

ALAT CUCI TANGAN OTOMATIS BERBASIS IOT

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan
Diploma III Politeknik Manufaktur Negeri Bangka Belitung



Disusun oleh:

Selly Rahmawanti	NIM	0031825
Zalpian	NIM	0031830

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG
TAHUN 2021**

LEMBAR PENGESAHAN

JUDUL PROYEK AKHIR

ALAT CUCI TANGAN OTOMATIS BERBASIS IOT

Oleh :

Selly Rahmawati /0031825
Zalpian /0031830

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan Diploma III Politeknik Manufaktur Negeri Bangka Belitung

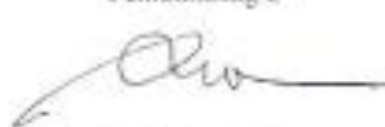
Menyetujui,

Pembimbing 1



Indra Dwisaputra, M.T.

Pembimbing 2



Irwan, M.Sc., Ph.D.

Penguji 1



Aan Febriansyah, M.T.

Penguji 2



Dr. Parliyan Silalahi, M.Pd.

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini:

Nama Mahasiswa 1 : Selly Rahmawanti NIM : 0031825
Nama Mahasiswa 2 : Zalpian NIM : 0031830

Dengan Judul : Alat Cuci Tangan Otomatis Berbasis IoT.

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 20 Agustus 2021

Nama Mahasiswa

Tanda Tangan

1. Selly Rahmawanti



2. Zalpian



ABSTRAK

Mencuci tangan dengan menggunakan sabun dapat menghilangkan bakteri dan kuman pembawa penyakit hanya dengan durasi 20 detik. Untuk mempermudah proses mencuci tangan tanpa menyentuh kran air dan sabun maka dibuat alat cuci tangan dengan mesin otomatis. Proyek Akhir dengan judul “Alat Cuci Tangan Otomatis Berbasis IoT” ini dirancang menggunakan Sensor Proximity, Sensor Infrared, dan Sensor Suara Voice Recognition V3 yang bertujuan untuk mendeteksi tangan ataupun perintah suara. Untuk memantau volume air dan sabun, digunakan sensor Ultrasonik yang dikendalikan menggunakan Arduino Uno, selanjutnya dihubungkan dengan jaringan Wifi menggunakan NodeMCU, sehingga pembacaan sensor ultrasonik dapat ditampilkan pada aplikasi Monitoring Volume. Penggunaan sensor Voice Recognition V3 digunakan untuk mendeteksi 3 perintah suara, yaitu Air Hidup, Air Mati, dan Sabun On, dengan suara laki-laki maupun perempuan. Pada saat penggunaan sensor infrared dan proximity terdapat batasan jarak dalam penggunaannya, sehingga sensor tidak bisa mendeteksi pada jarak terjauh, terutama wastafel cuci tangan . Pada alat ini batasan jarak sensor proximity adalah 8 cm dan jarak sensor infrared adalah 3 cm.

Kata kunci : *Alat cuci tangan, volume , Internet of Things, Monitoring Volume*

ABSTRACT

Washing your hands with soap can remove disease-causing bacteria and germs in just 20 seconds. To facilitate the process of washing hands without touching the faucet with water and soap, a hand washing machine with an automatic machine is made. This Final Project with the title "IoT-Based Automatic Hand Washing Tool" is designed using Proximity Sensors, Infrared Sensors, and Voice Recognition V3 Sensors which aim to detect hands or voice commands. To unify the volume of air and soap, an ultrasonic sensor is used which is controlled using Arduino Uno, then with a Wifi network using the NodeMCU, so that the ultrasonic sensor reading can be displayed in the Volume Monitoring application. The use of the Voice Recognition V3 sensor is used to detect 3 voice commands, namely Water On, Water Off, and Soap On, with male and female voices. When using infrared and proximity sensors there is a distance limit in their use, so the sensor cannot detect the farthest distance, especially at the hand washing sink. In this tool the proximity sensor limit is 8 cm and the infrared distance sensor is 3 cm.

Keywords : *Hand washing tools, volume, Internet of Things, Volume Monitoring*

KATA PENGANTAR

Puji syukur penulis panjatkan atas kehadiran Allah SWT, karena atas berkat dan rahmatNya penulis dapat menyelesaikan Laporan Proyek Akhir.

Laporan Proyek Akhir ini disusun sebagai salah satu syarat dan kewajiban mahasiswa dalam menyelesaikan kurikulum program pendidikan Dipolma III di Politeknik Manufaktur Negeri Bangka Belitung.

Laporan Proyek Akhir ini dapat diselesaikan berkat adanya usaha dan kerja sama tim yang baik serta adanya bantuan dan dukungan dari berbagai pihak. Pada kesempatan ini penulis ingin mengucapkan terima kasih kepada pihak-pihak tersebut, diantaranya :

1. Ibunda dan Ayahanda tercinta serta seluruh keluarga yang dengan penuh keikhlasan dan kesungguhan hati memberikan bantuan moral dan spiritual yang tak ternilai harganya
2. Bapak I Made Andik Setiawan, M.Eng., Ph.D. selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
3. Bapak Indra Dwisaputra, M.T. selaku pembimbing I yang telah meluangkan waktu, tenaga dan pikiran dalam mengarahkan pengarah dalam penulisan Laporan Proyek Akhir.
4. Bapak Irwan, M.Sc., Ph.D. selaku pembimbing II yang telah memberikan saran-saran dan solusi dari masalah-masalah yang dihadapi selama proses perencanaan dan pembuatan alat serta penyusunan laporan.
5. Bapak Ocsirendi M. T. selaku wali kelas III Elektronika A, D3 Teknik Elektronika Politeknik Manufaktur Negeri Bangka Belitung.
6. Seluruh staf pengajar di Politeknik Manufaktur Negeri Bangka Belitung.
7. Rekan-rekan mahasiswa Politeknik Manufaktur Negeri Bangka Belitung, yang telah banyak membantu selama pengerjaan Proyek Akhir.
8. Keluarga besar yang selalu senantiasa memberi dukungan dan doa selama proses pembuatan Proyek Akhir.

9. Pihak-pihak lain yang telah memberikan bantuan baik secara langsung dalam pembuatan Proyek Akhir yang tidak dapat disebutkan satu persatu.

Dalam pembuatan Laporan Proyek Akhir ini penulis menyadari bahwa laporan ini masih jauh dari kata sempurna, baik dari segi isi maupun penulisannya. Oleh karena itu penulis mengharapkan kritik dan saran dari semua pihak yang bersifat membangun demi kebaikan penulisan laporan kedepannya.

Akhir kata penulis ucapkan terima kasih, semoga laporan ini dapat berguna dalam menambah pengetahuan dan wawasan bagi penulis dan pembaca.

Sungailiat, 20 Agustus 2021

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	Error! Bookmark not defined.
PERNYATAAN BUKAN PLAGIAT	ii
ABSTRAK	iv
<i>ABSTRACT</i>	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah dan Batasan Masalah Proyek Akhir.....	2
1.2.1 Rumusan Masalah	2
1.2.2 Batasan Masalah.....	2
1.3 Tujuan.....	3
BAB II DASAR TEORI	4
2.1 Alat Cuci Tangan.....	4
2.2 Volume	4
2.3 Sensor <i>Ultrasonik</i> HC-SR04 sebagai Penentu Ketinggian (t).....	5
2.4 IoT (<i>Internet of Things</i>).....	7
BAB III METODE PELAKSANAAN	8

3.1	<i>Studi Literature</i>	8
3.2	Perancangan sistem	8
3.2.1	Perancangan <i>Hardware</i>	8
3.2.2	Perancangan <i>Software</i>	10
3.3	Pengujian Sistem	10
3.3.1	Uji Coba <i>Hardware</i>	10
3.3.2	Uji Coba <i>Software</i>	11
3.3.3	Uji Coba Keseluruhan	11
3.4	Analisis Data	12
3.5	Pembuatan Laporan Proyek Akhir	12
BAB IV PEMBAHASAN.....		13
4.1	Deskripsi Alat.....	13
4.2	Perancangan <i>Hardware</i>	15
4.3	Perakitan dan Pengujian <i>Hardware</i> Elektrik.....	18
4.3.1	Pengujian Sensor <i>Ultrasonik</i> pada Wadah Sabun cair.....	18
4.3.2	Pengujian Sensor <i>Ultrasonik</i> pada Wadah Air.....	24
4.3.3	Pengujian Sensor <i>Proximity</i> Menggunakan <i>Relay</i>	30
4.3.4	Pengujian Sensor <i>Infrared</i> Menggunakan <i>Relay</i>	34
4.3.5	Pengujian Sensor <i>Voice Recognition V3</i>	38
4.3.6	Uji Coba Menampilkan Data di LCD 20x4	45
4.3.7	Program Mengirim Data dari Arduino ke NodeMCU	46
4.3.8	Program Mengirim Data NodeMCU ke Android.....	46
4.4	Perakitan <i>Hardware</i> Keseluruhan	47
4.5	Perancangan dan Perakitan <i>Software</i>	47
4.6	Uji Coba <i>Software</i>	49

BAB V PENUTUP.....	53
5.1 Kesimpulan.....	53
5.2 Saran.....	53
DAFTAR PUSTAKA	54

DAFTAR GAMBAR

Gambar 2. 1 Volume Tabung.....	5
Gambar 2. 2 Cara Kerja Sensor <i>Ultrasonik</i> HC-SR04.....	6
Gambar 3. 1 Rancangan Kontruksi Alat	9
Gambar 3. 2 Blok Diagram <i>Hardware</i>	9
Gambar 3. 3 Tampilan Layar Utama Aplikasi <i>Monitoring</i> Volume.....	10
Gambar 3. 4 <i>Flowchart</i> Uji Coba Keseluruhan	11
Gambar 4. 1 Blok Diagram	14
Gambar 4. 2 Rangkaian Kontrol	18
Gambar 4. 3 Rangkaian <i>Hardware</i> Sensor <i>Ultrasonik</i> Pendeteksi Sabun Cair	19
Gambar 4. 4 Takaran Ukur Sabun Cair dengan Satuan mL.....	22
Gambar 4. 5 Grafik Data Hasil Pengukuran Volume Sabun Cair	24
Gambar 4. 6 Rangkaian <i>Hardware</i> Sensor <i>Ultrasonik</i> Pendeteksi Air.....	25
Gambar 4. 7 Takaran Ukur Air 2 L dan 5 L.....	28
Gambar 4. 8 Grafik Pengukuran Volume Air dengan Pengurangan 2L	30
Gambar 4. 9 Grafik Pengukuran Volume Air dengan Pengurangan 5 L	30
Gambar 4. 10 Rangkaian <i>Hardware</i> Sensor <i>Proximity</i>	31
Gambar 4. 11 Grafik Pembacaan Sensor <i>Proximity</i>	34
Gambar 4. 12 Rangkaian <i>Hardware</i> Sensor <i>Infrared</i>	35
Gambar 4. 13 Grafik Pembacaan Sensor <i>Infrared</i>	38
Gambar 4. 14 Rangkaian <i>Hardware</i> Sensor <i>Voice Recognition V3</i>	39
Gambar 4. 15 Perakitan <i>Hardware</i> Keseluruhan	47
Gambar 4. 16 Menu <i>Get Started</i>	48
Gambar 4. 17 Masuk dengan Akun <i>Google</i>	48
Gambar 4. 18 Tampilan Layar <i>Login</i>	49
Gambar 4. 19 Tampilan pada <i>FireBase</i>	50
Gambar 4. 20 Tampilan <i>Screen Menu</i>	50
Gambar 4. 21 Tampilan Aplikasi	51

DAFTAR TABEL

Tabel 4.1 Tabel Komponen Rangkaian Kontrol untuk Satu Alat	16
Tabel 4. 2 Skema Rangkaian <i>Hardware</i> Sensor <i>Ultrasonik</i> Pada Sabun Cair.....	20
Tabel 4. 3 Hasil Pengujian Volume Sabun Cair	23
Tabel 4. 4 Skema Rangkaian <i>Hardware</i> Sensor <i>Ultrasonik</i> pada Air	25
Tabel 4. 5 Hasil Uji Volume Air dengan Pengurangan 2 L.....	28
Tabel 4. 6 Hasil Uji Volume Air dengan Pengurangan 5 L.....	29
Tabel 4. 7 Skema Rangkaian <i>Hardware</i> Sensor <i>Proximity</i> pada Arduino.....	32
Tabel 4. 8 Skema Rangkaian <i>Hardware Relay</i> pada Arduino	32
Tabel 4. 9 Hasil Pengujian Sensor <i>Proximity</i>	33
Tabel 4. 10 Skema Rangkaian <i>Hardware Sensor Infrared</i> pada Arduino	36
Tabel 4. 11 Skema Rangkaian <i>Hardware Relay</i> pada Arduino	36
Tabel 4. 12 Hasil Pengujian Sensor <i>Infrared</i>	37
Tabel 4. 13 Skema Rangkaian <i>Hardware</i> Sensor <i>Voice Recognition V3</i>	40
Tabel 4. 14 Data Rekam Tanpa Kebisingan.....	40
Tabel 4. 15 Data Rekam Kebisingan (76db).....	41
Tabel 4. 16 Rekam Tanpa Kebisingan untuk Air Hidup (Laki-laki)	41
Tabel 4. 17 Rekam Tanpa Kebisingan untuk Air Mati (Laki-laki).....	42
Tabel 4. 18 Rekam Tanpa Kebisingan untuk Sabun On (Laki-laki).....	42
Tabel 4. 19 Data Rekam Tanpa Kebisingan untuk Air Hidup (Perempuan)	43
Tabel 4. 20 Data Rekam Tanpa Kebisingan untuk Air Mati (Perempuan).....	43
Tabel 4. 21 Data Rekam Tanpa Kebisingan untuk Sabun On (Perempuan).....	44
Tabel 4. 22 Skema Rangkaian Pengiriman Data dari Arduino ke NodeMCU.....	46
Tabel 4. 23 Uji Coba pada Alat 1	51
Tabel 4. 24 Uji Coba pada Alat 2.....	51

DAFTAR LAMPIRAN

- Lampiran 1 : DAFTAR RIWAYAT HIDUP
Lampiran 2 : PROGRAM

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sekarang ini banyak ditemukan berbagai macam jenis virus, termasuk virus *Coronavirus Disease 2019* (Covid-19). Virus Covid-19 adalah virus menular yang disebabkan oleh *Severe Acute Syndrome Coronavirus 2* (SARS-CoV-2) dapat menyerang sistem pernafasan. Virus ini mudah menular salah satunya adalah dengan menyentuh virus tersebut secara langsung dari manusia yang terpapar. Untuk mencegah penularan virus corona, dapat dilakukan dengan beberapa upaya salah satunya mencuci tangan dengan sabun selama kurang lebih 20 detik [1].

Namun keran air dan wadah sabun yang digunakan saat ini, umumnya masih bersentuhan langsung dengan tangan sehingga dirasa kurang efektif, maka dikembangkanlah sebuah alat pencuci tangan otomatis dengan mengeluarkan air dan sabun tanpa disentuh [2]. Pengeluaran air dan sabun cair secara otomatis ini dapat mengurangi faktor tersentuhnya keran air dan sabun secara bergantian untuk mengurangi jumlah penyebaran bakteri melalui tangan [3]. Sehingga dibuat penelitian alat cuci tangan yang dirancang dengan sistem menggunakan sebuah sensor yang dapat mendeteksi keberadaan tangan kita seperti sensor *optocpler* dan sebagainya [4].

