

RANCANG BANGUN SISTEM MONITORING WATER METER BERBASIS WIRELESS SENSOR NETWORK

by Dimico Mazatama

Submission date: 22-Dec-2023 03:22PM (UTC+0300)

Submission ID: 2264066033

File name: Skripsi_Dimico_Mazatama_3332170047.pdf (3.08M)

Word count: 20898

Character count: 116148

**RANCANG BANGUN SISTEM *MONITORING WATER METER*
BERBASIS *WIRELESS SENSOR NETWORK***

3
SKRIPSI

Disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik (S.T)



Disusun oleh :

DIMICO MAZATAMA

NPM. 3332170047

**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS SULTAN AGENG TIRTAYASA
2023**

3
LEMBAR PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya sebagai penulis Skripsi berikut:

Judul : Rancang Bangun Sistem *Monitoring Water Meter*
Berbasis *Wireless Sensor Network*
Nama Mahasiswa : Dimico Mazatama
NPM : 3332170047
Fakultas / Jurusan : Fakultas Teknik / Jurusan Teknik Elektro

Menyatakan dengan sesungguhnya bahwa skripsi tersebut di atas benar-benar hasil karya saya dan tidak memuat hasil karya orang lain, kecuali dinyatakan melalui rujukan yang benar dan dapat dipertanggungjawabkan. Apabila di kemudian hari ditemukan hal - hal yang menunjukkan bahwa sebagian atau seluruh karya ini bukan karya saya, maka saya bersedia dituntut melalui hukum yang berlaku. Saya juga bersedia menanggung segala akibat hukum yang timbul dari pernyataan yang secara sadar dan sengaja saya nyatakan melalui lembar ini.

Cilegon, 08 September 2022



Dimico Mazatama

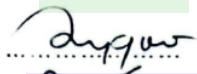


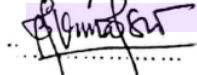
NPM. 3332170047

3 LEMBAR PENGESAHAN

Dengan ini ditetapkan bahwa Skripsi berikut:

Judul : Rancang Bangun Sistem *Monitoring Water Meter*
Berbasis *Wireless Sensor Network*
Nama Mahasiswa : Dimico Mazatama
NPM : 3332170047
Fakultas / Jurusan : Fakultas Teknik / Jurusan Teknik Elektro

41 Telah diuji dan dipertahankan pada tanggal 08 September 2022 melalui Sidang Skripsi di Fakultas Teknik Universitas Sultan Ageng Tirtayasa Cilegon dan dinyatakan LULUS

	Dewan Penguji	Tanda Tangan
Pembimbing I	: Anggoro Suryo Pramudyo, M.Kom.	
Pembimbing II	: Prof. Dr. Alimuddin, S.T, M.M,M.T	
Penguji I	: Dr.Eng. Rocky Alfanz, M.Sc.	
Penguji II	: Dr. Irma Saraswati, S.Si, M.T.	

Mengetahui,

Ketua Jurusan



14/9/23
Dr. Romi Wiryadinata, M.Eng

NIP. 198307032009121006

PRAKATA

Puji dan syukur atas kehadiran Allah Subhanahu Wata'ala berkat limpahan Rahmat dan Karunia-Nya penulis dapat menyelesaikan skripsi yang berjudul Rancang Bangun Sistem *Monitoring Water Meter* Berbasis *Wireless Sensor Network* sebagai salah satu syarat dalam menyelesaikan Program Sarjana (S1) dan memperoleh gelar Sarjana Teknik di Universitas Sultan Ageng Tirtayasa.

Penulis menyadari, penulisan skripsi ini tidak dapat terselesaikan tanpa dukungan dan bantuan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan rasa terima kasih kepada semua pihak yang memberikan bantuan dan dukungan dalam penulisan, penyusunan dan penyelesaian skripsi ini. Adapun penulis sampaikan banyak terima kasih yang sebesar - besarnya kepada :

1. Kedua orang tua, adik dan kerabat yang selalu memberikan doa maupun dukungan sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Dr. Romi Wiryadinata, M.Eng. selaku Ketua Jurusan Teknik Elektro Universitas Sultan Ageng Tirtayasa.
3. Bapak Anggoro Suryo Pramudyo, M.Kom. selaku Dosen Pembimbing 1 yang telah membimbing dalam penyusunan skripsi.
4. Bapak Prof. Dr. Alimuddin, S.T, M.M, M.T selaku Dosen Pembimbing 2 yang telah membimbing dalam penyusunan skripsi.
5. Bapak dan Ibu dosen yang telah memberikan pengetahuan dan ilmu yang sangat bermanfaat dalam perkuliahan.
6. Teman-teman Angkatan 2017 Jurusan Teknik Elektro Universitas Sultan Ageng Tirtayasa yang telah memberikan semangat dan memberikan dukungan moral.

