

BAB III METODE PENELITIAN

Sistem ini dirancang untuk mengetahui nilai suhu, nilai kelembaban tanah dan tinggi air di bak penampungan serta mengatur kelembaban tanah secara otomatis dengan memberikan air sesuai kebutuhan media ternak.

3.1 ALAT DAN BAHAN

Dalam penelitian ini, penulis membuat perancangan alat yang digunakan untuk memantau lingkungan media budidaya cacing tanah melalui teknologi *internet of things*. Alat ini mampu memantau suhu lingkungan, kelembaban tanah serta ketinggian air pada penampungan sebagai penyiram air ke media budidaya cacing tanah yang dimana dapat dipantau secara jarak jauh melalui aplikasi android. Daftar alat dan bahan yang digunakan dalam membuat sistem tersebut, dapat dilihat pada tabel 3.1.

Tabel 3.1 Daftar Alat dan Bahan

No	Alat dan Bahan	Jumlah
1	Laptop	1
2	<i>Smartphone</i>	1
3	NodeMCU ESP8266	1
4	Sensor Dallas DS18B20	1
5	Sensor <i>Soil Moisture</i>	1
6	Sensor Ultrasonik HC-SR04	1
7	<i>Water Pump Mini</i>	1
8	Relay	1
9	Termometer Digital	1
10	<i>Soil Moisture Meter</i>	1
11	Meteran	1
12	Arduino IDE	1
13	MIT App Inventor	1
14	Wireshark	1
15	Google Firebase	1

3.1.1 Laptop

Selain untuk pengolahan data, laptop juga digunakan untuk mengembangkan aplikasi android dan memprogram perangkat yang akan digunakan serta sebagai perantara dalam pengambilan hasil data.

3.1.2 Smartphone

Dalam penelitian ini, *smartphone* digunakan untuk memantau kondisi media ternak pada budidaya cacing tanah yang sudah di pasang aplikasi android, yang telah dirancang melalui MIT App Inventor.

3.1.3 NodeMCU ESP8266

Dalam penelitian ini, mikrokontroler NodeMCU ESP8266 digunakan sebagai pengolah komponen yang tersambung ke NodeMCU dan juga sebagai pengirim data sensor ke *google firebase* melalui komunikasi WiFi.

3.1.4 Sensor Dallas DS18B20

Sebagai pendeteksi suhu pada media budidaya cacing tanah. Nilai suhu tersebut dapat dipantau melalui aplikasi android.

3.1.5 Sensor *Soil Moisture*

Berfungsi untuk mendeteksi kelembaban tanah pada media budidaya cacing tanah. Nilai kelembaban tanah tersebut dapat dipantau melalui aplikasi android.

3.1.6 Sensor Ultrasonik HC-SR04

Sebagai pengukur tinggi air atau ketersediaan air pada penampungan. Tinggi air tersebut dapat dimonitoring menggunakan aplikasi android.

3.1.7 *Water Pump Mini*

Dalam penelitian ini, *water pump mini* berfungsi sebagai pemompa air dari penampungan ke media budidaya cacing tanah.

3.1.8 Relay

Pada penelitian ini, relay digunakan sebagai pemutus dan penghubung arus listrik yang terhubung dengan komponen *water pump mini*.

3.1.9 Termometer Digital

Dalam penelitian ini, alat ukur termometer digital berfungsi sebagai pembanding nilai alat ukur suhu dengan nilai sensor dallas DS18B20, dengan tujuan untuk mencari nilai akurasi dan *error* yang didapatkan dari sensor dallas DS18B20.

3.1.10 Soil Moisture Meter

Dalam penelitian ini, soil moisture meter berfungsi sebagai pembanding nilai alat ukur kelembaban tanah dengan nilai sensor *soil moisture*, dengan tujuan untuk mencari nilai akurasi dan *error* yang didapatkan dari sensor *soil moisture*.

3.1.11 Meteran

Dalam penelitian ini, alat ukur meteran berfungsi sebagai pembanding jarak antara alat ukur meteran dengan sensor ultrasonik HC-SR04, dengan tujuan untuk mencari nilai akurasi dan *error* yang didapatkan dari sensor ultrasonik HC-SR04.

3.1.12 Arduino IDE

Arduino IDE digunakan dalam desain sistem untuk menulis kode yang akan digunakan mikrokontroler NodeMCU ESP8266.

3.1.13 MIT App Inventor

Sebagai aplikasi android untuk memantau nilai suhu, kelembaban tanah, dan ketinggian air pada penampungan.

3.1.14 Wireshark

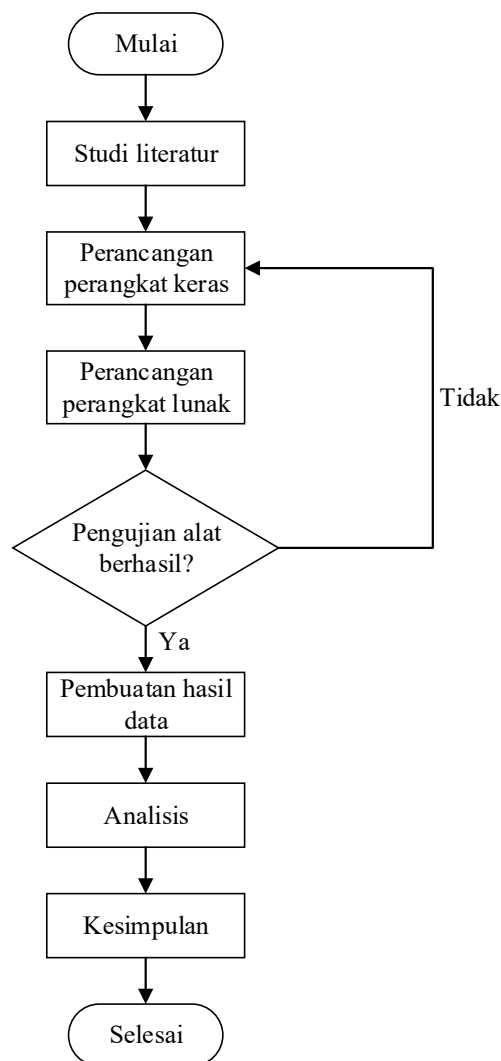
Dalam penelitian ini, *wireshark* berfungsi untuk melihat hasil kualitas QoS (*Quality of Service*) dari jaringan internet yang digunakan.

3.1.15 Google Firebase

Sebagai penyimpanan data yang telah dikirimkan dari alat secara *realtime* melalui jaringan internet, yang nantinya data akan diteruskan ke aplikasi android.

3.2 ALUR PENELITIAN

Dalam penelitian ini, ada beberapa tahapan yang dilakukan dalam pembuatan sistem pemantauan lingkungan media budidaya cacing tanah menggunakan *internet of things*. Proses tahapan penelitian ini dapat dilihat pada gambar 3.1.



