek Akhir
Subjek Buku	:	Elektronika

ABSTRAK

Pemupukan adalah proses penting dalam pertanian yang berkontribusi secara signifikan terhadap pertumbuhan dan hasil tanaman. Nutrisi yang tepat dalam tanah dapat meningkatkan kesehatan tanaman, produktivitas, dan kualitas hasil tanaman. Pemupukan yang tepat waktu dan dengan dosis yang sesuai dapat memenuhi kebutuhan nutrisi tanaman, mengoptimalkan pertumbuhan vegetatif maupun generatif, dan meminimalkan risiko defisiensi nutrisi.

Pada Proyek Akhir melakukan perancangan sistem otomatisasi penyiraman pupuk cair dengan menggunakan aplikasi android untuk pengendalian sistem. Tujuan utama dari pembuatan aplikasi ini adalah memberikan kemudahan dalam mengontrol dan memonitor proses penyiraman pupuk cair pada tanaman secara otomatis.

Hasil keluaran dari aplikasi ini yaitu pengguna dapat mengatur jadwal penyiraman, memantau kondisi air, serta mengendalikan perangkat secara langsung melalui antarmuka yang mudah digunakan. Hasil dari aplikasi mencakup informasi tingkat air dalam tangki, serta informasi mengenai jadwal otomatis. Sistem ini diharapkan dapat meningkatkan efisiensi penyiraman tanaman dan mengurangi campur tangan Manusia secara manual, sehingga tanaman dapat tumbuh dengan baik. Hasil pengujian bahwa sistem memiliki tingkat akurasi yang baik dengan error rata-rata 2.88% dalam pengukuran volume air dan tingkat akurasi mencapai 97.12%.

Kata Kunci: *Pemupukan, Penyiraman otomatis, Aplikasi Android, Sistem control, Kondisi air.*

ABSTRACT

Fertilization is an important process in agriculture that contributes significantly to crop growth and yield. The right nutrients in the soil can improve plant health, productivity and yield quality. Timely and properly dosed fertilization can meet the nutritional needs of plants, optimize vegetative and generative growth, and minimize the risk of nutrient deficiencies.

In the Final Project, the design of a liquid fertilizer watering automation system using an android application for system control. The main purpose of making this application is to provide convenience in controlling and monitoring the process of watering liquid fertilizer on plants automatically.

The output of this application is that users can set a watering schedule, monitor water conditions, and control the device directly through an easy-to-use interface. The results of the application include information on the water level in the tank, as well as information on the automatic schedule. This system is expected to increase the efficiency of plant watering and reduce manual human intervention, so that plants can grow well. The test results show that the system has a good level of accuracy with an average error of 2.88% in measuring water volume and an accuracy rate of 97.12%.

Keywords: *Fertilization, Automatic watering, Android application, Control system, Water condition.*

KATA PENGANTAR

Assalamualaikum warahmatullahi wabarakatuh.

Puji syukur alhamdulillah, penulis persembahkan kehadiran Allah SWT yang senantiasa mencurahkan taufik, hidayah dan inayah-Nya sehingga penulis dapat menyusun Proyek Akhir ini. Sholawat serta salam semoga selalu tercurahkan kepada junjungan Rasulullah SAW, yang akan kita nantikan safa'atnya di hari kiamat nanti.

Proyek Akhir ini dibuat untuk memenuhi syarat kelulusan tahap Ahli Madya pada program studi D3 Teknologi Telekomunikasi Fakultas Ilmu Terapan Universitas Telkom. Judul yang dibahas dalam Proyek Akhir ini adalah “**Desain dan Implementasi Sistem Otomatisasi Penyiraman Pupuk Cair AB Mix ada Tanaman menggunakan Aplikasi Mobile Berbasis Flutter**”

Penulis menyadari bahwa Proyek Akhir ini masih jauh dari kesempurnaan, untuk itu saran dan kritik yang bertujuan membangun dari pembaca sangat diharapkan demi perbaikan di masa yang akan datang. Dengan segala kerendahan hati, penulis berharap semoga Proyek Akhir ini dapat bermanfaat bagi kita semua.

Wassalamualaikum warahmatullahi wabarakatuh.

Bandung, 22 Juli 2024



Penulis

UCAPAN TERIMAKASIH

Dalam mengerjakan Proyek Akhir ini, tentu saja merupakan hal yang tidak mungkin apabila penulis berjalan sendiri tanpa berhubungan dengan pihak – pihak yang telah dengan ikhlas memberikan bimbingan, bantuan, dukungan, dan pengarahan baik dalam bentuk materil maupun moril. Karena itu, penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah S.W.T., berkat Rahmat dan Hidayah Nya, penulis diberikan kesehatan dan kelancaran dalam melaksanakan setiap proses demi proses dalam pengerjaan Proyek Akhir ini.
2. Kedua orang tua yang telah memberikan doa dan dukungan yang sangat besar sehingga penulis termotivasi untuk menyelesaikan Proyek Akhir ini.
3. Bapak Unang Sunarya, S.T., M.T., Ph.D. selaku pembimbing I yang telah memberikan arahan dan motivasi kepada penulis agar dapat mengerjakan Proyek Akhir ini dengan terencana dan sesuai dengan target.
4. Bapak Denny Darlis S.Si., M.T selaku pembimbing II yang telah memberikan dukungan dan bimbingan dalam penyelesaian Proyek Akhir
5. Seluruh dosen D3 Teknologi Telekomunikasi selaku pengajar dan pendidik bagi penulis, karena berkat bantuan dan ilmu yang diberikan, sehingga penulis dapat menyelesaikan Proyek Akhir tepat waktu.
6. Mengucapkan terima kasih yang sebesar-besarnya kepada Bapak Endri Suyanto dan Ibu Dian Oktaviani. Berkat jasa dan dukungan mereka, saya dapat melanjutkan kuliah di sini dan menyelesaikan proyek akhir ini. Terima kasih atas segala bimbingan dan dukungan yang telah diberikan.

Kesempurnaan hanya milik Allah SWT. Oleh karena itu, penulis memohon maaf sebesar-besarnya apabila masih terdapat kekurangan serta kesalahan dalam penyelesaian Proyek Akhir ini. Semoga dapat bermanfaat bagi semua pihak.

Bandung, 22 Juli 2019



Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	i
LEMBAR PERNYATAAN ORISINALITAS	ii
IDENTITAS BUKU	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR	vi
UCAPAN TERIMAKASIH	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	ii
DAFTAR TABEL	iii
DAFTAR ISTILAH	iv
DAFTAR SINGKATAN	1
BAB I PENDAHULUAN	2
1.1 Latar Belakang.....	2
1.2 Tujuan dan Manfaat.....	3
1.3 Rumusan Masalah.....	3
1.4 Batasan Masalah	4
1.5 Metodologi.....	4
1.6 Sistematika Penulisan	5
BAB II DASAR TEORI	6
2.1 Sistem Pertanian	6
2.2 Mobile Aplikasi	6
2.3 Visual Studio.....	6
2.4 Android Studio.....	7
2.4.1 Android Emulator.....	8
2.5 Flutter.....	8
2.6 Firebase.....	9
2.7 NodeMCU ESP32 WROOM.....	11

2.8	Modul RTC	12
2.9	Sensor Ultrasonik HC-SR04	13
2.10	Selonoid Valve.....	14
BAB III PERANCANGAN APLIKASI DAN ALAT FERTILIZER		16
3.1	Deskripsi Proyek Akhir	16
3.1.1	Sistem otomatisasi penyiraman pupuk menggunakan aplikasi mobile	17
3.2	Flowchart Sistem	18
3.3	Perancangan Mikrokontroller	19
3.4	Perancangan Aplikasi	21
3.4.1	Use Case Diagram.....	21
3.4.2	Diagram Activity.....	22
3.4.3	<i>Wireframe</i>	23
3.4.4	Mockup.....	26
BAB IV HASIL DAN PENGUJIAN SISTEM.....		27
4.1	Hasil Perancangan Sistem (Alat)	27
4.1.1	Hasil Pengujian Sistem menggunakan Realtime Database	27
4.1.2	Pengujian Sensor Ultrasonik	28
4.2	Hasil Perancangan Aplikasi menggunakan Flutter	31
4.2.1	Pengujian Kompatibilitas AR.....	31
4.2.2	Pengujian Fungsionalitas Aplikasi Fertilizer	32
4.2.3	Pengujian Integrasi Alat Penyiraman Pupuk dengan Aplikasi.....	33
BAB V KESIMPULAN DAN SARAN		39
5.1	Kesimpulan	39
5.2	Saran	39
DAFTAR PUSTAKA		40
LAMPIRAN		42

DAFTAR GAMBAR

Gambar 2. 1 Flutter.....	9
Gambar 2. 2 Firebase.....	9
Gambar 2. 3 ESP32 WROOM -32	11
Gambar 2. 4 RTC	12
Gambar 2. 5 Ultrasonic HC-SR04.....	13
Gambar 2. 6 Selonoid valve	14
Gambar 3. 1 Diagram sistem	17
Gambar 3. 2 Gambaran Sistem.....	18
Gambar 3. 3 Flowchart Sistem	19
Gambar 3. 4 Perancangan Mikrokontroler	19
Gambar 3. 5 Use case diagram	22
Gambar 3. 6 Diagram activity pengendali.....	22
Gambar 3. 7 Diagram activity Monitoring	23
Gambar 3. 8 Halaman Splash Screen	24
Gambar 3. 9 Wireframe Login	24
Gambar 3. 10 Halaman Utama	24
Gambar 3. 11 Wireframe Semprotan Otomatis	25
Gambar 3. 12 Wireframe Status Refill	25
Gambar 3. 13 Wireframe Penjadwalan Otomatis	25
Gambar 3. 14 Wireframe Halaman Pengaturan.....	26
Gambar 3. 15 Mock-up Aplikasi	26
Gambar 4. 1 Hasil Implementasi Alat	27
Gambar 4. 2 Pengujian Integrasi Alat dengan Firebase	28
Gambar 4. 3 Tampilan Hasil Framework Flutter Aplikasi Fertilizer	31
Gambar 4. 4 Pengujian aplikasi semprotan otomatis dengan realtime database	34
Gambar 4. 5 source code menghubungkan DatabaseReference di <code>_getControlValue()</code> :	34
Gambar 4. 6 source code mengubah nilai di Firebase pada <code>_onChangeValveOne()</code> :	35
Gambar 4. 7 source code mengubah nilai di Firebase pada <code>_onChangeValvetwo()</code> :	35
Gambar 4. 8 pengujian aplikasi penjadwalan dengan realtime database	36
Gambar 4. 9 Source code Mengambil Data dari Firebase ' <code>_getScheduleValue()</code> '	36
Gambar 4. 10 Source code Menyimpan Data ke Firebase <code>_onPressSave()</code>	37
Gambar 4. 11 informasi level air pada tangki	38