Penulis menyadari bahwa penelitian ini masih jauh dari kata sempurna dan masih banyak kekurangan. Penulis berharap semoga penelitian ini akan memberi manfaat bagi pembaca dan penelitian di masa depan. Penulis memohon maaf sebesar - besarnya apabila terjadi kesalahan dalam penulisan skripsi ini. Penulis menerima kritik dan saran dari para pembaca untuk kemajuan penelitian ini.

Cilegon, 08 September 2022

Penulis

ABSTRAK

Dimico Mazatama
Teknik Elektro

Rancang Bangun Sistem *Monitoring Water Meter* Berbasis *Wireless Sensor Network*

Air bersih merupakan kebutuhan pokok yang diperlukan masyarakat dan memiliki peranan penting dalam meningkatkan kesehatan lingkungan. Masalah umum yang timbul dari meteran analog adalah sulitnya pembacaan meteran air. Masalah ini dapat diatasi menggunakan teknologi sensor dan *wireless sensor network*. Penelitian ini bertujuan mengimplementasikan sistem *monitoring* penggunaan air yang dapat dilakukan secara nirkabel. Sistem *Monitoring Water Meter* memiliki tingkat akurasi pembacaan yang tinggi yaitu 0,56% dan dapat membaca air dengan kecepatan relatif lambat yaitu 0,17 Liter/menit. Dengan memanfaatkan teknologi WSN, *monitoring* dapat dilakukan dalam jangkauan luas dengan jarak maksimum antar *node* adalah 30 meter. Hasil pembacaan air dapat dilihat melalui *website*.

Kata Kunci: Sistem *Monitoring*, *Water Meter*, WSN, *Flow sensor*

ABSTRACT

Dimico Mazatama
Electrical Engineering

Design of Water Meter Monitoring System Based On Wireless Sensor Network

Clean water is a basic need needed by the community and has an important role in improving environmental health. A common problem with analog meters is the difficulty of reading the water meter. This problem can be overcome using sensor technology and wireless sensor networks. This study aims to implement a water use Monitoring system that can be done wirelessly. The Water Meter Monitoring system has a high reading accuracy rate of 0.56% and can read water at a relatively slow speed of 0.17 Liters/minute. By utilizing WSN technology, Monitoring can be carried out in a wide range with a maximum distance between nodes of 30 meters. The results of water readings can be seen through the website.

Keyword: Monitoring System, Water Meter, WSN, Flow sensor

3 DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PERNYATAAN KEASLIAN SKRIPSI.....	ii
LEMBAR PENGESAHAN.....	iii
PRAKATA.....	iv
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	viii
DAFTAR TABEL.....	ix
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	4
1.3 Tujuan Penelitian.....	4
1.4 Manfaat Penelitian.....	4
1.5 Batasan Masalah.....	4
1.6 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA	
2.1 Sistem <i>Monitoring</i>	6
2.2 <i>Wireless Sensor Network</i>	6
2.2.1 Topologi <i>Star</i>	7
2.2.2 Topologi <i>Tree</i>	8
2.2.3 Topologi <i>Mesh</i>	9
2.3 NodeMCU.....	10
2.4 <i>Water Meter</i>	11
2.5 <i>World Wide Web</i>	13
2.5.1 Fungsi <i>Web</i>	14
2.5.2 Pengertian <i>Web</i>	14
2.5.3 Unsur-unsur <i>Web</i>	15
2.6 <i>Database Management System</i>	16
2.7 <i>Software XAMPP</i>	17