Gambar 3.1 Flowchart alur penelitian

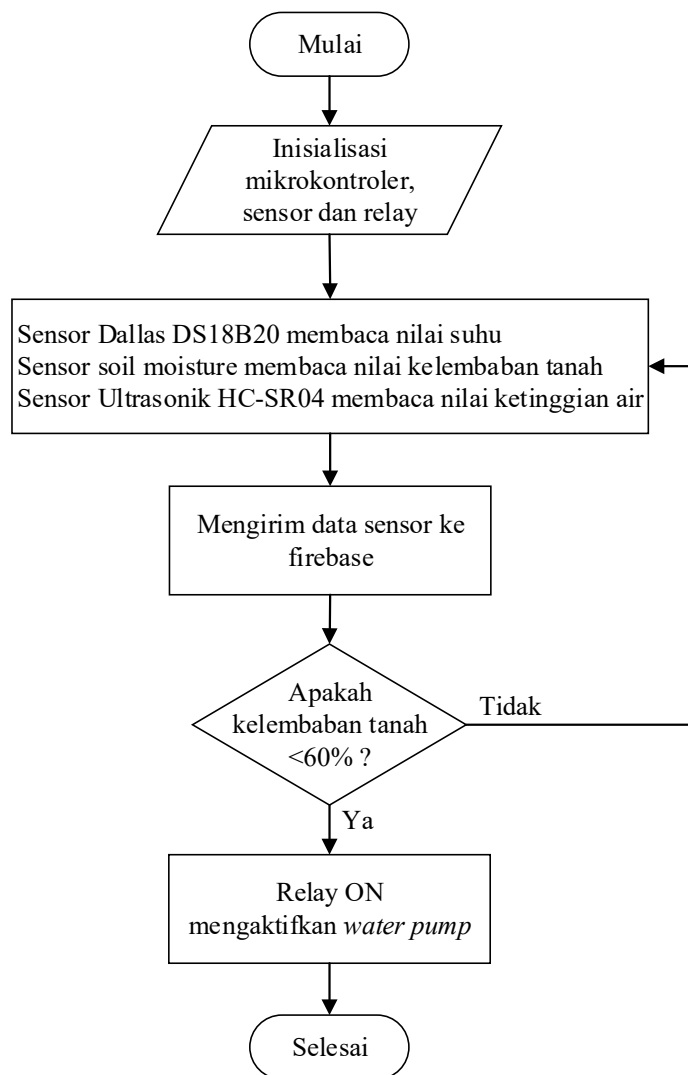
Pada gambar 3.1 langkah pertama dalam penelitian ini adalah mencari materi publikasi tentang desain sebelumnya. Studi literatur dilakukan dengan cara membaca jurnal, buku dan artikel di web yang berhubungan dengan penyusunan

skripsi. Kemudian, harus merencanakan dengan mengumpulkan alat dan bahan yang diperlukan misalnya mikrokontroler NodeMCU ESP8266 sebagai pengolah informasi, sensor Dallas DS18B20 untuk melihat nilai suhu, sensor *soil moisture* untuk melihat nilai kelembaban tanah, sensor ultrasonik HC-SR04 untuk membaca ketinggian air di tempat penyimpanan, relay, dan pompa mini untuk mengalirkan air dari penampungan ke media ternak dengan mengirimkan informasi nilai sensor ke aplikasi android menggunakan komunikasi WiFi. Desain perangkat lunak adalah langkah berikutnya. Pada langkah ini, mikrokontroler NodeMCU ESP8266 digunakan untuk membuat atau mengkonfigurasi program dalam aplikasi Arduino IDE, MIT App Inventor untuk membuat aplikasi android, dan pembuatan database firebase. Langkah selanjutnya adalah melakukan pengujian sistem yang telah dirancang, apakah sistem tersebut berfungsi sebagaimana mestinya dan berhasil atau tidak. Selanjutnya yang perlu dilakukan adalah mengambil hasil data yang didapatkan dari hasil pengujian. Langkah selanjutnya adalah menganalisa data-data yang telah dikumpulkan dari sistem yang dibuat, dan langkah terakhir adalah menarik kesimpulan dan memberikan rekomendasi bagaimana perbaikannya terkait dengan skripsi ini pada pengembangan selanjutnya.

3.3 PERANCANGAN ALAT

Komunikasi data dilakukan melalui jaringan WiFi yang terhubung juga dengan mikrokontroler NodeMCU ESP8266. Sensor yang digunakan meliputi sensor Dallas DS18B20 sebagai pengukur nilai suhu, sensor *soil moisture* sebagai pengukur kelembaban tanah dalam media ternak cacing tanah, dan sensor ultrasonik HC-SR04 untuk mengukur tingkat tersedianya air dalam penampungan. Hasil nilai sensor ini akan dikirimkan pada aplikasi android yang dibuat oleh penulis menggunakan MIT App Inventor, dengan menggunakan database Firebase. Sistem yang dirancang oleh penulis dapat memonitoring dan mengatur kelembaban tanah secara otomatis dengan memberikan air sesuai dengan nilai kelembaban yang telah ditentukan sebelumnya.

- a. *Flowchart* alur perangkat mikrokontroler

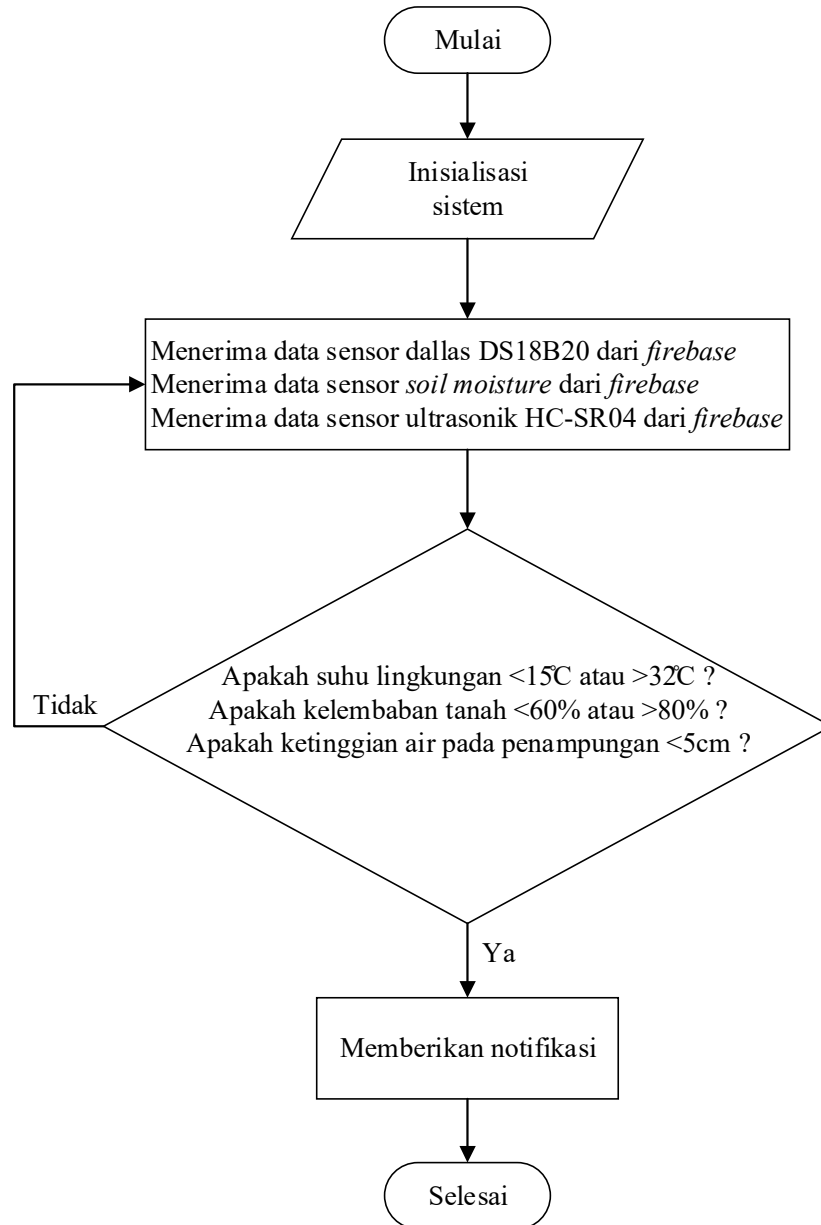


Gambar 3.2 Flowchart alur perangkat mikrokontroler

Gambar 3.2 merupakan *flowchart* alur perangkat mikrokontroler, yang dimulai dari mikrokontroler menjalankan tugasnya sesuai program yang dimasukkan untuk menginisialisasi variabel, sensor, dan relay. Proses selanjutnya yaitu sensor Dallas DS18B20 akan membaca nilai suhu, sensor *soil moisture* membaca nilai kelembaban tanah, dan sensor ultrasonik HC-SR04 untuk membaca ketinggian air atau tersedianya air dalam penampungan. NodeMCU ESP8266 menggunakan komunikasi WiFi untuk mengirimkan hasil data sensor ke Firebase setelah semua datanya dibaca. Aplikasi android yang dikembangkan menggunakan website MIT App Inventor akan menerima data sensor dari Firebase. Selanjutnya relay akan mengaktifkan pompa air untuk mengalirkan air pada penampungan ke media ternak cacing tanah apabila kelembaban tanah yang terdeteksi kurang dari

