**SISTEM PENGUKUR VOLUME AIR BERBASIS *INTERNET OF***

***THINGS (IOT)* MENGGUNAKAN SENSOR ULTRASONIK HC-SR04**

***CAPSTONE PROJECT***

**Sebagai Salah Satu Syarat Memenuhi Matakuliah *Capstone Project***

**di Fakultas Teknik**

**Universitas Wiralodra**



**Oleh:**

|  |  |
| --- | --- |
| Adi Mulyana | (562020122024) |
| Deden Nurhidayah Hardadi | (562020122007) |
| Ainu Fahad Mansiz | (562020122022) |
| Setyo Ramdhani | (562020122054) |
| Muchamad Fikriy | (562020122063) |

**PROGRAM STUDI TEKNIK KOMPUTER**

**FAKULTAS TEKNIK**

**UNIVERSITAS WIRALODRA**

**HALAMAN PENGESAHAN**

**SISTEM PENGUKUR VOLUME AIR BERBASIS *INTERNET OF***

***THINGS (IOT)* MENGGUNAKAN SENSOR ULTRASONIK HC-SR04**

**Oleh:**

|  |  |
| --- | --- |
| **Adi Mulyana** | **562020122024** |
| **Deden Nurhidayah Hardadi** | **562020122007** |
| **Ainu Fahad Mansiz** | **562020122022** |
| **Setyo Ramdhani** | **562020122054** |
| **Muchamad Fikriy** | **562020122063** |

**Menyetujui**

|  |  |
| --- | --- |
| **Dosen Pembimbing 1** | **Dosen Pembimbing 2** |
|  |  |
| **Darsanto, S.T, M.Kom** | **Muh Pauzan, S.Si, M.Sc** |
| **NIDN. 0401097203** | **NIDN. 0408089003** |

**Mengetahui**

|  |
| --- |
| **Ketua Program Studi Teknik Komputer** |
|  |
| **Muh Pauzan, S.Si, M.Sc** |
| **NIDN. 0408089003**  **IDENTITAS TIM PENGUJI**   |  |  | | --- | --- | | Judul | : SISTEM PENGUKUR VOLUME AIR BERBASIS *INTERNET OF THINGS (IOT)* MENGGUNAKAN SENSOR ULTRASONIK HC-SR04 | | Nama Mahasiswa | : Adi Mulyana | | NPM | : 562020122024 | | Nama Mahasiswa | : Deden Nurhidayah Hardadi | | NPM | : 562020122007 | | Nama Mahasiswa | : Ainu Fahad Mansiz | | NPM | : 562020122022 | | Nama Mahasiswa | : Setyo Ramdhani | | NPM | : 562020122054 | | Nama Mahasiswa | : Muchamad Fikriy | | NPM | : 562020122063 | | PENGUJI PEMBIMBING |  | | Penguji 1 | : Darsanto, S.T, M.Kom. | | Penguji 2 | : Muh Pauzan, S.Si, M.Sc. | | Tanggal Ujian | : 15 Juli 2025 | |

# KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan *Capstone Project* yang berjudul "Sistem Pengukur Volume Air Berbasis *Internet of Things (IoT)* Menggunakan Sensor Ultrasonik HC-SR04" ini dengan baik, Laporan ini disusun sebagai salah satu syarat untuk menyelesaikan matakuliah *Capstone Project* pada program studi Teknik Komputer Fakultas Teknik Universitas Wiralodra. Dalam proses penyusunan dan pelaksanaan *Capstone Project* ini, penulis banyak mendapatkan bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Darsanto, S.T, M.Kom selaku dosen pembimbing 1 yang telah memberikan bimbingan, saran, dan dukungan dalam pengerjaan *Capstone Project* ini.
2. Muh Pauzan, S.Si, M.Sc selaku dosen pembimbing 2 yang telah memberikan masukan dan panduan dalam pengerjaan *Capstone* *Project* ini.
3. Orangtua dan keluarga yang selalu memberikan do’a dan dukungan moral kepada penulis.
4. Teman-teman mahasiswa Teknik Komputer yang sudah berbagi ilmu dan pengalaman berharga dalam mengerjakan *Capstone* *Project* ini.

Tim penulis menyadari bahwa laporan *Capstone* *Project* yang kami buat masih jauh dari kata sempurna, baik dalam segi penyusunan kata maupun format laporan ini yang masih banyak kekurangan dan kesalahan, untuk itu kami menerima masukan berupa kritik dan saran untuk kedepannya agar lebih baik, penulis berharap karya ilmiah ini dapat bermanfaat bagi semua orang.

|  |
| --- |
| Indramayu 15 Juli 2025 |
|  |
| Tim Penulis. |

**DAFTAR ISI**

[KATA PENGANTAR iii](#_Toc205286039)

[DAFTAR GAMBAR vi](#_Toc205286040)

[DAFTAR TABEL vii](#_Toc205286041)

[DAFTAR LAMPIRAN viii](#_Toc205286042)

[BAB I PENDAHULUAN 1](#_Toc205286043)

[1. 1. Latar Belakang 1](#_Toc205286044)

[1. 2. Rumus Masalah 1](#_Toc205286045)

[1. 3. Tujuan 2](#_Toc205286046)

[1. 4. Batasan 2](#_Toc205286047)

[BAB II DASAR TEORI 4](#_Toc205286048)

[2. 1. NodeMCU ESP32 Devkitc V4 4](#_Toc205286049)

[2. 2. PCB 5](#_Toc205286050)

[2. 3. Ultrasonic HC-SR04 5](#_Toc205286051)

[2. 4. Layar LCD 16x2 I2C 6](#_Toc205286052)

[2. 5. Modul RTC DS3231 7](#_Toc205286053)

[2. 6. Visual Studio Code 7](#_Toc205286054)

[2. 7. KiCad 8](#_Toc205286055)

[2. 8. Kabel Jumper 9](#_Toc205286056)

[2. 9. Step UP Converter MT3608 9](#_Toc205286057)

[2. 10. Arduino IDE 10](#_Toc205286058)

[2. 11. yEd *Graph Editor* 11](#_Toc205286059)

[2. 12. HTML 11](#_Toc205286060)

[2. 13. CSS 12](#_Toc205286061)

[2. 14. PHP 12](#_Toc205286062)

[2. 15. JavaScript 13](#_Toc205286063)

[*2. 16.* Diagram Alir *(Flowchart)* 14](#_Toc205286064)

[BAB III METODOLOGI 17](#_Toc205286065)

[3. 1. Komponen dan alat yang digunakan 17](#_Toc205286066)

[3. 2. Cara Kerja Alat 17](#_Toc205286067)

[3. 3. Perancangan Perangkat Keras 18](#_Toc205286068)

[3.3.1. Skematik 18](#_Toc205286069)

[3.3.2. Desain PCB 20](#_Toc205286070)

[3. 4. Perancangan Perangkat Lunak 20](#_Toc205286071)

[3. 5. Casing Alat 22](#_Toc205286072)

[3. 6. Rencana Pengujian 23](#_Toc205286073)

[BAB IV HASIL DAN PEMBAHASAN 24](#_Toc205286074)

[4.1. Pengujian Alat dan ambil data 24](#_Toc205286075)

[*4.2.* *Website* 27](#_Toc205286076)

[4.2. Pembahasan 29](#_Toc205286077)

[BAB V PENUTUP 31](#_Toc205286078)

[5.1. Kesimpulan 31](#_Toc205286079)

[5.2. Saran 31](#_Toc205286080)

[DAFTAR PUSTAKA 32](#_Toc205286081)

[LAMPIRAN 33](#_Toc205286082)

# DAFTAR GAMBAR

[Gambar 2. 1, NodeMCU ESP32 Devkitc V4 4](#_Toc204671541)

[Gambar 2. 2, Papan PCB 5](#_Toc204671542)

[Gambar 2. 3, Ultrasonik HC-SR04 5](#_Toc204671543)

[Gambar 2. 4, LCD 16x2 I2C 6](#_Toc204671544)

[Gambar 2. 5, RTC DS3231 7](#_Toc204671545)

[Gambar 2. 6, Visual Studio Code 7](#_Toc204671546)

[Gambar 2. 7, KiCad 8](#_Toc204671547)

[Gambar 2. 8, Kabel Jumper 9](#_Toc204671548)

[Gambar 2. 9, Step up MT3608 9](#_Toc204671549)

[Gambar 2. 10 Arduino IDE 10](#_Toc204671550)

[Gambar 2. 11 yEd Graph Editor 11](#_Toc204671551)

[Gambar 2. 12 HTML *(Hypertex Markup Language)* 11](#_Toc204671552)

[Gambar 2. 13, CSS *(Cascading Style Sheet)* 12](#_Toc204671553)

[Gambar 2. 14, PHP *(Hypertext Processor)* 12](#_Toc204671554)

[Gambar 2. 15 JavaScript 13](#_Toc204671555)

[Gambar 3. 3. 1, Skematik Perangkat Keras *(Hardware)* 19](#_Toc204671556)

[Gambar 3. 3. 2, *Layout* desain PCB 20](#_Toc204671557)

[Gambar 4. 1 Tampilan awal *website* 27](#_Toc204671560)

[Gambar 4. 2, Tampilan *Dashboard* 28](#_Toc204671561)

[Gambar 4. 3, Kondisi air rendah 28](#_Toc204671562)

[Gambar 4. 4, Kondisi air sedang 29](#_Toc204671563)

[Gambar 4. 5, Kondisi air penuh 29](#_Toc204671564)

# DAFTAR TABEL

[Table 1, *Flowchart* 14](#_Toc204671777)

[Table 2, Komponen dan Alat 17](#_Toc204671778)

[Table 3, Konektivitas ESP32 ke Ultrasonik HC-SR04 19](#_Toc204671779)

[Table 4, Konektivitas ESP32 ke LCD 16x2 I2C 19](#_Toc204671780)

[Table 5, Konektivitas ESP32 ke RTC DS3231 19](#_Toc204671781)

[Table 6, Kalibrasi Sensor 24](#_Toc204671782)

[Table 7, Pengambilan data volume air 26](#_Toc204671783)

# DAFTAR LAMPIRAN

[Lampiran 1, Coding ESP32 pada Arduino IDE 33](#_Toc204713329)

[Lampiran 2, Coding Konfig.php 37](#_Toc204713330)

[Lampiran 3, Coding menampilkan informasi 38](#_Toc204713331)

[Lampiran 4, Coding menampilkan chart Presentase 38](#_Toc204713332)

[Lampiran 5, Coding menampilkan chart Volume Air 39](#_Toc204713333)

[Lampiran 6, Dokumentasi pengambilan data 40](#_Toc204713334)

# BAB I PENDAHULUAN

## Latar Belakang

Air merupakan kebutuhan dasar yang sangat penting dalam kehidupan manusia, baik untuk konsumsi, kegiatan rumah tangga, pertanian, maupun industri. Oleh karena itu, pemantauan volume air menjadi hal yang krusial, terutama dalam situasi di mana ketersediaan air terbatas atau penggunaan air harus dikelola secara efisien. Namun, metode pengukuran volume air yang masih dilakukan secara manual dinilai kurang praktis, tidak efisien, dan rawan kesalahan.

Seiring dengan perkembangan teknologi, penerapan sistem berbasis *Internet of Things (IoT)* semakin luas dalam berbagai bidang, termasuk pengelolaan sumber daya air. IoT memungkinkan perangkat saling terhubung dan bertukar data secara *real-time* melalui koneksi internet. Dalam konteks pengukuran volume air, teknologi ini dapat dimanfaatkan untuk mengembangkan sistem yang mampu memantau kondisi volume air secara otomatis dan jarak jauh, sehingga memberikan efisiensi baik dari segi waktu, tenaga, maupun akurasi data.