DAFTAR TABEL

Tabel 2. 1 Spesifikasi ESP32.....	12
Tabel 2. 2 spesifikasi RTC	13
Tabel 2. 3 Spesifikasi Ultrasonic.....	14
Tabel 2. 4 Spesifikasi Solenoid valve.....	15
Tabel 3. 1 Wiring Perangkat.....	20
Tabel 4. 1 Hasil fitur AR	32
Tabel 4. 2 Hasil scenario pengujian aplikasi	33

DAFTAR ISTILAH

<i>Linux</i>	:	Sistem operasi open-source berbasis kernel Unix yang populer untuk komputer dan server.
<i>C++</i>	:	Bahasa pemrograman tingkat tinggi yang digunakan untuk pengembangan perangkat lunak.
<i>C#</i>	:	Bahasa pemrograman yang dikembangkan oleh Microsoft, terutama digunakan untuk pengembangan aplikasi berbasis Windows dan .NET.
<i>Python</i>	:	Bahasa pemrograman tingkat tinggi yang sering digunakan untuk pengembangan aplikasi web, ilmiah, dan scripting.
<i>Go</i>	:	Bahasa pemrograman yang dikembangkan oleh Google, dikenal dengan kesederhanaan dan kecepatannya, umumnya digunakan untuk pengembangan aplikasi server.
<i>Java</i>	:	Bahasa pemrograman yang populer digunakan untuk pengembangan aplikasi desktop, web, mobile (Android), dan enterprise.
<i>Github</i>	:	Platform pengembangan perangkat lunak berbasis web yang menggunakan sistem kontrol versi Git.
<i>Gradle</i>	:	Alat otomatisasi build yang digunakan terutama untuk pengembangan perangkat lunak berbasis Java.
<i>App Engine</i>	:	Layanan komputasi cloud dari Google yang memungkinkan pengembang untuk melakukan hosting aplikasi secara skalabel.
<i>Kivy</i>	:	Framework Python open-source untuk pengembangan aplikasi multitouch.

- React Native* : Framework pengembangan aplikasi mobile open-source dari Facebook, menggunakan JavaScript dan React.
- Kotlin* : Bahasa pemrograman yang kompatibel dengan Java, sering digunakan untuk pengembangan aplikasi Android.
- Back as a service* : Layanan cloud yang menyediakan backend (server-side) untuk aplikasi tanpa memerlukan pengelolaan infrastruktur secara langsung.

DAFTAR SINGKATAN

ESP32	: <i>Espressif System Platform 32</i>
RTC	: <i>Real Time-Clock</i>
PLC	: <i>Programmable Logic Controller</i>
IDE	: <i>Integrated Development Environment</i>
ADT	: <i>Android Development Tools</i>
NDK	: <i>Native Development Kit</i>
AVD	: <i>Android Virtual Device</i>
GPS	: <i>Global Positioning System</i>
iOS	: <i>iPhone Operating System</i>
SDK	: <i>Software Development Kit</i>
API	: <i>Application Programming Interface</i>
JSON	: <i>JavaScript Object Notation</i>
CDN	: <i>Content Delivery Network</i>
HTTP	: <i>Hypertext Transfer Protocol</i>
CSS	: <i>Cascading Style Sheets</i>
LCD	: <i>Liquid Crystal Display</i>

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pemupukan adalah usaha yang dilakukan oleh petani untuk memperbaiki tanah dan menambah unsur hara di dalamnya, dengan tujuan meningkatkan kualitas tanah sebagai penyedia unsur hara bagi tanaman. Proses pemupukan juga memiliki peran penting dalam keberhasilan produksi tanaman, karena pemupukan dapat meningkatkan baik kualitas maupun kuantitas tanaman tersebut. Petani biasanya melakukan pemberian pupuk menggunakan naluri insting dan dilakukan secara manual. Meskipun demikian, metode pemupukan dengan penyiraman dapat menjadi sedikit melelahkan bagi petani. Oleh karena itu, petani pada umumnya perlu membawa alat *cocole* untuk menyuburkan tanaman mereka dengan cara mengelilingi lahan pertanian. Maka dikembangkan sistem otomatisasi penyiraman pupuk cair dengan menggunakan aplikasi android sebagai pengendalian sistem [1].

Kemajuan dalam teknologi elektronik sebagai pengontrol otomatis, seperti mikrokontroler, mikroprosesor, PLC, dan berbagai jenis perangkat lainnya, telah menunjukkan kinerja yang sangat baik. Dalam konteks ini, salah satu sistem pengontrol yang menarik untuk dikembangkan adalah dengan memanfaatkan mikrokontroler ESP32. Keunggulan ESP32 sebagai komponen pengendali dapat memberikan fleksibilitas dan efisiensi yang tinggi dalam implementasi sistem otomatisasi [2].

Pada proyek akhir ini, dibangun sebuah sistem otomatisasi penyiraman pupuk yang dioperasikan melalui aplikasi ponsel berbasis android. Dengan memanfaatkan aplikasi seluler yang dirancang menggunakan Flutter, petani dapat mengatur jadwal penyiraman, memantau kondisi air, serta mengendalikan perangkat secara langsung melalui antarmuka yang mudah digunakan. Hasil dari aplikasi mencakup informasi tingkat air dalam tangki, serta informasi mengenai jadwal otomatis. Sistem ini diharapkan dapat meningkatkan efisiensi penyiraman tanaman dan mengurangi campur tangan Manusia secara manual, sehingga tanaman dapat tumbuh dengan baik.

1.2 Tujuan dan Manfaat

Adapun tujuan dari penulisan Proyek Akhir ini, sebagai berikut.

1. Merancang sistem otomatisasi penyiraman pupuk cair AB *mix* pada tanaman
2. Merancang aplikasi android sebagai pengendali sistem otomatisasi penyiraman pupuk cair AB *mix* pada tanaman
3. Mengimplementasikan sistem otomatisasi penyiraman pupuk cair AB *mix* pada tanaman
4. Menguji sistem otomatisasi penyiraman pupuk cair AB *mix* pada tanaman, menggunakan mobile app berbasis Flutter

Manfaat dari penulisan Proyek Akhir ini, sebagai berikut.

1. Meningkatkan efisiensi dalam proses penyiraman pupuk cair AB *mix* pada tanaman secara otomatis.
2. Memudahkan petani dalam mengendalikan sistem penyiraman pupuk cair AB *mix* melalui aplikasi Android, sehingga lebih praktis dan efisien.
3. Memberikan kemudahan dalam memantau dan mengontrol sistem penyiraman pupuk cair AB *mix* secara *real-time* melalui *mobile app* berbasis flutter.

1.3 Rumusan Masalah

Adapun rumusan masalah dari Proyek Akhir ini, sebagai berikut.

1. Apa saja kendala teknis yang mungkin timbul dalam perancangan dan penerapan sistem otomatisasi untuk penyiraman pupuk AB *Mix* pada tanaman?
2. Apa saja hambatan yang mungkin dihadapi dalam menerapkan kontrol jarak jauh untuk penyiraman pupuk AB *Mix* pada tanaman melalui aplikasi mobile berbasis Flutter?
3. Bagaimana desain dan implementasi sistem dapat mengintegrasikan teknologi Flutter untuk menyajikan antarmuka pengguna yang efisien dan mudah digunakan?

1.4 Batasan Masalah

Adapun batasan masalah dari Proyek Akhir ini, sebagai berikut.

1. Pengembangan aplikasi android dilakukan menggunakan *platform* Flutter dengan bahasa pemrograman Dart.
2. Software yang digunakan untuk mengembangkan aplikasi Android yaitu Visual Studio Code.
3. Emulator yang digunakan untuk menjalankan dan menguji aplikasi adalah perangkat handphone yang terhubung dengan Android Studio.
4. Database yang digunakan dalam pengembangan aplikasi Android adalah Firebase Realtime Database.
5. Menggunakan ESP32 sebagai mikrokontroler untuk mengirim data dari sensor dan mengendalikan aktuator.
6. Hasil keluaran aplikasi Android mencakup informasi dari, kontrol katup selonoid, serta jadwal penyiraman otomatis.

1.5 Metodologi

Adapun metodologi pada penelitian Proyek Akhir ini, sebagai berikut.

1. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan literatur-literatur dan kajian-kajian yang berkaitan dengan permasalahan yang ada pada penelitian Proyek Akhir ini, baik berupa buku referensi, artikel, maupun *e-journal* yang berhubungan dengan perancangan sistem otomatisasi penyiraman pupuk cair melalui aplikasi Android

2. Pengumpulan Data

Pengumpulan data dilakukan dengan menentukan jenis data yang diperlukan, seperti data sensor (ketinggian air), jadwal penyiraman, dan kontrol katup selonoid. Hal ini dilakukan dengan tujuan untuk mengetahui kebutuhan sistem otomatisasi.

3. Perencanaan

Perencanaan untuk merancang sistem secara detail, termasuk pemilihan perangkat keras (ESP32, sensor-sensor), pengembangan aplikasi Flutter untuk Android, dan integrasi dengan Firebase Realtime Database untuk sinkronisasi data.

4. Simulasi Perencanaan

Simulasi perencanaan dilakukan untuk uji coba sistem sebelum diterapkan. Menjalankan program di ESP32 dan sensor-sensor untuk memastikan semuanya berfungsi dengan baik. Pengujian aplikasi Android dengan Flutter untuk memastikan dapat mengendalikan sistem penyiraman pupuk cair dan menyinkronkan data dengan Firebase Realtime Database.

5. Analisis Perencanaan

Dilakukan proses analisa untuk melihat bagaimana sistem berjalan. Mencari permasalahannya dan perbaiki jika perlukan. Tujuannya adalah agar sistem penyiraman pupuk cair bisa bekerja dengan baik sesuai yang diinginkan.