60%. Apabila kelembaban tanah tidak kurang dari 60% maka sensor akan kembali membaca nilai kelembaban tanah.

b. *Flowchart* alur perangkat lunak (*software*)



Gambar 3.3 *Flowchart* alur perangkat lunak (*software*)

Pada Gambar 3.3 tahapan yang pertama diawali dengan inisialisasi sistem untuk memberikan kondisi nilai awal pada saat membuka aplikasi android. Selain itu, data dari sensor dallas DS18B20, sensor *soil moisture*, dan sensor ultrasonik HC-SR04 diperoleh dari database Firebase. Jika suhu turun di bawah 15°C atau naik melebihi 32°C, aplikasi android akan mengirimkan notifikasi, jika tidak akan mengambil data sensor dallas DS18B20 dari firebase. Dalam hal kelembaban tanah,

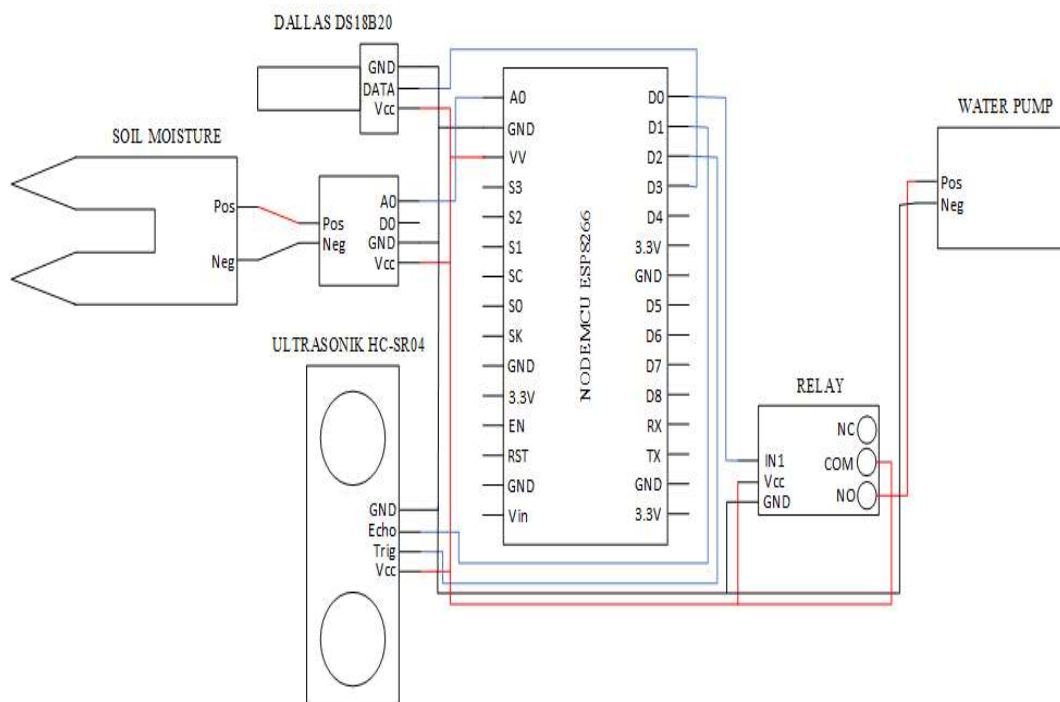
aplikasi android akan memberi tahu pengguna jika tingkat kelembaban tanah dibawah 60% atau lebih tinggi dari 80%, jika tidak akan memperoleh data sensor kelembaban tanah dari firebase. Langkah terakhir adalah mendapatkan data dari sensor ultrasonik HC-SR04 dari firebase. Jika ketinggian air di persediaan dibawah 5cm atau air hampir habis maka aplikasi android akan mengirimkan notifikasi, jika tidak maka akan kembali mendapatkan informasi sensor ultrasonik HC-SR04 dari firebase.

3.3.1 Perancangan Perangkat Keras

Pada bagian perancangan perangkat keras membahas mengenai skematik perangkat keras, blok diagram sistem perangkat keras dan wiring rangkaian perangkat keras.

3.3.1.1 Skematik Perangkat Keras

Bagian ini mencakup desain skematik berbasis internet of things (IOT) dari sistem pemantauan kualitas lingkungan untuk media ternak dalam produksi cacing tanah. Skematik perangkat keras dapat dilihat pada gambar 3.4.



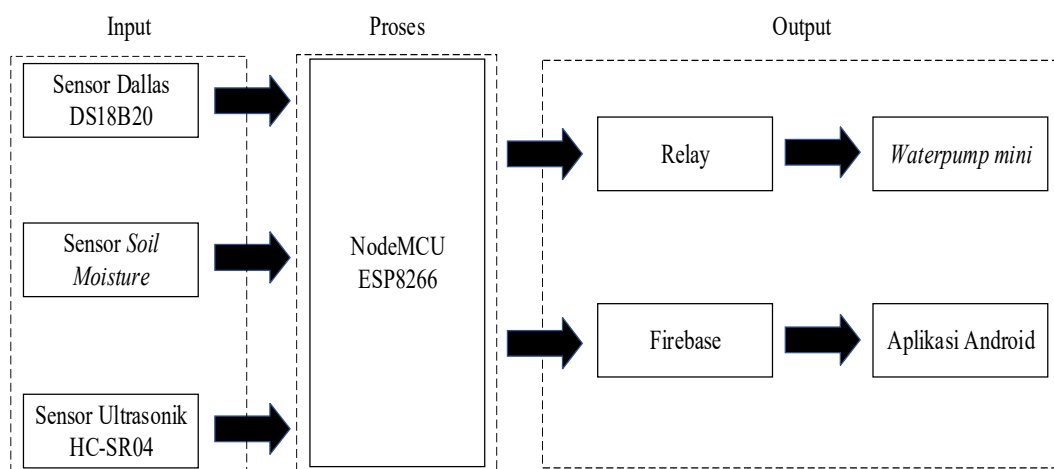
Gambar 3.4 Skematik perangkat keras

Jalur pin yang digunakan oleh penulis untuk membuat sistem pemantauan kualitas lingkungan media ternak berbasis internet of things (IOT) untuk peternakan

cacing tanah ditunjukkan pada Gambar 3.4. Skema pin GND dari DS18B20 sensor Dallas terhubung ke ESP8266 pin GND NodeMCU, pin data terpasang ke pin D3, dan pin Vcc terhubung ke pin catu daya NodeMCU ESP8266 5V. Pin Vcc terhubung ke catu daya NodeMCU ESP8266 5V, pin GND ke GND, dan pin A0 dari sensor kelembaban tanah ke pin A0. Pin GND dari sensor ultrasonik HC-SR04 harus dihubungkan ke GND, pin Echo harus terhubung ke D1, pin Trigger harus dipasang ke D2, dan pin Vcc harus terhubung ke pin catu daya 5V NodeMCU ESP8266 mikrokontroler. Pada relay, pin D0 ESP8266 mikrokontroler NodeMCU terhubung ke pin IN1, pin catu daya NodeMCU, ESP8266 5V terhubung ke pin Vcc, pin GND terhubung ke GND, dan COM relay terhubung ke pin catu daya 5V NodeMCU ESP8266. Pada pompa air mini, pin negatif dihubungkan ke pin GND relay dan pin positif akan menuju ke pin NO relay.

3.3.1.2 Blok Diagram Sistem Perangkat Keras

Perancangan sistem pemantauan kualitas lingkungan berbasis internet of things (IOT) untuk media ternak dalam budidaya cacing tanah tercakup dalam bagian ini. Blok diagram sistem perangkat keras dapat dilihat pada gambar 3.5.