Pada penelitian yang berjudul “Perancangan Mesin Cuci Tangan Otomatis dan Higienis Berbasis Kamera”, masih memiliki kekurangan yaitu air akan terus mengalir saat pengguna masih berada di depan kamera, sehingga penghematan air tidak berjalan maksimal [5].

Selain itu, penelitian yang berjudul “Rancang Bangun Mesin Cuci Tangan Otomatis *Cutato-Unhan* Berbasis Karakteristik *Antropometri*” juga masih kurang efektif dikarenakan air dan sabun bercampur pada wadah dan keran yang sama serta pada waktu yang bersamaan pula, sehingga proses mencuci tangan akan lebih lama [6].

Air dan sabun pada alat cuci tangan merupakan cairan yang dapat habis, sehingga perlu adanya pemantauan terhadap volume air dan sabun cair. Pemantauan secara manual dilakukan dengan melihat meteran air secara langsung, memerlukan waktu dan tenaga sehingga dirasa kurang efektif apalagi wadah air dan sabun diletakkan pada gedung bertingkat atau tempat yang berjauhan, jika dilakukan pemantauan setiap hari. Sehingga penelitian ini bertujuan untuk memudahkan seseorang memantau volume air dan sabun dari kejauhan sebagai prediksi adanya air dan sabun di dalam wadah, dengan memanfaatkan *Internet of Things* (IoT) [7]. Sehingga dibuatlah alat cuci tangan yang berjudul “Alat Cuci Tangan Otomatis Berbasis IoT” dengan menggunakan sensor *proximity* untuk mendeteksi keberadaan tangan kita, sensor suara (*Voice Recognition module V3*) untuk mendeteksi perintah suara yang digunakan, serta sensor *ultrasonik* untuk memantau volume cairan.

Dengan adanya alat ini dapat memudahkan seseorang dalam mencuci tangan dan memantau volume air dan sabun dari kejauhan yang berbasis internet menggunakan android apabila seseorang tidak berada didekat alat, dan juga dapat memantau volume air dan sabun dengan LCD yang terpasang pada alat apabila seseorang berada di dekat alat.

1.2 Rumusan Masalah dan Batasan Masalah Proyek Akhir

1.2.1 Rumusan Masalah

Rumusan masalah yang diangkat berdasarkan latar belakang proyek akhir ini adalah sebagai berikut:

1. Bagaimana mengembangkan alat cuci tangan otomatis berbasis IoT?
2. Bagaimana efektifitas alat cuci tangan otomatis berbasis IoT pada jarak yang berjauhan?

1.2.2 Batasan Masalah

Batasan masalah yang diangkat berdasarkan latar belakang proyek akhir ini adalah sebagai berikut:

1. Penelitian ini tidak membahas tingkat kebersihan dalam mencuci tangan terhadap kuman, virus, dan bakteri.
2. Parameter yang diukur pada proyek akhir ini hanyalah volume air dan sabun cair.
3. Alat ini tidak dapat bekerja jika tidak ada sumber listrik.
4. Jenis sabun yang di gunakan adalah sabun cair.
5. Sistem tidak mengatur proses isi ulang air dan sabun.

1.3 Tujuan

1. Mengembangkan alat cuci tangan berbasis IoT dengan menggunakan sensor jarak dan sensor suara.
2. Menguji efektivitas alat cuci tangan otomatis berbasis IoT dengan sebuah aplikasi *monitoring*.

BAB II

DASAR TEORI

2.1 Alat Cuci Tangan

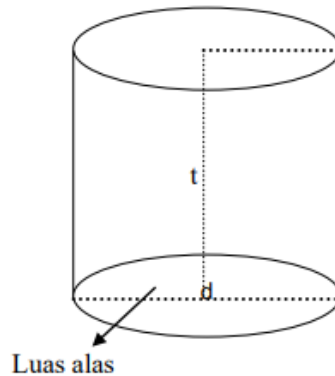
Menurut WHO (*World Health Organization*) tahun 2009, mencuci tangan dengan menggunakan sabun dapat menghilangkan bakteri dan kuman pembawa penyakit hanya dengan durasi 20-30 detik. Sehingga sekarang ini banyak dibuat alat cuci tangan terutama dengan mesin otomatis. Perancangan mesin cuci tangan otomatis sudah dilakukan sejak tahun 1999, yang terdiri dari kran air otomatis dan sabun otomatis dengan menggunakan sensor *infrared*. Dalam beberapa tahun terakhir, mulai dikembangkan alat cuci tangan menggunakan kamera, *fotodiode*, dan sensor PIR. Hal ini dilakukan untuk mempermudah proses mencuci tangan tanpa menyentuh kran dan tempat sabun [8].

Kegiatan mencuci tangan sangat memerlukan air dan sabun, sehingga hal ini harus selalu dipastikan ketersediaannya, sehingga sangat bergantung terhadap volume air dan sabun.

2.2 Volume

Volume atau kapasitas merupakan penghitungan seberapa banyak ruang yang bisa ditempati dalam suatu objek. Objek bisa dikatakan bervolume apabila didalamnya terdapat ruang. Objek ini biasanya berupa benda yang bentuknya beraturan dan tidak beraturan. Volume digunakan untuk menentukan *massa* dari suatu benda [9].

Salah satu contoh benda bervolume adalah tabung atau silinder. Tabung merupakan bangun ruang dengan 3 dimensi. Tabung mempunyai 3 sisi dan 4 rusuk. Volume tabung dapat dihitung dengan rumus sebagai berikut:



Gambar 2. 1 Volume Tabung [10]

Volume (V) dihitung dari phi kali radius/jari-jari(r) kali tinggi(t), ditunjukkan pada persamaan:

$$V = \text{Luas Alas} \times \text{Tinggi} \dots \dots \dots (2.1)$$

$$V = \pi r^2 t \dots \dots \dots (2.2)$$

V : Volume Tabung (cm^3)

π : Phi = 3,14

r : Jari-jari (cm)

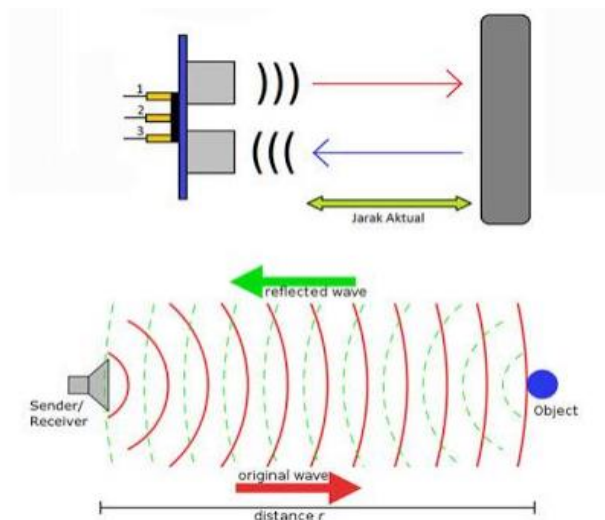
t : Tinggi (cm)

2.3 Sensor Ultrasonik HC-SR04 Sebagai Penentu Ketinggian (t)

Sensor ultrasonik ini memiliki 2 prinsip kerja, yaitu sebagai pendeteksi gelombang ultrasonik dan pembangkit gelombang ultrasonik. Sensor ini mampu mendeteksi gelombang ultrasonik, yaitu gelombang suara yang memiliki frekuensi diatas frekuensi pendengaran manusia. Alat ini bisa mendeteksi jarak benda dari 2 cm sampai 400 cm [9]. Untuk karakteristik dan cara kerja sensor dapat dilihat pada tabel 2.1 dan gambar 2.2

Tabel 2. 1 Karakteristik Sensor *Ultrasonik* HC-SR04

No	Karakteristik
1	Tegangan sumber operasi tunggal 5.0 V
2	Konsumsi arus 15 Ma
3	Frekuensi operasi 40 kHz
4	Minimum pendeteksi jarak 0.02 m (2 cm)
5	Maksimum pendeteksian jarak 4 m
6	Jangkauan deteksi 2 cm sampai 400 cm
7	Sudut pantul gelombang pengukuran 15 derajat
8	Minimum waktu penyulutan 10 <i>mikrodetik</i> dengan pulsa berlevel <i>Transistor Transistor Logic</i> (TTL)
9	Pulsa deteksi berlevel TTL dengan durasi yang bersesuaian dengan jarak deteksi
10	Dimensi 45 x 20 x 15 mm



Gambar 2. 2 Cara Kerja Sensor *Ultrasonik* HC-SR04 [9]

Pada gambar 2.2 dapat diketahui bahwa cara kerja sensor *ultrasonik* HC-SR04 adalah ketika sensor diberikan tegangan *positif* pada pin *trigger* selama

10uS, maka sensor akan mengirimkan 8 step sinyal ultrasonik dengan *frekuensi* 40KHz. Setelah itu sinyal akan diterima oleh pin *Echo*. Untuk menghitung jarak benda yang dipantulkan, maka selisih waktu ketika mengirim dan menerima sinyal (*delay*) digunakan untuk menentukan jarak [9].

Untuk mendapatkan Volume menggunakan sensor *ultrasonik* digunakan rumus:

$$V = \text{Luas Alas} \times \text{Tinggi} \dots \dots \dots (2.1)$$

$$V = \pi r^2 t \dots \dots \dots (2.2)$$

$$t = t \text{ tabung} - t \text{ sensor} \dots \dots \dots (2.3)$$

sehingga:

$$V = \pi r^2 (t \text{ tabung} - t \text{ sensor}) \dots \dots \dots (2.4)$$

2.4 IoT (*Internet of Things*)

Internet of Things bertujuan untuk memperluas manfaat dan konektivitas internet yang tersambung secara terus menerus. *Internet of Things* mampu berbagi data bahkan mengontrol kerja alat dengan sambungan internet. Cara kerja *Internet of Things* yaitu memanfaatkan sebuah pemrograman dengan setiap perintah menghasilkan sebuah interaksi antara alat yang saling terhubung secara otomatis tanpa campur tangan manusia dengan jarak berapa pun selama masih terhubung ke jaringan internet. Internet menjadi penghubung di antara kedua interaksi alat tersebut, sementara manusia hanya bertugas sebagai pengatur dan pengawas bekerjanya alat tersebut secara langsung [11].

BAB III

METODE PELAKSANAAN

3.1 *Studi Literature*

Pada tahap pertama, hal yang dilakukan yaitu mencari dan mengumpulkan artikel dan jurnal dari penelitian sebelumnya mengenai alat cuci tangan otomatis. *Studi literatur* mempelajari referensi-referensi yang bersumber dari buku-buku maupun internet. Referensi-referensi ini bertujuan sebagai acuan dalam pembuatan proyek akhir sehingga dapat mengetahui perkembangan alat cuci tangan otomatis yang telah dibuat sebelumnya.

3.2 Perancangan sistem

Pada tahapan perancangan sistem pembuatan Alat Cuci Tangan Otomatis Berbasis IoT dibagi menjadi 2, yaitu perancangan *hardware* dan perancangan *software*.

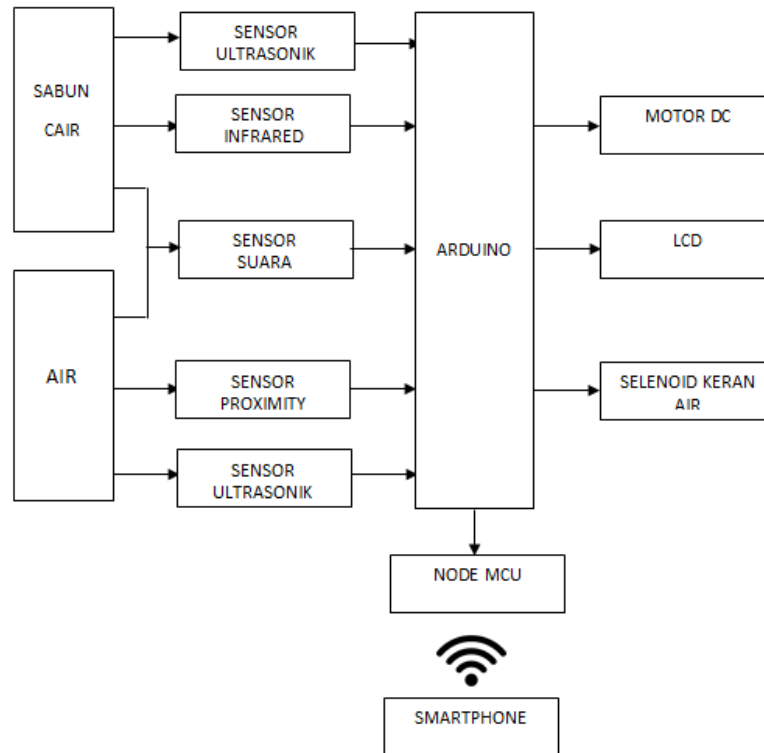
3.2.1 Perancangan *hardware*

Perancangan *hardware* dimulai dengan membuat rancangan konstruksi, dan menentukan komponen yang akan digunakan untuk pembuatan proyek akhir ini, seperti sensor *proximity*, sensor suara, sensor *ultrasonik*, NodeMCU, *solenoid valve*, *relay* dan LCD. Membuat rancangan konstruksi bertujuan untuk mendapatkan gambaran mengenai bentuk dan peralatan yang dibutuhkan dalam pembuatan konstruksi, seperti wadah air, *wastafel* dan rangka baja ringan.



Gambar 3. 1 Rancangan Kontruksi Alat

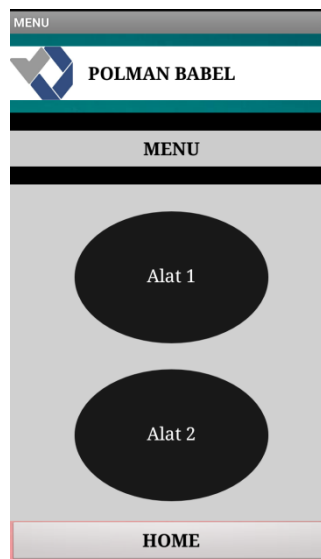
Setelah membuat rancangan kontruksi dan menentukan komponen, selanjutnya membuat blok diagram *hardware*.



Gambar 3. 2 Blok Diagram *Hardware*

3.2.2 Perancangan *software*

Perancangan *software* yaitu membuat aplikasi pemantauan volume air dan sabun cair. Aplikasi ini di buat dan di program sederhana menggunakan *MIT App Inventor*, yang di fungsikan melalui *API Blynk*. Berikut tampilan aplikasi layar utama *monitoring* volume air dan sabun cair dapat dilihat pada gambar 3.3



Gambar 3. 3 Tampilan Layar Utama Aplikasi *Monitoring* Volume

Setelah perancangan sistem sudah dilakukan, selanjutnya ke tahap pengujian sistem.

3.3 Pengujian Sistem

Pengujian sistem dilakukan setelah proses perancangan dan perakitan *hardware* dan *software*. Pengujian ini bertujuan untuk mengetahui komponen yang digunakan dapat bekerja dengan baik dan sesuai fungsi atau tidak. Uji coba ini terdiri dari 3 tahapan yaitu:

3.3.1 Uji Coba *hardware*

Pengujian *hardware* elektrik setiap komponen dilakukan satu persatu, sehingga dapat mengetahui komponen yang rusak.