*Capstone Project* ini bertujuan untuk merancang dan mengimplementasikan sistem pengukur volume air berbasis IoT menggunakan sensor ultrasonik HC-SR04 sebagai alat utama untuk membaca ketinggian permukaan air. Sensor ini mengukur jarak antara sensor dan permukaan air, lalu data tersebut dikonversi menjadi volume berdasarkan ketinggian wadah dan jarak sensor. Mikrokontroler NodeMCU ESP32 digunakan sebagai pusat kendali sistem karena memiliki konektivitas Wi-Fi untuk mengirimkan data ke *server* atau aplikasi monitoring secara *real-time*. Untuk mendukung tampilan lokal, digunakan LCD 16x2 I2C, dan pencatatan waktu yang presisi dilakukan oleh modul RTC DS3231, Data yang diperoleh kemudian ditampilkan melalui antarmuka *website*.

## Rumus Masalah

Rumusan masalah yang ditemukan pada *Capstone Project* ini diantaranya sebagai berikut :

1. Bagaimana merancang alat pengukur volume air?
2. Apa saja komponen utama yang dibutuhkan untuk membangun alat pengukur volume air?
3. Bagaimana cara mengintegrasikan sensor ultrasonik HC-SR04 untuk mengukur volume air dengan sistem pemantauan berbasis *IoT*?
4. Bagaimana cara memvisualisasikan data volume air secara *real-time* agar mudah dipahami pengguna?

## Tujuan

*Capstone* *Project* ini bertujuan untuk merancang dan membangun sebuah alat pengukur volume air yang dapat memberikan informasi secara *real-time* kepada pengguna. Adapun tujuan spesifik dari projek ini adalah sebagai berikut:

1. Merancang dan mengembangkan alat pengukur volume air yang dapat melakukan pengukuran volume air sehingga memudahkan pengguna dalam memantau volume air.
2. Mengimplementasikan sensor ultrasonik HC-SR04 dalam mengukur volume air dengan sistem pemantauan berbasis *Internet of things (IoT).*
3. Menyajikan data volume air dalam bentuk informasi yang mudah dipahami pengguna dan ditampilkan dalam *website*.

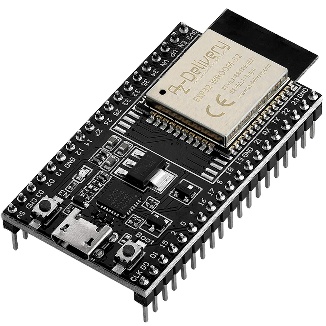
## Batasan

*Capstone Project* ini dilakukan untuk merancang dan mengembangkan alat pengukur volume air berbasis *IoT* menggunakan sensor ultrasonik dan mikrokontroler NodeMCU ESP32 V4. Mengingat luasnya ruang lingkup dalam pengembangan teknologi pemantauan volume air, penelitian ini memiliki batasan pada aspek tertentu. adapun batasan penelitian ini adalah sebagai berikut:

1. Sistem hanya menggunakan sensor ultrasonik HC-SR04 untuk mengukur ketinggian permukaan air sebagai dasar perhitungan volume. Sensor ini bekerja berdasarkan waktu pantulan gelombang ultrasonik dari permukaan air.
2. Penelitian ini dibatasi pada pengukuran volume air dalam wadah berbentuk atau dengan dimensi tertentu yang sudah diketahui sebelumnya. Sistem tidak secara otomatis menyesuaikan perhitungan volume berdasarkan bentuk wadah yang berbeda.
3. Tampilan informasi volume air secara lokal hanya menggunakan LCD 16x2 I2C, sementara informasi waktu diperoleh dari modul RTC DS3231. Dan hasil pengukuran, informasi waktu ditampilkan dalam tampilan berbasis *website*.

# BAB II DASAR TEORI

## NodeMCU ESP32 Devkitc V4



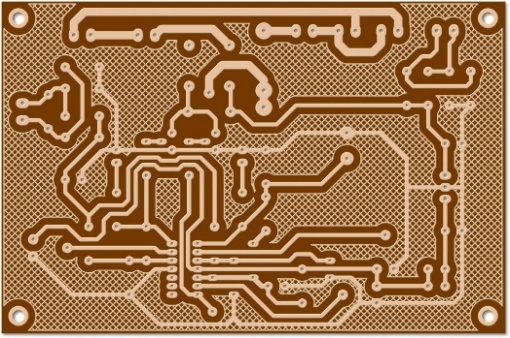
Gambar 2. , NodeMCU ESP32 Devkitc V4

NodeMCU ESP32 adalah mikrokontroler yang dikenalkan oleh *Espressif* *System* merupakan penerus dari mikrokontroler ESP8266 .Perbedaan yang menjadi keunggulan mikrokontroler ESP32 dibanding dengan mikrokontroler yang lain, mulai dari *pin* *out* nya yang lebih banyak, pin analog lebih banyak, memori yang lebih besar, terdapat *bluetooth* 4.0 *low* *energy* serta tersedia WiFi yang memungkinkan untuk mengaplikasikan *Internet* *of* *Things* dengan mikokontroler ESP32. (Suriana et al., 2022)

Mikrokontroler ESP32 juga merupakan mikrokontroler SoC *(System on Chip)* terpadu dengan dilengkapi *WiFi* 802.11 b/g/n, *Bluetooth* versi 4.2, dan berbagai peripheral. ESP32 adalah chip yang cukup lengkap, terdapat prosesor, penyimpanan dan akses pada GPIO *(General Purpose Input Output)*. ESP32 bisa digunakan untuk rangkaian pengganti pada Arduino, ESP32 memiliki kemampuan untuk mendukung terkoneksi ke *WiFi* secara langsung.

Adapun spesifikasi dari ESP32 adalah sebagai berikut: Board ini memiliki dua versi, yaitu 30 GPIO dan 36 GPIO. Keduanya memiliki fungsi yang sama tetapi versi yang 30 GPIO dipilih karena memiliki dua pin GND. Semua pin diberi label dibagian atas board sehingga mudah untuk dikenali. *Board* ini memiliki *interface* USB to UART yang mudah diprogram dengan program pengembangan aplikasi seperti Arduino IDE. Sumber daya *board* bisa diberikan melalui konektor micro USB (Nizam et al., 2022)

## PCB



Gambar 2. , Papan PCB

PCB *(Print Circuit Board)* merupakan *circuit* atau jalur-jalur rangkaian elektronik yang memiliki konduktivitas dari bahan konduktor seperti tembaga, dibuat pada sebuah *circuit* *board* atau papan sirkuit guna untuk penghubung antar komponen-komponen elektronik.

Penggunaan PCB yang digunakan secara luas mengandung banyak bahan kimia berbahaya, salah satu contoh bahan tersebut adalah FR4. Bahaya dari bahan-bahan yang terdapat dalam PCB bisa mengakibatkan kerusakan lingkungan jika alat elektronik yang sudah tidak terpakai dibuang pada sembarang tempat, contoh kecil pembuangan sebuah alat elektronik yang sudah usang pada sebuah sungai. Hal tersebut bisa menyebabkan pencemaran ekosistem sungai yang bisa membahayakan makhluk hidup yang menggunakan air sungai.

## Ultrasonic HC-SR04

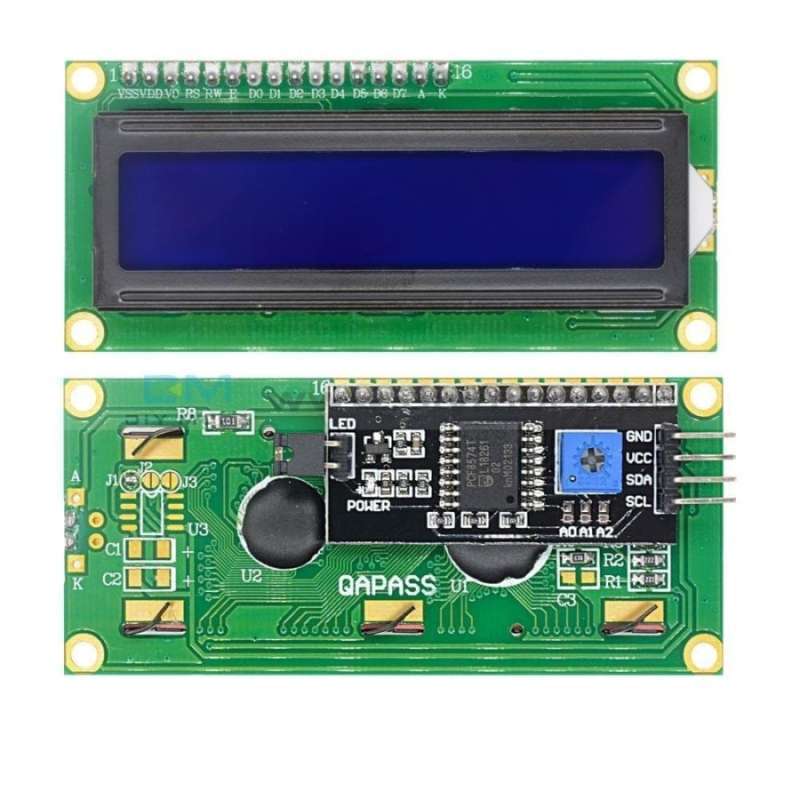


Gambar 2. , Ultrasonik HC-SR04

Sensor ultrasonik adalah sebuah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. Gelombang ultrasonik adalah gelombang bunyi yang mempunyai frekuensi 20.000 Hz. Bunyi ultrasonik tidak dapat di dengar oleh telinga manusia. Bunyi ultrasonik bisa merambat melalui zat padat, cair dan gas. Reflektivitas bunyi ultrasonik di permukaan zat padat hampir sama dengan reflektivitas bunyi ultrasonik di permukaan zat cair namun, gelombang bunyi ultrasonik akan diserap oleh tekstil.

HC-SR04 merupakan sensor ultrasonik siap pakai, satu alat yang berfungsi sebagai pengirim, penerima, dan pengontrol gelombang ultrasonik. Alat ini bisa digunakan untuk mengukur jarak benda dari 2cm – 4m dengan akurasi 3mm. Cara kerja Ultrasonik HC-SR04 adalah dengan cara memancarkan gelombang ultrasonik pada frekuensi 40.000 Hz yang merambat melalui udara dan jika ada suatu benda atau halangan pada range pancaran gelombang, gelombang ultrasonik tersebut akan memantul kembali ke modul (Yudha & Sani, 2017).

## Layar LCD 16x2 I2C



Gambar 2. , LCD 16x2 I2C

LCD I2C adalah modul tampilan yang menggunakan komunikasi I2C *(Inter-Integrated Circuit)* untuk menghubungkan layar LCD, seperti LCD 16x2, dengan mikrokontroler seperti Arduino atau ESP32. Modul ini menyederhanakan koneksi karena hanya memerlukan dua pin data utama, yaitu SDA *(Serial Data)* dan SCL *(Serial Clock),* dibandingkan dengan koneksi paralel yang membutuhkan lebih banyak pin. Dengan I2C, pengguna dapat menghemat pin mikrokontroler dan tetap menampilkan data seperti teks, angka, atau informasi sensor dengan jelas pada layar.

Selain itu, modul I2C biasanya memiliki potensiometer bawaan untuk mengatur kontras layar, serta alamat I2C yang dapat dikonfigurasi jika menggunakan lebih dari satu perangkat I2C dalam sistem. Hal ini menjadikan LCD I2C pilihan yang efisien dan praktis dalam berbagai proyek elektronika dan IoT.