6. Tahap Kesimpulan

Pada tahap ini dilakukan analisis keseluruhan dan menyimpulkan hasil dari penelitian yang telah dilakukan.

1.6 Sistematika Penulisan

Dalam penulisan Proyek Akhir terdiri atas lima bab, dengan keterangan sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini berisi latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah, metodologi penelitian, serta sistematika penulisan.

BAB II DASAR TEORI

Pada bab ini membahas tentang teori pendukung pengerjaan Proyek Akhir, dan aplikasi yang digunakan dalam merancang sistem otomatisasi penyiraman pupuk cair AB mix pada tanaman menggunakan aplikasi mobile berbasis Flutter.

BAB III PERANCANGAN APLIKASI

Pada bab ini membahas tentang deskripsi Proyek Akhir, alur pengerjaan Proyek Akhir, Perancangan aplikasi.

BAB IV SIMULASI DAN ANALISIS

Pada bab ini membahas tentang simulasi dan analisis perencanaan.

BAB V PENUTUP

Pada bab ini membahas tentang kesimpulan dari pengerjaan Proyek Akhir dan saran untuk pembaca yang akan mengambil penelitian dengan topik yang sama.

BAB II

DASAR TEORI

2.1 Sistem Pertanian

Pertanian konvensional merupakan salah satu sistem bercocok tanam dengan media tanam tanah, pertanian konvensional terbagi ke dalam dua jenis yaitu pertanian konvensional tradisional dan pertanian konvensional modern. Pertanian konvensional tradisional yaitu pertanian konvensional yang masih menggunakan alat-alat pertanian tradisional sehingga semua tahapan proses budidaya tanaman masih manual dan membutuhkan banyak tenaga kerja. Ciri dari pertanian konvensional tradisional adalah penggunaan cangkul sebagai alat pengolahan tanah, handsprayer yang dipompa manual oleh tangan untuk penyemprotan pestisida, serta selang atau embrat sebagai alat penyiram tanaman. Sedangkan pertanian konvensional modern yaitu pertanian konvensional yang sudah menerapkan berbagai inovasi teknologi untuk memberi kemudahan bagi petani dalam membudidayakan tanaman. Ciri-ciri pertanian konvensional modern adalah penggunaan rumah kaca sebagai teknik iklim sederhana, penggunaan traktor untuk pengolahan tanah pada lahan tanpa rumah kaca, serta alat penyiram sebagai alat untuk menyiram tanaman [3].

2.2 Mobile Aplikasi

Mobile Aplikasi seluler adalah aplikasi perangkat lunak yang dapat berjalan di perangkat seluler (smartphone, tablet, iPod, dll) dan memiliki sistem operasi yang secara otomatis mendukung perangkat lunak tersebut. Platform distribusi aplikasi seluler yang tersedia biasanya dikelola oleh pemilik sistem operasi seluler, seperti Apple App, Google Play, Windows Phone [3].

2.3 Visual Studio

Visual Studio Code (VS Code) adalah teks editor yang ringan dan andal yang dikembangkan oleh Microsoft untuk berbagai sistem operasi, termasuk Linux, Mac, dan Windows. Editor ini mendukung bahasa pemrograman seperti JavaScript, Typescript, dan Node.js secara langsung, serta berbagai bahasa lainnya melalui pemasangan plugin yang tersedia di marketplace Visual Studio Code, seperti C++, C#, Python, Go, dan Java.

VS Code menawarkan banyak fitur seperti Intellisense, integrasi Git, debugging, dan dukungan ekstensi yang menambah fungsionalitas editor. Fitur-fitur ini terus berkembang seiring dengan pembaruan versi VS Code yang dirilis secara rutin setiap bulan, memberikan keunggulan tersendiri dibandingkan dengan teks editor lainnya.

Selain itu, VS Code bersifat open source, sehingga kode sumbernya dapat dilihat dan dikontribusikan oleh siapa saja melalui Github. Hal ini menjadikan VS Code sebagai pilihan favorit di kalangan pengembang aplikasi, karena mereka dapat turut serta dalam pengembangan dan peningkatan kemampuan editor ini di masa depan [4].

2.4 Android Studio

Android Studio adalah perangkat lunak IDE (Integrated Development Environment) atau Lingkungan Pengembangan Terpadu yang digunakan untuk membuat aplikasi Android. Android Studio mendukung bahasa pemrograman Java dan Kotlin, yang juga digunakan oleh sistem operasi Android. Berbasis pada editor kode IntelliJ dan alat pengembangan serba guna, Android Studio menggantikan IDE sebelumnya, Eclipse ADT (Android Development Tools) [5]

Android Studio menyediakan berbagai fitur yang meningkatkan produktivitas dalam pengembangan aplikasi Android, seperti:

1. Memanfaatkan sistem *build* berbasis *Gradle* yang fleksibel.
2. Emulator yang mudah, cepat, dan kaya akan fitur.
3. Bisa digunakan untuk mengembangkan aplikasi semua jenis perangkat Android.
4. Edit Live untuk memperbarui komponen secara langsung di emulator dan perangkat fisik secara real-time.
5. Mempermudah impor kode melalui template kode yang terintegrasi dengan GitHub.
6. Framework dan perangkat pengujian yang lengkap.
7. Alat lint untuk memonitor performa, kegunaan, kompatibilitas versi, serta permasalahan lainnya.
8. Dukungan untuk C++ dan NDK.
9. Dukungan bawaan untuk *Google Cloud Platform*, memfasilitasi integrasi *Google Cloud Messaging* dan *App Engine*.

Setiap proyek di Android Studio terdiri dari satu atau beberapa modul yang berisi file kode sumber dan sumber daya. Jenis-jenis modul termasuk:

- a. Modul aplikasi Android
- b. Modul perpustakaan
- c. Modul Google App Engine

2.4.1 Android Emulator

Android Emulator adalah sebuah mesin virtual Android yang biasa disebut AVD (Android Virtual Device). AVD ini merepresentasikan perangkat Android tertentu di komputer tanpa memerlukan perangkat asli. Pada dasarnya, Android Emulator berfungsi sebagai platform target untuk menjalankan, debugging, dan pengujian aplikasi Android yang dibuat. AVD atau mesin virtual Android ini tidak hanya merepresentasikan ponsel, tetapi juga dapat merepresentasikan Tablet, Wear OS, Android TV. Emulator Android mampu mensimulasikan hampir semua fitur dari perangkat asli: seperti simulasi panggilan masuk, SMS masuk, lokasi GPS, kecepatan internet yang berbeda-beda, dan berbagai macam sensor perangkat. Dengan demikian, mendebug aplikasi menggunakan emulator Android bisa menjadi pilihan yang baik, meskipun membutuhkan sumber daya yang cukup besar untuk performanya [5].

2.5 Flutter

Flutter adalah kerangka UI mobile dari Google yang gratis dan sumber terbuka yang memberikan pendekatan cepat dan ekspresif bagi pengembang untuk membuat aplikasi asli untuk iOS dan Android. Ada banyak alat yang tersedia untuk mengembangkan aplikasi mobile, seperti Python Kivy, React Native, dan Android Studio yang menggunakan Java atau Kotlin. Satu-satunya kerangka kerja dengan SDK mobile yang menawarkan gaya reaktif tanpa menggunakan jembatan Javascript adalah flutter. SDK ini bersifat sumber terbuka dan tersedia secara gratis, memungkinkan pengembang untuk bereksperimen dan membangun aplikasi pelacakan yang tangguh. Flutter menjadi inspirasi di balik program dan antarmuka pengguna berbasis flutter. Flutter menggunakan GPU, dibangun dari satu kode sumber, dan mengakses API dan layanan platform. Ini juga dikompilasi langsung

menjadi kode arm native. Oleh karena itu, untuk penelitian ini, menggunakan Flutter [6].



Gambar 2. 1 Flutter

2.6 Firebase

Firebase adalah layanan yang dimiliki oleh Google dan digunakan untuk menyederhanakan pengembangan aplikasi bagi para pengembang. Sebagai penyedia layanan *cloud* dan *backend as a service*, Firebase menyediakan solusi yang ditujukan untuk mempermudah pekerjaan pengembang dalam mengembangkan aplikasi dan web. Firebase memungkinkan pengembang aplikasi untuk menitikberatkan perhatian pada pengembangan fitur inti tanpa harus memulai dari awal dalam membangun *backend* dan infrastruktur. Hal ini memungkinkan para pengembang untuk lebih berkonsentrasi dalam mengembangkan aplikasi yang lebih besar [7].



Gambar 2. 2 Firebase

Fitur Firebase tertera pada gambar yaitu:

a. *Firebase Realtime*

Firebase Realtime Database yaitu, *Firebase Realtime Database* adalah database yang di-host di cloud. Data disimpan dalam format JSON dan

disinkronkan secara real-time ke setiap klien yang terhubung. Saat membangun aplikasi lintas platform menggunakan SDK untuk iOS, Android, dan JavaScript, semua klien berbagi satu instance Realtime Database dan menerima pembaruan data terbaru secara otomatis.

b. Penyimpanan data

Firestore memungkinkan penggunanya untuk menyimpan berbagai jenis file langsung ke Google Cloud Storage dari sisi klien. Dengan menggunakan Firestore, pengguna dapat dengan mudah mengunggah file seperti gambar, video, dan dokumen lainnya ke cloud. Penyimpanan Firestore juga dilengkapi dengan aturan keamanan khusus yang dirancang untuk melindungi data pengguna dari akses yang tidak sah oleh pengguna lain. Ini memastikan bahwa setiap file yang disimpan tetap aman dan hanya dapat diakses oleh pengguna yang berwenang, memberikan lapisan perlindungan tambahan dan kepercayaan bagi pengembang yang menggunakan layanan ini dalam aplikasi mereka. Aturan keamanan ini sangat penting dalam menjaga integritas dan privasi data, terutama dalam aplikasi yang menangani informasi sensitif. Firestore Storage tidak hanya menawarkan solusi penyimpanan yang andal tetapi juga memberikan kontrol penuh kepada pengembang atas siapa yang dapat mengakses dan memanipulasi data yang disimpan.

c. Autentikasi

Firestore *Authentication* adalah salah satu layanan back-end yang menyediakan SDK untuk Android dan iOS, serta antarmuka siap pakai yang mudah digunakan untuk mengautentikasi pengguna aplikasi. Firestore Authentication mendukung autentikasi melalui nomor telepon, sandi, serta penyedia identitas populer seperti Google, Facebook, dan lainnya. Layanan ini juga terintegrasi dengan berbagai fitur lain dari Firestore.

d. Firestore *Hosting*

Firestore menyediakan layanan hosting yang mudah digunakan untuk semua dokumen statis dengan memanfaatkan jaringan CDN global yang mendukung HTTP/2. Selain itu, Firestore menggunakan Superstatic, sebuah web server statis, yang dapat dijalankan secara lokal untuk

pengujian secara menyeluruh. Dengan layanan ini, pengembang dapat dengan mudah meng-host situs web dan aplikasi mereka, memastikan konten statis seperti HTML, CSS, dan JavaScript dapat diakses dengan cepat dan efisien dari berbagai lokasi di seluruh dunia. Hosting Firebase juga memungkinkan pengujian lokal yang menyeluruh sebelum deployment, sehingga memastikan bahwa aplikasi berjalan dengan lancar dan tanpa masalah sebelum diakses oleh pengguna akhir.