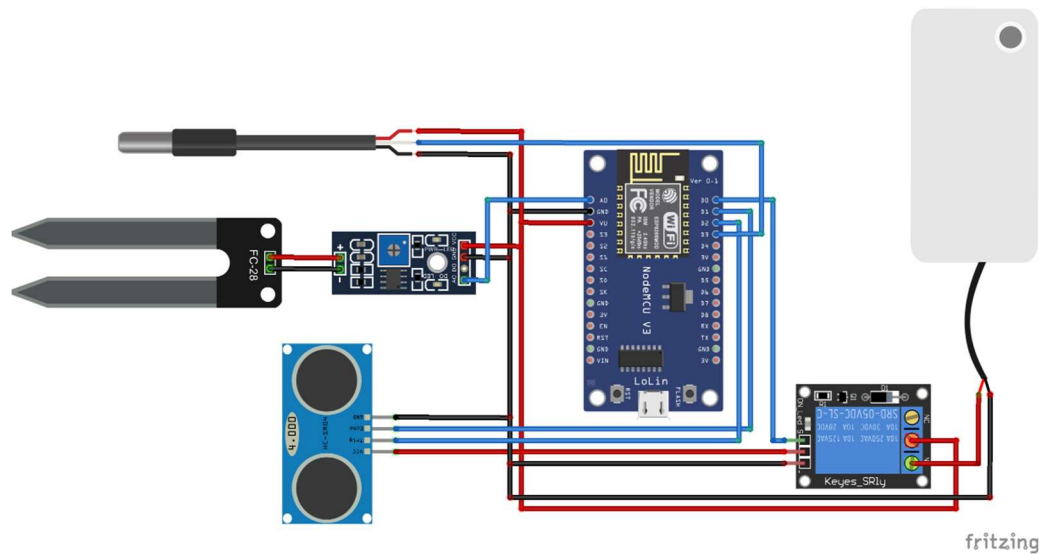
Gambar 3.5 Blok diagram sistem perangkat keras

Pada gambar 3.5 sensor dallas DS18B20, sensor *soil moisture*, dan sensor ultrasonik HC-SR04 adalah sensor yang digunakan dalam blok diagram. Sensor dallas DS18B20 untuk mengukur suhu, sensor ultrasonik HC-SR04 untuk mengukur kadar air tanah, dan sensor ultrasonik HC-SR04 untuk mengukur ketinggian air atau ketersediaan air penampungan. Pembacaan data sensor akan diolah oleh mikrokontroler NodeMCU ESP8266 dan mengirimkan hasil

pembacaan sensor ke platform firebase menggunakan komunikasi WiFi. Nilai pembacaan sensor juga dapat dipantau melalui aplikasi android. Apabila pada nilai kelembaban tanah dibawah nilai standar maka relay akan menghidupkan *water pump mini* untuk mengalirkan air pada penampungan ke media budidaya cacing tanah.

3.3.1.3 Wiring Rangkaian Perangkat Keras

Pembuatan desain perangkat keras dilakukan berdasarkan komponen yang telah ditentukan. Rangkaian perangkat keras dapat dilihat pada gambar 3.6.



Gambar 3.6 Wiring rangkaian perangkat keras

Tabel 3.2 Jalur pin sensor dallas DS18B20 dengan NodeMCU ESP8266

Pin Sensor Dallas DS18B20	Pin NodeMCU ESP8266
Vcc	VU (Catu daya 5V)
GND	GND
Data	D3

Tabel 3.3 Jalur pin sensor *soil moisture* dengan NodeMCU ESP8266

Pin Sensor <i>Soil Moisture</i>	Pin NodeMCU ESP8266
A0	A0
Vcc	VU (Catu daya 5V)
GND	GND

Tabel 3.4 Jalur pin sensor ultrasonik HC-SR04 dengan NodeMCU ESP8266

Pin Sensor Ultrasonik HC-SR04	Pin NodeMCU ESP8266
Echo	D1
Trigger	D2
Vcc	VU (Catu daya 5V)
GND	GND

Tabel 3.5 Jalur pin relay dengan NodeMCU ESP8266

Pin Relay	Pin NodeMCU ESP8266
IN1	D0
Vcc	VU (Catu daya 5V)
GND	GND
COM	VU (Catu Daya 5V)

Tabel 3.6 Jalur pin *water pump mini* dengan relay

Pin <i>Water pump mini</i>	Pin Relay
+ (positif)	NO
- (negatif)	GND

Rangkaian perangkat keras yang menggunakan mikrokontroler NodeMCU ESP8266 untuk mengelola setiap komponen digambarkan pada Gambar 3.6. Sensor Dallas DS18B20, sensor ultrasonik HC-SR04, sensor kelembaban tanah, dan relay yang terhubung ke pompa air mini semuanya dibaca oleh ESP8266 NodeMCU, yang diprogram menggunakan Arduino IDE.

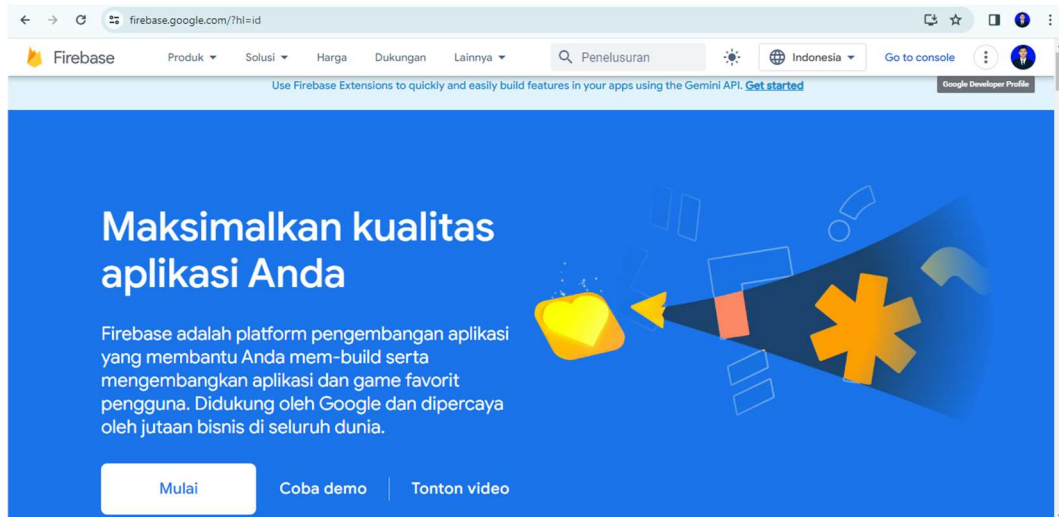
3.3.2 Perancangan Perangkat Lunak

Pada bagian perancangan perangkat lunak membahas mengenai pembuatan *database* di firebase dan pembuatan aplikasi android menggunakan MIT App Inventor.

3.3.2.1 Pembuatan *Database* Pada Google Firebase

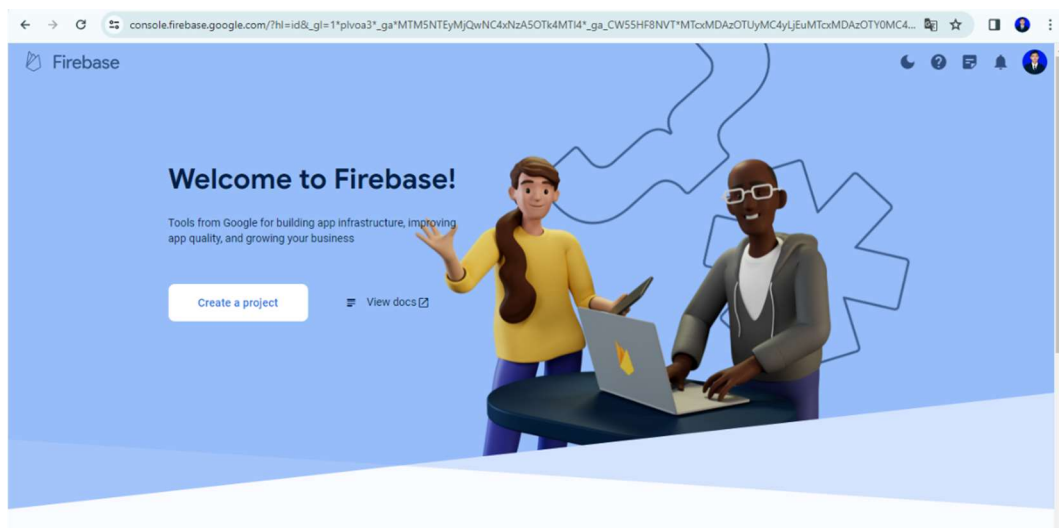
Dalam membuat *database*, penulis menggunakan firebase untuk menyimpan data dari sensor dallas DS18B20, sensor *soil moisture* dan sensor ultrasonik HC-SR04. Firebase dapat membantu untuk membangun aplikasi mobile

secara cepat, tanpa perlu melakukan pengaturan infrastruktur server. Firebase juga mendukung pengembangan aplikasi iOS, android, web, Unity, dan C++. Salah satu fitur yang dapat dimanfaatkan adalah *Realtime Database*. *Firestore Database* dapat digunakan untuk penyimpanan dan sinkronisasi data aplikasi dalam satuan milidetik. Pembuatan autentikasi pengguna juga cukup sederhana dan aman.