## Modul RTC DS3231

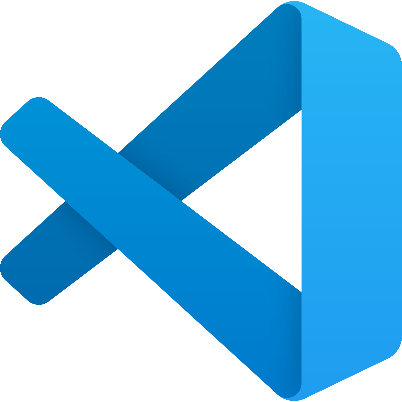


Gambar 2. , RTC DS3231

Modul RTC *(Real Time Clock)* adalah suatu rangkaian terpadu (IC) yang berperan sebagai penyimpan informasi tentang waktu dan tanggal. RTC DS3231 merupakan varian dari RTC yang mampu menyimpan data detik, menit, jam, tanggal, bulan, hari dalam seminggu, dan tahun, dengan masa berlaku yang berlangsung hingga tahun 2100 RTC DS3231 menggunakan jalur data paralel dan memiliki antarmuka serial *two-wire* (I2C) Komunikasi I2C menggunakan dua saluran, yaitu saluran Data Serial (SDA) dan saluran *Clock Serial* (SCL), untuk membaca konten *register* dari RTC (Safrilly & Badarudin, 2024).

Dalam perancangan sistem ini, RTC DS3231 berperan sebagai pengatur waktu, yang memungkinkan perangkat untuk mencatat waktu saat pengukuran dilakukan dan mengatur penyimpanan data berdasarkan waktu yang tercatat tersebut

## Visual Studio Code



Gambar 2. , Visual Studio Code

Visual Studio Code (VSCode) adalah editor kode sumber terbuka *(Open Source)* yang dikembangkan oleh Microsoft. VSCode telah menjadi pilihan utama bagi banyak pengembang karena fleksibilitas, kinerja yang ringan, dan ekosistem ekstensinya. VSCode juga menyediakan lingkungan pengembangan yang ideal untuk proyek-proyek besar dan kompleks karena kemampuannya untuk mendukung berbagai bahasa pemrograman dan alat pengembangan (Aprilia et al., 2024).

salah satu keunggulan utama VSCode adalah kemampuannya untuk mengelola proyek besar secara efisien. VSCode menyediakan alat navigasi seperti ekstensi serta manajemen file yang memungkinkan pengembang untuk mengatur dan mengakses berbagai komponen proyek dengan mudah.

## KiCad



Gambar 2. , KiCad

KiCad adalah perangkat lunak sumber terbuka *(Open Source)* yang dirancang untuk membantu dalam merancang dan membuat papan sirkuit cetak (PCB) secara profesional. Dengan kompatibilitas lintas *platform*, KiCad dapat dijalankan di berbagai sistem operasi, termasuk Windows, macOS, dan Linux (Asep Burhan Subagja, Ema, 2024).

KiCad juga dirancang untuk membantu pengguna dalam proses desain sirkuit elektronik dan papan sirkuit cetak (PCB). Dengan KiCad, pengguna dapat membuat skematik rangkaian elektronik menggunakan editor skematik, yang memungkinkan mereka untuk menggambar dan mengatur komponen dengan mudah. Setelah skematik selesai, pengguna dapat beralih ke editor PCB untuk merancang tata letak fisik papan sirkuit, di mana mereka dapat menempatkan komponen, menghubungkan jalur, dan memastikan desain memenuhi spesifikasi yang diinginkan.

KiCad juga dilengkapi dengan manajemen komponen yang memungkinkan pengguna untuk mengelola dan menggunakan pustaka komponen elektronik yang luas, sehingga memudahkan dalam pemilihan dan penempatan komponen. Selain itu, perangkat lunak ini dapat menghasilkan file Gerber, yang merupakan format standar yang diperlukan untuk produksi PCB, sehingga memudahkan transisi dari desain ke manufaktur. Fitur tambahan seperti *3D viewer* yang memungkinkan pengguna untuk melihat desain PCB dalam tampilan tiga dimensi, memberikan gambaran yang lebih jelas tentang bagaimana papan akan terlihat secara fisik

## Kabel Jumper

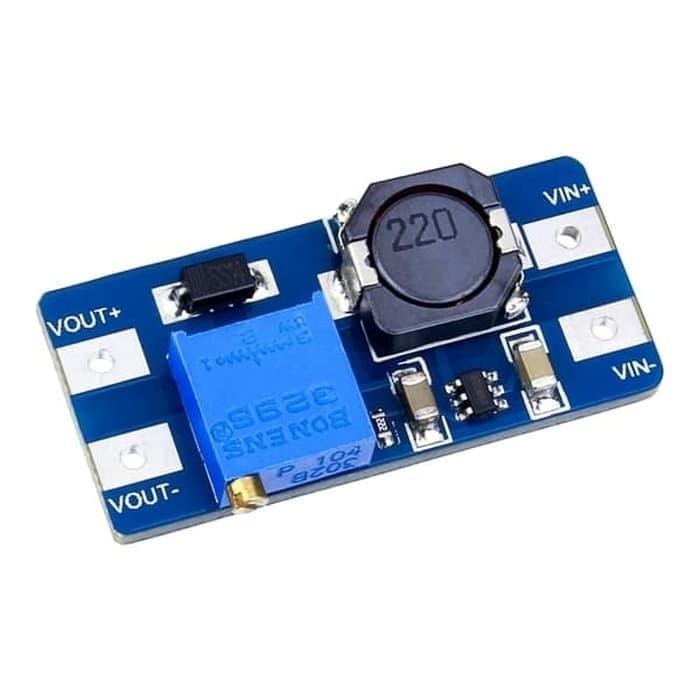


Gambar 2. , Kabel Jumper

Kabel jumper adalah kabel kecil yang digunakan untuk membuat sambungan listrik sementara antara komponen atau titik-titik pada papan sirkuit, seperti breadboard atau modul mikrokontroler. Kabel ini terdiri dari konduktor tembaga yang dilapisi oleh isolasi plastik, dan tersedia dalam berbagai warna untuk memudahkan identifikasi sambungan.

Kabel jumper terdapat dalam tiga jenis utama berdasarkan ujung konektornya: *male-to-male* (jarum ke jarum), *female-to-female* (lubang ke lubang), dan *male-to-female* (jarum ke lubang), yang masing-masing digunakan sesuai dengan jenis pin atau soket pada komponen yang dihubungkan. Kabel jumper umumnya digunakan dalam *prototyping* atau pengujian rangkaian elektronik karena memungkinkan sambungan tanpa perlu menyolder, sehingga mudah dilepas atau diubah.

## Step UP Converter MT3608



Gambar 2. , Step up MT3608

*Step-up converter* MT3608 adalah modul penguat *tegangan (boost converter)* berbasis *chip* MT3608 yang dirancang untuk meningkatkan tegangan *input* (biasanya dari baterai atau sumber tegangan rendah) menjadi tegangan *output* yang lebih tinggi. Modul ini bekerja menggunakan prinsip konversi DC-DC, di mana energi disimpan dalam induktor selama siklus ON dan dilepaskan dengan tegangan yang lebih tinggi selama siklus OFF, dengan frekuensi *switching* tinggi (hingga 1.2 MHz). MT3608 sangat efisien, mampu mengubah tegangan serendah 2V hingga mencapai *output* sekitar 28V, tergantung pada beban dan inputnya, dengan arus *output* maksimum sekitar 2A (meskipun direkomendasikan di bawah 1.5A untuk stabilitas jangka panjang).

Modul ini memiliki potensiometer kecil untuk mengatur tegangan *output* secara manual, dan banyak digunakan dalam proyek elektronik seperti *power* *supply* portabel, *charger* DIY, serta sistem IoT yang memerlukan tegangan lebih tinggi dari sumber daya terbatas seperti baterai Li-ion 3.7V. MT3608 juga memiliki fitur proteksi terhadap suhu tinggi dan arus lebih, menjadikannya pilihan praktis dan ekonomis dalam berbagai aplikasi peningkat tegangan.

## Arduino IDE



Gambar 2. Arduino IDE

Arduino IDE adalah *software* yang digunakan untuk membuat sketch pemrogaman atau dengan kata lain arduino IDE sebagai media untuk pemrogaman pada *board* yang ingin diprogram. Arduino IDE ini berguna untuk mengedit, membuat, meng-upload ke *board* yang ditentukan, dan meng-coding program tertentu. Arduino IDE dibuat dari bahasa pemrogaman JAVA, yang dilengkapi dengan *library* C/C++*(wiring),* yang membuat operasi *input/output* lebih mudah (Kamal et al., 2023).

## yEd *Graph Editor*



Gambar 2. yEd *Graph Editor*

yEd adalah aplikasi desktop yang kuat yang dapat digunakan untuk menghasilkan diagram berkualitas tinggi secara cepat dan efektif. Buat diagram secara manual, atau impor data eksternal untuk analisis. Algoritma tata letak otomatis dengan mengatur bahkan kumpulan data besar hanya dengan menekan tombol. Dengan menggunakan *tools* tersebut, dapat menggambarkan beberapa pemodelan dalam pembuatan diagram *flowchart* (Nugraha et al., 2019).

## HTML



Gambar 2. HTML *(Hypertex Markup Language)*

*Hypertext Markup Language* (HTML) adalah bahasa standar yang digunakan untuk menampilkan konten pada halaman *website*. Fungsi-fungsi yang dapat dilakukan dengan bahasa programan HTML adalah Mengatur serta mendesain tampilan isi halaman *website*, (2) Membuat tabel pada halaman *website*, Mempublikasikan halaman *website* secara *online*, Membuat form yang dapat menjadi *input* serta menangani registrasi dan transaksi via *website*, Menampilkan area gambar pada browser.

Penanda perintah pada bahasa pemrograman *Hypertext* *Markup* *Language* (HTML) disebut dengan tag. Tag digunakan untuk menentukan tampilan dari dokumen HTML. Tag HTML berfungsi untuk mendefinisikan bahwa isi dalam file tersebut adalah dokumen. Element *head* merupakan kepala dari dokumen HTML. Elemen *head* digunakan untuk menempatkan identitas file, sedangkan tag *body* digunakan untuk menentukan konten yang akan ditampilkan pada halaman *website* (Mariko, 2019).

## CSS



Gambar 2. , CSS *(Cascading Style Sheet)*

CSS *(Cascading Style Sheets)* adalah bahasa yang digunakan untuk mengatur tampilan dan tata letak elemen-elemen dalam halaman web yang ditulis menggunakan HTML. CSS memungkinkan pengembang web untuk memisahkan konten dari desain visual, sehingga struktur halaman dan gaya tampilannya bisa dikelola secara terpisah. Dengan CSS, kita dapat menentukan warna, ukuran teks, jenis *font*, *margin*, *padding*, posisi elemen, dan banyak aspek lainnya yang berkaitan dengan estetika sebuah situs web.

CSS juga mendukung penggunaan selektor dan kelas untuk menargetkan elemen tertentu secara spesifik, serta memungkinkan penggunaan media *query* untuk membuat tampilan *website* yang responsif terhadap berbagai ukuran layar. Dengan demikian, CSS memainkan peran penting dalam menciptakan halaman web yang menarik, konsisten, dan mudah diakses di berbagai perangkat.