2.7 NodeMCU ESP32 WROOM

ESP32 adalah mikrokontroler yang diperkenalkan oleh Espressif Systems sebagai penerus dari mikrokontroler ESP8266. Mikrokontroler ini memiliki modul WiFi terintegrasi dalam chip, menjadikannya sangat cocok untuk pengembangan aplikasi Internet of Things (IoT). Pada Gambar 2.3, ditunjukkan pin out dari ESP32. Pin-pin tersebut dapat digunakan sebagai input atau output untuk berbagai keperluan, seperti menyalakan LCD, lampu, atau menggerakkan motor DC [8].



Gambar 2. 3 ESP32 WROOM -32

Tabel 2. 1 Spesifikasi ESP32

SPESIFIKASI	
Tegangan Masukan	5V
Tegangan Kerja	3.3 V
Pin kontrol	EN,RST
Pin Analog	A0
Resolusi Analog	10 bit
Flash Memory	4MB
SRAM	64KB
Clock Speed	80MHz
Pin GPIO	GPIO1 – GPIO16
Pin SPI	SD1, CMD, SD0, CLK
UART Pins	TXD0, RXD0, TXD0

2.8 Modul RTC

Gambar 2.4 menunjukkan bentuk fisik dari modul RTC. Real time clock (RTC) adalah jam elektronik berbentuk chip yang mampu menghitung waktu secara akurat (dari detik hingga tahun) dan menyimpan data waktu tersebut secara real time. Karena modul RTC berkerja secara real time, setelah penghitungan waktu selesai, data segera disimpan atau dikirim ke perangkat lain melalui antarmuka sistem [9].



Gambar 2. 4 RTC

Chip RTC biasanya terdapat pada motherboard PC (biasanya terletak di dekat chip BIOS). Semua komputer menggunakan RTC karena fungsinya untuk menyimpan informasi jam terkini dari komputer yang bersangkutan. RTC dilengkapi dengan baterai untuk memberi daya pada chip sehingga jam akan tetap diperbarui meskipun komputer dimatikan. RTC dianggap sebagai pengatur waktu yang cukup akurat karena menggunakan osilator kristal.

Tabel 2. 2 spesifikasi RTC

SPESIFIKASI	
Tegangan Input	3.3V-5.5V
Komunikasi	12C
Akurasi	±2
Rentang Suhu	0 - 70°C

2.9 Sensor Ultrasonik HC-SR04

Sensor ultrasonik adalah perangkat elektronik yang dapat mengubah energi listrik menjadi energi mekanik dalam bentuk gelombang suara ultrasonik. Pada Gambar 2.5 merupakan salah satu sensor ultrasonik yang populer digunakan untuk mengukur jarak benda adalah Sensor HC-SR04. Sensor ini terdiri dari dua bagian utama, yaitu pemancar ultrasonik (*transmitter*) dan penerima ultrasonik (*receiver*). [10] Sensor HC-SR04 mampu mengukur jarak dengan rentang antara 2 cm hingga 400 cm dengan tingkat presisi sekitar 0,3 cm. Sudut deteksi sensor ini tidak lebih dari 15°. Sensor ini membutuhkan arus kurang dari 2mA dan tegangan operasional sebesar +5V. Total pin yang digunakan oleh sensor ini adalah 4 [11].



Gambar 2. 5 Ultrasonic HC-SR04

Tabel 2. 3 Spesifikasi Ultrasonic

SPESIFIKASI	
Tegangan	5V DC
Arus statis	< 2mA
Level output	5V – 0V
Sudut sensor	< 15°
Jarak yang bisa dideteksi	2cm-450cm (4.5m)
Tingkat keakuratan	Up to 0.3cm (3mm)

2.10 Solenoid Valve

Solenoid valve adalah katup yang menggunakan arus listrik, baik AC maupun DC, untuk mengontrol aliran fluida melalui kumparan atau solenoida. Solenoid valve merupakan komponen kontrol yang sangat umum digunakan dalam sistem-sistem fluida, seperti sistem pneumatik, hidrolik, dan kontrol mesin yang memerlukan otomatisasi. Dalam sistem pneumatik, solenoid valve berperan dalam mengatur aliran udara bertekanan yang menuju aktuator pneumatik seperti silinder [12].



Gambar 2. 6 Solenoid valve

Tabel 2. 4 Spesifikasi Solenoid valve

SPESIFIKASI	
Tegangan	24V DC
Arus statis	400mA
suhu	0-55°
Tekanan	0.02(0.8Mpa)
Model Operasi	Normally closed
Size In-Out	3/8 selang

BAB III

PERANCANGAN APLIKASI DAN ALAT *FERTILIZER*

3.1 Deskripsi Proyek Akhir

Pada Proyek Akhir ini akan dilakukan perancangan dan implementasi sistem monitoring dan pengendalian otomatis untuk penyiraman pupuk cair, dengan memanfaatkan teknologi mikrokontroler dan aplikasi mobile berbasis Flutter. Proyek ini ditujukan untuk menciptakan solusi yang efisien dan mudah digunakan bagi petani dalam mengelola penyiraman pupuk cair pada tanaman mereka. Sistem ini terdiri dari beberapa komponen utama, yakni Mikrokontroler ESP32 yang berfungsi sebagai pusat kendali, Sensor Ultrasonik HC-SR04 untuk mengukur ketinggian air dalam tangki pupuk, serta Pompa Air dan Katup Solenoid yang mengatur aliran pupuk sesuai kebutuhan. Selain itu, Real-Time Clock (RTC) digunakan untuk memastikan jadwal penyiraman berjalan tepat waktu. Aplikasi mobile berbasis Flutter dirancang sebagai antarmuka bagi pengguna, memungkinkan mereka untuk mengatur jadwal penyiraman, memantau kondisi sistem, dan menerima notifikasi secara *real-time* terkait status penyiraman dan volume pupuk yang tersisa. Dengan integrasi teknologi ini, diharapkan sistem dapat meningkatkan efisiensi dan produktivitas dalam pertanian, serta memberikan kemudahan bagi pengguna dalam melakukan pengelolaan pupuk cair.

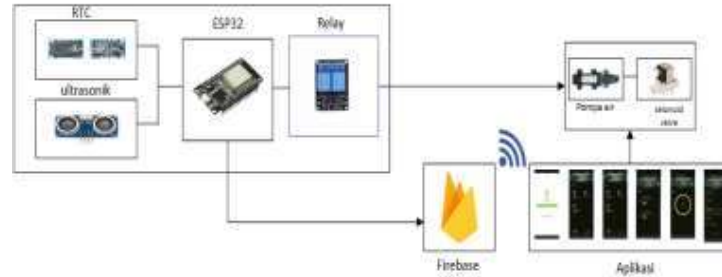
Pada proyek akhir ini, mengembangkan sebuah sistem penyiraman otomatis yang dilengkapi dengan berbagai fitur untuk meningkatkan efisiensi dan kemudahan dalam merawat tanaman. Sistem ini dirancang untuk memberikan kemudahan bagi pengguna dalam mengatur jadwal penyiraman melalui aplikasi mobile. Pengguna dapat dengan mudah menetapkan jadwal penyiraman, yang kemudian akan dijalankan secara otomatis oleh sistem. Selain itu, aplikasi mobile ini menyediakan pemantauan *real-time* yang memberikan informasi mengenai volume pupuk cair yang tersisa dalam tangki. Pengguna juga memiliki kemudahan untuk mengontrol penyiraman secara manual apabila dibutuhkan.

Implementasi sistem ini memanfaatkan koneksi Wi-Fi untuk menghubungkan mikrokontroler ESP32 dengan Firebase Realtime Database. Firebase berperan sebagai penyimpan jadwal penyiraman dan status perangkat keras yang dapat diakses oleh

aplikasi mobile berbasis Flutter. Aplikasi ini dirancang untuk menyediakan antarmuka yang user-friendly, memudahkan pengguna dalam mengatur dan memantau aktivitas penyiraman dengan mudah dan efisien. Dengan sistem penyiraman otomatis yang komprehensif ini, diharapkan dapat membantu pengguna untuk lebih efektif dalam merawat tanaman mereka, meringankan beban pekerjaan manual, dan memastikan tanaman selalu mendapatkan penyiraman yang optimal.