Uji coba *hardware* yang dilakukan meliputi:

- Uji coba koneksi antara Arduino dan sensor *ultrasonik*

- Uji coba koneksi antara Arduino dan sensor *proximity*
- Uji coba koneksi antara Arduino dan sensor *infrared*
- Uji coba koneksi antara Arduino dan sensor suara (*voice recognition*)
- Uji coba koneksi antara Arduino dan *Liquid Crystal Display* (LCD)
- Uji coba koneksi serial komunikasi antara Arduino dan NodeMCU
- Uji coba koneksi antara NodeMCU dan Android

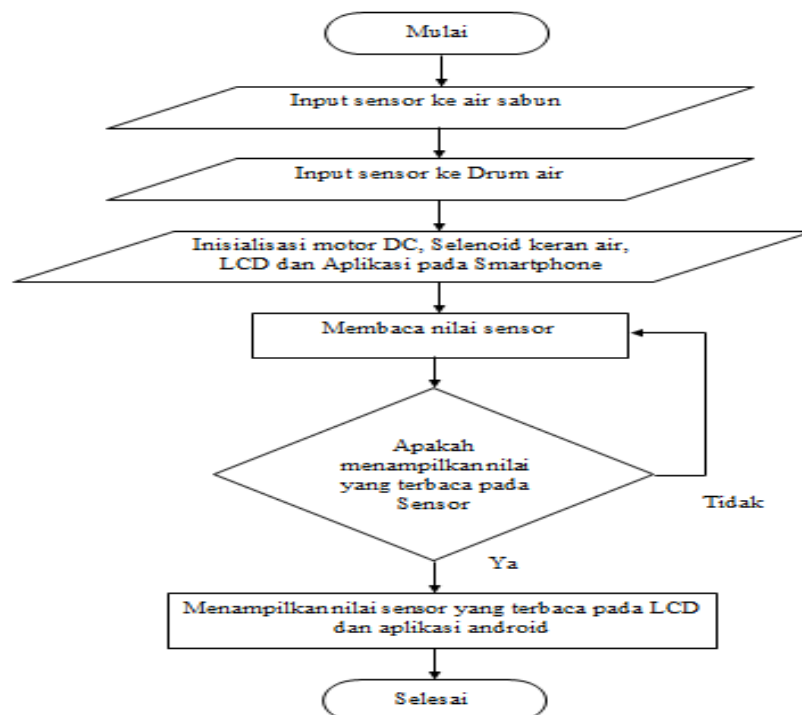
Setelah pengujian *hardware* telah dilakukan satu per satu, selanjutnya ke tahap perakitan *hardware* elektrik keseluruhan untuk menghubungkan semua komponen.

3.3.2 Uji Coba Software

Pengujian aplikasi *monitoring* pada Android dengan menggunakan aplikasi yang telah dibuat menggunakan *MIT App Inventor*.

3.3.3 Uji Coba Keseluruhan

Uji coba keseluruhan dilakukan untuk mengetahui hasil pengujian dari keseluruhan alat.



Gambar 3. 4 *Flowchart* Uji Coba Keseluruhan

3.4 Analisis Data

Pada tahap ini merupakan tahap analisis data pada hasil pendeteksian sensor suara dan pemantauan volume air dan sabun cair. Hal ini bertujuan untuk melihat kekurangan terhadap alat yang telah dibuat baik dari segi konstruksi, rangkaian kontrol, program dan aplikasi yang dibuat.

3.5 Pembuatan Laporan Proyek Akhir

Tahap pembuatan laporan merupakan tahap terakhir dalam pembuatan proyek akhir. Bertujuan untuk merangkum keseluruhan yang berhubungan dengan proyek akhir dan memberikan informasi yang didapat pada proyek akhir yang telah dilakukan.

BAB IV

PEMBAHASAN

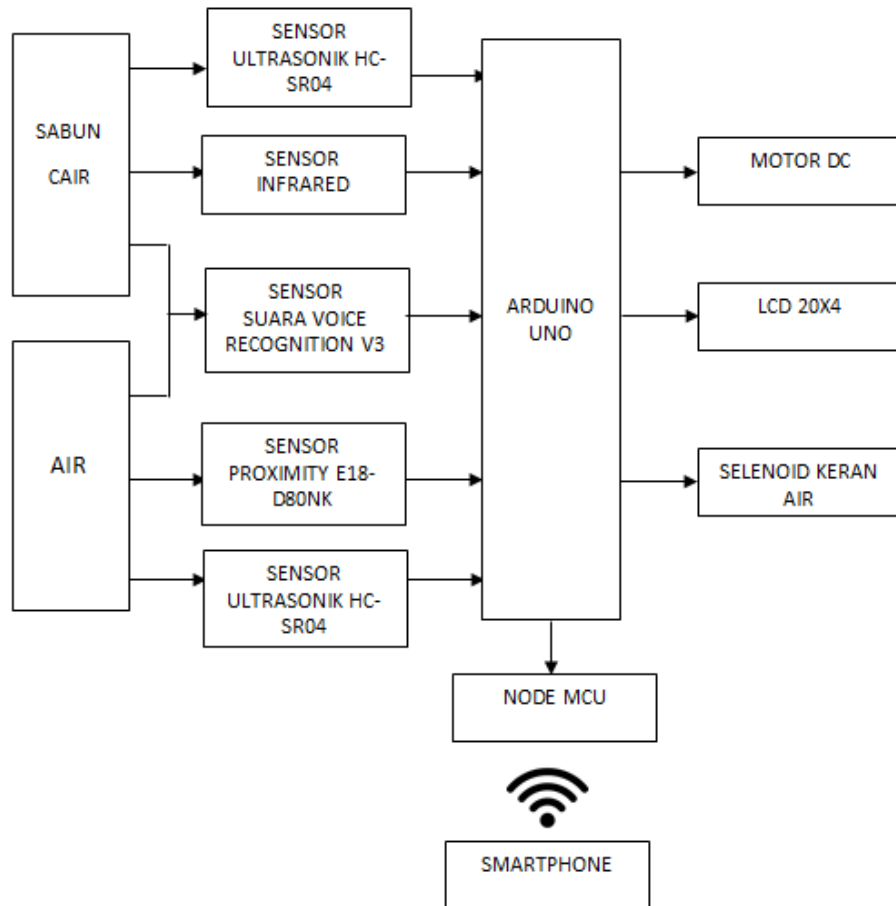
Bab ini menguraikan proses pembuatan proyek akhir dengan judul “Alat Cuci Tangan Otomatis Berbasis IoT” berdasarkan metode yang telah dipaparkan pada bab sebelumnya. Secara umum bab ini menjelaskan tentang:

1. Deskripsi alat
2. Perancangan *hardware*
3. Perakitan dan pengujian *hardware* elektrik
4. Perakitan keseluruhan *hardware* elektrik
5. Perancangan *software*
6. Uji coba *software*

4.1 Deskripsi Alat

Alat cuci tangan otomatis berbasis IoT adalah alat yang digunakan untuk mempermudah proses mencuci tangan tanpa harus menyentuh keran air dan sabun, dengan menggunakan sensor *proximity*, sensor *infrared* dan sensor suara (*voice recognition*) sehingga dapat mengurangi resiko tertularnya virus dan bakteri dari tangan seseorang. Alat ini dilengkapi dengan sistem IoT yang berfungsi untuk memantau volume air dan sabun cair dengan *smartphone* menggunakan aplikasi buatan.

Berikut ini merupakan blok diagram alat cuci tangan otomatis berbasis IoT dapat dilihat pada gambar 4.1



Gambar 4. 1 Blok Diagram

Alat ini akan beroperasi apabila semua komponen terhubung pada sumber listrik dengan catu daya 12 volt. Setelah semua komponen terhubung ke sumber listrik, sensor *proximity*, sensor *infrared* dan sensor suara mulai mendeteksi keberadaan tangan dan suara seseorang dengan jarak tertentu. Sensor ini akan mengetahui keberadaan manusia, jika keberadaan tangan terdeteksi sensor *infrared* atau suara terdeteksi oleh sensor suara (*voice recognition*) dengan perintah sabun cair, yang telah di program pada Arduino Uno, maka secara otomatis akan memicu *relay* 1 untuk menghidupkan Motor DC yang berfungsi untuk mengeluarkan sabun cair dari wadah sabun. Apabila keberadaan tangan terdeteksi sensor *proximity* atau suara terdeteksi oleh sensor suara (*voice recognition*) dengan perintah air, maka secara otomatis akan memicu *relay* 2 untuk menghidupkan *solenoid* keran air yang berfungsi untuk mengeluarkan air

dari keran. Selanjutnya sensor *ultrasonik* yang terdapat pada tedmond dan wadah sabun akan mendeteksi volume air dan sabun cair, menampilkannya pada LCD dan *smartphone* menggunakan aplikasi buatan yang di program dengan kode *API Blynk* untuk menampilkan data. Pemantauan ini menggunakan sebuah Arduino Uno yang telah terprogram beserta sebuah NodeMCU sebagai pengirim data yang akan ditampilkan pada Android melalui aplikasi buatan. Pembacaan sensor juga dapat dilihat pada *link* berikut:

<http://blynk-cloud.com/Token/get/Pin>

Pin: V1 = Sabun Cair 1

V2 = Air 1

V3 = Sabun Cair 2

V4 = Air 2

4.2 Perancangan *Hardware*

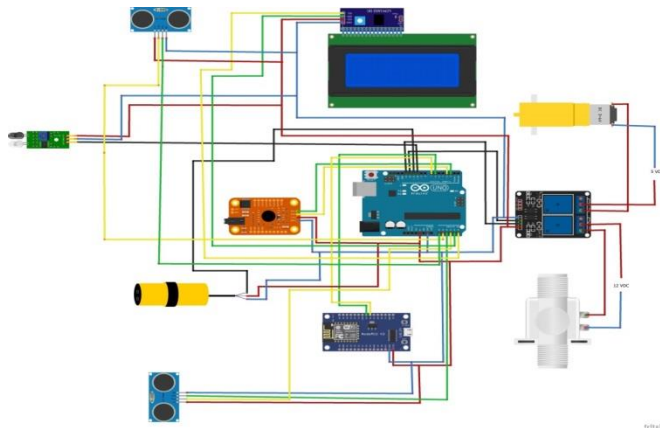
Perancangan *hardware* dilakukan dengan membuat rancangan kontruksi alat terlebih dahulu. Kontruksi alat terdiri dari tedmond air sebagai wadah penyimpanan air, *wastafel* sebagai tempat mencuci tangan, serta meja untuk meletakkan tedmond air. Setelah rancangan kontruksi selesai, selanjutnya pembuatan rancangan rangkaian kontrol. Pada rangkaian kontrol terdapat sebuah *Arduino Uno* yang digunakan untuk mengolah data nilai sensor. Rangkaian kontrol “Alat Cuci Tangan Otomatis Berbasis IoT” dirangkai di dalam sebuah *box* dengan ukuran 30x20. Pembuatan alat ini menggunakan beberapa komponen. Adapun komponen yang digunakan untuk membuat alat cuci tangan otomatis berbasis IoT dapat dilihat pada tabel 4.1

Tabel 4.1 Tabel Komponen Rangkaian Kontrol untuk satu alat

No	Peralatan yang Digunakan	Jumlah	Keterangan	Fungsi
1	Catu Daya	1 Buah		Sebagai sumber listrik untuk menghidupkan komponen
2	<i>Arduino Uno</i>	1 Buah		Digunakan untuk menghubungkan semua komponen dalam mengontrol prinsip kerja alat
3	<i>Relay</i>	2 Buah		Digunakan untuk menyalakan Motor DC dan selenoid keran air
4	<i>voice recognition</i>	1 Buah		Digunakan untuk mendeteksi perintah suara berupa Air dan Sabun
5	<i>NodeMCU</i>	1 Buah		Digunakan untuk menghubungkan IoT
6	<i>Sensor Ultrasonik</i>	2 Buah		Digunakan untuk membaca volume air dan sabun

No	Peralatan Yang Digunakan	Jumlah	Keterangan	Fungsi
7	Sensor <i>Infrared</i>	1 Buah		Untuk mendeteksi keberadaan tangan seseorang pada jarak tertentu untuk mengeluarkan sabun
8	LCD 20x4	1 Buah		Untuk menampilkan pembacaan sensor
9	Sensor <i>Proximity</i>	1 Buah		Untuk mendeteksi keberadaan tangan seseorang pada jarak tertentu untuk mengeluarkan air

Komponen-komponen tersebut disambungkan ke Arduino Uno sebagai bagian dari sistem kontrol. Setiap komponen memiliki jalur pin tersendiri untuk disambungkan ke Arduino Uno. Jika kode pin pada *hardware* tidak sama dengan yang tertulis di dalam program, maka alat juga tidak bisa berfungsi sebagaimana mestinya. Oleh karena itu, diperlukannya ketelitian dan konsentrasi yang baik dalam perangkaian komponen serta pembuatan program Arduino. Rangkaian kontrol dapat dilihat pada gambar 4.2



Gambar 4. 2 Rangkaian Kontrol

Pada gambar 4.2 merupakan rangkaian kontrol yang menjelaskan tahapan penyambungan sensor dan Arduino Uno sehingga alat dapat bekerja sebagaimana mestinya.

4.3 Perakitan dan Pengujian *Hardware* Elektrik

Selanjutnya adalah perakitan *hardware* elektrik yang merupakan proses merakit komponen untuk mengetahui sebuah komponen dalam keadaan baik atau tidak. Hal ini dapat dilakukan dengan cara menyambungkan komponen dengan Arduino Uno dan melakukan pengujian pada setiap komponen.

Pengujian komponen dilakukan dengan menguji masing-masing komponen dengan menggunakan *software* Arduino, yang hasilnya akan muncul pada layar monitor. Hal ini bertujuan untuk melihat apakah komponen bekerja dengan baik dan sesuai fungsi yang diinginkan atau tidak.

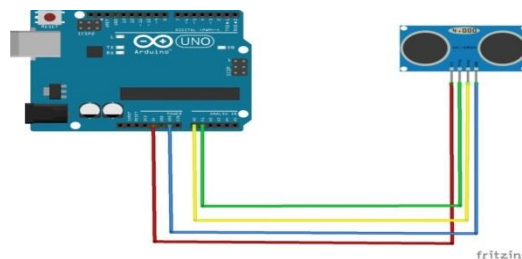
4.3.1 Pengujian Sensor *Ultrasonik* pada Wadah Sabun cair

Pengujian sensor *ultrasonik* pada sabun cair merupakan proses pembacaan jarak dan kalibrasi pada durasi tertentu. Pengujian ini dilakukan sebanyak 5 kali, yaitu dengan mengurangi volume sabun cair sejumlah 100 mL pada setiap percobaan. Cara menguji sensor ultrasonik pada sabun cair:

1. Hubungkan sensor *ultrasonik* dan Arduino Uno sesuai dengan warna kabel. Sensor *ultrasonik* terdiri dari 4 warna kabel yaitu warna merah,

warna hijau, warna kuning dan warna biru. Warna merah untuk VCC (sumber tegangan positif sensor), sehingga VCC pada sensor *ultrasonik* dihubungkan pada 5 V Arduino Uno. Warna hijau untuk *Trlg* (*Trigger* atau pin yang digunakan sebagai pembangkit sinyal ultrasonik), sehingga *Trlg* dihubungkan pada Pin A1. Warna kuning untuk *Echo* (*Receive* atau pin yang digunakan sebagai pendeteksi sinyal pantulan ultrasonik), sehingga *Echo* dihubungkan pada Pin A0. Warna biru untuk GND, sehingga GND pada sensor *ultrasonik* dihubungkan pada GND Arduino Uno.