## PHP

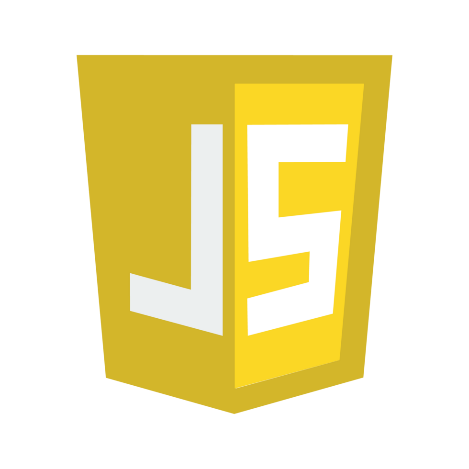


Gambar 2. , PHP *(Hypertext Processor)*

PHP *(Hypertext Preprocessor)* adalah bahasa pemrograman *server-side* yang dirancang khusus untuk pengembangan web dan dapat disisipkan ke dalam HTML. PHP pertama kali dikembangkan oleh Rasmus Lerdorf pada tahun 1994 dan sejak itu berkembang menjadi salah satu bahasa pemrograman paling populer untuk membangun situs web dinamis dan aplikasi berbasis web. Keunggulan utama PHP terletak pada kemudahannya untuk dipelajari, fleksibilitasnya, dan kompatibilitasnya dengan berbagai sistem manajemen basis data seperti MySQL, PostgreSQL, dan SQLite.

PHP dapat digunakan untuk berbagai tugas seperti mengelola formulir, mengakses dan memanipulasi *database*, mengatur sesi *(session),* hingga membuat sistem autentikasi pengguna. Selain itu, PHP didukung oleh komunitas pengembang yang sangat besar, menyediakan berbagai *framework* seperti Laravel, CodeIgniter, dan Symfony untuk mempercepat pengembangan aplikasi. PHP juga dapat berjalan di berbagai sistem operasi seperti Windows, Linux, dan macOS serta kompatibel dengan berbagai *server* web seperti Apache. Karena sifat *open source*-nya, PHP dapat digunakan secara gratis dan terus diperbarui oleh komunitasnya. Dengan kombinasi antara kemudahan penggunaan, performa yang baik, serta dukungan luas, PHP menjadi pilihan yang populer untuk membangun *website* dinamis, *e-commerce*, sistem manajemen konten (CMS) seperti *WordPress,* dan berbagai aplikasi web lainnya.

## JavaScript



Gambar 2. JavaScript

JavaScript adalah bahasa pemrograman tingkat tinggi yang bersifat dinamis, berbasis prototipe, dan dijalankan di sisi klien *(client-side),* meskipun kini juga umum digunakan di sisi *server* *(server-side)* melalui *platform* seperti Node.js. Awalnya dikembangkan oleh Netscape pada tahun 1995 dengan tujuan untuk memberikan interaktivitas pada halaman *website*, JavaScript kini telah menjadi salah satu teknologi inti dalam pengembangan *website* bersama dengan HTML dan CSS. JavaScript memungkinkan pengembang untuk membuat halaman web yang dinamis, responsif, dan interaktif, seperti menangani input pengguna, memanipulasi elemen HTML dan CSS secara *real-time*, membuat animasi, dan melakukan komunikasi dengan *server* tanpa harus memuat ulang halaman (teknik AJAX),

JavaScript bersifat *interpreter-based*, artinya kode dieksekusi baris demi baris oleh mesin JavaScript yang terdapat di browser seperti Google Chrome, dan lainnya. Bahasa ini mendukung paradigma pemrograman berorientasi objek, fungsional, dan imperatif. Saat ini, JavaScript tidak hanya digunakan untuk mengembangkan aplikasi web, tetapi juga aplikasi mobile (dengan React Native), aplikasi desktop hingga pengembangan *Internet of Things (IoT).* Dengan komunitas yang besar serta framework yang luas seperti React, Angular, dan Vue.js, JavaScript telah menjadi salah satu bahasa pemrograman paling populer dan berpengaruh di dunia pengembangan perangkat lunak.

## Diagram Alir *(Flowchart)*

*Flowchart* atau diagram alir adalah representasi grafis dari langkah-langkah atau alur kerja dalam suatu proses, sistem, atau algoritma. *Flowchart* menggunakan simbol-simbol standar, seperti persegi panjang untuk menunjukkan proses, belah ketupat untuk keputusan, dan panah untuk menunjukkan aliran logika atau urutan langkah.

*Flowchart* memiliki fungsi yang sangat penting dalam berbagai bidang seperti analisis sistem, pemrograman, dan manajemen proses. Fungsi utamanya adalah untuk memvisualisasikan alur kerja atau logika suatu proses secara jelas dan sistematis, sehingga lebih mudah dipahami dibandingkan dengan penjelasan dalam bentuk teks. *Flowchart* juga berperan dalam menyederhanakan proses yang kompleks dengan menggambarkannya langkah demi langkah, sehingga bagian-bagian penting dari sistem dapat diidentifikasi dengan mudah.

dan berikut merupakan tabel yang berisi diagram alir *(flowchart)*

Table , *Flowchart*

| **No** | **Gambar** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| 1 |  | *Terminal* | Simbol ini memiliki bentuk oval atau kapsul dan digunakan untuk menandai titik awal (*start*) dan akhir (*end*) dari suatu alur proses atau program. Simbol ini sangat penting karena menunjukkan kapan proses dimulai dan kapan proses tersebut selesai. Dalam *flowchart*, setiap diagram pasti memiliki minimal satu simbol terminator di awal dan satu di akhir untuk menunjukkan batasan alur logika yang digambarkan. |
| 2 |  | *Data* | Simbol *input/output* berbentuk jajar genjang dan digunakan untuk mewakili aktivitas masukan (*input*) atau keluaran (*output*). *Input* mencakup tindakan seperti memasukkan data dari keyboard atau sensor, sementara output bisa berupa tampilan di layar monitor, hasil cetak, atau data yang dikirim ke perangkat lain. Simbol ini membantu mengidentifikasi titik interaksi antara pengguna dan sistem. |
| 3 |  | *Decision* | Simbol keputusan berbentuk belah ketupat dan digunakan untuk menggambarkan proses pengambilan keputusan atau percabangan logika. Di dalam simbol ini biasanya terdapat pertanyaan atau kondisi yang harus dievaluasi, dan alur selanjutnya tergantung pada hasil evaluasi tersebut, misalnya “ya” atau “tidak”. Simbol ini penting dalam *flowchart* karena menentukan arah alur program berdasarkan kondisi tertentu. |
| 4 |  | *Proccess* | Simbol proses berbentuk persegi panjang dan digunakan untuk menggambarkan tindakan atau aktivitas yang dilakukan dalam sistem. Aktivitas ini bisa berupa perhitungan, pengolahan data, penyimpanan data, atau langkah-langkah operasional lainnya. Setiap langkah yang memerlukan kerja sistem atau manusia untuk memproses sesuatu biasanya digambarkan dengan simbol ini. |
| 5 |  | *Flow Direction* | Yaitu symbol yang digunakan untuk menghubungkan antara simbol yang satu dengan symbol yang lain. |

Simbol-simbol diatas memiliki jenis dan fungsi yang berbeda. Dalam sistem pengukur volume air menggunakan sensor ultrasonik HC-SR04 ini terdapat *flowchart* dimana *flowchart* ini berisikan cara kerja alat hingga menghasilkan data dan mengirimkan tampilan layer dan informasi mengenai volume air, ketika sensor mengirimkan sinyal dan menangkapnya kembali lalu hasil pembacaan sensor dihitung berdasarkan dimensi wadah tempat air dan memberikan informasi kepada pengguna.

# BAB III METODOLOGI

## Komponen dan alat yang digunakan

Berikut ini adalah perangkat yang digunakan dalam pengembangan sistem pengukur volume air menggunakan sensor ultrasonik HC-SR04. Setiap perangkat memiliki fungsi spesifik dalam mendukung proses dalam mengambil data.

Sensor ultrasonik berfungsi untuk mengukur berapa volume air yang ada pada tempat penampungan. Data dari sensor ini kemudian dihitung menggunakan program dan hasil dari perhitungan tersebut data akan dikirimkan ke *server* dan ditampilkan dalam web dan ditampilkan pada LCD I2C. dan berikut merupakan komponen dan alat yang digunakan untuk membangun sistem pengukur volume air.

Table , Komponen dan Alat

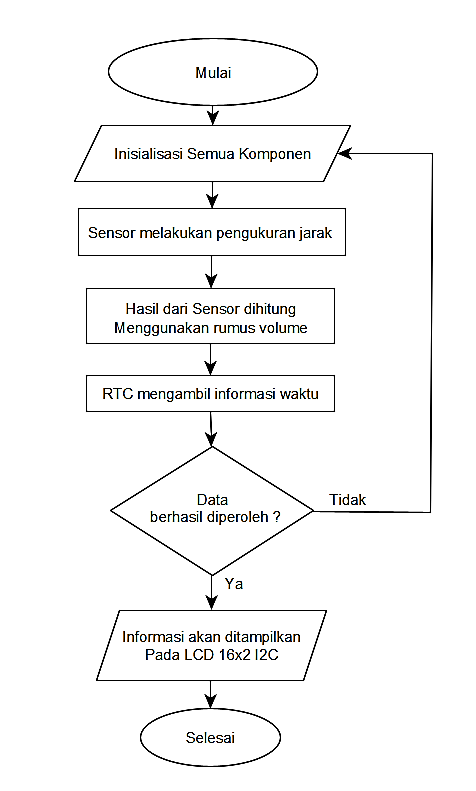
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Komponen dan Alat** | **Jumlah** |
| 1 | NodeMCU ESP32 Devkitc V4 | 1 |
| 2 | Ultrasonik HC-SR04 | 1 |
| 3 | LCD *(Liquid Crystal Display)* 16x2 I2C | 1 |
| 5 | Modul RTC DS3231 | 1 |
| 6 | Kabel Jumper *(Female to Female)* | - |

## Cara Kerja Alat

Untuk mengetahui cara kerja alat, maka dibuatkan sebuah *flowchart* sederhana, pada dasarnya *flowchart* atau diagram alur merupakan diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu program, setiap langkah digunakan dalam bentuk diahram dan dihubungkan dengan garis atau arah panah.

.

Untuk membuat *flowchart* sederhana diharuskan untuk mengetahui setiap simbol dan juga fungsinya. Berikut ini merupakan *flowchart* sederhana untuk cara kerja Sistem Pengukur Volume Air Menggunakan Sensor Ultrasonik HC-SR04, terdapat 2 *flowchart* yang dibuat, diantaranya merupakan *flowchart* cara kerja *hardware*, *flowchart* cara kerja *website*, dan *flowchart* *hardware* ke *Software*:



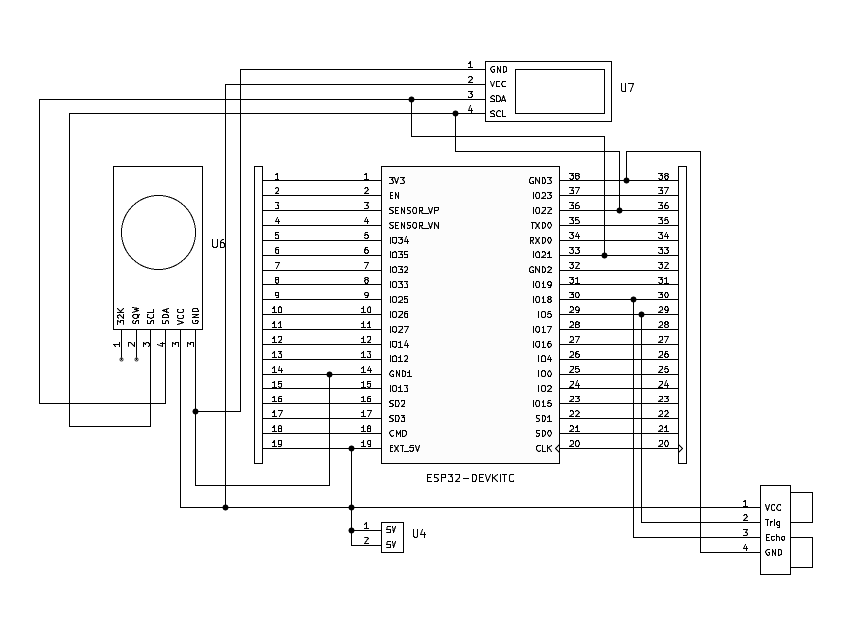
Gambar 3. *Flowchart Hardware*

Pada *flowchart* diatas menunjukkan cara kerja alat perangkat keras *(Hardware)* tanpa ada *server* maupun *website*.