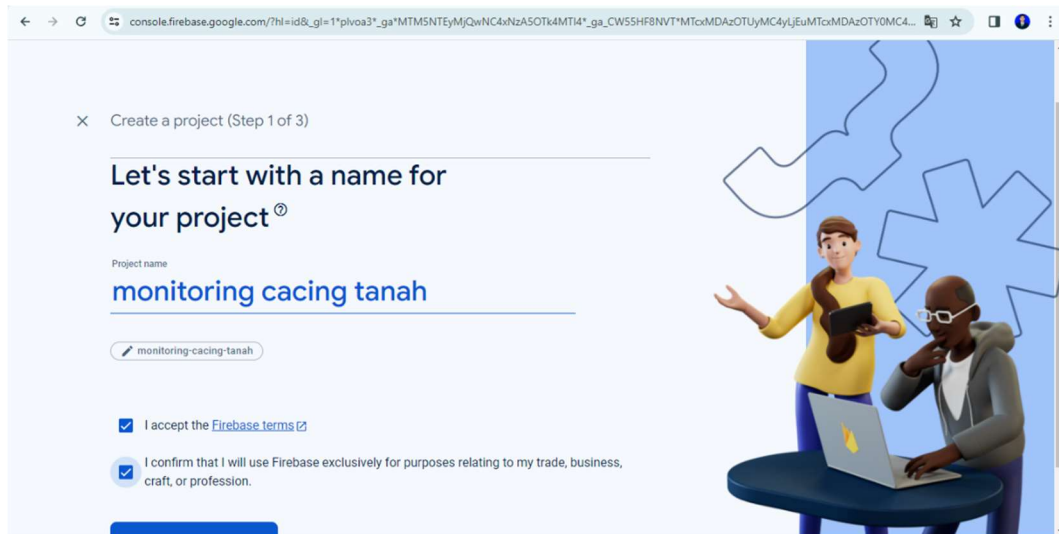
Gambar 3.7 Tampilan awal halaman Firebase

Gambar 3.7 menunjukkan tampilan awal halaman website Google Firebase, pembuatan *database* dilakukan dengan menggunakan akun gmail. Langkah awal untuk membuat *database* yaitu dengan klik tombol mulai yang tertera pada website Google Firebase.



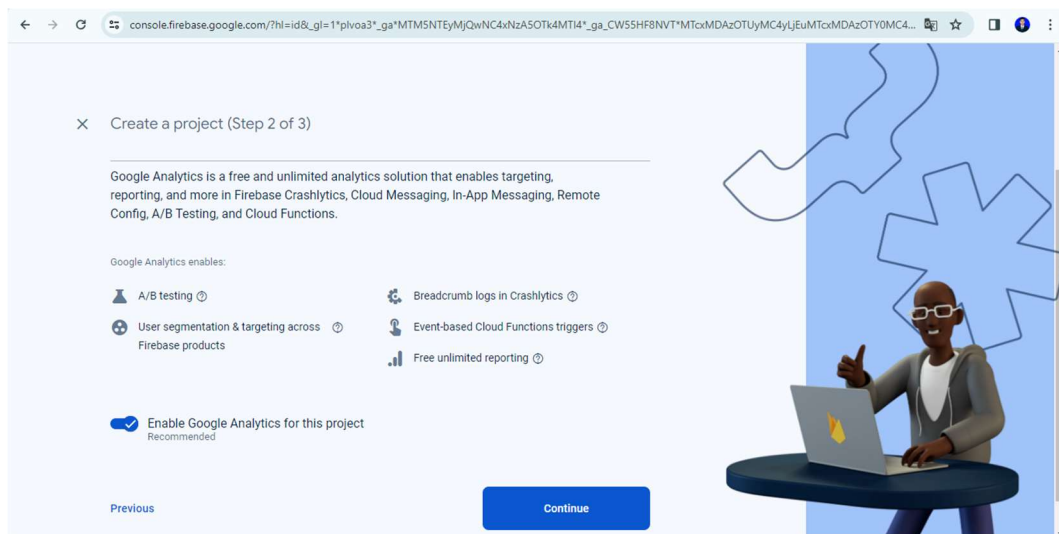
Gambar 3.8 Tampilan pembuatan proyek Firebase

Gambar 3.8 menunjukkan tampilan pembuatan proyek Firebase. Untuk membuat proyek baru, klik *Create a project* yang tertera pada tampilan tersebut.



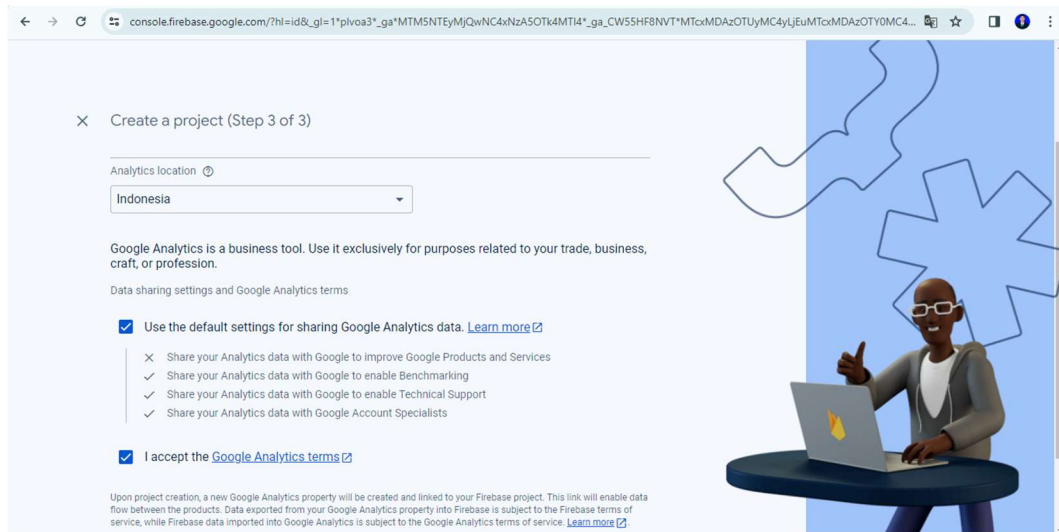
Gambar 3.9 Tampilan pemberian nama proyek Firebase

Gambar 3.9 menunjukkan tampilan saat membuat proyek baru. Di bagian ini, pengguna diminta untuk memasukkan nama proyek dan menyetujui semua persyaratan yang diminta oleh layanan Google Firebase.