2. Pasang sensor *ultrasonik* pada tutup wadah sabun cair. Setelah itu, *upload* program ke Arduino Uno. Tunggu sampai sensor *ultrasonik* bernilai 0.
3. Setelah nilai sensor *ultrasonik* 0, masukkan 500 mL sabun cair dengan gelas ukur dan *upload* kembali program Arduino Uno.
4. Setelah nilai sensor akurat, lakukan percobaan sebanyak 5 kali, dengan mengurangi volume sabun cair 100 mL pada setiap percobaan.



Gambar 4. 3 Rangkaian *Hardware* Sensor *Ultrasonik* Pendeteksi Sabun Cair

Pada gambar 4.3 merupakan rangkaian *hardware* sensor *ultrasonik* yang terpasang pada Arduino Uno untuk mendeteksi volume sabun cair. Berikut skema rangkaian dapat dilihat pada tabel 4.2

Tabel 4. 2 Skema Rangkaian *Hardware* Sensor *Ultrasonik* pada Sabun Cair

Pin Sensor Ultrasonik	Pin Arduino
VCC	5V
GND	GND
ECHO	A0
TRIG	A1

Pada tabel 4.2 merupakan skema *hardware* sensor *ultrasonik* pada sabun cair. Pengujian sensor ini dilakukan dengan menggunakan program Arduino sebagai berikut:

```

int echo = A0;
int trig = A1;
float durasi, jarak ,nilai ;
void setup() {
    Serial.begin(9600);
    pinMode(trig, OUTPUT);
    pinMode(echo, INPUT);
}

void loop() {
    digitalWrite(trig, LOW);
    delayMicroseconds(5);
    digitalWrite(trig, HIGH);
    delayMicroseconds(5);

```

} Deklarasi variabel

} Mode pin yang digunakan (*INPUT* atau *OUTPUT*)

} Berfungsi untuk memberikan nilai 0 atau 1 pada pin digital dengan waktu 5 microseconds

<pre>digitalWrite(trig, LOW);</pre>	}	pulse mentotal waktu tunggu dari <i>echo</i>
<pre> delayMicroseconds(5);</pre>	}	
<pre> durasi = pulseIn(echo, HIGH);</pre>	}	
<pre> jarak = 417 - durasi;</pre>	}	Rumus kalibrasi meng- nolkan pada saat wadah kosong
<pre>if (jarak<=0){</pre>	}	jika jarak kurang dari 0 maka jarak = 0
<pre> jarak=0;</pre>	}	
<pre>}</pre>	}	
<pre> if (jarak>=0){</pre>	}	jika jarak lebih dari 0 maka jarak = jarak
<pre> jarak=jarak;</pre>	}	
<pre>}</pre>	}	
<pre>nilai = (jarak / 285) * 500;</pre>	}	Rumus pada saat wadah full 500 mL
<pre> Serial.print("durasi");</pre>	}	
<pre> Serial.println(durasi);</pre>	}	
<pre> Serial.print("jarak");</pre>	}	
<pre> Serial.println(jarak);</pre>	}	
<pre> Serial.print("nilai");</pre>	}	Menampilkan nilai pada serial monitor
<pre> Serial.println(nilai);</pre>	}	
<pre> Serial.println("mL");</pre>	}	
<pre>}</pre>		

Untuk mendapatkan nilai pembacaan sensor *ultrasonik* pada Arduino Uno, digunakan 2 rumus dalam program yaitu:

1. $\text{jarak} = 417 - \text{durasi}$

Rumus ini digunakan untuk kalibrasi meng-nolkan pada saat wadah sabun cair kosong, sehingga pembacaan yang muncul pada nilai sensor

ultrasonik adalah 0, digunakan untuk mencari nilai jarak pada saat wadah kosong.

2. nilai = $(\text{jarak} / 285) * 500$

Rumus ini digunakan pada saat wadah *full* 500 mL, sehingga pembacaan yang muncul pada nilai sensor *ultrasonik* adalah 500 mL. Penambahan nilai 285 merupakan jarak dari sensor *ultrasonik* ke sabun pada saat sabun dalam keadaan penuh digunakan untuk memperoleh hasil pembacaan yang akurat.



Gambar 4. 4 Takaran Ukur Sabun Cair dengan Satuan mL

Pada gambar 4.4 merupakan takaran ukur sabun cair dengan satuan mL mulai dari 500 mL, 400 mL, 300 mL, 200 mL, dan 100 mL tanpa sensor *ultrasonik*. Pengujian sensor dilakukan sebanyak 5 kali dengan volume sabun cair yang berbeda-beda. Berikut tabel pengujian volume sabun cair:

Tabel 4. 3 Hasil Pengujian Volume Sabun Cair

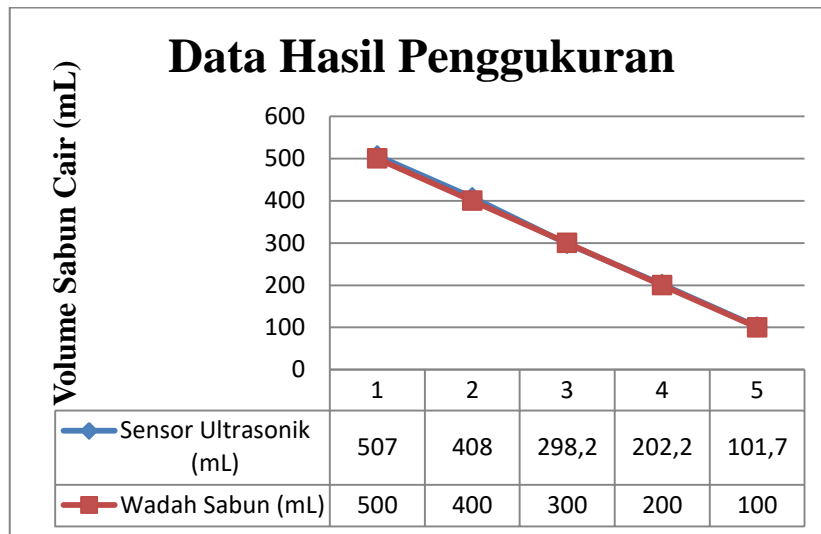
NO	Sensor Ultrasonik (mL)	Wadah Sabun Cair (mL)	Selisih Volume (mL)	Persentase Error (%)
1.	507.0	500	7	0.014
2.	408.7	400	8	0.02
3.	298.2	300	2	0.006
4.	202.2	200	2.2	0.011
5.	101.7	100	1.7	0.017
Rata-rata				0.0136

Tabel 4.3 merupakan hasil pengujian volume sabun cair dengan sensor *ultrasonik*. Didapatkan 5 selisih pengukuran dengan 5 volume yang berbeda beda, setiap pengukurannya dikurangi 100 mL. Untuk mencari nilai selisih, digunakan rumus sebagai berikut:

$$\text{Selisih volume} = \text{nilai sensor} - \text{nilai pada wadah sabun cair} \dots\dots\dots(4.1)$$

Perhitungan persentase nilai *error* (%) pada pengukuran volume sabun cair dapat dihitung dengan rumus sebagai berikut:

$$\% \text{ error volume} = \left| \frac{\text{Selisih volume}}{\text{nilai pada wadah}} \right| \times 100\% \dots\dots\dots(4.2)$$



Gambar 4. 5 Grafik Data Hasil Pengukuran Volume Sabun Cair

Dari hasil pengujian sensor *ultrasonik* pengukuran sabun cair, dapat disimpulkan bahwa, presentase *error* rata-rata sebesar 0.0136%. Artinya pengukuran hampir mendekati akurat. *Error* yang diperoleh dikarenakan adanya *nois*, sistem pembulatan perhitungan pada saat pembuatan program bahkan *human error*.

4.3.2 Pengujian Sensor *Ultrasonik* pada Wadah Air

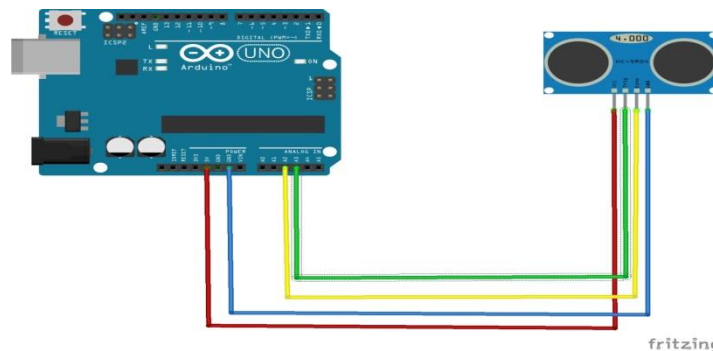
Pengujian sensor *ultrasonik* pada air dilakukan sebanyak 2 kali, yaitu pada volume air 2 Liter dan 5 Liter yang masing-masing 3 kali pengambilan data.

Cara menguji sensor *ultrasonik* pada sabun air:

1. Hubungkan sensor *ultrasonik* dan Arduino Uno sesuai dengan warna kabel. Sensor *ultrasonik* terdiri dari 4 warna kabel yaitu warna merah, warna hijau, warna kuning dan warna biru. Warna merah untuk VCC (sumber tegangan positif sensor), sehingga VCC pada sensor *ultrasonik* dihubungkan pada 5 V Arduino Uno. Warna hijau untuk *Trlg* (*Trigger* atau pin yang digunakan sebagai pembangkit sinyal ultrasonik), sehingga *Trlg* dihubungkan pada Pin A3. Warna kuning untuk *Echo* (*Receive* atau pin yang digunakan sebagai pendeteksi sinyal pantulan ultrasonik), sehingga *Echo* dihubungkan pada Pin A3. Warna biru untuk GND,

sehingga GND pada sensor *ultrasonik* dihubungkan pada GND Arduino Uno.

2. Pasang sensor *ultrasonik* pada wadah air bagian atas. Setelah itu, *upload* program ke Arduino Uno. Tunggu sampai sensor *ultrasonik* bernilai 0.
3. Setelah nilai sensor *ultrasonik* 0, masukkan 200 L air dengan takaran ukur dan *upload* kembali program Arduino Uno.
4. Setelah nilai sensor akurat, lakukan percobaan sebanyak 2 kali masing-masing 3 kali pengambilan data.



Gambar 4. 6 Rangkaian *Hardware* Sensor *Ultrasonik* Pendeteksi Air

Pada gambar 4.6 merupakan rangkaian *hardware* sensor *ultrasonik* yang terpasang pada Arduino Uno untuk mendeteksi volume air. Berikut skema rangkaian dapat dilihat pada tabel 4.4

Tabel 4. 4 Skema Rangkaian *Hardware* Sensor *Ultrasonik* pada Air

Pin Sensor Ultrasonik	Pin Arduino
VCC	5V
GND	GND
ECHO	A2
TRIG	A3

Pada tabel 4.4 merupakan skema *hardware* sensor ultrasonik pada pengukuran volume air. Pengujian sensor ini dilakukan dengan menggunakan program Arduino sebagai berikut:

<pre>int echo1 = A2;</pre>	}	Deklarasi variabel
<pre>int trig1 = A3;</pre>		
<pre>float durasi1, jarak1 ,nilai1;</pre>		
<pre>void setup() {</pre>	}	Mode pin yang digunakan (<i>INPUT</i> atau <i>OUTPUT</i>)
<pre> Serial.begin(9600);</pre>		
<pre> pinMode(trig1, OUTPUT);</pre>		
<pre> pinMode(echo1, INPUT);</pre>		
<pre>}</pre>		
<pre>void loop() {</pre>	}	Berfungsi untuk memberikan nilai 0 atau 1 pada pin digital dengan waktu 5 microseconds
<pre> digitalWrite(trig1, LOW);</pre>		
<pre> delayMicroseconds(5);</pre>		
<pre> digitalWrite(trig1, HIGH);</pre>		
<pre> delayMicroseconds(5);</pre>		
<pre> digitalWrite(trig1, LOW);</pre>	}	pulseIn mentotal waktu tunggu dari <i>echo</i>
<pre> delayMicroseconds(5);</pre>		
<pre> durasi = pulseIn(echo1, HIGH);</pre>	}	Rumus kalibrasi meng-nolkan pada saat wadah kosong
<pre> jarak1 = 3535 - durasi1;</pre>		
<pre> if (jarak1<=0){</pre>	}	jika jarak kurang dari 0 maka jarak = 0
<pre> jarak1=0;</pre>		
<pre> }</pre>		

```

if (jarak1>=0){
    jarak1=jarak1;
}

nilai1 = (jarak1 / 2579) * 200;

Serial.print("durasil");
Serial.println(durasil);
Serial.print("jarak1");
Serial.println(jarak1);
Serial.print("nilai1");
Serial.println(nilai1);
Serial.println("L");
}

```

jika jarak lebih dari 0 maka jarak = jarak

Rumus pada saat wadah full 200 L

Menampilkan nilai pada serial monitor

Untuk mendapatkan nilai pembacaan sensor *ultrasonik* untuk pengukuran volume air pada Arduino Uno, digunakan 2 rumus dalam program yaitu:

1. $\text{jarak} = 3535 - \text{durasi}$

Rumus ini digunakan untuk kalibrasi meng-nolkan pada saat wadah air kosong, sehingga pembacaan yang muncul pada nilai sensor *ultrasonik* adalah 0, digunakan untuk mencari nilai jarak pada saat wadah kosong.

2. $\text{nilai} = (\text{jarak} / 2579) * 200$

Rumus ini digunakan pada saat wadah *full* 200 L, sehingga pembacaan yang muncul pada nilai sensor *ultrasonik* adalah 200 L. Penambahan nilai 2579 merupakan jarak dari sensor *ultrasonik* ke air pada saat air dalam keadaan penuh digunakan untuk memperoleh hasil pembacaan yang akurat.



Gambar 4. 7 Takaran Ukur Air 2 L dan 5 L

Pada gambar 4.7 merupakan takaran ukur untuk melakukan pengukuran air yang dilakukan sebanyak 2 kali pengukuran dan 3 kali pengambilan data, yang terdiri dari 2 takaran, dengan pengurangan 2 L dan 5 L

Tabel 4. 5 Hasil Uji Volume Air dengan Pengurangan 2 L

NO	Volume Awal (L)	Pengurangan 2 L(L)	Selisih Volume (L)	Persentase <i>Error (%)</i>
1.	202	200.08	1,92	0.08
2.	200	198.53	1.47	0.53
3.	198	196.12	1.88	0.12
Rata-rata				0.24

Dari hasil pengujian sensor *ultrasonik* pengukuran air, dapat disimpulkan bahwa, presentase *error* rata-rata sebesar 0.24%. *Error* yang diperoleh dikarenakan adanya *nois*, sistem pembulatan perhitungan pada saat pembuatan program.

Percobaan dilakukan dengan mengurangi 2 L pada setiap pengukuran. Untuk mencari nilai selisih, digunakan rumus sebagai berikut:

$$\text{Selisih volume} = \text{nilai volume awal} - \text{nilai hasil pengurangan} \dots \dots \dots (4.1)$$

Perhitungan persentase nilai *error* (%) pada pengukuran volume air dapat dihitung dengan rumus sebagai berikut:

$$\% \text{ error volume} = \left| \frac{\text{Selisih volume}}{\text{nilai pengurangan sesungguhnya}} \right| \times 100\% \dots\dots\dots(4.2)$$

Dengan nilai pengurangan sesungguhnya adalah 2 L.