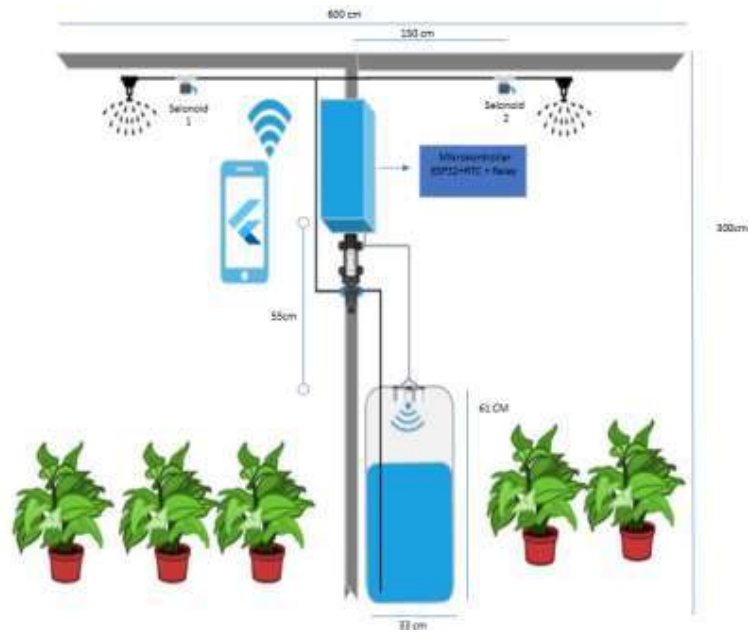
3.1.1 Sistem otomatisasi penyiraman pupuk menggunakan aplikasi mobile

Pada Gambar 3.1 menunjukkan Diagram keseluruhan Proyek Akhir yaitu ESP32 mengambil data dari sensor (misalnya, level air) dan mengirimkannya ke Firebase. ESP32 juga memonitor Firebase untuk perubahan status solenoid valve dan pompa air, dan mengambil tindakan berdasarkan perintah yang diterima (misalnya, mengaktifkan atau menonaktifkan). Hal ini menunjukkan bagaimana ESP32 berfungsi sebagai penghubung antara sensor fisik, Firebase untuk penyimpanan data, dan pengendalian perangkat yang dikendalikan secara otomatis melalui aplikasi Flutter atau perubahan otomatis berdasarkan kondisi sistem yang diprogram.



Gambar 3. 1 Diagram sistem

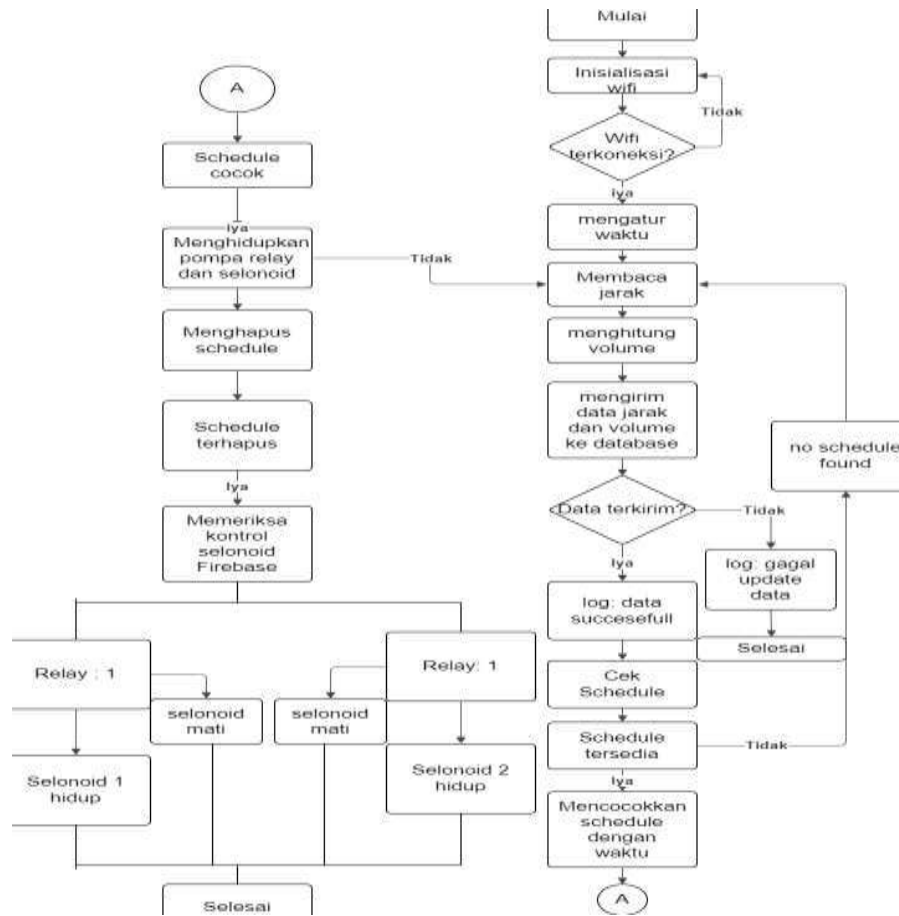
Gambar 3.2 menunjukkan bagaimana sistem akan diterapkan atau dijalankan.



Gambar 3. 2 Gambaran Sistem

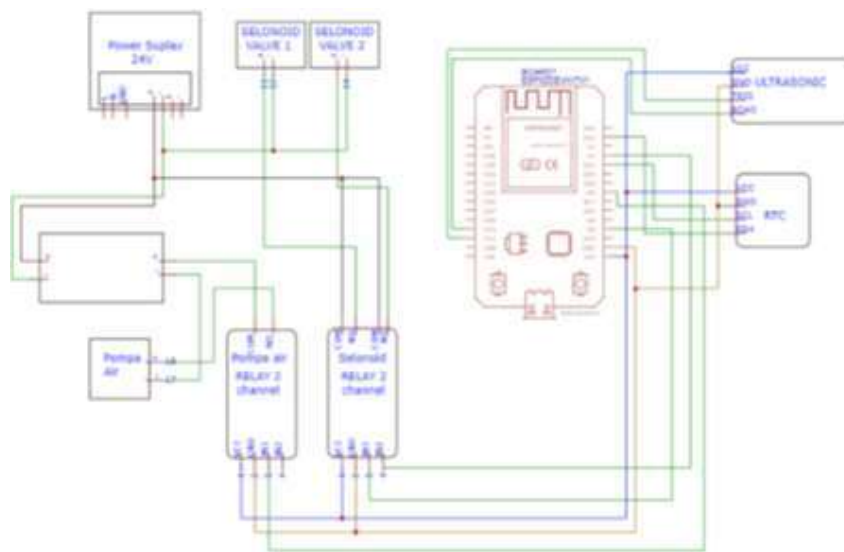
3.2 Flowchart Sistem

Pada gambar 3.3 dalam pengerjaan Proyek Akhir proses pengerjaan dimulai dari inialisasi wifi dan sensor ultrasonic akan membaca jarak juga menghitung volume yang akan dikirim ke database jika data terkirim maka data succesfull. Lalu membuat schedule dan membuat jadwal dan mencocokkan schedule dengan waktu. Ketika schedule cocok akan menghidupkan pompa relay dan solenoid jika sudah selesai maka schedule akan terhapus. Untuk penyiraman pupuk secara manual dengan memeriksa kontrol solenoid pada firebase dan dapat mengaktifkan solenoid 1 atau 2 juga menonaktifkannya.



Gambar 3. 3 Flowchart Sistem

3.3 Perancangan Mikrokontroller



Gambar 3. 4 Perancangan Mikrokontroller

Tabel 3. 1 Wiring Perangkat

Sensor Ultrasonik	PINOUT
VCC	3V3
GND	GND
TRIG	Pin 13
ECHO	Pin 12
RTC (Real-Time Clock)	PINOUT
VCC	3V3
GND	GND
SCL	Pin 21
SDA	Pin 22
Relay 2 channel (Selonoid valve)	PINOUT
VCC	3V3
GND	GND
IN1	Pin 2
IN2	Pin 4
Relay 2 channel (pompa air)	PINOUT
VCC	3V3
GND	GND
IN1	Pin 5

Gambar di atas merupakan gambaran rangkaian yang digunakan dalam sistem mikrokontroller untuk penyiraman otomatis. Sistem ini terdiri dari beberapa komponen utama yaitu ESP32 WROOM-32, Modul RTC DS1307, dua solenoid valve (24VDC), Relay 2 Channel, sensor ultrasonik HC-SR04, pompa air 12VDC, step down, dan power supply 24VDC.

ESP32 WROOM-32 berperan sebagai mikrokontroller utama yang mengontrol seluruh sistem. Pin 3.3V dan GND ESP32 digunakan untuk memberi daya ke sensor dan modul lainnya, serta terhubung ke Wi-Fi untuk komunikasi dengan Firebase. Modul RTC DS1307 terhubung ke pin I2C pada ESP32 untuk menjaga waktu yang akurat, sedangkan relay 2 channel mengontrol dua solenoid valve yang mengatur aliran air ke dua saluran berbeda. Sensor ultrasonik HC-SR04 digunakan untuk mengukur ketinggian air dalam tangki, yang datanya digunakan oleh ESP32 untuk menghitung volume air.

Pompa air 12VDC dikontrol oleh relay dan dayanya diambil dari power supply 24V melalui step down yang menurunkan tegangan menjadi 12V. Solenoid

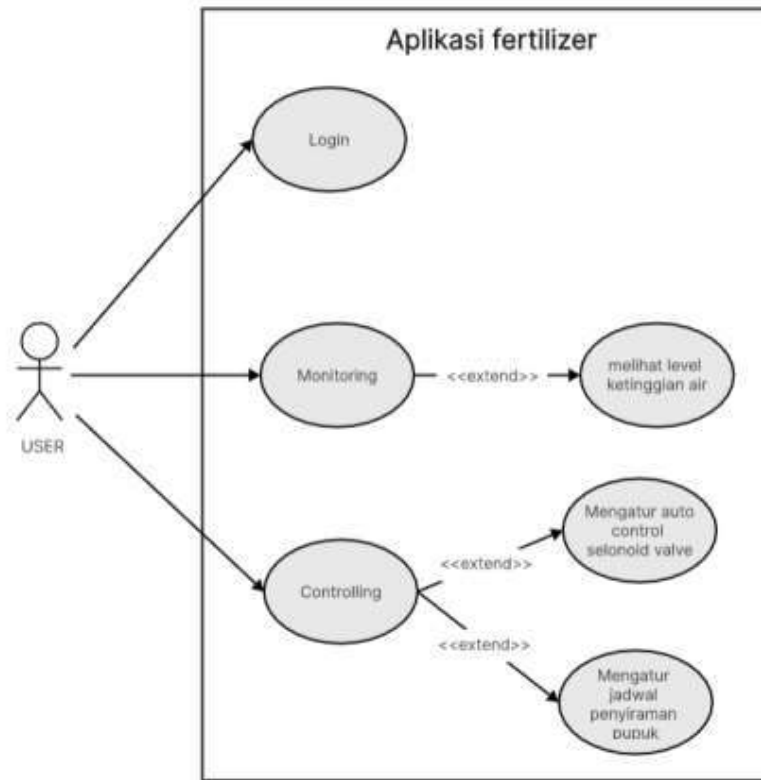
valve terhubung ke power supply 24V untuk daya dan dikontrol oleh relay. Sistem ini dapat mengendalikan penyiraman baik secara otomatis berdasarkan jadwal yang disimpan di Firebase maupun secara manual melalui aplikasi mobile yang terhubung ke Firebase. Data seperti volume air dan status relay diupdate secara berkala ke Firebase untuk pemantauan dan pengendalian jarak jauh. Dengan perancangan ini, sistem penyiraman otomatis dapat berjalan secara efisien dan terkontrol, memastikan tanaman mendapatkan air dan nutrisi yang cukup sesuai dengan jadwal atau kontrol manual dari pengguna.