2.8 <i>Flow Sensor</i>	18
2.9 LCD 16x2 I2C	18
2.10 Kajian Pustaka	19
BAB III METODOLOGI PENELITIAN	
3.1 Perancangan Perangkat Keras	21
3.2 Perancangan Perangkat Lunak	24
3.2.1 Perancangan <i>Database</i>	26
3.2.2 Perancangan <i>Website</i>	27
3.2.3 Perancangan Topologi <i>Wireless Sensor Network</i>	30
BAB IV HASIL DAN ANALISIS	
4.1 Hasil Perancangan <i>Hardware</i>	31
4.2 Hasil Perancangan <i>Software</i>	31
4.2.1 Hasil Perancangan <i>Database</i>	31
4.2.2 Tampilan <i>Dashboard Web</i>	32
4.2.3 Tampilan Halaman <i>Detail</i>	33
4.2.4 Tampilan Halaman <i>About Us</i>	34
4.2.5 Tampilan Halaman <i>Contact</i>	34
4.3 Pengujian <i>Hardware</i>	35
4.4 Pengujian <i>Software</i>	37
4.5 Pengujian Jaringan	38
4.5.1 Pengujian Jarak <i>Node</i>	38
4.5.2 Pengujian Topologi <i>Mesh</i>	39
4.6 Pengujian Keseluruhan	41
BAB V PENUTUP	
5.1 Kesimpulan	50
5.2 Saran	50
DAFTAR PUSTAKA	
LAMPIRAN A DOKUMENTASI PENGUJIAN	
LAMPIRAN B HASIL PENGUJIAN	
LAMPIRAN C LISTING PROGRAM	

DAFTAR GAMBAR

Gambar 2.1 <i>Wireless Sensor Network</i>	7
Gambar 2.2 Topologi <i>Star</i>	8
Gambar 2.3 Topologi <i>Tree</i>	9
Gambar 2.4 Topologi <i>Mesh</i>	9
Gambar 2.5 NodeMCU	10
Gambar 2.6 <i>Software XAMPP</i>	12
Gambar 2.7 LCD 16x2.....	18
Gambar 3.1 <i>Flowchart</i> Perangkat Keras <i>Water Meter</i>	21
Gambar 3.2 Diagram Blok Sistem	22
Gambar 3.3 Rangkaian Listrik Sistem	23
Gambar 3.4 Rancangan Model 3D Alat	23
Gambar 3.5 <i>Flowchart Sistem</i>	24
Gambar 3.6 <i>Use Case Diagram</i>	25
Gambar 3.7 Rancangan <i>Database Monitoring Smart Water Meter</i>	26
Gambar 3.8 Rancangan Tampilan <i>Landing Page</i>	27
Gambar 3.9 Rancangan Tampilan <i>Detail User</i>	28
Gambar 3.10 Rancangan Tampilan <i>About Us</i>	29
Gambar 3.11 Rancangan Tampilan <i>Contact</i>	29
Gambar 3.12 Topologi <i>Mesh</i>	30
Gambar 4.1 Tampilan Dalam <i>Hardware</i>	31
Gambar 4.2 Tampilan <i>Database Sistem Monitoring</i>	32
Gambar 4.3 Tampilan Halaman <i>Dashboard Web</i>	32
Gambar 4.4 Tampilan Halaman <i>Detail Node</i>	33
Gambar 4.5 Tampilan Halaman <i>About Us</i>	34
Gambar 4.6 Tampilan Halaman <i>Contact</i>	35
Gambar 4.7 <i>Access Point Node 1</i>	37
Gambar 4.8 Kondisi Node Terhubung	38
Gambar 4.9 Kondisi 1 <i>Node 2 Off</i>	40
Gambar 4.10 Kondisi 2 <i>Node 5 Off</i>	41
Gambar 4.11 Topologi <i>Mesh</i> Pengujian Keseluruhan	42

Gambar 4.12 Grafik <i>Monitoring</i> Penggunaan Air <i>Node</i> 2	42
Gambar 4.13 Grafik <i>Monitoring</i> Penggunaan Air <i>Node</i> 3	44
Gambar 4.14 Grafik <i>Monitoring</i> Penggunaan Air <i>Node</i> 4	45
Gambar 4.15 Grafik <i>Monitoring</i> Penggunaan Air <i>Node</i> 5	46
Gambar 4.16 Grafik Interval Pengiriman Data	49