Tabel 4. 6 Hasil Uji Volume Air dengan Pengurangan 5 L

NO	Volume Awal (L)	Pengurangan 5 L(L)	Selisih Volume (L)	Persentase <i>Error</i> (%)
1.	193	188,20	4.8	0.2
2.	188	181.47	6,53	1,53
3.	183	177.59	5.41	0.41
Rata-rata				0.71

Dari hasil pengujian sensor *ultrasonik* pengukuran air, dapat disimpulkan bahwa, presentase *error* rata-rata sebesar 0.71%. *Error* yang diperoleh dikarenakan adanya *nois*, sistem pembulatan perhitungan pada saat pembuatan program.

Percobaan dilakukan dengan mengurangi 5 L pada setiap pengukuran. Untuk mencari nilai selisih, digunakan rumus sebagai berikut:

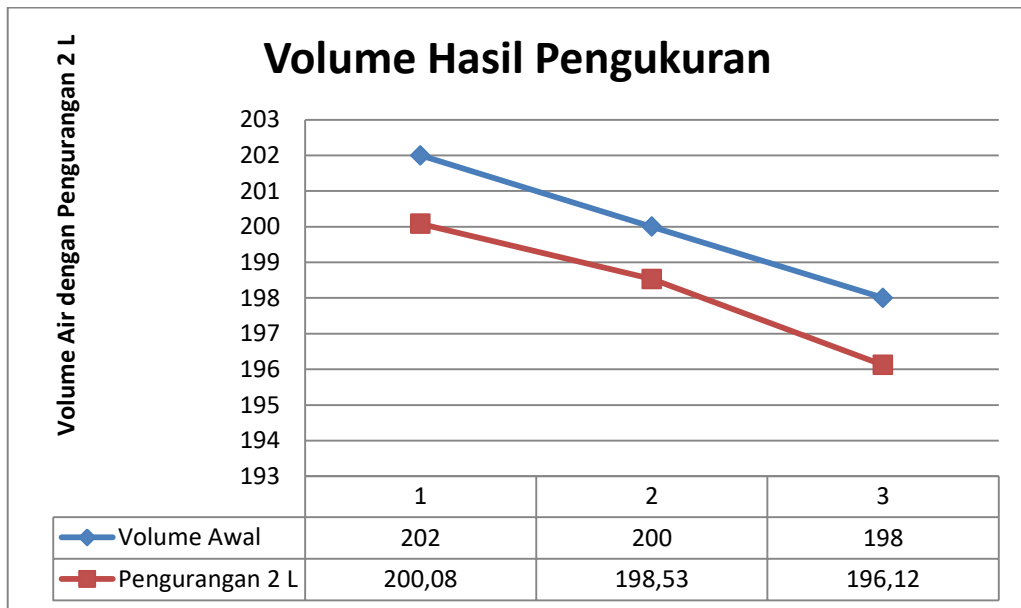
$$\text{Selisih volume} = \text{nilai volume awal} - \text{nilai hasil pengurangan} \dots\dots\dots(4.1)$$

Perhitungan persentase nilai *error* (%) pada pengukuran volume air dapat dihitung dengan rumus sebagai berikut:

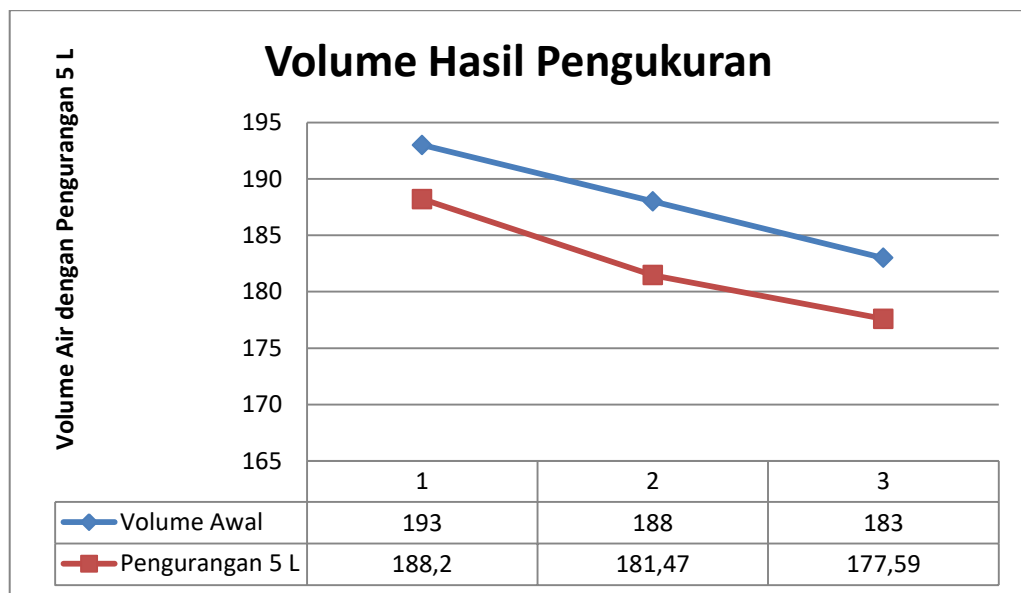
$$\% \text{ error volume} = \left| \frac{\text{Selisih volume}}{\text{nilai pengurangan sesungguhnya}} \right| \times 100\% \dots\dots\dots(4.2)$$

Dengan nilai pengurangan sesungguhnya adalah 5 L.

Sehingga grafik hasil uji pengurangan volume air sebanyak 2 L dan 5 L dapat dilihat pada gambar berikut:



Gambar 4. 8 Grafik Pengukuran Volume Air dengan Pengurangan 2L



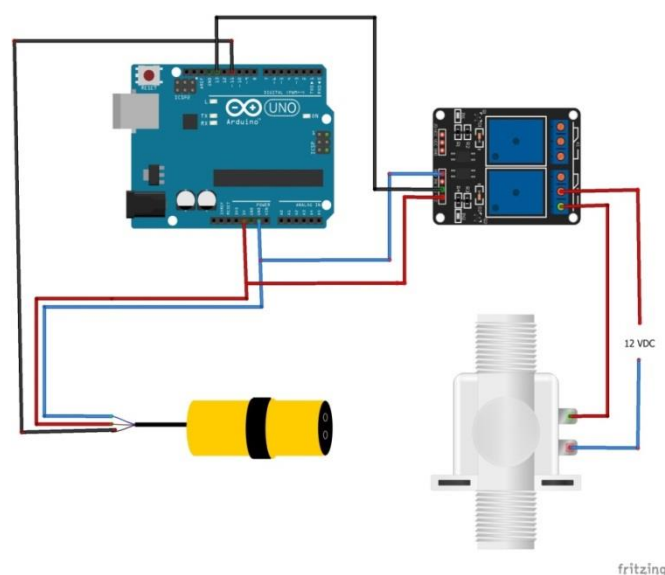
Gambar 4. 9 Grafik Pengukuran Volume Air dengan Pengurangan 5 L

4.3.3 Pengujian Sensor *Proximity* Menggunakan Relay

Pengujian sensor *proximity* bertujuan untuk mengaktifkan *solenoid valve*. Dalam spesifikasi, sensor ini mampu mendeteksi pada jarak 80 cm, tetapi pada alat cuci tangan otomatis berbasis IoT, pembacaan jarak diatur lebih dekat sehingga sensor *proximity* tidak mendeteksi *wastafel* cuci tangan. Pada pengujian

jarak, dilakukan sebanyak 10 kali percobaan dengan jarak yang berbeda. Cara menguji sensor *proximity* untuk mengaktifkan *Solenoid Valve*:

1. Hubungkan sensor *proximity* dan Arduino Uno sesuai dengan warna kabel. Sensor *proximity* terdiri dari 3 warna kabel yaitu warna merah, warna biru dan warna hitam. Warna merah untuk VCC (sumber tegangan *positif* sensor), sehingga VCC pada sensor *proximity* dihubungkan pada 5 V Arduino Uno. Warna biru untuk GND, sehingga GND pada sensor *proximity* dihubungkan pada GND Arduino Uno. Warna hitam untuk *output* sehingga *Out* dihubungkan pada Pin 11.
2. Hubungkan *relay* dan Arduino Uno sesuai dengan warna kabel. *Relay* terdiri dari 3 warna kabel yaitu warna merah, warna biru dan warna hitam. Warna merah untuk VCC (sumber tegangan *positif* relay), sehingga VCC pada *relay* dihubungkan pada 5 V Arduino Uno. Warna biru untuk GND, sehingga GND pada *relay* dihubungkan pada GND Arduino Uno. Warna hitam untuk input sehingga IN2 dihubungkan pada Pin 13.
3. Hubungkan NO dan COM pada *relay* dengan *positif* dan *negatif* pada *Solenoid valve*. Setelah itu, *upload* program ke Arduino Uno.
4. Setelah pembacaan sensor telah akurat, lakukan percobaan dengan jarak yang berbeda



Gambar 4. 10 Rangkaian *Hardware* Sensor *Proximity*

Pada gambar 4.10 merupakan rangkaian *hardware* sensor *proximity* yang terpasang pada Arduino Uno sebagai pendeteksi jarak untuk mengaktifkan *solenoid valve*. Berikut skema rangkaian dapat dilihat pada tabel 4.7 dan tabel 4.8

Tabel 4. 7 Skema Rangkaian *Hardware* Sensor *Proximity* pada Arduino

Pin Sensor <i>Proximity</i>	Pin Arduino
VCC	5V
GND	GND
OUT	11

Tabel 4. 8 Skema Rangkaian *Hardware* *Relay* pada Arduino

Pin Relay	Pin Arduino
VCC	5V
GND	GND
IN2	13

Pada Tabel 4.7 dan 4.8 merupakan skema *hardware* sensor *proximity* untuk mengaktifkan *solenoid valve*. Pengujian sensor ini dilakukan dengan menggunakan program Arduino sebagai berikut:

```

int air = 13;
int sensor1 = 11;
void setup() {
    Serial.begin(9600);
    pinMode(sensor1, OUTPUT);
    pinMode(air, INPUT);
}

```

} Deklarasi variabel

} Mode pin yang digunakan (*INPUT* atau *OUTPUT*)

```

void loop() {
    int baca1 = digitalRead(11);
    Serial.println(baca1);
}
if (baca1 == 0)
{
    digitalWrite(air, LOW);
}
else
{
    digitalWrite(air, HIGH);
}

```

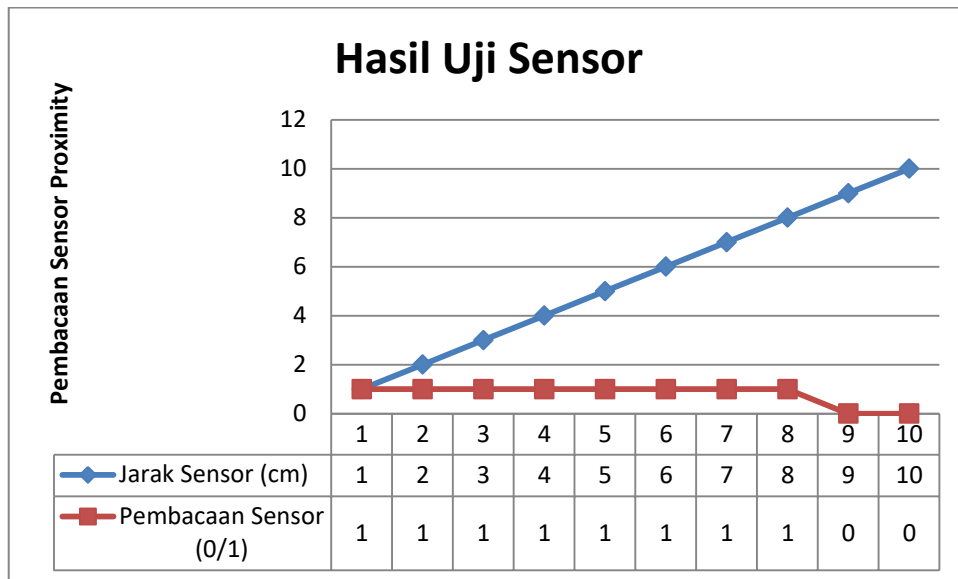
Membaca nilai sensor dan menampilkannya di serial monitor

Kondisi setelah sensor aktif dan tidak aktif, jika *LOW* 1 dan jika *HIGH* 0

Untuk mengetahui sensor bekerja sesuai fungsi yang diinginkan atau tidak, maka dilakukan pengujian pada sensor *proximity* berdasarkan nilai jarak. Pengujian pembacaan pada sensor *proximity* dapat dilihat pada tabel 4.9

Tabel 4. 9 Hasil Pengujian Sensor *Proximity*

NO	Jarak Sensor (cm)	Pembcaan Sensor (0/1)	Kedadaan Selenoid (Off/On)
1.	1	1	On
2.	2	1	On
3.	3	1	On
4.	4	1	On
5.	5	1	On
6.	6	1	On
7.	7	1	On
8.	8	1	On
9.	9	0	Off
10.	10	0	Off



Gambar 4. 11 Grafik Pembacaan Sensor *Proximity*

Dari tabel 4.11 dan grafik, dapat disimpulkan bahwa batas jarak sensor dapat diatur sesuai keinginan. Kondisi *solenoid valve* akan aktif atau sensor bernilai 1 apabila jarak sensor masih terpenuhi. Apabila jarak sensor tidak terpenuhi atau sensor tidak mendeteksi keberadaan tangan seseorang maka pembacaan sensor bernilai 0. Pada percobaan ini, dapat dilihat batas maksimum deteksi sensor adalah jarak 8 cm.

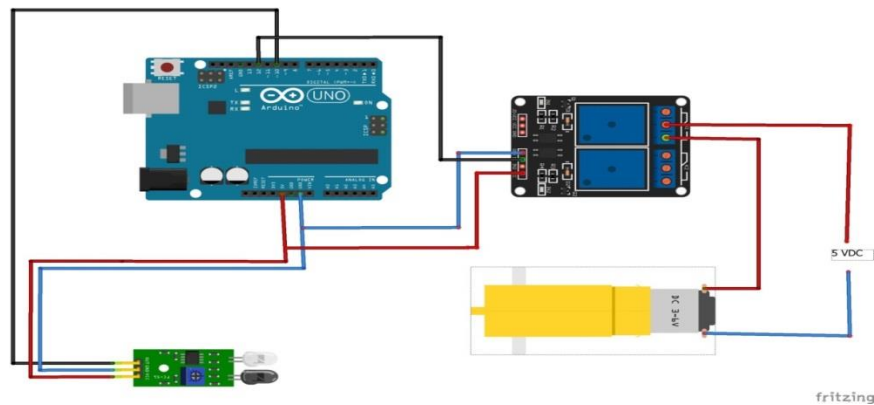
4.3.4 Pengujian Sensor *Infrared* Menggunakan *Relay*

Pengujian sensor *Infrared* bertujuan untuk mengaktifkan motor DC. Dalam spesifikasi, sensor ini mampu mendeteksi pada jarak 15 cm, tetapi pada alat cuci tangan otomatis berbasis IoT, pembacaan jarak diatur lebih dekat sehingga sensor *infrared* tidak mendeteksi baja. Pada pengujian jarak, dilakukan sebanyak 10 kali percobaan dengan jarak yang berbeda. Cara menguji sensor *infrared* untuk mengaktifkan motor DC:

1. Hubungkan sensor *infrared* dan Arduino Uno sesuai dengan warna kabel. Sensor *infrared* terdiri dari 3 warna kabel yaitu warna merah, warna biru dan warna hitam. Warna merah untuk VCC (sumber tegangan positif sensor), sehingga VCC pada sensor *infrared* dihubungkan pada 5 V

Arduino Uno. Warna biru untuk GND, sehingga GND pada sensor *infrared* dihubungkan pada GND Arduino Uno. Warna hitam untuk *output* sehingga *Out* dihubungkan pada Pin 10.