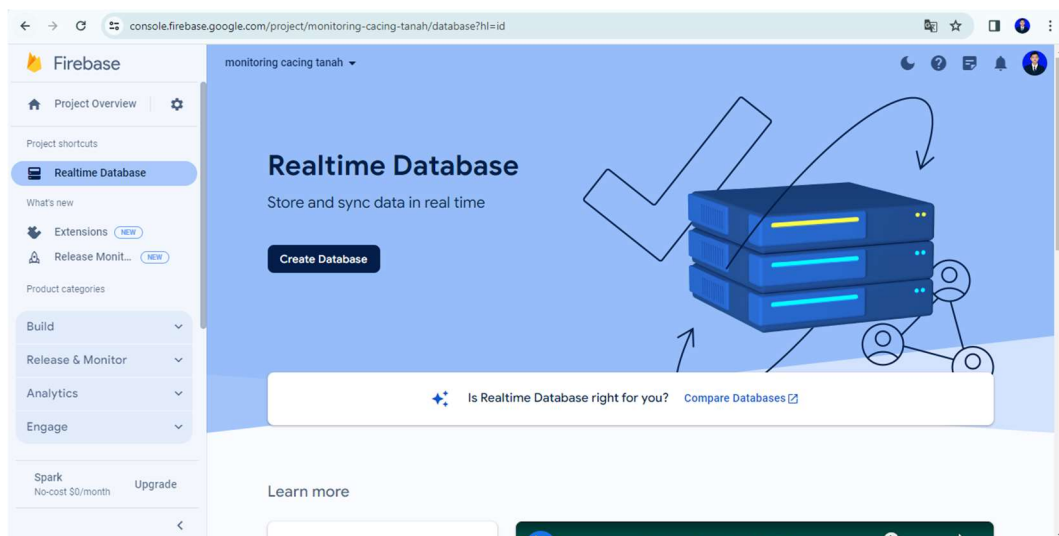
Gambar 3.10 Tampilan langkah ke 2 dari pembuatan proyek Firebase

Pada gambar 3.10 menunjukkan tampilan langkah ke 2 pembuatan *project* firebase, yang dimana pada tampilan ini berisi informasi dari fitur *Google Analytics* yang ada di firebase. Pengguna menautkan proyek dengan *Google Analytics*, yang memungkinkan pelacakan dan analisis penggunaan aplikasi secara mendalam. Setelah proyek dibuat, firebase menyediakan berbagai alat dan layanan, termasuk *database real-time*, autentikasi pengguna, dan penyimpanan *cloud*, yang dapat diintegrasikan dengan aplikasi untuk meningkatkan fungsionalitas dan performa.



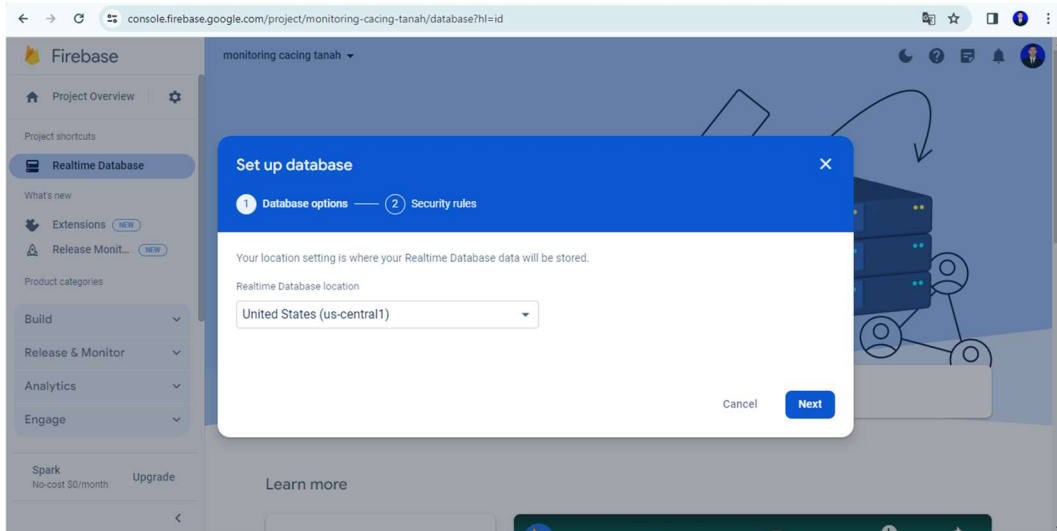
Gambar 3.11 Tampilan pemilihan lokasi *analytics*

Gambar 3.11 menunjukkan tampilan pemilihan lokasi *analytics*. Pada langkah ini pengguna diminta untuk memilih lokasi *analytics* berdasarkan negara atau wilayah organisasi pengguna. Setelah memilih lokasi *analytics* dan menyetujui persyaratan firebase, pengguna dapat mengklik *Create project*.



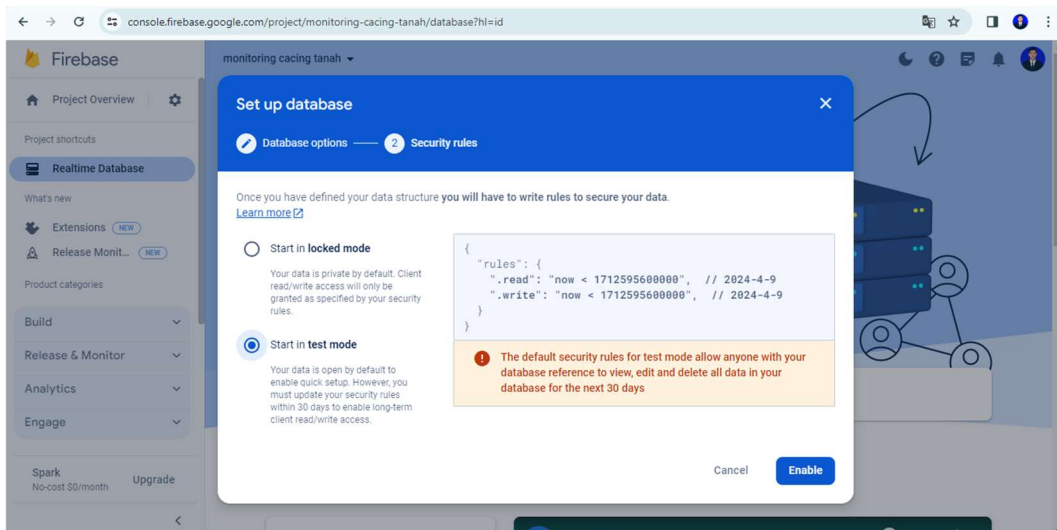
Gambar 3.12 Tampilan pembuatan database secara *realtime*

Gambar 3.12 menunjukkan antarmuka untuk pembuatan database. *Realtime Database* merupakan jenis *database* yang dapat dimanfaatkan karena sensor dapat dipantau langsung secara *realtime*. Pengguna dapat masuk ke bagian *Realtime Database* dan memilih opsi *Create Database* untuk membuat database menjadi *realtime*.



Gambar 3.13 Tampilan *Set up database* bagian *Database options*

Pada Gambar 3.13 merupakan *set up database options* yang dimana pengguna memilih lokasi *realtime database*.



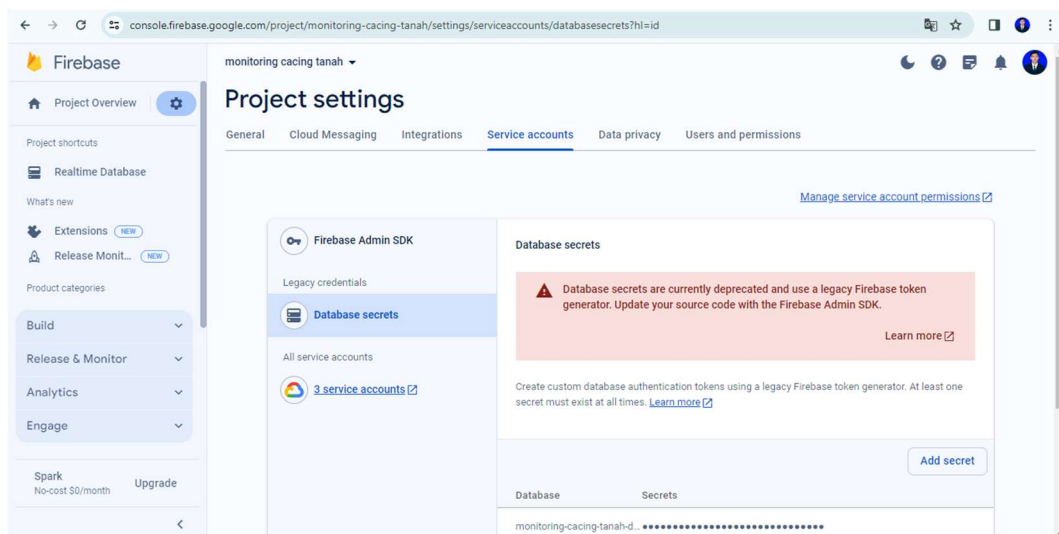
Gambar 3.14 Tampilan *Set up database* bagian *Security rules*

Pada gambar 3.14 merupakan tampilan *set up security rules* dari *database*. Ada dua mode untuk aturan keamanan, yaitu mode lock dan mode test. Karena penelusuran informasi sensor ini akan ditampilkan dalam aplikasi android yang dibuat menggunakan MIT App Inventor, maka database dibuat menggunakan mode test. Dalam mode test ini, nilai sensor yang muncul di firebase dapat diteruskan ke aplikasi android.