## Perancangan Perangkat Keras

Rancangan alat keras yang kami buat terdiri dari NodeMCU ESP32 Wroom V4, pin header, *Pin* *Socket* untuk komponen, perancangan alat sebagai berikut :

### Skematik



Gambar 3. 3. , Skematik Perangkat Keras *(Hardware)*

Berikut merupakan konektivitas antar komponen yang saling terhubung

Table , Konektivitas ESP32 ke Ultrasonik HC-SR04

|  |  |
| --- | --- |
| **ESP32** | **Ultrasonik HC-SR04** |
| GPIO5 | Trigh Pin |
| GPIO18 | Echo Pin |
| 5V | VCC |
| GND | GND |

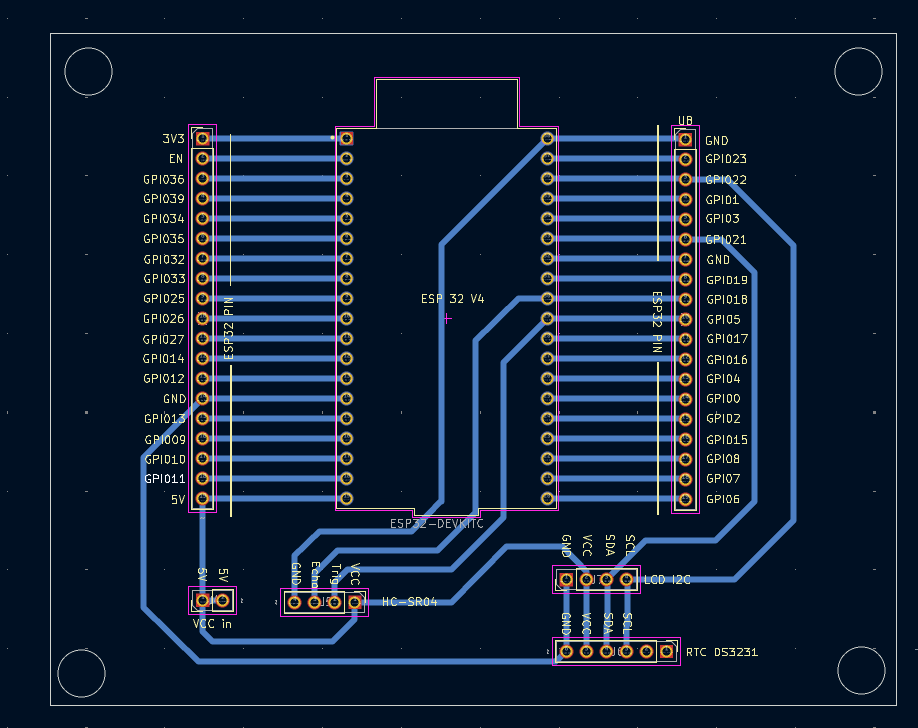
Table , Konektivitas ESP32 ke LCD 16x2 I2C

|  |  |
| --- | --- |
| **ESP32** | **LCD 16x2 I2C** |
| GPIO21 | SDA Pin |
| GPIO22 | SCL Pin |
| 5V | VCC |
| GND | GND |

Table , Konektivitas ESP32 ke RTC DS3231

|  |  |
| --- | --- |
| **ESP32** | **RTC DS3231** |
| - | 32K |
| - | SQW |
| GPIO21 | SDA |
| GPIO22 | SCL |
| 5V | VCC |
| GND | GND |

### Desain PCB



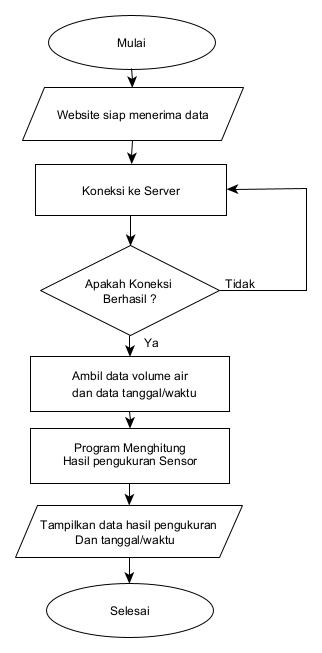
Gambar 3. 3. , *Layout* desain PCB

Pembuatan Desain PCB memberikan gambaran mengenai tata letak komponen elektronik serta bagaimana jalur koneksi antar komponen yang dirancang. Proses ini membantu dalam menentukan dimensi dan bentuk papan sirkuit, yang nantinya menjadi acuan dalam perancangan casing perangkat. Dengan demikian, casing dapat dibuat sesuai dengan kebutuhan dan memastikan setiap komponen memiliki posisi yang ideal untuk dipasang.

Selain itu, pertimbangan dalam menentukan pin-pin yang digunakan berperan penting dalam penempatan setiap komponen pada PCB. Pemilihan posisi pin yang tepat memungkinkan koneksi antar komponen lebih efisien serta mengurangi kemungkinan kesalahan pemasangan. Hal ini juga membantu dalam optimasi ruang pada PCB agar perangkat dapat berfungsi dengan baik tanpa mengalami gangguan pada sistem kelistrikan dan komunikasi antar komponen.

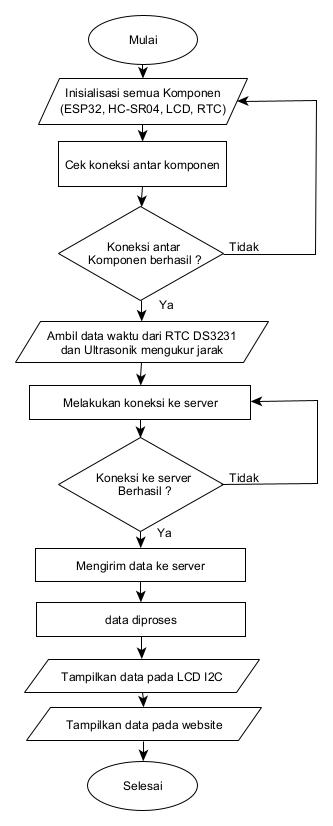
## Perancangan Perangkat Lunak

Perancangan Perangkat Lunak atau *Software* adalah sebuah perangkat yang didalamnya berisi tentang serangkaian instruksi, program, prosedur, pengendalian pendukung dan beberapa aktivitas pengolahan perintah pada sebuah sistem di komputer. Proses pembuatan dan perancangan perangkat lunak (*Software*) pada penelitian ini akan membahas aplikasi yang digunakan dalam pembuatan Sistem pengukur volume air, dapat dilihat pada *flowchart* dibawa ini.



Gambar 3. , *Flowchart Software*

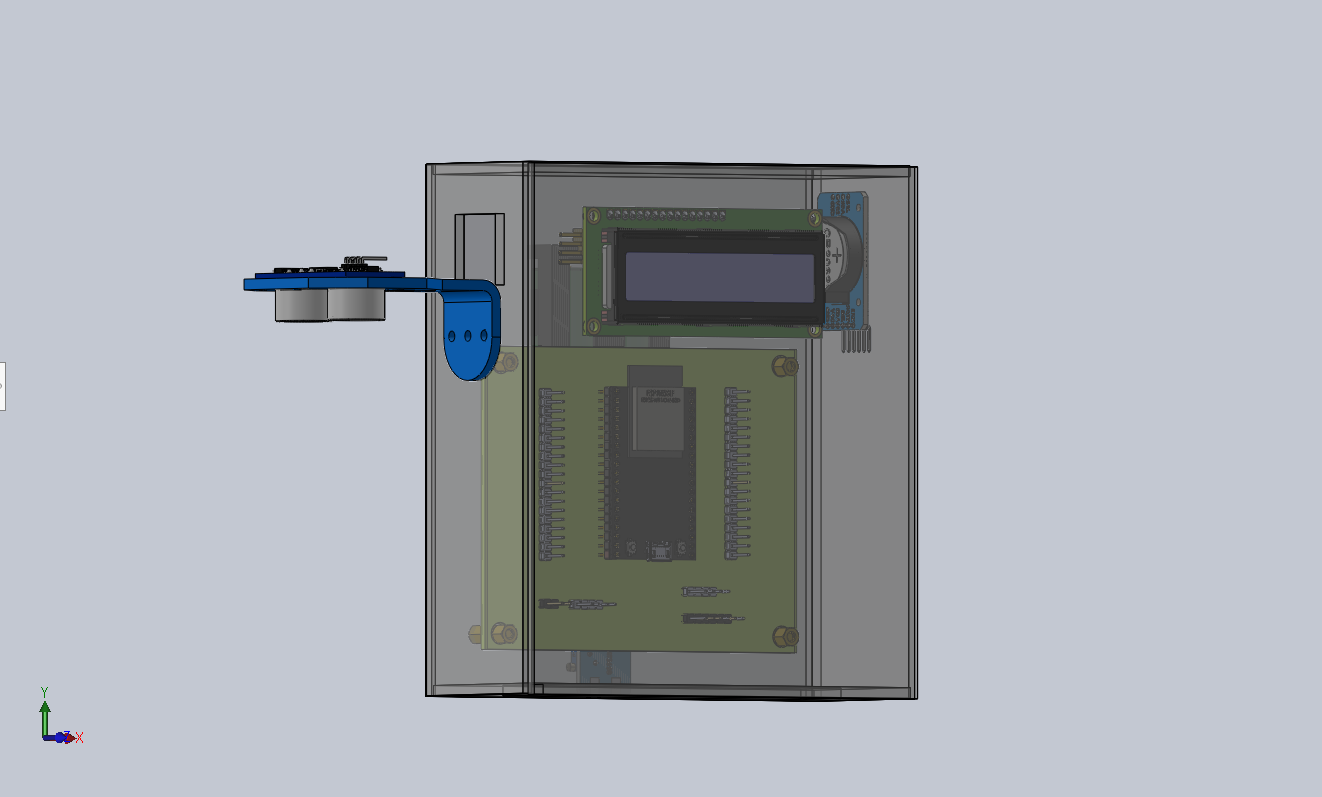
Dan berikut merupakan *flowchart* gabungan antara *software* dan *hardware:*



Gambar 3. , *Flowchart* *Hardware* dan *Software*

## Casing Alat

Berikut adalah rancangan casing yang dibuat versi 3D beserta bagian-bagian komponen yang digunakan untuk perangkat kerasnya, terdapat penempatan beberapa komponen pada tiap sisi casing dan beberapa lubang untuk jalur kabel dan berfungsi juga sebagai ventilasi udara agar panasnya alat bisa keluar melalui lubang seperti yang tertera pada gambar.



Gambar 3. , Desain 3D Casing Alat

## Rencana Pengujian

Rencana Pengujian ini bertujuan untuk mengevaluasi alat pengukuran volume air, kestabilan pengiriman data ke *server*, dan keandalan menampilkan data secara *real-time* pada *website*. Dengan melakukan pengujian ini, diharapkan sistem dapat bekerja secara optimal sesuai dengan perancangan.