2. Hubungkan *relay* dan Arduino Uno sesuai dengan warna kabel. *Relay* terdiri dari 3 warna kabel yaitu warna merah, warna biru dan warna hitam. Warna merah untuk VCC (sumber tegangan *positif relay*), sehingga VCC pada *relay* dihubungkan pada 5 V Arduino Uno. Warna biru untuk GND, sehingga GND pada *relay* dihubungkan pada GND Arduino Uno. Warna hitam untuk *input* sehingga IN1 dihubungkan pada Pin 12.
3. Hubungkan *NO* dan *COM* pada *relay* dengan *positif* dan *Negatif* pada motor DC. Setelah itu, *upload* program ke Arduino Uno.
4. Setelah itu, *upload* program ke Arduino Uno.
5. Setelah pembacaan sensor telah akurat, lakukan percobaan dengan jarak yang berbeda



Gambar 4. 12 Rangkaian *Hardware Sensor Infrared*

Pada gambar 4.12 merupakan rangkaian *hardware sensor infrared* yang terpasang pada Arduino Uno sebagai pendeteksi jarak untuk mengaktifkan motor DC. Berikut skema rangkaian dapat dilihat pada tabel 4.10 dan tabel 4.11

Tabel 4. 10 Skema Rangkaian *Hardware Sensor Infrared* pada *Arduino*

Pin Sensor Infrared	Pin Arduino
VCC	5V
GND	GND
OUT	10

Tabel 4. 11 Skema Rangkaian *Hardware Relay* pada *Arduino*

Pin Relay	Pin Arduino
VCC	5V
GND	GND
IN2	12

Pada tabel 4.10 dan tabel 4.11 merupakan skema *hardware sensor infrared* untuk mengaktifkan motor DC. Pengujian sensor ini dilakukan dengan menggunakan program Arduino sebagai berikut:

```

int sabun = 12;
int sensor = 10;
void setup() {
    Serial.begin(9600);
    pinMode(sabun, OUTPUT);
    pinMode(10, INPUT);
}

void loop() {
    int baca1 = digitalRead(10);
    Serial.println(baca);
}

```

} Deklarasi variabel

} Mode pin yang digunakan (*INPUT* atau *OUTPUT*)

} Membaca nilai sensor dan menampilkannya di serial monitor

```

if (bacal == 0|)
{
digitalWrite(sabun, LOW);
}
else
{
digitalWrite(sabun, HIGH);
}
}

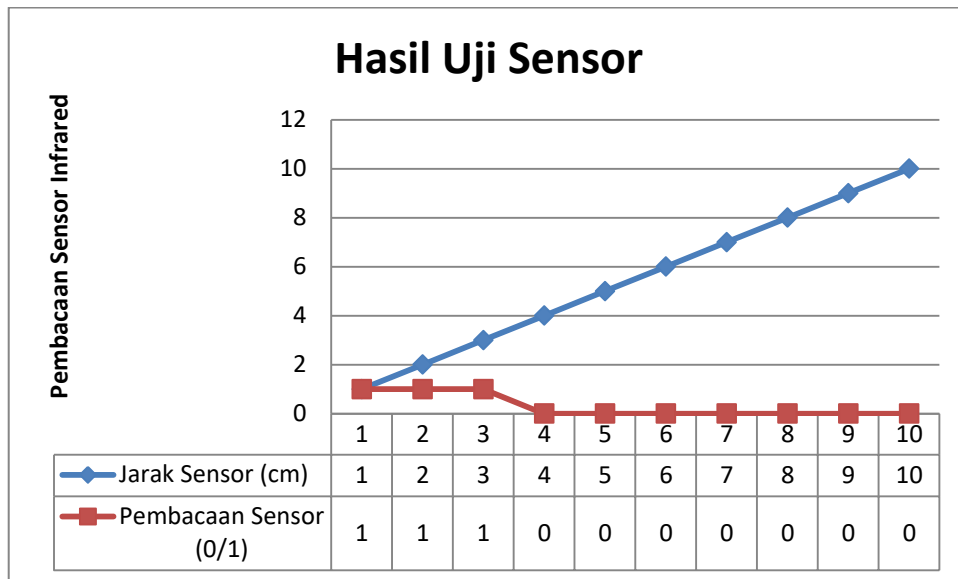
```

Kondisi setelah sensor aktif dan tidak aktif, jika *LOW* 1 dan jika *HIGH* 0

Untuk mengetahui sensor bekerja sesuai fungsi yang diinginkan atau tidak, maka dilakukan pengujian pada sensor *infrared* berdasarkan nilai jarak. Pengujian pembacaan pada sensor *infrared* dapat dilihat pada tabel 4.12

Tabel 4. 12 Hasil Pengujian Sensor *Infrared*

NO	Jarak Sensor (cm)	Pembacaan Sensor (0/1)	Kondisi Motor DC (Off/On)
1.	1	1	On
2.	2	1	On
3.	3	1	On
4.	4	0	Off
5.	5	0	Off
6.	6	0	Off
7.	7	0	Off
8.	8	0	Off
9.	9	0	Off
10.	10	0	Off



Gambar 4. 13 Grafik Pembacaan Sensor *Infrared*

Dari gambar 4.13 dapat disimpulkan bahwa batas jarak sensor dapat diatur sesuai jarak yang diinginkan. Kondisi motor DC akan aktif atau sensor bernilai 1 apabila jarak sensor masih terpenuhi. Apabila jarak sensor tidak lagi terpenuhi atau sensor tidak lagi mendeteksi keberadaan benda atau tangan seseorang maka pembacaan sensor bernilai 0. Pada percobaan ini, dapat dilihat batas maksimum deteksi sensor adalah jarak 3 cm.

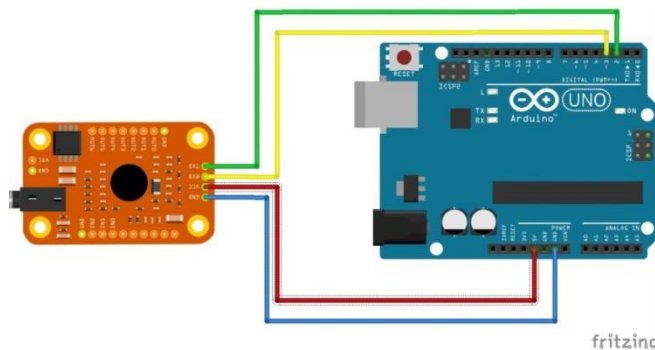
4.3.5 Pengujian Sensor *Voice Recognition V3*

Pengujian sensor *voice recognition V3* bertujuan untuk melihat apakah sensor ini mampu mendeteksi suara manusia. Dalam penggunaan sensor *voice recognition* terdapat langkah-langkah sebagai berikut:

1. Hubungkan sensor *voice recognition V3* dengan arduino sesuai gambar. Sensor ini terdiri dari 4 warna kabel yaitu warna merah, warna biru, warna kuning dan warna hijau. Warna merah untuk VCC (sumber tegangan positif sensor), sehingga VCC pada sensor suara dihubungkan pada 5 V Arduino Uno. Warna hijau untuk TX, sehingga kabel ini dihubungkan pada Pin 2. Warna kuning untuk RX, sehingga kabel ini dihubungkan

Apada Pin 3. Warna biru untuk GND, sehingga GND pada sensor suara dihubungkan pada GND Arduino Uno.

2. Unggah arduino, tunggu sampai arduino selesai mengunggah. Setelah itu, buka serial monitor dan atur *baud rate* 115200, dan *setting* kirim dengan *newline*
3. Latih modul pengenalan suara dengan cara, mengirimkan perintah *sigtrain* 0 On untuk melatih *record* 0 dengan *signature* “On”. Apabila serial monitor menampilkan tulisan “ Bicara sekarang”, maka kita perlu bersuara, dengan mengucapkan “Sabun On” atau sebagainya sesuai kata yang ada dalam program. Pada saat serial monitor menampilkan tulisan “ Bicara lagi” maka kita harus mengulangi kata tersebut. Apabila suara cocok, maka serial monitor akan menampilkan tulisan “*Succes*”, tetapi apabila suara tidak cocok, maka serial monitor akan menampilkan ulangi berbicara hingga suara bisa terdeteksi.



Gambar 4. 14 Rangkaian *Hardware* Sensor *Voice Recognition V3*

Pada gambar 4.14 merupakan rangkaian *hardware* sensor *voice recognition V3* yang terpasang pada Arduino Uno sebagai pendeteksi perintah suara untuk mengaktifkan *relay*. Berikut skema rangkaian dapat dilihat pada tabel 4.13

Tabel 4. 13 Skema Rangkaian *Hardware Sensor Voice Recognition V3*

Pin Sensor voice recognition V3	Pin Arduino
VCC	5V
GND	GND
TX	2
RX	3

Pada tabel 4.13 merupakan skema *hardware sensor voice recognition V3* untuk mengaktifkan *relay*. Pengujian sensor ini dilakukan dengan menggunakan program Arduino pada lampiran 2 poin 1.

Untuk mengetahui sensor bekerja sesuai fungsi yang diinginkan atau tidak, maka dilakukan pengujian pada sensor *Voice Recognition* berdasarkan jenis suara dan kebisingan. Pada proses uji coba *Voice Recognition* dilakukan pengujian dengan membandingkan rekam tanpa kebisingan, rekam dengan kebisingan, 3 suara laki-laki, dan 4 suara perempuan. Volume suara akan diukur menggunakan aplikasi *Sound Meter*. Sehingga diperoleh data sebagai berikut:

Tabel 4. 14 Data Rekam Tanpa Kebisingan

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Air Hidup (76db)	✓	✗	✓	✓	✓	80%
2	Air Mati (76db)	✓	✗	✓	✓	✓	80%
3	Sabun On (75db)	✓	✓	✓	✓	✓	100%

Dari data hasil percobaan rekam tanpa kebisingan, dapat disimpulkan bahwa dalam pemanggilan perintah suara Air Hidup dan Air Mati didapatkan persentase keberhasilan sebanyak 80% dengan 5 kali percobaan, hal ini

dikarenakan salah satu suara tidak terdeteksi dan pengucapannya tidak jelas. Sedangkan untuk npengucapan Sabun On diperoleh persentase keberhasilan sebanyak 100% dengan 5 kali percobaan, hal ini dikarenakan pengucapan yang jelas dan suara yang stabil.

Tabel 4. 15 Data Rekam Kebisingan (76db)

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Air Hidup (79db)	✓	✓	✗	✓	✓	80%
2	Air Mati (79db)	✓	✓	✗	✓	✓	80%
3	Sabun On (81db)	✓	✗	✓	✓	✓	80%

Dari data hasil percobaan rekam dengan kebisingan (76db), dapat disimpulkan bahwa dalam pemanggilan perintah suara Air Hidup, Air Mati, dan Sabun On didapatkan persentase keberhasilan sebanyak 80% dengan 5 kali percobaan, hal ini dikarenakan *Voice recognition* tidak hanya menangkap suara perekam, tetapi juga suara kebisingan atau suara pengganggu, sehingga suara yang diucapkan oleh perekam harus dengan volume yang besar.

Tabel 4. 16 Rekam Tanpa Kebisingan untuk Air Hidup (Laki-laki)

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Zalpian (76db)	✓	✓	✗	✓	✓	80%
2	Deri (78db)	✓	✓	✗	✗	✗	40%
3	Buslowi (73db)	✓	✓	✓	✗	✓	80%

Dari data hasil percobaan rekam dengan suara orang yang berbeda, dapat disimpulkan bahwa dalam pemanggilan perintah suara Air Hidup yang dilakukan oleh Zalpian dan Buslowi didapatkan persentase keberhasilan sebanyak 80% dengan 5 kali percobaan, hal ini dikarenakan suara yang dideteksi memiliki

kesamaan, pada suara Deri, *Voice Recognition* masih bisa mendeteksi suara walaupun dengan presentasi keberhasilan 40%. Hal ini dikarenakan suara yang dideteksi memiliki suara yang berbeda. Sedangkan apabila percobaan dilakukan dengan pengaruh kebisingan 76db maka persentase keberhasilannya berkurang.

Tabel 4. 17 Rekam Tanpa Kebisingan untuk Air Mati (Laki-laki)

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Zalpian (76db)	✓	✓	✗	✓	✓	80%
2	Deri (78db)	✗	✓	✗	✗	✗	20%
3	Buslowi (73db)	✓	✓	✓	✗	✓	80%

Dari data hasil percobaan rekam dengan suara orang yang berbeda, dapat disimpulkan bahwa dalam pemanggilan perintah suara Air Mati yang dilakukan oleh Zalpian dan Buslowi didapatkan persentase keberhasilan sebanyak 80% dengan 5 kali percobaan, hal ini dikarenakan suara yang dideteksi memiliki kesamaan, pada suara Deri, *Voice Recognition* masih bisa mendeteksi suara walaupun dengan presentasi keberhasilan 20%. Hal ini dikarenakan suara yang dideteksi memiliki suara yang berbeda. Sedangkan apabila percobaan dilakukan dengan pengaruh kebisingan 76db maka persentase keberhasilannya berkurang.

Tabel 4. 18 Rekam Tanpa Kebisingan untuk Sabun On (Laki-laki)

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Zalpian (76db)	✓	✓	✓	✓	✓	100%
2	Deri (78db)	✓	✗	✗	✓	✗	40%
3	Buslowi (73db)	✓	✓	✓	✓	✓	100%

Dari data hasil percobaan rekam dengan suara orang yang berbeda, dapat disimpulkan bahwa dalam pemanggilan perintah suara Sabun On yang dilakukan

oleh Zalpian dan Buslowi didapatkan persentase keberhasilan sebanyak 100% dengan 5 kali percobaan, hal ini dikarenakan suara yang dideteksi memiliki kesamaan, pada suara Deri, *Voice Recognition* masih bisa mendeteksi suara walaupun dengan presentasi keberhasilan 40%. Hal ini dikarenakan suara yang dideteksi memiliki suara yang berbeda. Sedangkan apabila percobaan dilakukan dengan pengaruh kebisingan 76db maka persentase keberhasilannya berkurang.

Tabel 4. 19 Data Rekam Tanpa Kebisingan untuk Air Hidup (Perempuan)

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Selly (65db)	✓	✓	✓	✓	✓	100%
2	Veni (63db)	✓	✓	✓	✓	✓	100%
3	Riza (73db)	✓	✗	✓	✗	✓	60%
4	Maryani (68db)	✓	✗	✓	✓	✓	80%

Dari data hasil percobaan rekam dengan suara orang yang berbeda, dapat disimpulkan bahwa dalam pemanggilan perintah suara Air Hidup yang dilakukan oleh Selly dan Veni didapatkan persentase keberhasilan sebanyak 100% dengan 5 kali percobaan, hal ini dikarenakan suara yang dideteksi memiliki kesamaan dengan suara perekam. Pada suara Riza, *Voice Recognition* masih bisa mendeteksi suara walaupun dengan presentasi keberhasilan 60%, karena suara yang dideteksi memiliki suara yang berbeda dari suara perekam. Sedangkan Maryani memiliki persentase keberhasilan sebesar 80%. Apabila percobaan dilakukan dengan pengaruh kebisingan 76db maka persentase keberhasilannya berkurang.