Gambar 3.15 Tampilan Rules Database

Gambar 3.15 merupakan tampilan dari *rules database*. Pada bagian *rules* pengguna perlu mengedit kondisi *read* dan *write* yang awalnya berisi default menjadi *true*. Arti *true* pada mode ini akan memberikan akses ke semua *platform* atau pihak ketiga selain dari firebase.

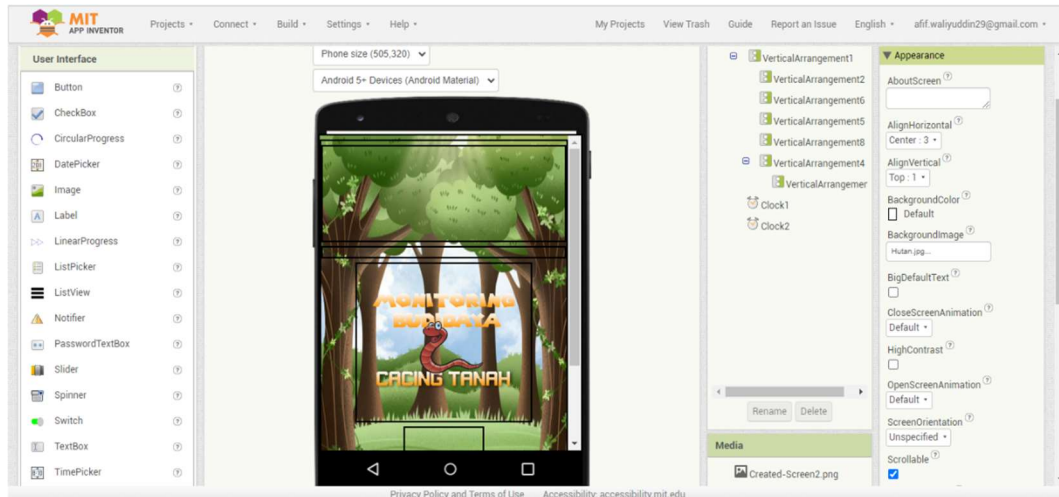


Gambar 3.16 Tampilan Service Accounts

Pada gambar 3.16 menunjukkan antarmuka *service accounts* yang berisi token database firebase. Token ini digunakan untuk menghubungkan database firebase dengan program yang dibuat di Arduino IDE, dimana token ini dimasukkan ke firebase auth dan bagian token firebase saat membuat aplikasi android di MIT App Inventor.

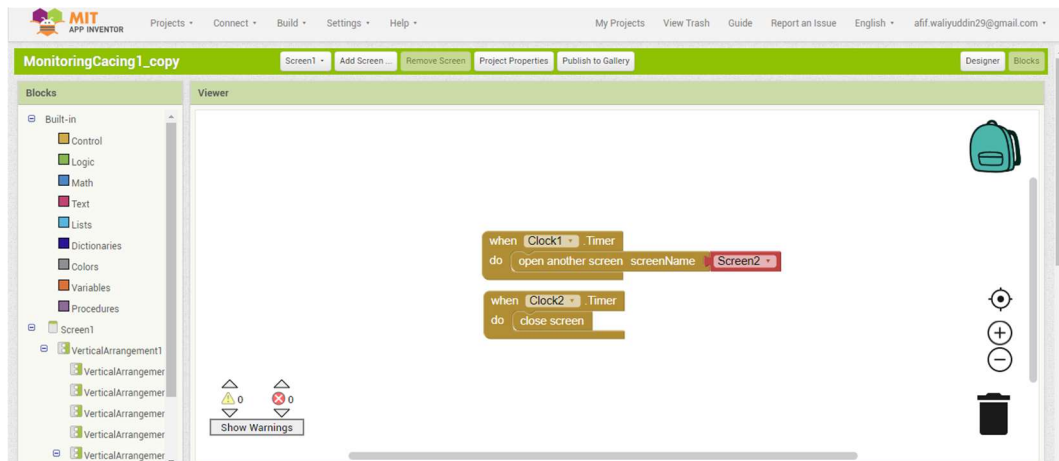
3.3.2.2 Pembuatan Aplikasi Android Pada MIT App Inventor

Dalam penelitian ini, penulis mengembangkan aplikasi android menggunakan MIT App Inventor. MIT App Inventor adalah tahap pengembangan aplikasi android sederhana yang dapat diakses secara online melalui situs web, cocok untuk pemula dalam pembuatan aplikasi.



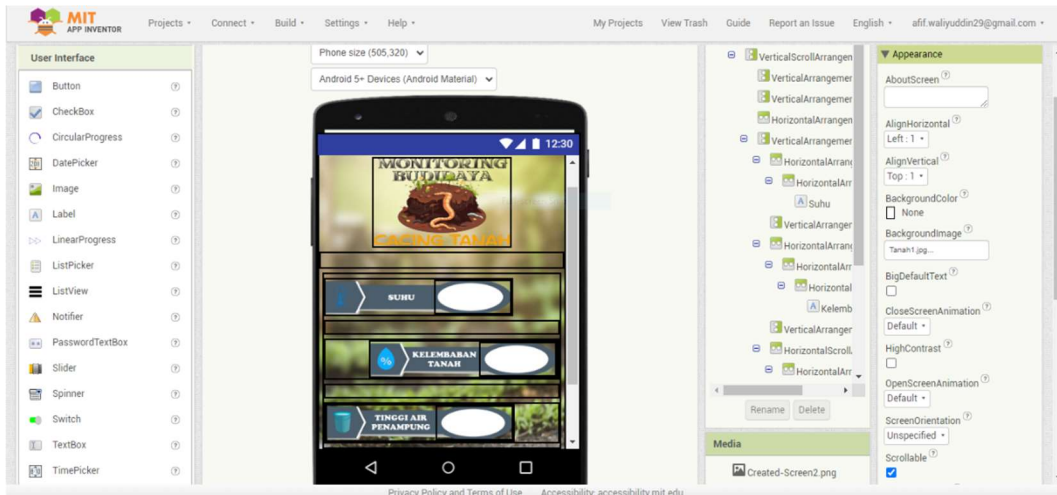
Gambar 3.17 Tampilan desain *screen 1*

Gambar 3.17 menampilkan desain tampilan awal dari *screen 1* pada aplikasi android yang dibuat menggunakan MIT App Inventor. Tampilan *screen 1* difungsikan sebagai layar pembuka yang akan beralih ke layar berikutnya.



Gambar 3.18 Tampilan *Block Editor Screen 1*

Gambar 3.18 menunjukkan tampilan *block editor screen 1*, yang merupakan proses untuk membuka layar 1 dengan rentang waktu 3 detik. Layar 2 akan terbuka secara otomatis setelah aplikasi menjalankan layar pembuka atau layer pertama selama tiga detik.



Gambar 3.19 Tampilan desain *screen 2*

Gambar 3.19 menunjukkan desain tampilan *screen 2* pada aplikasi android yang telah dibuat menggunakan MIT App Inventor. *Screen 2* digunakan untuk menampilkan nilai yang didapatkan secara *realtime* dari database firebase.