3.4 Perancangan Aplikasi

Tahap perancangan aplikasi melibatkan perancangan proses kerja hingga desain antarmuka aplikasi. Proses ini memerlukan use case dan diagram aktivitas menggunakan UML (Unified Modeling Language). Use case digunakan untuk secara ringkas menggambarkan siapa yang akan menggunakan sistem dan apa yang dapat mereka lakukan. Diagram aktivitas berfungsi untuk menunjukkan aliran proses kerja sistem. Desain antarmuka aplikasi dirancang dengan membuat wireframe dan mock-up, yang kemudian menjadi panduan dalam pengembangan aplikasi.

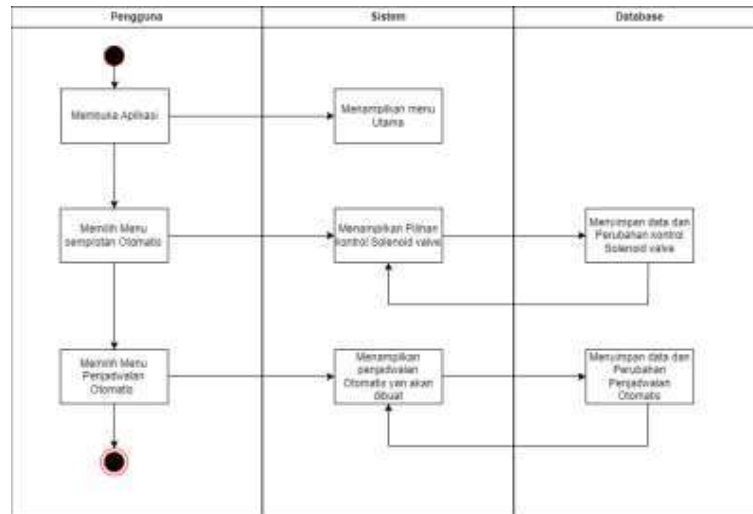
3.4.1 Use Case Diagram

Pada gambar 3.5 dalam use case menjelaskan hanya terdapat satu actor yaitu pengguna aplikasi. Saat membuka aplikasi penyiraman pupuk pengguna diarahkan memberikan identitas diri untuk login atau masuk untuk mengakses aplikasi, Setelah berhasil pengguna akan diarahkan ke halaman utama aplikasi. Halaman ini berisi kontrol dan informasi. Interaksi yang dapat dilakukan oleh pengguna aplikasi penyiraman pupuk adalah mengontrol dua buah solenoid valve yang mengatur aliran pupuk cair, pengguna dapat mengaktifkan dan menonaktifkan solenoid valve untuk memulai atau menghentikan penyiraman sesuai kebutuhan dan menambahkan pengaturan jadwal penyiraman pupuk secara otomatis. Lalu memantau tingkat air pupuk atau level air pupuk dalam tangki.



Gambar 3. 5 Use case diagram

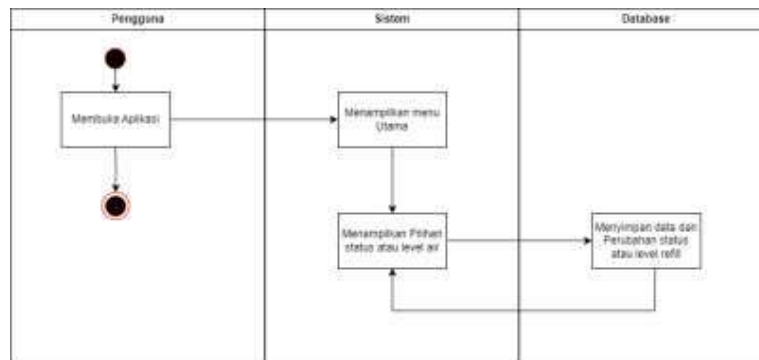
3.4.2 Diagram Activity



Gambar 3. 6 Diagram activity pengendali

Gambar 3.6 yang merupakan *activity* Pengendali pada aplikasi Penyiraman pupuk menjelaskan bahwa pengguna dapat memilih mengontrol

dua solenoid valve yang mengatur aliran air ke dua saluran berbeda. Membuat penjadwalan otomatis pada aplikasi.



Gambar 3. 7 Diagram activity Monitoring

Gambar 3.7 merupakan *activity* Monitoring pada aplikasi Penyiraman pupuk menjelaskan bahwa pengguna dapat melihat status atau level air yang tersisa pada tangki air pupuk.

3.4.3 Wireframe

Wireframe adalah kerangka dasar yang digunakan untuk mengatur sistem dalam aplikasi sebelum diubah menjadi *mock-up*. *Wireframe* menggambarkan struktur dan tata letak dasar dari antarmuka pengguna, tanpa memasukkan elemen desain visual seperti warna, gambar, atau tipografi. Ini membantu dalam perencanaan dan pengorganisasian komponen-komponen utama aplikasi, seperti *header*, menu navigasi, konten utama, dan *footer*. *Wireframe* berfungsi sebagai panduan awal bagi tim pengembang dan desainer untuk memahami bagaimana aplikasi akan bekerja dan berinteraksi dengan pengguna.

1. Halaman *Splash Screen*

Pada Halaman splash screen menampilkan logo dari Lab Green Tech beserta loading bar di bagian bawah. Gambar menunjukkan tampilan halaman splash screen tersebut.



Gambar 3. 8 Halaman Splash Screen

2. Halaman Login

Pada Halaman Login pengguna akan memberikan identitas diri menggunakan Google. Gambar menunjukkan tampilan halaman Login tersebut dengan tombol untuk mengakses akun Google.



Gambar 3. 9 Wireframe Login

3. Halaman Utama

Pada Halaman Utama Pada Halaman Utama aplikasi, pengguna akan melihat berbagai menu yang mencakup penyemprotan otomatis, status isi ulang, penjadwalan otomatis, dan pengaturan.



Gambar 3. 10 Halaman Utama

4. Halaman Semprotan Otomatis

Pada Halaman Semprotan Otomatis menampilkan pilihan mengaktifkan dan menonaktifkan katup selonoid 1 dan 2.



Gambar 3. 11 Wireframe Semprotan Otomatis

5. Halaman Status Refill

Pada Halaman Status Refill menampilkan status atau level air pupuk pada tangki.



Gambar 3. 12 Wireframe Status Refill

6. Halaman Penjadwalan Otomatis

Pada Halaman Penjadwalan Otomatis, pengguna memiliki opsi untuk membuat dan mengatur jadwal penyiraman pupuk secara otomatis.



Gambar 3. 13 Wireframe Penjadwalan Otomatis

7. Halaman Pengaturan

Pada Halaman Pengaturan menampilkan pilihan untuk keluar dari aplikasi.



Gambar 3. 14 Wireframe Halaman Pengaturan

3.4.4 Mockup

Mock-up merupakan tampilan nyata terkait sistem pada aplikasi. Mock-up aplikasi penyiraman pupuk dibuat menggunakan Figma. Berikut merupakan gambar mock-up aplikasi penyiraman pupuk yang ditunjukkan oleh Gambar .3.16



Gambar 3. 15 Mock-up Aplikasi

BAB IV

HASIL DAN PENGUJIAN SISTEM

4.1 Hasil Perancangan Sistem (Alat)

Tampilan hasil perancangan alat penyiraman pupuk pada gambar 4.1 menunjukkan beberapa komponen penting. Terdapat solenoid valve yang dipasang di atas tiang, yang berfungsi untuk mengendalikan penyiraman pupuk. Modul RTC ditempatkan dalam kotak yang digunakan untuk menjaga waktu secara akurat. Sensor ultrasonik yang dipasang pada tutup tangki berfungsi untuk mengukur jarak antara tutup tangki dan permukaan air, sehingga dapat memantau sisa air dalam tangki dalam satuan liter. Pompa air berada dibawah box untuk mengalirkan air dari tangki keatas.



Gambar 4. 1 Hasil Implementasi Alat

4.1.1 Hasil Pengujian Sistem menggunakan Realtime Database

Pada tahap Pengujian, dilakukan pengujian integrasi alat dengan realtime database dimana alat terhubung dengan firebase. Pengujian ini untuk mengetahui integrasi antara alat dan realtime database.



Gambar 4. 2 Pengujian Integrasi Alat dengan Firebase

4.1.2 Pengujian Sensor Ultrasonik

Pengujian sensor ultrasonik untuk mengukur level air dan volume air di dalam tangki, memahami perbedaan antara level air dan volume air yaitu level air merujuk pada ketinggian air dari dasar tangki hingga permukaan air yang diukur. Sensor ultrasonik mengukur waktu yang dibutuhkan gelombang suara untuk kembali setelah memantul dari permukaan air. Waktu ini kemudian dikonversi menjadi jarak. Diukur dalam satuan panjang, biasanya dalam sentimeter (cm) atau meter (m). Volume air adalah jumlah ruang yang ditempati oleh air di dalam tangki. Diukur dalam satuan volume, biasanya dalam liter (L) atau meter kubik (m³). Volume dihitung menggunakan dimensi geometris tangki (seperti tinggi dan diameter) dan level air.

- Menghitung tinggi air yang terisi:

Tinggi terisi: Tinggi tangki - level air yang diukur

- Menghitung volume air

Volume: $\pi \times \left(\frac{\text{diameter}}{2}\right)^2 \times \text{tinggi terisi}$

Level Air: Diukur oleh sensor ultrasonik sebagai jarak dari sensor ke permukaan air. Dipigunakan langsung dalam perhitungan volume.