Alat dan bahan yang akan digunakan dalam pengujian ini terdiri dari beberapa komponen utama, yaitu NodeMCU ESP32 V4 sebagai mikrokontroler, sensor ultrasonik HC-SR04 untuk mengukur ketinggian permukaan air, LCD 16x2 I2C untuk menampilkan hasil pengukuran, dan modul RTC DS3231 untuk mengambil informasi tanggal dan waktu secara *real-time*, Uji pengukuran dilakukan menggunakan botol air minum berukuran 15 liter sebagai media penampungan air untuk mensimulasikan volume yang diukur oleh sistem.

Data yang dihasilkan dari sensor dan modul RTC lalu diproses pada NodeMCU lalu dikirimkan ke *server*, dan diproses lalu ditampilkan ke *website* yang berisi informasi.

# BAB IV HASIL DAN PEMBAHASAN

## Pengujian Alat dan ambil data

Setelah proses perancangan dan pembuatan sistem pengukur volume air berbasis IoT selesai dilakukan, dilakukan beberapa tahap pengujian untuk memastikan bahwa alat dapat bekerja sesuai dengan yang direncanakan. Pengujian ini mencakup uji fungsional sensor, LCD, RTC, koneksi internet, serta pengiriman data ke *server*.

* Pengujian sensor ultrasonik

Kalibrasi sensor ultrasonik HC-SR04 setiap 1 cm adalah proses penting untuk meningkatkan akurasi pengukuran, terutama dalam aplikasi yang membutuhkan ketelitian tinggi seperti sistem pengukur volume air. Proses ini dilakukan dengan cara membandingkan pembacaan sensor dengan jarak sebenarnya yang diukur menggunakan alat ukur standar seperti penggaris atau mistar. Langkah pertama dalam kalibrasi adalah menyiapkan perangkat sensor yang terhubung ke mikrokontroler (seperti ESP32), kemudian menulis program dasar untuk membaca jarak menggunakan prinsip waktu tempuh gelombang ultrasonik. Setelah itu, objek pantul diletakkan secara bertahap pada jarak 1 cm, 2 cm, 3 cm, dan seterusnya hingga batas tertentu. Untuk kalibrasi ini kami mengambil maksimal 50 cm atau 50 titik pengukuran, hasil pembacaan sensor dicatat dan dibandingkan dengan jarak sebenarnya. Selisih yang ditemukan pada tiap titik akan menjadi acuan koreksi, berikut merupakan tabel kalibrasi sensor ultrasonik HC-SR04 :

Table , Kalibrasi Sensor

| **No** | **Jarak Aktual (cm)** | **Terbaca Sensor (cm)** | ***Error* (cm)** | **Koreksi** |
| --- | --- | --- | --- | --- |
| 1 | 1 | 1.66 | 0.66 | -0.66 |
| 2 | 2 | 2.32 | 0.32 | -0.32 |
| 3 | 3 | 2.97 | -0.03 | 0.03 |
| 4 | 4 | 3.94 | -0.06 | 0.06 |
| 5 | 5 | 4.27 | -0.73 | 0.73 |
| 6 | 6 | 5.57 | -0.43 | 0.43 |
| 7 | 7 | 6.55 | -0.45 | 0.45 |
| 8 | 8 | 7.85 | -0.15 | 0.15 |
| 9 | 9 | 9.21 | 0.21 | -0.21 |
| 10 | 10 | 9.55 | -0.45 | 0.45 |
| 11 | 11 | 10.92 | -0.08 | 0.08 |
| 12 | 12 | 11.95 | -0.05 | 0.05 |
| 13 | 13 | 13.33 | 0.33 | -0.33 |
| 14 | 14 | 13.99 | -0.01 | 0.01 |
| 15 | 15 | 15.04 | 0.04 | -0.04 |
| 16 | 16 | 16.38 | 0.38 | -0.38 |
| 17 | 17 | 16.69 | -0.31 | 0.31 |
| 18 | 18 | 17.65 | -0.35 | 0.35 |
| 19 | 19 | 18.62 | -0.38 | 0.38 |
| 20 | 20 | 19.59 | -0.41 | 0.41 |
| 21 | 21 | 20.55 | -0.45 | 0.45 |
| 22 | 22 | 21.51 | -0.49 | 0.49 |
| 23 | 23 | 23.05 | 0.05 | -0.05 |
| 24 | 24 | 24.22 | 0.22 | -0.22 |
| 25 | 25 | 25.09 | 0.09 | -0.09 |
| 26 | 26 | 25.67 | -0.33 | 0.33 |
| 27 | 27 | 26.84 | -0.16 | 0.16 |
| 28 | 28 | 27.41 | -0.59 | 0.59 |
| 29 | 29 | 28.86 | -0.14 | 0.14 |
| 30 | 30 | 29.41 | -0.59 | 0.59 |
| 31 | 31 | 30.72 | -0.28 | 0.28 |
| 32 | 32 | 31.76 | -0.24 | 0.24 |
| 33 | 33 | 32.29 | -0.71 | 0.71 |
| 34 | 34 | 33.6 | -0.40 | 0.40 |
| 35 | 35 | 34.11 | -0.89 | 0.89 |
| 36 | 36 | 35.93 | -0.07 | 0.07 |
| 37 | 37 | 37.1 | 0.10 | -0.10 |
| 38 | 38 | 37.32 | -0.68 | 0.68 |
| 39 | 39 | 38.71 | -0.29 | 0.29 |
| 40 | 40 | 39.17 | -0.83 | 0.83 |
| 41 | 41 | 40.32 | -0.68 | 0.68 |
| 42 | 42 | 41.71 | -0.29 | 0.29 |
| 43 | 43 | 43.06 | 0.06 | -0.06 |
| 44 | 44 | 43.27 | -0.73 | 0.73 |
| 45 | 45 | 44.88 | -0.12 | 0.12 |
| 46 | 46 | 45.28 | -0.72 | 0.72 |
| 47 | 47 | 46.9 | -0.10 | 0.10 |
| 48 | 48 | 47.28 | -0.72 | 0.72 |
| 49 | 49 | 48.29 | -0.71 | 0.71 |
| 50 | 50 | 49.68 | -0.32 | 0.32 |

Setelah dilakukan kalibrasi sensor sebanyak 50 titik pada tiap titik berjarak 1 cm, setelah didaapat hasil koreksi tiap titik, lalu hasil koreksi tersebut dalam bentuk program.

* Proses Pengukuran dan Ambil data

Tabel di bawah menunjukkan hasil pengambilan data volume air menggunakan botol air minum berkapasitas 15 liter. Pengisian dilakukan secara bertahap dari kondisi kosong (0 liter) hingga penuh (15 liter) menggunakan gelas yang memiliki takaran 1 liter, dengan kenaikan setiap 1 liter. Setiap tahapan pengisian dicatat tinggi airnya dalam satuan sentimeter serta jarak antara permukaan air ke sensor ultrasonik HC-SR04 yang terpasang di bagian atas botol air.

Table , Pengambilan data volume air

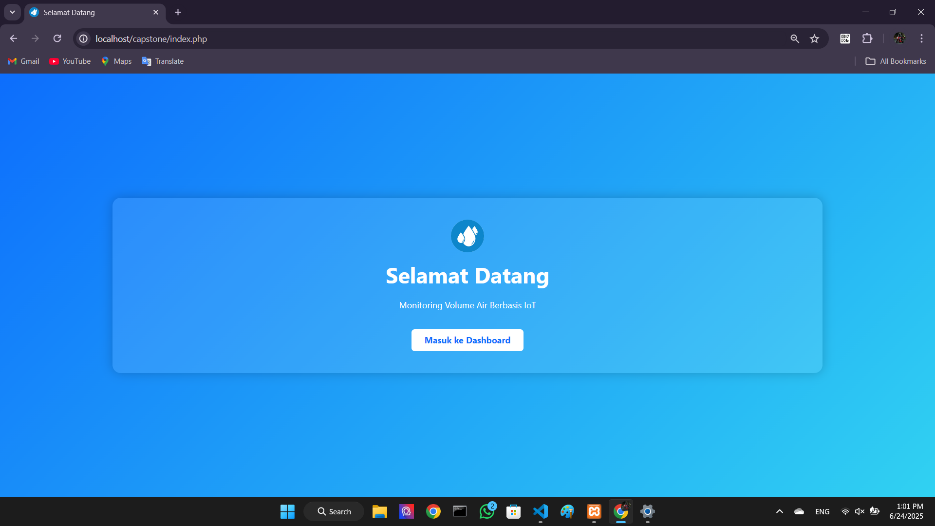
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Isi Air (L)** | **Tinggi air (cm)** | **Terbaca Sensor (L)** | **Selisih** | **Tanggal dan Waktu** |
| 1 | 0 | 0.0 | 0 | 0 | 6/11/25, 13:20 |
| 2 | 1 | 2.8 | 1.2 | 0,3 | 6/11/25, 13:29 |
| 3 | 2 | 5.6 | 2.2 | 0,2 | 6//11/25, 13:36 |
| 4 | 3 | 8.4 | 3,1 | 0,1 | 6//11/25, 13:40 |
| 5 | 4 | 11.3 | 4,1 | 0,1 | 6//11/25, 13:44 |
| 6 | 5 | 14.1 | 5.2 | 0,2 | 6//11/25, 13:49 |
| 7 | 6 | 16.9 | 6.0 | 0 | 6//11/25, 13:51 |
| 8 | 7 | 19.7 | 6.8 | 0,2 | 6//11/25, 14:00 |
| 9 | 8 | 22.5 | 7.9 | 0,1 | 6//11/25, 14:03 |
| 10 | 9 | 25.3 | 9.0 | 0 | 6//11/25, 14:05 |
| 11 | 10 | 28.1 | 9.8 | 0,2 | 6//11/25, 14:08 |
| 12 | 11 | 30.9 | 10.8 | 0,2 | 6//11/25, 14:15 |
| 13 | 12 | 33.8 | 11.7 | 0,3 | 6//11/25, 14:18 |
| 14 | 13 | 36.6 | 12.7 | 0,3 | 6//11/25, 14:32 |
| 15 | 14 | 39.4 | 14 | 0 | 6//11/25, 16:22 |
| 16 | 15 | 42.2 | 15 | 0 | 6//11/25, 16:29 |

## *Website*

*Website* pada sistem ini dirancang sebagai media monitoring volume air secara *real-time* yang dapat diakses dari jarak jauh melalui koneksi internet. Fitur ini menjadi bagian penting dalam penerapan konsep *Internet of Things* *(IoT),* di mana data yang diperoleh dari sensor tidak hanya ditampilkan secara lokal pada LCD, tetapi juga dikirimkan ke *server* dan divisualisasikan dalam bentuk web yang informatif. *Website* ini dibuat menggunakan kombinasi bahasa pemrograman HTML, CSS, JavaScript untuk tampilan, serta PHP untuk proses *backend* dan koneksi ke *database*.