Tabel 4. 20 Data Rekam Tanpa Kebisingan untuk Air Mati (Perempuan)

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Selly (65db)	✓	✓	✓	✓	✓	100%
2	Veni (63db)	✓	✓	✗	✓	✓	80%

3	Riza (73db)	✓	✗	✓	✓	✗	60%
4	Maryani (68db)	✓	✓	✓	✗	✓	80%

Dari data hasil percobaan rekam dengan suara orang yang berbeda, dapat disimpulkan bahwa dalam pemanggilan perintah suara Air Mati yang dilakukan oleh Selly didapatkan persentase keberhasilan sebanyak 100% dengan 5 kali percobaan. Pada suara Riza, *Voice Recognition* masih bisa mendeteksi suara walaupun dengan presentasi keberhasilan 60%, karena suara yang dideteksi memiliki suara yang berbeda dari suara perekam. Sedangkan Maryani dan Veni memiliki persentase keberhasilan sebesar 80% hal ini dikarenakan suara yang dideteksi hampir sama dengan suara perekam. Apabila percobaan dilakukan dengan pengaruh kebisingan 76db maka persentase keberhasilannya berkurang.

Tabel 4. 21 Data Rekam Tanpa Kebisingan untuk Sabun On (Perempuan)

No	Rekaman	Percobaan					Persentase Keberhasilan
		1	2	3	4	5	
1	Selly (65db)	✓	✓	✓	✓	✓	100%
2	Veni (63db)	✓	✓	✓	✓	✓	100%
3	Riza (73db)	✗	✓	✓	✗	✓	60%
4	Maryani (68db)	✓	✓	✓	✓	✗	80%

Dari data hasil percobaan rekam dengan suara orang yang berbeda, dapat disimpulkan bahwa dalam pemanggilan perintah suara Air Hidup yang dilakukan oleh Selly dan Veni didapatkan persentase keberhasilan sebanyak 100% dengan 5 kali percobaan, hal ini dikarenakan suara yang dideteksi memiliki kesamaan dengan suara perekam. Pada suara Riza, *Voice Recognition* masih bisa mendeteksi suara walaupun dengan presentasi keberhasilan 60%, karena suara yang dideteksi memiliki suara yang berbeda dari suara perekam. Sedangkan Maryani memiliki persentase keberhasilan sebesar 80%. Apabila percobaan dilakukan dengan pengaruh kebisingan 76db maka persentase keberhasilannya berkurang.

Dari seluruh pengujian yang telah dilakukan, dapat di simpulkan bahwa:

1. *Voice Recognition* hanya dapat mendeteksi perintah suara berupa Air Hidup, Air Mati, dan Sabun On.
2. *Voice Recognition* sangat berpengaruh terhadap gangguan kebisingan.
3. *Voice Recognition* hanya dapat mendeteksi perintah dengan suara yang sama.

4.3.6 Uji Coba Menampilkan Data di LCD 20x4

LCD 20x4 digunakan untuk menampilkan hasil pembacaan volume air dan sabun cair dengan sensor *ultrasonik*. Tulisan yang akan ditampilkan oleh LCD berupa “DATA VOLUME”, “SABUN CAIR” dan “AIR”. Berikut adalah list program untuk menampilkan data di LCD:

```
lcd.init();  
lcd.backlight();  
  
lcd.setCursor(4,0);  
lcd.print("DATA VOLUME:");  
lcd.setCursor(0,1);  
lcd.print("SABUN CAIR =");  
digitalWrite(trig, HIGH);  
lcd.setCursor(13,1);  
lcd.print(nilai);  
lcd.setCursor(18,1);  
lcd.print("mL");  
lcd.setCursor(0,2);  
lcd.print("AIR =");  
digitalWrite(trig1, HIGH);
```

Mengaktifkan
LCD

Menampilkan
Data Sabun Cair
pada LCD 20x4

```

delay(50);

lcd.setCursor(7,2);

lcd.print(nilai1);

lcd.setCursor(18,2);

lcd.print("L");

```

} Menampilkan
Data Air pada
LCD 20x4

4.3.7 Program Mengirim Data Dari Arduino ke NodeMCU

Program pengiriman data dari arduino ke NodeMCU bertujuan untuk menyimpan data di NodeMCU untuk di kirimkan ke android. Pin yang digunakan pada arduino yaitu pin 6 dan pin 7, sedangkan pin yang digunakan pada NodeMCU adalah pin D5 dan D6. Sehingga skema rangkaian dapat dilihat pada tabel 4.22

Tabel 4. 22 Skema Rangkaian Pengiriman Data dari Arduino ke NodeMCU

Pin Arduino	Pin NodeMCU
6	D5
7	D6

Sehingga data yang akan dikirimkan dari Arduino ke NodeMCU adalah volume air dan sabun cair yang terbaca dari sensor *ultrasonik*.

4.3.8 Program Mengirim Data NodeMCU ke Android

Program pengiriman data dari NodeMCU ke Android, memerlukan jaringan *WiFi*. Proses ini dilakukan dengan menggunakan aplikasi buatan yang di desain melalui *MIT App Inventor*, difungsikan melalui *API Blynk*. Untuk menghubungkan NodeMCU dan android, diperlukan nama *WiFi* dan *password*-nya, sehingga aplikasi bisa terhubung ke NodeMCU. Untuk program lengkap pengiriman data ke android, dapat dilihat pada Lampiran 2 poin 2.

4.4 Perakitan *Hardware* Keseluruhan

Perakitan *hardware* keseluruhan merupakan perakitan konstruksi dan perakitan rangkaian kontrol. Perakitan *hardware* keseluruhan dilakukan apabila semua komponen telah mengalami uji coba dan akan dilakukan pengujian secara keseluruhan. sehingga semua komponen dipasang dalam satu tempat. Hal ini bertujuan supaya kabel dan komponen tertata rapi. Perakitan hardware secara keseluruhan dapat dilihat pada gambar 4.15



Gambar 4. 15 Perakitan Hardware Keseluruhan

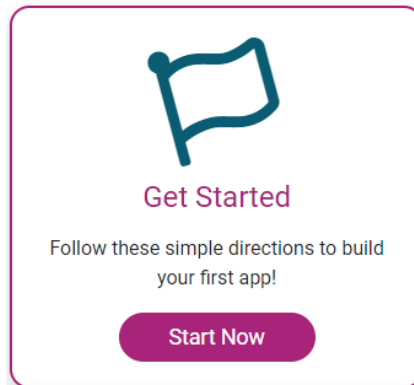
4.5 Perancangan dan Perakitan *Software*

Perancangan Software merupakan perancangan sebuah aplikasi yang digunakan untuk menampilkan data volume melalui *smartphone*. Aplikasi ini dibuat menggunakan sebuah aplikasi web yaitu *MIT App Inventor* dengan menggunakan server *API Blynk*. Dengan *MIT App Inventor*, kita dapat membuat aplikasi android sederhana tanpa harus membuat *coding* yang banyak.

Adapun langkah-langkah dalam penggunaan *MIT App Inventor*:

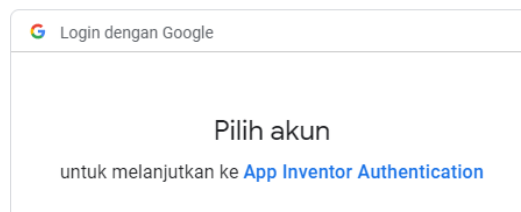
1. Mengakses *MIT App Inventor* online dengan cara mengunjungi <https://appinventor.mit.edu/>

2. Klik Menu *Get Started*



Gambar 4. 16 Menu *Get Started*

3. Setelah itu, masuk menggunakan akun *google*



Gambar 4. 17 Masuk dengan Akun *Google*

4. Buatlah proyek baru dengan cara klik Project, setelah itu pilih *Start New Project*, dan masukkan komponen sesuai tampilan yang diinginkan.

Pada saat membuat tampilan aplikasi, diperlukan program aplikasi sederhana. Dalam aplikasi ini memiliki 2 *Screen*, yang pertama adalah *Screen Login* dan yang kedua adalah *Screen Menu*.

1. *Screen Login*

Screen Login merupakan *Screen* awal yang menampilkan *Username* dan *Password* untuk masuk ke *Screen Menu*, supaya pemantauan volume dapat dilihat. Pada *Screen Login*, terdapat 2 *Button* yaitu *Button Login* dan *Button SignUp* yang dihubungkan dengan *Firebase*. Apabila *Username* dan *password* yang sudah terdaftar di *firebase* benar, maka *Screen Login* akan berpindah ke *Screen Menu*, tetapi apabila *Username* dan *Password*

salah, maka pada *Screen Login* akan muncul tulisan “*Username dan Password Salah*”.

2. *Screen Menu*

Screen Menu adalah tampilan utama pada aplikasi *Monitoring Volume*, yang terdapat 2 *Button* di dalamnya. *Button* tersebut berfungsi untuk memilih alat 1 atau alat 2 yang ingin di *monitoring* volumenya.

4.6 Uji Coba Software

Dalam uji coba *software*, aplikasi yang digunakan adalah aplikasi buatan “*Monitoring Volume*”. Langkah-langkah yang dapat dilakukan dalam uji coba *software* adalah:

1. Nyalakan data seluler atau nyalakan *hotspot*. Pastikan *hotspot* terhubung ke perangkat android.
2. Buka aplikasi “*Monitoring Volume*”. Maka tampilan akan muncul seperti Gambar 4.18



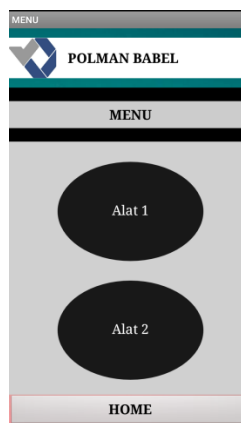
Gambar 4. 18 Tampilan Layar *Login*

3. Isi *Username* dan *Password* klik *Sign Up*, maka nama dan *password* anda akan terdaftar pada *FireBase*. Berikut tampilan pada *FireBase*:

```
login-4f8a0-default-rtdb
├── monitoring1
│   ├── rahma: "\"7777\""
│   ├── sell: "\"1234\""
│   └── selly: "\"se\""
```

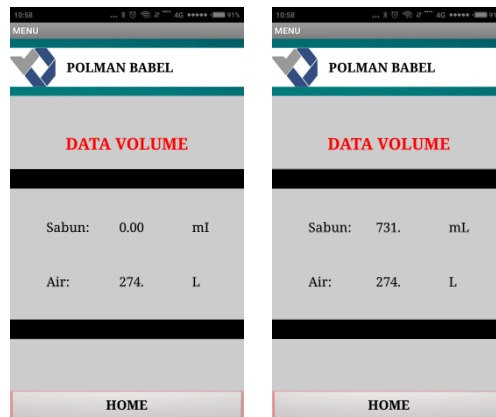
Gambar 4. 19 Tampilan pada *FireBase*

4. Kemudian klik *Login* untuk masuk ke *Screen Menu*. Tampilan *Screen Menu* dapat dilihat pada gambar 4.20



Gambar 4. 20 Tampilan *Screen Menu*



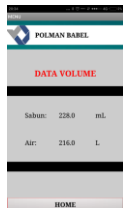

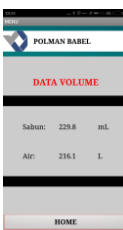

5. Hubungkan alat cuci tangan pada sumber listrik dengan mengisi terlebih dahulu air dan sabun cair.
6. Klik Alat 1 untuk *monitoring* alat yang pertama dan klik Alat 2 untuk *monitoring* alat kedua.
7. Sehingga akan muncul tampilan seperti ini:



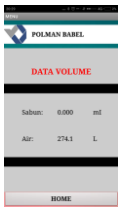







Gambar 4. 21 Tampilan Aplikasi

Uji Coba *Software*, dengan hasil sebagai berikut:

Tabel 4. 23 Uji Coba pada Alat 1

No	Tampilan Aplikasi	Tampilan LCD
1		
2		
3		

Tabel 4. 24 Uji Coba pada Alat 2

No	Tampilan Aplikasi	Tampilan LCD
1		
2		
3		
4		

Untuk hasil *screenshot* pada tampilan sangat sulit diambil, karena proses pergantian angka yang begitu cepat. Dari pengujian *software*, dapat disimpulkan bahwa volume air dan sabun cair bisa ditampilkan melalui android dengan koneksi jaringan yang baik.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian Alat Cuci Tangan Otomatis Berbasis IoT dapat diketahui bahwa:

1. Alat cuci tangan otomatis Berbasis IoT dapat digunakan sebagai alat cuci tangan yang bisa mendeteksi perintah suara, keberadaan tangan dan efektivitas dalam *monitoring* volume sabun dan air.
2. Alat ini berfungsi dengan baik, menggunakan 3 perintah suara yaitu air hidup, air mati dan sabun on dengan suara laki-laki (70-85db) dan perempuan (60-75db).
3. Uji coba sensor jarak dengan menggunakan sensor *proximity* dan sensor *infrared* dilakukan sebanyak 10 kali percobaan. Pada sensor *proximity*, batas jarak pembacaan sensor yaitu 8 cm sedangkan sensor *infrared* adalah 3 cm.
4. Volume air dan sabun cair di deteksi oleh sensor *ultrasonik*, ditampilkan pada LCD dan aplikasi *monitoring* volume pada android.

5.2 Saran

Apabila alat ini akan dikembangkan lebih lanjut, fungsi yang perlu diperbaiki dan ditambahkan adalah:

1. Diharapkan pada pengembangan alat selanjutnya ditambahkan *supply* daya cadangan apabila terjadi pemadaman listrik, sehingga alat tetap bisa digunakan.
2. Menambahkan sistem simpan data otomatis pada aplikasi sehingga data lama bisa ditampilkan.