Gambar 3.20 Tampilan *Block Editor Screen 2*

Gambar 3.20 menunjukkan tampilan *block editor screen 2*, yang dimana *block editor* ini berfungsi untuk menampilkan nilai sensor yang didapatkan dari firebase secara *realtime* dan dilengkapi dengan notifikasi pada aplikasi android yang telah dibuat. Notifikasi pada aplikasi android akan muncul ketika nilai sensor yang didapatkan dari firebase kurang atau melebihi dari nilai standar yang telah ditetapkan. Setiap blok memiliki perintah atau operasi tertentu yang dapat dihubungkan dengan blok lainnya untuk membentuk program yang kompleks. Dengan pendekatan berbasis blok ini, MIT App Inventor mempermudah pembuatan aplikasi dengan memungkinkan pengguna melihat hubungan antar bagian kode secara langsung dan meminimalkan risiko kesalahan sintaksis.

3.4 SKENARIO PENGUJIAN SISTEM

Rencana pengujian sistem pemantauan kualitas lingkungan media ternak pada budidaya cacing tanah, akan penulis uraikan pada subbab skenario pengujian sistem.

3.4.1 Skema Pengujian Sensor Dallas DS18B20

Dengan membandingkan nilai pengukuran sensor Dallas DS18B20 yang ditunjukkan pada layar Arduino IDE dengan nilai yang diperoleh dari alat pengukur termometer digital, nilai *error* dan keakuratan sensor dapat dicari dalam pengujian perangkat tersebut. Sensor Dallas DS18B20 diuji di tiga lingkungan yang berbeda, yaitu pada suhu dingin, suhu panas, dan suhu ruangan tertutup dengan total keseluruhan pengujian 30 data. Nilai *error* pada pengujian bisa didapatkan dengan rumus berikut :

$$\text{nilai error} = \frac{\text{nilai pembanding} - \text{nilai pengujian}}{\text{nilai pembanding}} \times 100\% \quad (3.1)$$

3.4.2 Skema Pengujian Sensor *Soil Moisture*

Keakuratan sensor kelembaban tanah yang menggunakan ADC 10-bit diuji dalam pengujian sensor kelembaban tanah. Pengujian akurasi sensor kelembaban tanah dilakukan untuk menemukan nilai *error* atau kesalahan dengan melihat nilai yang diperoleh dari alat ukur dan nilai pengukuran sensor, pada kondisi tanah lembab, kering, dan basah dengan total pengujian 30 data pengujian. Pengukur kelembaban tanah yang dapat mengukur nilai kelembaban tanah merupakan alat ukur *soil moisture* meter, yang digunakan untuk membandingkan nilai sensor. Hal ini akan menjadi acuan untuk mencari nilai kesalahan atau nilai *error* pada sensor *soil moisture*. Nilai *error* pada pengujian bisa didapatkan dengan rumus berikut :

$$\text{nilai error} = \frac{\text{nilai pembanding} - \text{nilai pengujian}}{\text{nilai pembanding}} \times 100\% \quad (3.2)$$

3.4.3 Skema Pengujian Sensor Ultrasonik HC-SR04

Dengan membandingkan nilai pengukuran sensor yang ditunjukkan pada serial monitor Arduino IDE dengan nilai sebenarnya dari alat pengukur meter, uji sensor ultrasonik HC-SR04 dilakukan untuk memastikan nilai kesalahan atau *error*. Hingga tiga puluh set data berbeda dengan berbagai keadaan ketinggian air diuji.

dimana ketinggian air yang diukur yaitu jarak sensor terhadap permukaan air yang ditempatkan pada penampung. Pengujian dilakukan dari jarak 2cm hingga 400cm dengan mengubah tinggi sensor dengan penampungan. Nilai *error* pada pengujian bisa didapatkan dengan rumus berikut :

$$\text{nilai error} = \frac{\text{nilai pembanding} - \text{nilai pengujian}}{\text{nilai pembanding}} \times 100\% \quad (3.3)$$

3.4.4 Skema Pengujian *Quality of Service* (QoS)

Penulis dapat mengetahui hasil pengujian QoS yang dihasilkan oleh WiFi.

3.4.4.1 Pengujian *Throughput*

Mengetahui ukuran sebenarnya dari transmisi data adalah tujuan dari pengujian *throughput*. Untuk pengujian *throughput*, pengiriman paket dalam *interval* waktu 1 menit diuji dengan berbagai jarak. Pengujian dalam penelitian ini dimulai dari jarak 1 meter hingga 10 meter dan berlanjut hingga jarak 30 meter dengan kelipatan 5 meter. Pengujian *throughput* menggunakan aplikasi wireshark untuk merekam dan menganalisis paket data yang dikirim.

3.4.4.2 Pengujian *Delay*

Untuk menentukan berapa lama waktu yang dibutuhkan data untuk mencapai tujuannya, dilakukan pengujian *delay*. Pengujian menggunakan jarak yang berbeda dan batas waktu 1 menit, paket dikirim untuk melakukan tes penundaan atau *delay*. Pengujian dalam penelitian ini dimulai dari jarak 1 meter hingga 10 meter dan berlanjut hingga jarak 30 meter dengan kelipatan 5 meter. Pengujian *delay* menggunakan aplikasi wireshark untuk merekam dan menganalisis paket data yang dikirim.

3.4.4.3 Pengujian *Packet Loss*

Tujuan dari tes *packet loss* adalah untuk menentukan berapa banyak data yang hilang dalam paket selama transmisi. Untuk menilai kehilangan paket, paket dikirim dengan jarak yang bervariasi selama 1 menit. Pengujian dalam penelitian ini dimulai dari jarak 1 meter hingga 10 meter dan berlanjut hingga jarak 30 meter dengan kelipatan 5 meter. Pengujian *packet loss* menggunakan aplikasi wireshark untuk merekam dan menganalisis paket data yang dikirim.

3.4.5 Skema Pengujian Keseluruhan Sistem

Untuk memastikan sistem dapat berfungsi sesuai dengan desain, pengujian sistem secara keseluruhan dilakukan. Pengujian dilakukan dengan menguji aplikasi Android yang telah dikembangkan dan dengan menguji transmisi data ke Google Firebase. Pada pengujian pengiriman data ke google firebase menggunakan 3 data, yaitu data suhu pada sensor dallas DS18B20, data kelembaban tanah pada sensor *soil moisture* dan data ketinggian air bak penampungan pada sensor ultrasonik HC-SR04. Pengujian pada masing – masing data dilakukan sebanyak 30 kali percobaan. Kemudian pada pengujian aplikasi android dilakukan untuk menampilkan nilai suhu, kelembaban tanah dan ketinggian air pada bak penampungan serta menampilkan peringatan pada aplikasi ketika nilai sensor melebihi dari batas yang ditentukan. Pada notifikasi aplikasi android, jika nilai suhu $<15^{\circ}\text{C}$ atau $>32^{\circ}\text{C}$ maka aplikasi android akan menampilkan peringatan “Suhu Lingkungan Tidak Ideal”. Pengujian suhu dilakukan dengan cara memberikan suhu dingin dan panas pada sensor dallas DS18B20. Pada pengujian notifikasi kelembaban tanah, apabila nilai kelembaban tanah $<60\%$ maka aplikasi akan menampilkan peringatan “Kelembaban Tanah Rendah” dan apabila nilai kelembaban tanah $>80\%$ maka aplikasi akan menampilkan peringatan “Kelembaban Tanah Tinggi”. Pengujian kelembaban tanah dilakukan dengan cara memberikan air ke media tanah. Pengujian notifikasi ketinggian air pada bak penampungan, apabila ketinggian air dibawah $<5\text{cm}$ maka aplikasi akan menampilkan peringatan “Air akan habis, segera isi ulang”. Setelah melakukan pengujian pengiriman data ke google firebase dan pengujian aplikasi android, langkah selanjutnya yaitu uji coba sistem yang telah dirancang ke media ternak budidaya cacing tanah. Uji coba sistem ini dilakukan diruangan tertutup untuk mengetahui apakah sistem yang dirancang sudah berjalan dengan baik atau tidak.