DAFTAR TABEL

Tabel 2.1 Spesifikasi <i>Pin</i> LCD 16x2.....	19
Tabel 4.1 Hasil Kalibrasi <i>Flow</i> Sensor	35
Tabel 4.2 Hasil Kalibrasi Meteran Analog	36
Tabel 4.3 Pengujian Pengiriman Data Ke <i>Website</i>	37
Tabel 4.4 Pengujian Jarak <i>Node</i> Sensor	39
Tabel 4.5 Hasil Pengujian <i>Mesh</i> Kondisi 1 <i>Node</i> 2 <i>Off</i>	40
Tabel 4.6 Hasil Pengujian <i>Mesh</i> Kondisi 2 <i>Node</i> 5 <i>Off</i>	41
Tabel 4.7 Tabel <i>Monitoring</i> Pembacaan Air Selama 1 Minggu	47
Tabel 4.8 Interval Pengiriman <i>Data Node</i> Menuju <i>Database</i>	48

BAB I

LATAR BELAKANG

1.1 Latar Belakang

Penyediaan air bersih merupakan peranan penting dalam meningkatkan kesehatan lingkungan dan masyarakat. Salah satu cara strategis untuk memenuhi kebutuhan dasar dan meningkatkan kualitas hidup masyarakat adalah ketersediaan air bersih yang cukup untuk berbagai macam aktifitas. Berdasarkan hal itu maka manajemen distribusi air bersih yang mumpuni dibutuhkan agar terdapat keseimbangan antara penyediaan dan kebutuhan air bagi masyarakat [1]. Berdasarkan penelitian yang sudah kelangkaan air dipengaruhi oleh jumlah air yang tersedia dan kualitas dari air yang tersedia [2]. Jumlah air yang sedikit jelas akan menimbulkan kelangkaan air, namun air yang terpolusi akan mempengaruhi siklus air sehingga akan mengganggu kualitas air yang diperoleh. Polusi namun dapat dikurangi dengan pengelolaan limbah air yang baik.

Indonesia tercatat mewakili sekitar 6% dari sumber daya air yang ada di dunia. Data ini menunjukkan bahwa sebenarnya sumber daya air di Indonesia cukup melimpah. Tetapi kenyataan menunjukkan bahwa pasokan air bersih bagi sebagian besar daerah di Indonesia justru mengalami kekurangan [3]. Ketersediaan air bersih juga memiliki pengaruh terhadap tingkat ekonomi masyarakat. Kenaikan 1% akses rumah tangga pada air bersih akan menurunkan indeks *gap* kemiskinan sebesar 0,022, artinya semakin banyak kuantitas air bersih yang dapat diakses oleh rumah tangga menjadikan air bersih mudah didapat sehingga harga dari air bersih akan menjadi semakin murah serta dapat mengurangi biaya hidup dan secara otomatis meningkatkan kesejahteraan. Kesimpulannya semakin tinggi akses air bersih maka akan menurunkan indeks *gap* kemiskinan secara signifikan [4].

Pengelolaan air bersih di daerah perkotaan pada umumnya dikelola oleh PDAM (Perusahaan Daerah Air Minum). Air akan didistribusikan ke setiap tempat yang membutuhkan air bersih dengan menggunakan saluran distribusi. Konsumen yang mendapatkan *supply* air bersih akan dipasang meteran air sebagai alat ukur

penggunaan air bersih dalam waktu tertentu. Pengelola dapat memberikan tarif air bersih sesuai dengan penggunaan tiap konsumen berdasarkan meteran airnya. Pengelola yang ahli sangat dibutuhkan dalam proses distribusi air karena kinerja dari pengelola air akan mempengaruhi kualitas dan kuantitas air yang akan disalurkan ke masyarakat. Fasilitas dalam pengelolaan air juga menjadi faktor tambahan yang dapat mempengaruhi kualitas distribusi air bersih. Contoh pengelolaan distribusi air yang baik dilakukan untuk mengaliri irigasi air ke sawah yang dilakukan oleh tenaga ahli berpengalaman di bidang distribusi air dan irigasi serta fasilitas *hydrotechnical* yang terawat dapat meningkatkan panen sebesar 15% serta penghematan penggunaan air sebanyak 22% [5]. Kualitas mutu air dengan menggunakan metode STORET yaitu dengan cara mengumpulkan data kualitas air dan menentukan nilai maksimum, minimum, dan rata-rata dari setiap parameter dan membandingkan hasil pengukuran setiap parameter dengan nilai baku mutu. Hasil pengukuran yang memenuhi nilai baku mutu maka akan diberi nilai 0 dan jika hasil pengukuran tidak memenuhi nilai baku mutu maka akan diberikan nilai sesuai dengan klasifikasi yang sudah ditentukan [6].