DAFTAR PUSTAKA

- [1] Kemkes, "Pedoman Pencegahan dan Pengendalian COVID-19," 2020. [Online]. <https://infeksiemerging.kemkes.go.id>
- [2] Hendri dan Hanifia, "Pembersih Tangan Otomatis Dilengkapi Air, Sabun, Handdryer dan LCD Menggunakan Sensor Infrared Berbasis Arduino," 2018.
- [3] N.I. Pradasari, I. Suanda dan R. Ruhibnur, "Rancang Bangun Keran Air dan Tempat Penyimpanan Sabun Otomatis Tanpa Disentuh untuk Pasar Tradisional Desa Paya Kumang Menggunakan Obstacle Sensor Guna Pencegahan Penyebaran Virus Covid-19," 2021. [Online]. <https://journal.yrpiiku.com/index.php/ceej>
- [4] Hendry dan Santoso, "Mesin Cuci Tangan Otomatis Menggunakan Sensor optocopler," 2018.
- [5] Hanifudin dan Sukri, "Perancangan Mesin Cuci Tangan Otomatis dan Higienis Berbasis Kamera," 2019. [Online]. <http://journal.trunojoyo.ac.id/rekayasa>
- [6] A. S. Jaya, A. Aulia, E. I. Bhiptime dan L. Y. W. Rukmono, "Rancang Bangun Mesin Cuci Tangan Otomatis Cutato-Unhan Berbasis Karakteristik Anthropolometri," 2020. [Online]. <https://scholar.google.com>
- [7] R. N. Rohmah, A. Budiman dan V. L. Rohmah, "Sistem Pemantauan dan Pengendalian Penggunaan Air Menggunakan Aplikasi Telegram Berbasis IoT," 2020. [Online]. <https://shcolar.google.com>
- [8] T. H. Saputra, H. A. Pamasaria, B. Wiratmoko, R. Hermawan dan R. Supriyono, "Rancang Bangun Mesin Cuci Tangan Otomatis Portabel Untuk Mengurangi Efek Pandemi Covid-19," 2020. [Online]. <http://logista.fateta.unand.ac.id/index.php/logista/article/view/441/222>
- [9] Ginting dan H. Oktaria, "Alat Ukur Volume Air dengan Menggunakan

- Sensor Ultrasonik berbasis At Mega 328P," 2018. [Online].
<http://repositori.usu.ac.id/bitstream/handle/123456789/7776/152411011.pdf?sequence=1&isAllowed=y>
- [10] Micco, "Cara Menghitung Volume Silinder (Tabung).," 2015. [Online].
<https://pendidikan.id/main/forum/diskusi-pendidikan/mata-pelajaran/556-cara-menghitung-volume-silinder-tabung>
- [11] A. Zamzami dan O. Fransisco, "Sistem Monitoring Kualitas Air Tambak Udang Berbasis Internet of Things (IoT)," 2021.

LAMPIRAN

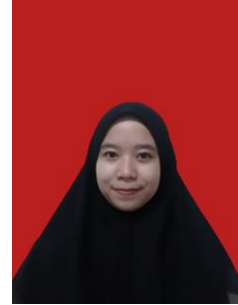
LAMPIRAN 1

DAFTAR RIWAYAT HIDUP

DAFTAR RIWAYAT HIDUP

I. Data Pribadi

Nama Lengkap : SELLY RAHMAWANTI
Tempat, Tanggal Lahir : SIMPANG PESAK, 30 JULI 2000
Alamat : DESA SIMPANG PESAK
Telp : -
HP : 087893908530
e-mail : rahmawantiselly77@gmail.com
Jenis kelamin : PEREMPUAN
Agama : ISLAM



II. Riwayat Pendidikan

Pendidikan	Tahun
SDN 1 SIMPANG PESAK	2012
SMPN 1 SIMPANG PESAK	2015
SMAN 1 SIMPANG PESAK	2018

III. Pendidikan Non Formal

-

Sungailiat, 20 Agustus 2021

Selly Rahmawanti

DAFTAR RIWAYAT HIDUP

I. Data Pribadi

Nama Lengkap : ZALPIAN
Tempat, Tanggal Lahir : BAKAM, 12 JULI 1999
Alamat : DESA BAKAM
Telp : -
HP : 083175216349 (WA)
082180467581 (Tlp)
e-mail : zalfiansamsung112233@gmail.com
Jenis kelamin : LAKI-LAKI
Agama : ISLAM



II. Riwayat Pendidikan

Pendidikan	Tahun
SDN 4 BAKAM	2011
SMPN 1 BAKAM	2014
SMKN 1 BAKAM	2017

III. Pendidikan Non Formal

-

Sungailiat, 20 Agustus 2021

Zalpian

LAMPIRAN 2

PROGRAM

1. PROGRAM ARDUINO

```
//kepala library

#include <SoftwareSerial.h>

#include "VoiceRecognitionV3.h"

#include<Wire.h>

#include<LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,20,4);

//pin serial komunikasi

SoftwareSerial ArduinoUno (7, 6);

//Deklarasi variabel

int air = 13;

int sabun =12;

int sensor = 10;

int sensor1 = 11;

int x = 0;

int echo = A0;

int trig = A1;

int echo1 = A2;

int trig1 = A3;

float durasi, jarak ,nilai,durasil, jarak1 ,nilail;
```

```

//program voice recognition

/**

Connection

Arduino    VoiceRecognitionModule

2    ----->    TX

3    ----->    RX

*/

VR myVR(2,3);    // 2:RX 3:TX, you can choose your favourite
pins.

uint8_t records[7]; // save record

uint8_t buf[64];

//pengalamatan rekaman suara

#define airhidupRecord    (0)

#define airmatiRecord    (1)

#define sabunonRecord    (2)

/**

@brief    Print signature, if the character is invisible,
          print hexible value instead.

@param    buf    --> command length

          len    --> number of parameters

*/

void printSignature(uint8_t *buf, int len)

```

```

{
    int i;

    for(i=0; i<len; i++){
        if(buf[i]>0x19 && buf[i]<0x7F){
            Serial.write(buf[i]);
        }
        else{
            Serial.print("[");
            Serial.print(buf[i], HEX);
            Serial.print("]");
        }
    }
}

/**
    @brief    Print signature, if the character is invisible,
             print hexible value instead.

    @param   buf  -->  VR module return value when voice is
recognized.

             buf[0]  -->  Group mode (FF: None Group, 0x8n:
User, 0x0n: System

             buf[1]  -->  number of record which is
recognized.

```

buf[2] --> Recognizer index(position) value
of the recognized record.

buf[3] --> Signature length

buf[4]~buf[n] --> Signature

*/

```
void printVR(uint8_t *buf)
```

```
{
```

```
    Serial.println("VR Index\tGroup\tRecordNum\tSignature");
```

```
    Serial.print(buf[2], DEC);
```

```
    Serial.print("\t\t");
```

```
    if(buf[0] == 0xFF){
```

```
        Serial.print("NONE");
```

```
    }
```

```
    else if(buf[0]&0x80){
```

```
        Serial.print("UG ");
```

```
        Serial.print(buf[0]&(~0x80), DEC);
```

```
    }
```

```
    else{
```

```
        Serial.print("SG ");
```

```
        Serial.print(buf[0], DEC);
```

```
    }
```

```
    Serial.print("\t")
```

```
Serial.print(buf[1], DEC);

Serial.print("\t\t");

if(buf[3]>0){

    printSignature(buf+4, buf[3]);

}

else{

    Serial.print("NONE");

}

Serial.println("\r\n");

}

void setup() {

    /** initialize */

    myVR.begin(9600);

    ArduinoUno.begin(4800);

    Serial.begin(115200);

    Serial.println("Elechouse Voice Recognition V3
Module\r\nControl LED sample");

    pinMode(air, OUTPUT);

    pinMode(sabun, OUTPUT);

    if(myVR.clear() == 0){

        Serial.println("Recognizer cleared.");

    }

}
```

```

    }else{

        Serial.println("Not find VoiceRecognitionModule.");

        Serial.println("Please check connection and restart
Arduino.");

        while(1);

    }

    if(myVR.load((uint8_t)airhidupRecord) >= 0){

        Serial.println("airhidupRecord loaded");

    }

    if(myVR.load((uint8_t)airmatiRecord) >= 0){

        Serial.println("airmatiRecord loaded");

    }

    if(myVR.load((uint8_t)sabunonRecord) >= 0){

        Serial.println("sabunonRecord loaded");

    }

//untuk mengaktifkan lcd

    lcd.init();

    lcd.backlight();

    // Mode pin yang digunakan (INPUT atau OUTPUT pada sensor
ultrasonic)

    pinMode(trig, OUTPUT);

    pinMode(echo, INPUT);

    pinMode(trig1, OUTPUT);

    pinMode(echo1, INPUT);

```

```

        // Mode pin yang digunakan (INPUT atau OUTPUT pada sensor
ultrasonic)

        pinMode(trig, OUTPUT);

        pinMode(echo, INPUT);

        pinMode(trig1, OUTPUT);

        pinMode(echo1, INPUT);

        // Mode pin yang digunakan (INPUT pada sensor proximity)

        pinMode(10, INPUT);

        pinMode(11, INPUT);

    }

    void SC() {

        //Berfungsi untuk memberikan nilai 0 atau 1 pada pin digital
            dengan waktu 5 microseconds

            digitalWrite(trig, LOW);

            delayMicroseconds(5);

            digitalWrite(trig, HIGH);

            delayMicroseconds(5);

        // pulseIn mentotal waktu tunggu dari echo

            digitalWrite(trig, LOW);

            delayMicroseconds(5);

            durasi = pulseIn(echo, HIGH);

        // Rumus kalibrasi meng-nolkan pada saat wadah kosong

            jarak = 417 - durasi;

            if (jarak<=0){

                jarak=0

```



```
    }

    if (jarak>=0){

        jarak=jarak;

    }

    nilai = (jarak / 285) * 500;    // Rumus pada saat wadah full
500 mL

//mengirim data dari Arduino ke NodeMCU

    ArduinoUno.print(nilai);

    ArduinoUno.println("A");

}

void AIR() {

    digitalWrite(trig1, LOW);

    delayMicroseconds(5);

    digitalWrite(trig1, HIGH);

    delayMicroseconds(5);

    digitalWrite(trig1, LOW);

    delayMicroseconds(5);

    durasil = pulseIn(echol, HIGH);

    jarak1 = 3535 - durasil;

    if (jarak1<=0){

        jarak1=0
```

```
    }

    if (jarak1>=0){

        jarak1=jarak1;

    }

    nilai1 = (jarak1 / 2579) * 200;

    ArduinoUno.print(nilai1);

    ArduinoUno.println("B");

}
```

```
void loop() {

    SC();

    AIR();

    int ret;

    ret = myVR.recognize(buf, 50);

    if(ret>0){

        switch(buf[1]){

            case airhidupRecord:

                x=0;

                /** turn on AIR */

                digitalWrite(air, LOW);

                break;

            case airmatiRecord:

                x=1;
```

```
        /** turn off AIR*/  
        digitalWrite(air, HIGH);  
  
        break;  
  
    case sabunonRecord:  
  
        /** turn on sabun*/  
  
        digitalWrite(sabun, LOW);  
  
        delay(1000);  
  
        /** turn off sabun*/  
  
        digitalWrite(sabun, HIGH);  
  
        break;  
  
    default:  
  
        Serial.println("Record function undefined");  
  
        break;  
  
}/** voice recognized */  
  
printVR(buf);  
  
}  
  
// Menampilkan nilai pada serial monitor  
  
Serial.print("durasi");  
  
Serial.println(durasi);  
  
Serial.print("jarak");  
  
Serial.println(jarak);
```

```
Serial.print("nilai");  
  
Serial.println(nilai);  
  
Serial.println("mL");  
  
Serial.print("durasil");  
  
Serial.println(durasil);  
  
Serial.print("jarak1");  
  
Serial.println(jarak1);  
  
Serial.print("nilail");  
  
Serial.println(nilail);  
  
Serial.println("L");  
  
// Menampilkan pada LCD 20X4  
  
lcd.setCursor(4,0);  
  
lcd.print("DATA VOLUME:");  
  
lcd.setCursor(0,1);  
  
lcd.print("SABUN CAIR =");  
  
digitalWrite(trig, HIGH);  
  
lcd.setCursor(13,1);  
  
lcd.print(nilai);  
  
lcd.setCursor(18,1);  
  
lcd.print("mL");  
  
  
lcd.setCursor(0,2);  
  
lcd.print("AIR =");
```

```

    digitalWrite(trig1, HIGH);

    delay(50);

    lcd.setCursor(7,2);

    lcd.print(nilai1);

    lcd.setCursor(18,2);

    lcd.print("L");

// Membaca nilai sensor dan menampilkannya di serial monitor

    int baca = digitalRead(10);

    Serial.println(baca);

    int baca1 = digitalRead(11);

    Serial.println(baca1);

// Kondisi setelah sensor aktif dan tidak aktif,jika LOW 1 dan
jika HIGH 0

    if (baca == 0)

    {

        digitalWrite(sabun, LOW);

    }

    else

    {

        digitalWrite(sabun, HIGH);

    }

    if (baca1 == 0 || x == 1)

```

```
{  
  digitalWrite(air, LOW);  
}  
else  
{  
  digitalWrite(air, HIGH);  
}  
}
```

2. PROGRAM NodeMCU

```
#define BLYNK_PRINT Serial //library Blynk

#include <SoftwareSerial.h>

#include <BlynkSimpleEsp8266.h>

#include <ESP8266WiFi.h>

BlynkTimer timer;

// Kode konfirmasi dari Email untuk pendaftaran blynk

char auth[] = "g17ogFhVICtyFV8KaSZxi5zEqObwSiCs";

//Isikan nama WiFi dan password nya

char ssid[] = "Redmi-4A";

char pass[] = "12345678";

//Deklarasi variabel

int A, B;

char c;

String str;

float s, air, SC, AIR;

//Pin serial komunikasi pada node mcu

SoftwareSerial NodeMCU(D5, D6);

void setup()

{

    Serial.begin(9600);

    NodeMCU.begin(4800);
```

```
Blynk.begin(auth, ssid, pass);

pinMode(D5, INPUT);

pinMode(D6, OUTPUT);

}

void loop()

{

    while (NodeMCU.available() > 0)

    {

//Program sabun cair

        {

            SC=NodeMCU.parseFloat();

            c=NodeMCU.read();

            if(c=='A')

            {

                s=SC;

                Serial.println(s);

                str = " ";

                c=0;

            }

            if(c!='A')

            {

                str+=c;

            }

        }

    }

}
```



```
    }

    Blynk.virtualWrite(V3, s);

}

//Program air

{

AIR=NodeMCU.parseFloat();

c=NodeMCU.read();

if(c=='B')

{

    air=AIR;

    Serial.println(air);

    str =" ";

    c=0;

}

if(c!='B')

{

    str+=c;

}

    Blynk.virtualWrite(V4, air);

}

}

Blynk.run();

timer.run();
```

1. PROGRAM APLIKASI

Screen 1

```
when SignUpBTN . Click
do
  call FirebaseDB1 . StoreValue
  tag TextBox1 . Text
  valueToStore PasswordTextBox1 . Text

when loginBTN . Click
do
  call FirebaseDB1 . GetValue
  tag TextBox1 . Text
  valueIfTagNotThere "NA"

when FirebaseDB1 . GotValue
tag value
do
  if get tag == TextBox1 . Text
  then
    if get value == PasswordTextBox1 . Text
    then
      set VerticalArrangement1 . Visible to false
      open another screen screenName "Screen2"
    else
      set Label2 . Text to "Username dan Password Salah"
```

Screen 2

```
to Hiden
do
  set Screenhome . Visible to false
  set AI1 . Visible to false
  set AI2 . Visible to false

when Alat1btn . Click
do
  set global state to false
  call Hiden
  set AI1 . Visible to true

when Alat2btn . Click
do
  set global state to false
  call Hiden
  set AI2 . Visible to true

initialize global state to false

when Screen2 . BackPressed
do
  if get global state == true
  then
    set global state to false
    call Hiden
    set Screenhome . Visible to true
  else
    close application

when HomeBTN . Click
do
  set global state to false
  call Hiden
  set Screenhome . Visible to true
```

```
initialize global data1 to ""
initialize global data2 to ""
initialize global data3 to ""
initialize global data4 to ""

initialize global getV1 to "/get/V1"
initialize global getV2 to "/get/V2"
initialize global getV3 to "/get/V3"
initialize global getV4 to "/get/V4"

initialize global server to "http://blynk-cloud.com"

initialize global token to "/g17ogFhVICtyFV8KaSZxi5zEqObwSiCs"
```