Volume Air: Dihitung berdasarkan level air yang diukur dan dimensi geometris tangki. Memberikan informasi tentang berapa banyak air yang ada di dalam tangki dalam satuan volume.

Tabel 4. 1 Data pengujian ultrasonik

NO	Level air(cm) - manual	Level air(cm) - sensor	Volume (L) - Manual	Volume (L) - Sensor	Error (%)
1.	10.5	10	43	43.59	1.37 %
2.	15.2	15	39	39.32	0.82 %
3.	20.3	20	34.5	35.049	1.59 %
4.	30.2	30	25.5	26.5	3.92 %
5.	40.1	40	17	17.95	5.59 %
6.	50.2	50	9	9.4	4.44 %
7.	54.5	55	5	5.12	2.4 %
Rata rata error					2.88 %
Tingkat Akurasi					97.12 %

Level Air (cm) - Manual vs Sensor: Data menunjukkan perbedaan kecil antara pengukuran manual dan sensor dengan rata-rata error yang terukur berkisar antara terendah 0.82% tertinggi 5.59%.

Volume (L) - Manual vs Sensor: Perbedaan antara volume yang diukur secara manual dan sensor juga cukup kecil, dengan rata-rata error 2.88%. Tingkat akurasi sistem adalah 97.12%, yang menunjukkan bahwa sensor berfungsi dengan baik dalam mengukur volume air yang tepat.

Tabel 4. 2 Data pengujian waktu dan volume air

NO	Durasi Penyiraman (detik)	Level Air Awal (cm)	Level Air Akhir (cm)	Volume Air Keluar (L)
1.	30	50	47	2.5
2.	60	48	44	4
3.	90	45	38	5.8
4.	120	39	30	7.9

Rumus Perhitungan:

$$\text{Volume: } \pi \times r^2 \times t$$

Ket:

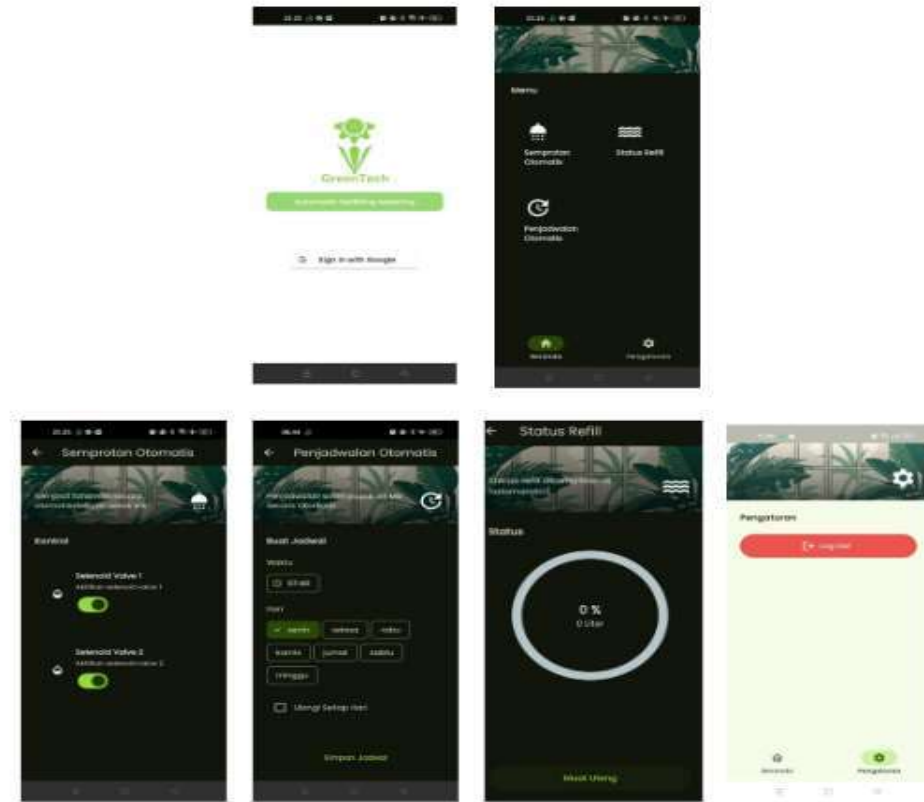
$$\Pi : \frac{22}{7} \text{ atau } 3,14$$

r^2 : Jari- jari diagram

t : level air awal – level air akhir

Pengujian waktu dan volume air yang keluar selama penyiraman pupuk. Diuji menggunakan durasi detik dan melihat berapa volume air yang dikeluarkan pompa melalui selang dan di keluarkan menggunakan *spraying* sesuai durasinya. Tabel 4.2 menunjukkan bahwa sistem penyiraman dapat mengatur volume air yang dikeluarkan dengan presisi, sesuai dengan durasi yang diprogramkan, dan memberikan informasi yang akurat mengenai perubahan level air dalam tangki selama proses penyiraman.

4.2 Hasil Perancangan Aplikasi menggunakan Flutter



Gambar 4. 3 Tampilan Hasil Framework Flutter Aplikasi Fertilizer

Pada Gambar 4.3 merupakan hasil perancangan aplikasi Fertilizer menggunakan framework Flutter.

4.2.1 Pengujian Kompatibilitas AR

Pengujian dilakukan dengan menggunakan smartphone dengan spesifikasi merek yang berbeda, untuk mengenai fitur AR berjalan sesuai harapan.

Tabel 4. 3 Hasil fitur AR

Pabrikasi	Model Perangkat	Fitur AR	Hasil
Xiomi	11T	Berjalan	Sesuai
OPPO	A96	Berjalan	Sesuai
OPPO	A75	Berjalan	Sesuai
OPPO	Reno 4F	Berjalan	Sesuai
Vivo	V23	Tidak Berjalan	Sesuai
OPPO	Reno 6	Berjalan	Sesuai
Vivo	19	Tidak Berjalan	Sesuai

4.2.2 Pengujian Fungsionalitas Aplikasi Fertilizer

Pada tahap pengujian fungsionalitas bertujuan untuk mengetahui apakah Aplikasi Fertilizer sudah berjalan dengan baik dan sesuai yang diharapkan atau tidak. Pengujian fungsionalitas dilakukan sesuai dengan skenario login. Switch Penyemprotan otomatis Selonoid 1 atau 2, Penjadwalan otomatis. Tabel adalah tahapan skenario pegujian Aplikasi Fertilizer dengan cara menjalanka fitur yang terdapat pada aplikasi.

Tabel 4. 4 Hasil scenario pengujian aplikasi

Skenario Pengujian	Data Masukan	Hasil yang diharapkan	Kesimpulan
Masuk pada aplikasi	Mengeklik aplikasi Fertilizer	Masuk kehalaman splash screen dan login	Sesuai
Halaman Login	Menekan tombol SignIn With Google	Dapat melihat menu Beranda atau home dan Pengaturan	sesuai
Masuk ke halaman Semprotan otomatis	Menekan salah satu switch selonoid valve	Memberikan indikasi dengan switch apakah selonoid aktif atau tidak	sesuai
Masuk ke halaman Penjadwalan otomatis	Membuat jadwal untuk penyiraman otomatis	Dapat membuat dan menyimpan penjadwalan	sesuai
Masuk ke halaman Status Refill	Menekan menu Status Refill	Dapat melihat status air pada tangki	sesuai
Masuk ke Pengaturan	Halaman untuk logout	Menekan tombol logout dan kembali pada halaman Login	Sesuai

4.2.3 Pengujian Integrasi Alat Penyiraman Pupuk dengan Aplikasi

Pada tahap pengujian, dilakukan pengujian integrasi alat dengan aplikasi dimana alat dan aplikasi terhubung dengan layanan firebase yang sama. Pengujian ini untuk mengetahui integrasi alat dan aplikasi.



Gambar 4. 4 Pengujian aplikasi semprotan otomatis dengan realtime database

Pada gambar 4.4 Menunjukkan hasil pengujian integrasi antara database Firebase dan aplikasi. Pengujian ini memperlihatkan bahwa data dalam database untuk kontrol solenoid 1 dan solenoid 2 masing-masing bernilai 1. Kontrol ini berhasil dilakukan melalui aplikasi.

```

void _getControlValue() async {
  DatabaseReference controlRef = FirebaseDatabase.instance.ref('control');
  subscription = controlRef.onValue.listen((DatabaseEvent event) {
    final data = event.snapshot.value as Map<dynamic, dynamic>;
    bool relay1 = data['relay1'] == 1 ? true : false;
    bool relay2 = data['relay2'] == 1 ? true : false;
    setState(() {
      _isValveOneActive = relay1;
      _isValveTwoActive = relay2;
      _isInitDataLoading = false;
    });
  });
}

```

Gambar 4. 5 source code menghubungkan DatabaseReference di _getControlValue():

Di dalam `_getControlValue()`, dimulai dengan menghubungkan ke bagian dalam database Firebase yang disebut 'control' menggunakan `FirebaseDatabase.instance.ref('control')`. Kemudian, menggunakan `.onValue.listen` untuk mendengarkan perubahan nilai di sana. Setiap

kali ada perubahan, aplikasi akan mengambil nilai terbaru dari database dan memperbarui tampilannya sesuai dengan nilai yang baru tersebut.

Terlihat pada gambar 4.5

```
void _onChangeValveOne(bool active) async {
  print('toggle valve 1');
  setState(() {
    _isValveOneLoading = true;
  });
  DatabaseReference ref = FirebaseDatabase.instance.ref("control");
  await ref.update({
    "relay1": active ? 1 : 0,
  });
  await Future.delayed(const Duration(seconds: 10));
  setState(() {
    _isValveOneLoading = false;
  });
  if (active) {
    _showValveSnackBar(contentText: 'Solenoid valve 1 berhasil diaktifkan');
  } else {
    _showValveSnackBar(
      contentText: 'Solenoid valve 1 berhasil dinonaktifkan');
  }
}
```

Gambar 4. 6 source code mengubah nilai di Firebase pada `_onChangeValveOne()`:

```
void _onChangeValveTwo(bool active) async {
  print('toggle valve 1');
  setState(() {
    _isValveTwoLoading = true;
  });
  DatabaseReference ref = FirebaseDatabase.instance.ref("control");
  await ref.update({
    "relay2": active ? 1 : 0,
  });
  await Future.delayed(const Duration(seconds: 10));
  setState(() {
    _isValveTwoLoading = false;
  });
  if (active) {
    _showValveSnackBar(contentText: 'Solenoid valve 2 berhasil diaktifkan');
  } else {
    _showValveSnackBar(
      contentText: 'Solenoid valve 2 berhasil dinonaktifkan');
  }
}
```

Gambar 4. 7 source code mengubah nilai di Firebase pada `_onChangeValvetwo()`:

Ketika ingin mengubah status katup (valve) di dalam metode `_onChangeValveOne()` pada gambar 4.6 dan `_onChangeValveTwo()` digambar 4.7, menggunakan `FirebaseDatabase.instance.ref("control")` untuk merujuk ke bagian 'control' di Firebase Realtime Database. Menggunakan metode `.update()` untuk mengirim perintah dan memperbarui nilai relay1 atau relay2 di database. Dengan ini

memungkinkan kami untuk mengaktifkan atau menonaktifkan katup secara langsung dari aplikasi.