Meteran air yang digunakan umumnya masih menggunakan meteran air analog. Penggunaan meteran air analog memiliki beberapa masalah. Masalah yang timbul mulai dari akurasi pembacaan meteran air, konsumen yang tidak dapat membaca meteran air sehingga dapat menyebabkan tagihan air meningkat, dan banyak hal lain yang dapat menimbulkan kesalahan pencatatan dan penunggakan pembayaran. Pencatatan dalam meteran konvensional dilakukan dengan cara menghitung selisih dari jumlah pemakaian bulan saat ini dengan bulan sebelumnya [7]. Kesalahan pembacaan pada meteran analog dalam penelitian yang pernah dilakukan menjelaskan bahwa dari 158 meteran yang diuji terdapat 35 meteran air yang tidak dikalibrasi dengan benar, 21 meteran air memiliki kebocoran air *internal* akibat pipa *internal* yang rusak, sehingga hal ini dapat menimbulkan deviasi pembacaan air tiap bulan yang mengalami potensi kehilangan air sebesar 1,56% sampai dengan 46,73% atau setara dengan 8312,92 liter air per hari [8].

Masalah pada meteran air analog dapat diatasi menggunakan teknologi saat ini. Penelitian ini bertujuan untuk memodernisasikan alat dan sistem *monitoring* penggunaan air yang dapat dilakukan secara nirkabel sehingga petugas tidak perlu

datang kelapangan setiap bulan untuk mengecek penggunaan air setiap konsumen secara satu per satu. *Water meter* yang menggunakan teknologi *mechanical water meter* memiliki kelemahan yaitu pembacaan laju aliran air yang rendah sehingga membatasi jangkauan pengukuran, mengurangi ketelitian pembacaan pengukuran, dan kekurangan pada pembacaan data *real time*, sedangkan *water meter* yang menggunakan *flow sensor* dalam pembacaan airnya hanya memiliki kelemahan pada kebutuhan pengamanan lebih pada komponen elektronik yang digunakan [9]. Menggunakan teknologi *Wireless Sensor Network* (WSN) data penggunaan air akan dikirimkan ke *database* sehingga baik petugas atau konsumen dapat melihat penggunaan air dengan mudah. WSN yang optimal harus memenuhi beberapa parameter yaitu akurasi, biaya, daya, kerapatan antar *node*, dan tipe *node* [10]. Cara kerja dari WSN sendiri adalah dengan pemberian identitas unik pada tiap *node* dalam jaringan sehingga tiap *node* dapat berkomunikasi tergantung pada topologi yang digunakan. Urutan langkah kerja dari WSN adalah inisialisasi *node sensor*, pembacaan data yang dilakukan oleh *node sensor*, kemudian data akan diolah dan akan ditampilkan pada indikator atau dikirim ke *webservice* [11].

Beberapa penelitian terdahulu telah menyajikan perancangan *water meter* dalam bentuk digital dan sudah terhubung dengan jaringan *wireless*. Penelitian yang sudah dilakukan selanjutnya adalah menerapkan *prototype* sistem *monitoring* air dan energi listrik menggunakan IoT pada bangunan tempat tinggal menggunakan aplikasi android [12]. Penelitian selanjutnya adalah penerapan *water meter* menggunakan jaringan 4G pada *smart city*. Jaringan 4G digunakan sebagai media komunikasi *node* dengan *server* pada penelitian ini [13]. Penelitian selanjutnya melakukan perancangan dan mengimplementasikan jaringan WSN pada *water meter* menggunakan radio *wireless* nRF24101 dengan menerapkan topologi *mesh* pada jaringan WSN [14]. Penelitian lainnya menerapkan WSN pada *smart agriculture*. *Node sensor* terhubung pada modul ZigBee dan menggunakan mikrokontroler Arduino Nano [15]. Penelitian yang dilakukan sebelumnya masih memiliki beberapa kekurangan seperti masih menggunakan meteran analog dalam pembacaan air dan belum menggunakan *flow sensor*, serta tidak disediakan *website* untuk mempermudah *user* melihat penggunaan air. Berdasarkan hal itu, maka penelitian ini sistem *monitoring* penggunaan air menggunakan *flow sensor* agar

pembacaan air lebih akurat dan membuat *website* untuk mempermudah *user* melihat penggunaan air sehingga sistem monitoring *water meter* dapat lebih mudah digunakan dan efisien.