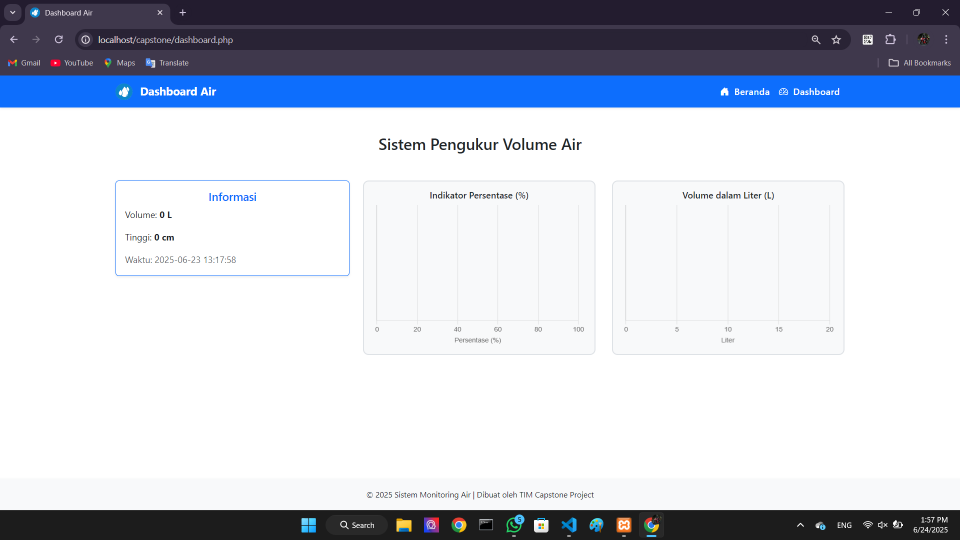
Saat alat aktif, ESP32 membaca data dari sensor ultrasonik HC-SR04 dan mengolahnya menjadi nilai volume berdasarkan tinggi permukaan air. Mikrokontroler ini kemudian menambahkan informasi waktu dari modul RTC DS3231, lalu mengirimkan data tersebut ke *server* melalui jaringan *Wi-Fi* menggunakan metode HTTP POST. Data yang diterima oleh *server* akan diproses dan disimpan ke dalam *database*, sehingga dapat ditampilkan dalam tampilan antarmuka *website*.

Tampilan halaman index.php atau awal ketika masuk *website*.



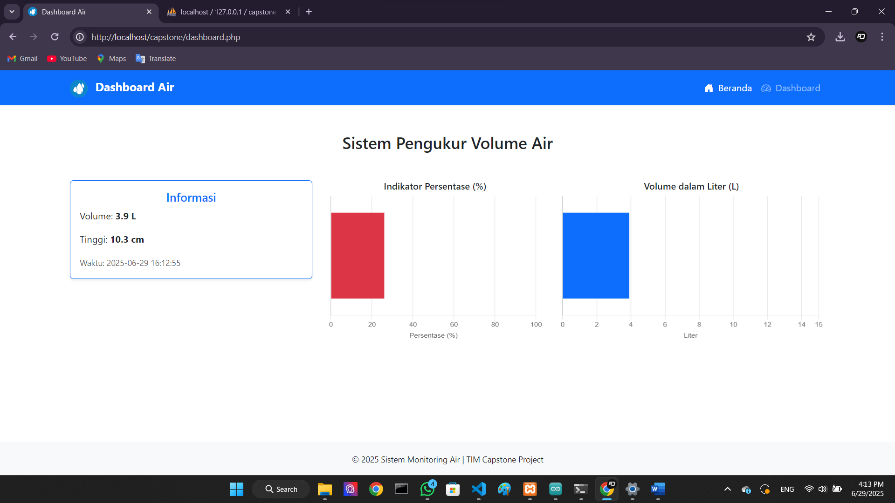
Gambar 4. Tampilan awal *website*

Berikut merupakan tampilan *dashboard* yang berisi informasi volume air dalam satuan liter (L) , informasi ketinggian permukaan air dalam satuan sentimeter (cm) dan informasi waktu terbaru pada *card* di sebelah kiri, serta penggunaan *chart* presentase di tengah dan insformasi volume air dalam bentuk *chart* di kanan, *chart* yang digunakan diambil melalui *link* CDN *(Content Delivery Network)*

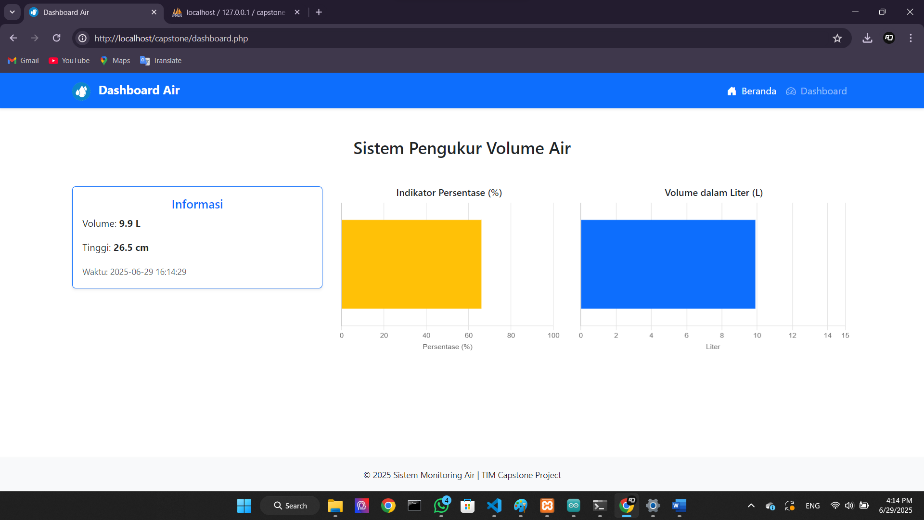


Gambar 4. , Tampilan *Dashboard*

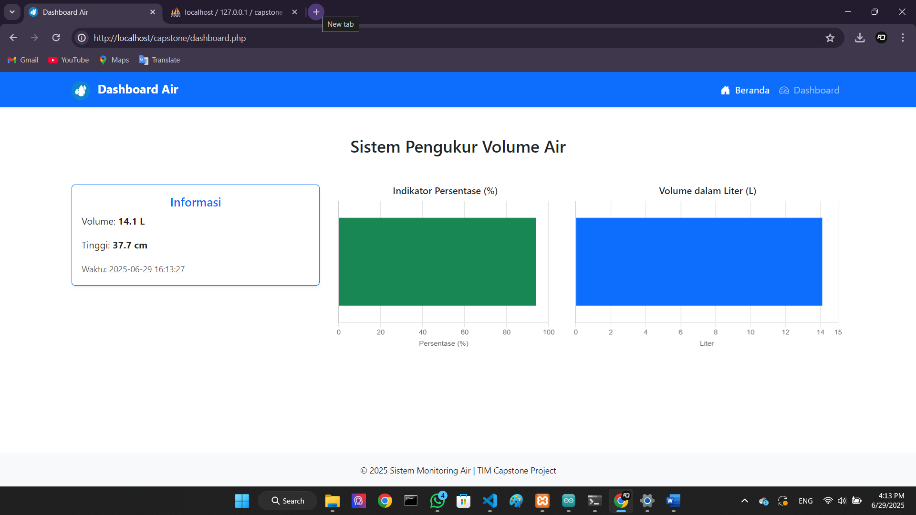
Gambar diatas merupakan tampilan *website* ketika sensor ultrasonik tidak membaca apapun, dan grafik *chart* akan muncul sesuai hasil volume air dalam bentuk *chart* presentase dan *chart* volume air, berikut merupakan tampilan *dashboard* ketika volume air naik, *chart* indikator presentase akan naik ketika air menunjukan 0 – 40 persen bar berwarna merah, ketika persen menunjukkan 40 – 80 persen bar akan berwarna kuning, dan bar akan berwarna hijau ketika persen menunjukan 80 – 100. dan berikut tampilannya.



Gambar 4. , Kondisi air rendah



Gambar 4. , Kondisi air sedang



Gambar 4. , Kondisi air penuh

## Pembahasan

Pengujian dilakukan dengan menggunakan botol air minum berkapasitas 15 liter sebagai media penampungan. Sistem pengukur volume air bekerja dengan mengandalkan sensor ultrasonik HC-SR04 yang dipasang di bagian atas botol untuk mendeteksi jarak antara sensor dan permukaan air. Berdasarkan nilai jarak tersebut, sistem menghitung volume air dengan pendekatan tinggi kolom air dalam botol. Diketahui bahwa tinggi maksimum air dalam botol tersebut adalah 45 cm sampai 50 cm ketika terisi penuh. Oleh karena itu, untuk menghitung volume air berdasarkan tinggi permukaan, digunakan Rumus matematika berikut:

* Rumus matematis untuk mendapatkan ketinggian air dan volume air

Jarak Terkoreksi:

*Jarak Terkoreksi = Jarak Sensor + Nilai Koreksi*

Tinggi Air :

*Tinggi Air = Tinggi botol air – Jarak sensor*

Volume Air

Digunakannya rumus diatas dikarenakan bentuk botol air berukuran 15 liter yang ingin diukur volumenya memiliki bentuk yang tidak rata dengan posisi samping rata dan bagian atasnya cembung.

Dengan pengujan ini, alat berhasil mengukur jarak menjadi estimasi volume air dan mengirimkannya ke *server* melalui koneksi *Wi-Fi*. Data kemudian ditampilkan secara lokal pada LCD 16x2 I2C serta secara *online* melalui *dashboard* *website*. Penggunaan modul RTC DS3231 memastikan data waktu yang dikirim akurat dan sesuai zona waktu lokal (WIB) atau waktu UTC+7.

# BAB V PENUTUP

## Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem pengukur volume air berbasis *Internet of Things (IoT)* menggunakan sensor ultrasonik HC-SR04, dapat disimpulkan bahwa sistem telah bekerja sesuai dengan yang dirancang. Sistem mampu mengukur jarak permukaan air, mengonversi data menjadi volume air berdasarkan tinggi wadah yang telah ditentukan, serta menampilkannya secara *real-time* melalui LCD dan *website*.

Pengujian dilakukan dengan mensimulasikan beberapa kondisi volume air dalam wadah berkapasitas 15 liter. Sistem berhasil mendeteksi perubahan volume pada setiap kondisi tersebut, dan menampilkan data yang cukup akurat serta stabil di antarmuka *website,* dan menampilkan-nya pada layar LCD 16x2, Informasi volume, ketinggian air, dan waktu pengukuran ditampilkan dengan jelas sehingga mempermudah pengukuran oleh pengguna.

## Saran

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem pengukur volume air berbasis *Internet of Things* (IoT) menggunakan sensor ultrasonik HC-SR04, terdapat beberapa hal yang perlu menjadi perhatian dan saran untuk pengembangan selantunya diiantaranya :

* Menggunakan sensor yang lebih tahan terhadap kelembaban dan cipratan air, terutama jika alat digunakan di luar ruangan atau lingkungan terbuka.
* Mengembangkan sistem pengukur volume air ke ranah yang lebih luas.

# DAFTAR PUSTAKA

Aprilia, S., Agustin, R., Pranatawijaya, V. H., & Sari, N. N. K. (2024). Penerapan Api Gemini Dalam Layanan Peminjaman Novel *Online* Pada *Website* Cozybook. *Jurnal Informatika Dan Teknik Elektro Terapan*, *12*(3). Https://Doi.Org/10.23960/Jitet.V12i3.4508

Asep Burhan Subagja, Ema, N. H. (2024). *Perancangan Dan Monitoring Modul Power Gps Tracker Menggunakan Ic Buck Converter*. *10*(5), 1255–1264.

Kamal, K., Tyas, U. M., Buckhari, A. A., & Pattasang, P. (2023). Implementasi Aplikasi Arduino Ide Pada Mata Kuliah Sistem Digital. *Jurnal Pendidikan Dan Teknologi (Teknos)*, *1*(1), 1–10.

Mariko, S. (2019). *Aplikasi Website Berbasis HTML Dan Javascript Untuk Menyelesaikan Fungsi Integral Pada Mata Kuliah Kalkulus Selli*. *6*(1), 80–91.

Nizam, M., Yuana, H., Informasi, F. T., Islam, U., Blitar, B., & Switch, M. D. (2022). *Mikrokontroler Esp 32 Sebagai Alat Monitoring Pintu Berbasis Web*. *6*(2), 767–772.

Nugraha, A. R., Sari, S. K., Pratondo, A., & Ph, D. (2019). *Aplikasi Pengelolaan Umkm Dan Pemasaran Produk Umkm Berbasis Web Studi Kasus : Dinas Koperasi Dan Usaha Kecil Provinsi Jawa Barat Umkm Management Application And Web Based Umkm Product Marketing Case Study : Cooperatives And Small Business Offices West Ja*. *5*(2), 1172–1182.