Gambar 4. 8 pengujian aplikasi penjadwalan dengan realtime database

Gambar 4.8 menunjukkan hasil pengujian integrasi antara database Firebase dan aplikasi. Pengujian ini memperlihatkan bahwa data penjadwalan waktu dan hari dalam database sesuai dengan yang ada di aplikasi.

```

void _getScheduleValue() async {
  DatabaseReference scheduleRef = FirebaseDatabase.Instance.ref('schedule');
  subscription = scheduleRef.onValue.listen((DatabaseEvent event) {
    final data = event.snapshot.value as Map<dynamic, dynamic>;
    print(data);
    bool isEveryday = data['repeat'] == 1 ? true : false;

    int scheduleHour = 0;
    int scheduleMinute = 0;

    data.forEach((key, value) {
      if (value is Map &&
          value.containsKey('hour') &&
          value.containsKey('minute')) {
        DayFilter? day = _dayFromString(key);
        if (day != null) {
          days.add(day);
          scheduleHour = value['hour'];
          scheduleMinute = value['minute'];
        }
      }
    });
    // print(scheduleHour);
    // print(scheduleMinute);
    final now = DateTime.now();

    setState(() {
      _scheduleTime = TimeOfDay(hour: scheduleHour, minute: scheduleMinute);
      scheduleTimeText = DateFormat('hh:mm')
        .format(DateTime(
          now.year, now.month, now.day, scheduleHour, scheduleMinute))
        .toString();
      isEveryday = isEveryday;
      _isInitDataLoading = false;
    });
  });
}

```

Gambar 4. 9 Source code Mengambil Data dari Firebase '_getScheduleValue()'

Pada Gambar 4.9 Metode `_getScheduleValue()` digunakan untuk memantau perubahan data di Firebase dan mengambil data jadwal yang telah tersimpan. Di dalam `_getScheduleValue()`, dimulai dengan menghubungkan ke bagian dalam database Firebase yang disebut 'schedule' menggunakan `FirebaseDatabase.instance.ref('schedule')`. Kemudian aplikasi memantau perubahan data pada lokasi ini dengan `scheduleRef.onValue.listen((DatabaseEvent event) { ... });`, yang akan menjalankan kode setiap kali ada perubahan data di Firebase. Data yang diterima dikonversi menjadi bentuk Map menggunakan `final data = event.snapshot.value as Map<dynamic, dynamic>`. Selanjutnya, data ini diiterasi dengan `data.forEach((key, value) { ... });` untuk mendapatkan nilai jam (hour) dan menit (minute) dari setiap hari yang disimpan. Terakhir, state aplikasi diperbarui dengan `setState(() { ... });`, sehingga tampilan aplikasi menampilkan jadwal yang baru dari Firebase.

```
void _onPressSave() async {
  try {
    debugPrint('Menyimpan..');

    setState(() {
      _isLoadingSave = true;
    });

    Map<String, dynamic> schedule = {};

    int scheduleHour = _scheduleTime.hour;
    int scheduleMinute = _scheduleTime.minute;

    for (var day in days) {
      String dayName = day.toString().split('.').last;
      schedule[dayName] = {
        "hour": scheduleHour,
        "minute": scheduleMinute,
      };
    }

    schedule["repeat"] = _isEveryday ? 1 : 0;

    DatabaseReference ref = FirebaseDatabase.instance.ref("schedule");
    await ref.set(schedule);

    _showScaffold(content: "Jadwal berhasil tersimpan");
  } catch (e) {
    print(e);
  } finally {
    setState(() {
      _isLoadingSave = false;
    });
  }
}
```

Gambar 4. 10 Source code Menyimpan Data ke Firebase `_onPressSave()`

Pada Gambar 4.10 `_onPressSave()` digunakan untuk menyimpan data jadwal ke Firebase Realtime Database. Pertama, state aplikasi diatur untuk menunjukkan bahwa proses penyimpanan sedang berlangsung.

Sebuah map kosong `schedule` dibuat untuk menyimpan data jadwal, dan hari-hari yang dipilih oleh pengguna diiterasi serta ditambahkan ke dalam map tersebut. Informasi apakah jadwal diulang setiap hari juga ditambahkan. Referensi ke bagian 'schedule' di Firebase kemudian dibuat, dan data jadwal disimpan menggunakan `await ref.set(schedule)`. Setelah penyimpanan berhasil, pesan sukses ditampilkan menggunakan `_showScaffold(content: "Jadwal berhasil tersimpan")`. Akhirnya, state aplikasi diatur kembali ke normal setelah proses penyimpanan selesai.



Gambar 4. 11 informasi level air pada tangki

Gambar 4.11 menunjukkan bahwa hasil pengujian data dari sensor ultrasonik menyatakan volume air dalam tangki dalam satuan liter, dan jarak antara tutup tangki dan permukaan air dalam satuan centimeter (cm).

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, pengujian dan analisa yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Berhasil mengintegrasikan mikrokontroler ESP32 dengan sensor ultrasonik dan RTC (Real-Time Clock), untuk menciptakan sistem pemantauan dan pengendalian penyiraman tanaman yang sesuai. Menggunakan Flutter sebagai framework pengembangan aplikasi, dapat mengembangkan aplikasi mobile yang user-friendly untuk pengaturan dan pengendalian sistem. Firebase Realtime Database digunakan sebagai *backend* yang memungkinkan penyimpanan dan pengambilan data secara real-time, memastikan bahwa informasi selalu *up-to-date* dan dapat diakses dengan cepat. Melalui aplikasi dapat mengatur dan Membuat jadwal pengiriman yang langsung terintegrasikan di realtime database.
2. Hasil pengujian menunjukkan bahwa sensor yang digunakan memiliki tingkat akurasi yang sangat baik, dengan rata-rata error sebesar 2.88% dan tingkat akurasi mencapai 97.12%. Data ini mengonfirmasi bahwa sensor mampu memberikan informasi yang akurat mengenai level air dan volume yang dikeluarkan.

5.2 Saran

Berdasarkan hasil pembangunan Proyek Akhir ini, dapat disampaikan beberapa saran untuk pengembangan selanjutnya yaitu:

1. Menambahkan fitur notifikasi real-time pada aplikasi mobile untuk memberi tahu pengguna tentang kondisi level air pada tangki.
2. Pengguna harus memiliki kuota internet untuk dapat menggunakan dan mengakses aplikasi Fertilizer.

DAFTAR PUSTAKA

- [1] J. Martin and E. Susanto, “KENDALI PH DAN KELEMBABAN TANAH BERBASIS LOGIKA FUZZY MENGGUNAKAN MIKROKONTROLLER (ARRANGEMENT PH AND HUMIDITY OF SOIL BASED ON FUZZY LOGIC USING MICROCONTROLLER).”
- [2] M. Akbar and R. Indra Borman, “OTOMATISASI PEMUPUKAN SAYURAN PADA BIDANG HORTIKULTURA BERBASIS MIKROKONTROLER ARDUINO,” *Jurnal Teknik dan Sistem Komputer (JTIKOM)*, vol. 2, no. 2, 2021.
- [3] A. Rayensyah and D. Hirawan, “PEMBANGUNAN SISTEM PEMELIHARAAN TANAMAN DAN PENGENDALIAN ORGANISME PENGGANGGU TANAMAN OTOMATIS BERBASIS INTERNET OF THINGS.”
- [4] Y. A. Permana and P. Romadlon, “PERANCANGAN SISTEM INFORMASI PENJUALAN PERUMAHAN MENGGUNAKAN METODE SDLC PADA PT. MANDIRI LAND PROSPEROUS BERBASIS MOBILE”.
- [5] R. Rezkini, D. Darlis, and A. Novianti, “PERANCANGAN APLIKASI ADADOKTER PADA ALAT SMART HEALTH MONITORING ADADOKTER APPLICATION DESIGN ON SMART HEALTH MONITORING TOOL.”
- [6] B. Sutrisno and W. S. Prasetya, “Rancang Bangun Mobile Apps E-tiket Bioskop Dengan Penerapan QR Code.”
- [7] R. A. Aziz, “Firebase Membangun Aplikasi Berbasis Android.”
- [8] A. Imran and M. Rasul, “PENGEMBANGAN TEMPAT SAMPAH PINTAR MENGGUNAKAN ESP32,” 2020.
- [9] V. S. Windyasari and P. Azas Bagindo, “Rancang Bangun Alat Penyiraman Dan Pemupukan Tanaman Secara Otomatis Dengan Sistem Monitoring Berbasis Internet Of Things.”
- [10] H. Purwanto, M. Riyadi, D. W. W. Astuti, and I. W. A. W. Kusuma, “KOMPARASI SENSOR ULTRASONIK HC-SR04 DAN JSN-SR04T UNTUK APLIKASI SISTEM DETEKSI KETINGGIAN AIR,” 2019.

- [11] I. G. Megantara, D. Darlis, and A. Novianti, “RANCANG BANGUN SMART HEALTH MONITORING YANG TERINTEGRASI DENGAN APLIKASI ADADOKTER Design and Build Smart Health Monitoring Integrated with The Adadokter Application.”
- [12] M. Nainggolan and D. Putra Caniago, “Desain Pengisian Tangki Penyimpanan Air Otomatis Menggunakan Solenoid Valve Berbasis Arduino Dan Sensor Air,” 2023.

LAMPIRAN

SISTEM PENYIRAMAN PUPUK OTOMATIS