Safrilly, S. M., & Badarudin, R. (2024). *Sistem Monitoring Suhu Secara Real-Time Berbasis Arduino Uno Untuk Pemantauan Lingkungan*. *3*(2).

Suriana, I. W., Setiawan, I. G. A., & Graha, I. M. S. (2022). Rancang Bangun Sistem Pengaman Kotak Dana Punia Berbasis Mikrokontroler Nodemcu Esp32 Dan Aplikasi Telegram. *Jurnal Ilmiah Telsinas Elektro, Sipil Dan Teknik Informasi*, *4*(2), 75–84. Https://Doi.Org/10.38043/Telsinas.V4i2.3198

Yudha, P. S. F., & Sani, R. A. (2017). Jurnal Einstein Jurnal Hasil Penelitian Bindang Fisika Implementasi Sensor Ultrasonik Hc-Sr04 Sebagai Sensor Parkir Mobil Berbasis Arduino. *Jurnal Einstein*, *5*(3), 19–26. Http://Jurnal.Unimed.Ac.Id/2012/Index.Php/Inpafie-Issn:2407-747x,P-Issn2338-1981

# LAMPIRAN

Lampiran , Coding ESP32 pada Arduino IDE

1. // Library
2. #include <WiFi.h>
3. #include <HTTPClient.h>
4. #include <Wire.h>
5. #include <RTClib.h>
6. #include <LiquidCrystal\_I2C.h>
7. // Konfigurasi AccessPoint
8. const char\* ssid = "SSID";
9. const char\* password = "Password";
10. // IP Static (ESP32) ke AccessPoint
11. IPAddress local\_IP(192, 168, x, xxx);
12. IPAddress gateway(192, 168, x, 1);
13. IPAddress subnet(255, 255, 255, 0);
14. IPAddress primaryDNS(8, 8, 8, 8);
15. IPAddress secondaryDNS(8, 8, 4, 4);
16. // Pin Ultrasonik
17. const int trigPin = 5;
18. const int echoPin = 18;
19. // Inisialisasi Komponen
20. RTC\_DS3231 rtc;
21. LiquidCrystal\_I2C lcd(0x27, 16, 2);
22. // Parameter yang ingin diukur
23. const float tinggi\_max\_cm = 45;        // Tinggi maks ke permukaan
24. const float volume\_max\_liter = 16;    // Volume Maksimal
25. // Server PHP
26. const char\* serverName = "http://192.168.x.xxx/capstone/konfig.php"; // Ganti IP server
27. // Koreksi kalibrasi Sensor (50 titik / 50cm)
28. float koreksi[] = {
29. -0.66, -0.32, 0.03, 0.06, 0.73, 0.43, 0.45, 0.15, -0.21, 0.45,
30. 0.08, 0.05, -0.33, 0.01, -0.04, -0.38, 0.31, 0.35, 0.38, 0.41,
31. 0.45, 0.49, -0.05, -0.22, -0.09, 0.33, 0.16, 0.59, 0.14, 0.59,
32. 0.28, 0.24, 0.71, 0.40, 0.89, 0.07, -0.10, 0.68, 0.29, 0.83,
33. 0.68, 0.29, -0.06, 0.73, 0.12, 0.72, 0.10, 0.72, 0.71, 0.32
34. };
35. // Volume terakhir (untuk pengiriman)
36. float lastVolume = -1;
37. // Fungsi untukkoreksi jarak berdasarkan kalibrasi
38. float koreksiJarak(float jarakSensor) {
39. int index = round(jarakSensor) - 1;
40. if (index >= 0 && index < 50) {
41. return jarakSensor + koreksi[index];
42. } else {
43. return jarakSensor;
44. }
45. }
46. void setup() {
47. Serial.begin(115200);
48. Wire.begin();
49. lcd.begin(16, 2);
50. lcd.backlight();
51. pinMode(trigPin, OUTPUT);
52. pinMode(echoPin, INPUT);
53. // Inisialisasi Modul RTC
54. if (!rtc.begin()) {
55. lcd.print("RTC Gagal");
56. while (1);
57. }
58. // rtc.adjust(DateTime(2025, 6, 26, 2, 4, 0)); // Hanya dijalankan sekali saat set waktu
59. // IP Static
60. if (!WiFi.config(local\_IP, gateway, subnet, primaryDNS, secondaryDNS)) {
61. Serial.println("Gagal atur IP statis");
62. }
63. // Melakukan Koneksi ke AccessPoint
64. WiFi.begin(ssid, password);
65. Serial.print("Menghubungkan ke WiFi");
66. int tries = 0;
67. while (WiFi.status() != WL\_CONNECTED && tries < 20) {
68. delay(500);
69. Serial.print(".");
70. tries++;
71. }
72. if (WiFi.status() == WL\_CONNECTED) {
73. lcd.clear();
74. lcd.print("WiFi Terhubung");
75. Serial.println("\nWiFi Terhubung");
76. Serial.print("IP ESP32: ");
77. Serial.println(WiFi.localIP());
78. } else {
79. lcd.clear();
80. lcd.print("WiFi Gagal");
81. Serial.println("Gagal konek WiFi");
82. }
83. }
84. void loop() {
85. // sensor ultrasonik melakukan pembacaaan
86. digitalWrite(trigPin, LOW);
87. delayMicroseconds(2);
88. digitalWrite(trigPin, HIGH);
89. delayMicroseconds(10);
90. digitalWrite(trigPin, LOW);
91. long duration = pulseIn(echoPin, HIGH);
92. float jarak\_cm = duration \* 0.034 / 2.0;
93. float jarak\_koreksi = koreksiJarak(jarak\_cm);
94. // Rumus Hitung tinggi air dan volume
95. float tinggi\_air\_cm = constrain(tinggi\_max\_cm - jarak\_koreksi, 0, tinggi\_max\_cm);
96. float volume\_liter = (tinggi\_air\_cm / tinggi\_max\_cm) \* volume\_max\_liter;
97. // Ambil waktu dari RTC (WIB)
98. DateTime now = rtc.now() + TimeSpan(0, 7, 0, 0);
99. char waktuBuffer[20];
100. sprintf(waktuBuffer, "%04d-%02d-%02d %02d:%02d:%02d",
101. now.year(), now.month(), now.day(), now.hour(), now.minute(), now.second());
102. // Tampilkan ke LCD
103. lcd.setCursor(0, 0);
104. lcd.printf("Vol: %.1f L    ", volume\_liter);
105. lcd.setCursor(0, 1);
106. lcd.printf("WIB %02d:%02d", now.hour(), now.minute());
107. // Debug Serial
108. Serial.printf("Kirim: %.1f L | %s\n", volume\_liter, waktuBuffer);
109. // Kirim ke server jika perubahan signifikan
110. if (abs(volume\_liter - lastVolume) >= 0.1 && WiFi.status() == WL\_CONNECTED) {
111. HTTPClient http;
112. String url = String(serverName) + "?waktu=" + urlencode(waktuBuffer) +
113. "&volume=" + String(volume\_liter, 1) +
114. "&ketinggian=" + String(tinggi\_air\_cm, 1);
115. http.begin(url);
116. int httpCode = http.GET();
117. Serial.print("Status kirim: ");
118. Serial.println(httpCode);
119. http.end();
120. lastVolume = volume\_liter;
121. }
122. delay(2000);
123. }
124. // Fungsi URL encode untuk HTTP
125. String urlencode(String str) {
126. String encoded = "";
127. char c;
128. char code0, code1;
129. for (int i = 0; i < str.length(); i++) {
130. c = str.charAt(i);
131. if (isalnum(c)) {
132. encoded += c;
133. } else {
134. code0 = (c >> 4) & 0xF;
135. code1 = c & 0xF;
136. encoded += '%';
137. encoded += "0123456789ABCDEF"[code0];
138. encoded += "0123456789ABCDEF"[code1];
139. }
140. }
141. return encoded;
142. }

Lampiran , Coding Konfig.php



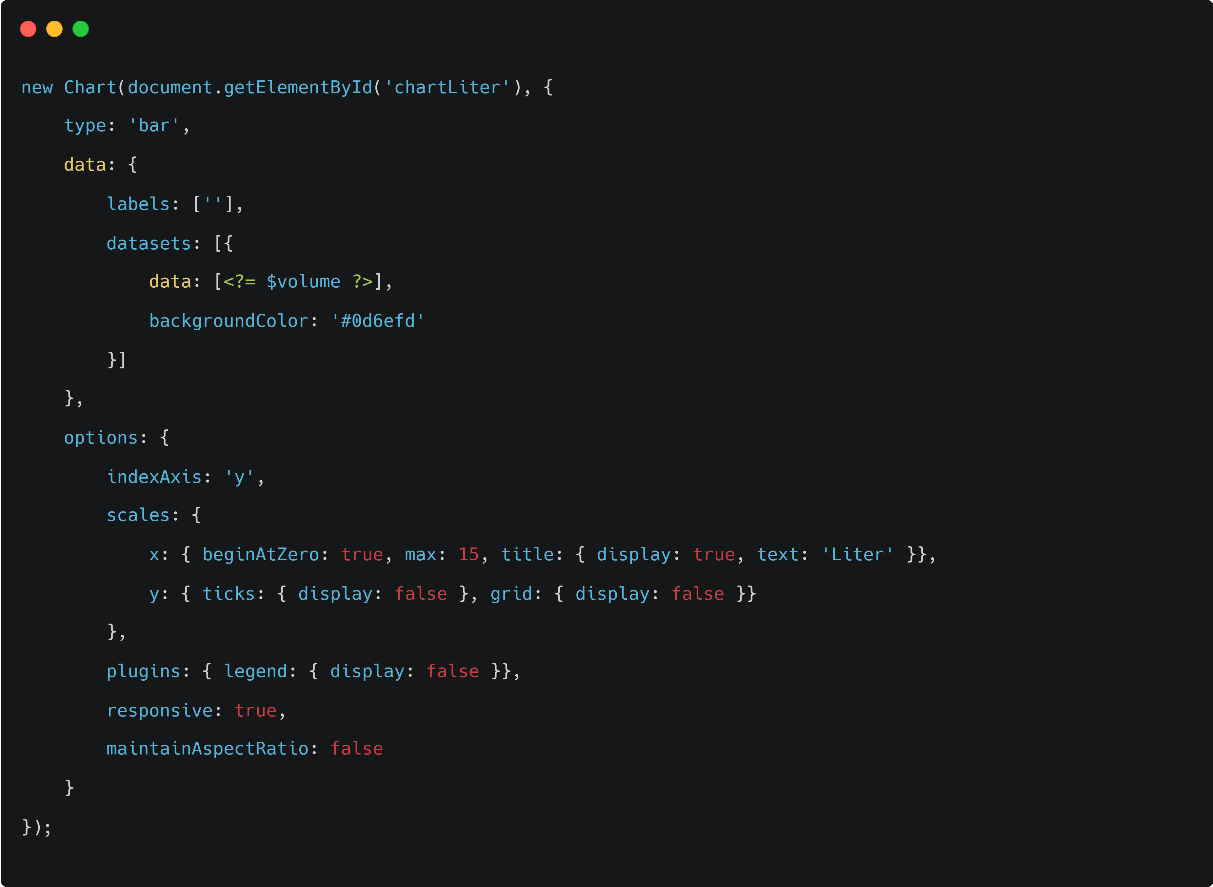
Lampiran , Coding menampilkan informasi



Lampiran , Coding menampilkan chart Presentase



Lampiran , Coding menampilkan chart Volume Air



Lampiran , Dokumentasi pengambilan data