34 1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana cara merancang sistem *monitoring water meter* berbasis *wireless sensor network* ?
2. Bagaimana cara mengimplementasikan rancangan sistem dalam kehidupan sehari-hari ?

98 1.3 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut:

1. Merancang sistem *monitoring water meter* berbasis *wireless sensor network*.
2. Mengimplementasikan perancangan sistem *monitoring smart water meter* dalam kehidupan sehari-hari.

102 1.4 Manfaat Penelitian

Manfaat penelitian ini adalah sebagai berikut:

1. Dapat mengatasi masalah pencatatan penggunaan air yang masih menggunakan meteran konvensional.
2. Dapat memudahkan petugas dan konsumen dalam pembacaan penggunaan air
3. Dapat menyadarkan masyarakat dalam menghemat penggunaan air.

8 1.5 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut:

1. Tidak menganalisis lebih lanjut tentang keamanan jaringan pada sistem yang dirancang.
2. Sistem tidak memonitoring kualitas air yang digunakan.
3. Tidak membahas lebih lanjut model mobile dari *software*.

¹⁷ **1.6 Sistematika Penulisan**

Sistematika penulisan skripsi pada penelitian ini adalah sebagai berikut.

BAB I PENDAHULUAN

Bab ini menjelaskan tentang latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan

BAB II TINJAUAN PUSTAKA

Bab ini menjelaskan teori-teori dasar pendukung pada penelitian.

BAB III METODE PENELITIAN

²²
Bab ini menjelaskan rangkaian alur penelitian serta implementasi rancangan sistem dalam bentuk perangkat keras dan perangkat lunak.

BAB IV ANALISIS

Bab ini menjelaskan hasil dari ¹⁶ perancangan dan ¹⁰⁴ pengujian dari perangkat keras dan perangkat lunak berdasarkan metode yang sudah ¹⁰⁴ dirancang.

BAB V PENUTUP

Bab ini menjelaskan kesimpulan dan saran dari keseluruhan pengujian perangkat keras dan perangkat lunak.

TINJAUAN PUSTAKA

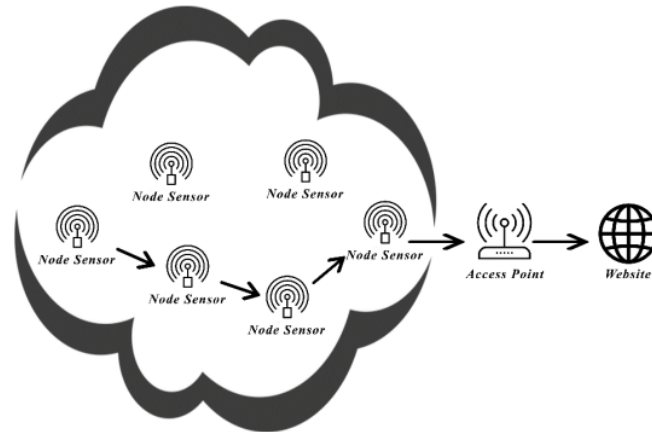
2.1 Sistem Monitoring

Sistem adalah kumpulan komponen dan element yang terintegrasi dan terorganisir dalam mewujudkan suatu tujuan tertentu [16]. Suatu model sistem sederhana dapat dikategorikan menjadi 3 bagian yaitu *input*, proses, dan *output*. *Monitoring* adalah langkah untuk mengkaji kegiatan yang telah dilaksanakan berdasarkan rencana yang telah dibuat, mengidentifikasi masalah yang timbul agar dapat langsung diatasi, dan melakukan penilaian pola kerja untuk mengetahui proses kegiatan berdasarkan tujuan yang sudah dirancang agar kegiatan dapat efektif, efisien, dan tepat sasaran [17]. Proses dari sistem *Monitoring* memiliki beberapa tujuan yaitu:

1. Membandingkan hasil yang diperoleh dengan hasil yang diharapkan
2. Identifikasi masalah sehingga dapat diperoleh solusi secara langsung
3. Evaluasi pola kerja terhadap tujuan sistem
4. Mendapatkan *feedback* kerja sistem
5. Adaptasi sistem terhadap lingkungannya

2.2 Wireless Sensor Network

Wireless sensor network adalah kelompok *node* sensor yang saling terhubung dan berada di lokasi yang berbeda bertujuan untuk melakukan pengukuran terhadap suatu besaran fisis dan kemudian mengandalakan *node* pusat atau yang disebut *node gateway* sebagai pengumpul informasi dari hasil pengukuran keseluruhan *node* [18]. Rangkaian komunikasi itu bertujuan untuk menyampaikan data yang berasal dari *node* sensor yang berada di posisi paling jauh sehingga dapat diterima oleh *user*. *Node* sensor saling dapat saling terhubung secara langsung tanpa perlu menggunakan *router* atau *access point*. *Wireless Sensor Network* juga memiliki *administrator* yang dapat melakukan instrumentasi, pengamatan, dan interaksi terhadap suatu fenomena di lingkungan tertentu [19]. Cara kerja dari *wireless sensor network* dapat dilihat pada Gambar 2.1.



Gambar 2.1 *Wireless Sensor Network*

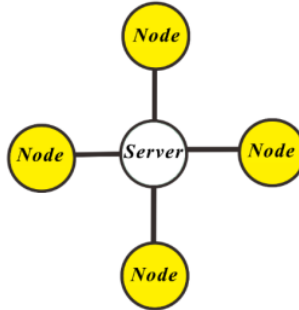
Gambar 2.1 terdapat sebuah jalur komunikasi antar *node sensor*. *Node sensor* biasanya memiliki sumber energi berupa baterai. Efisiensi dalam transmisi data pada WSN dapat dilakukan dengan mencapai 3 bagian berikut:

1. Operasi *low-cost*
2. Proses *local* bertujuan untuk mengurangi waktu transmisi data
3. Jaringan *multihop* untuk mengurangi kebutuhan transmisi jarak jauh

Terdapat tiga aspek penting yang harus dipenuhi dalam *wireless sensor network* yaitu *data processing*, *Media Access Control (MAC)*, dan *communication protocol* [20]. Bentuk topologi jaringan yang umumnya digunakan pada WSN adalah topologi *star*, topologi *tree*, dan topologi *mesh*.

2.2.1 Topologi Star

Topologi *star* merupakan topologi yang paling sederhana dibandingkan dengan topologi *tree* dan topologi *mesh*. Karakteristik dari topologi ini adalah *node* akan langsung terhubung dengan *server*. Contoh dari bentuk jaringan topologi *Star* dapat dilihat pada Gambar 2.2.



Gambar 2.2 Topologi *Star*

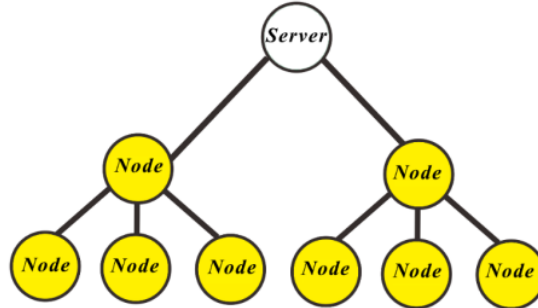
Gambar 2.2 merupakan topologi *star*. Komunikasi yang terjalin terbatas hanya antara *server* dan *node*, sedangkan tidak ada komunikasi antara *node* dengan *node*. Kelebihan pada topologi ini adalah arsitektur topologi yang sederhana dengan bentuk menghubungkan beberapa *client* secara langsung ke *server*. Kekurangan yang dimiliki topologi ini yaitu tingkat *rate of failure* pengiriman data yang tinggi [19].

2.2.2 Topologi *Tree*

Topologi *tree* banyak digunakan karena memiliki kelebihan dalam pengendalian serta manajemen data. Topologi *tree* lebih terstruktur dibandingkan dengan topologi yang lainnya. Karakteristik yang dimiliki oleh topologi *tree* adalah sebagai berikut:

1. Terdapat kabel utama atau yang dapat disebut *backbone*, yang digunakan sebagai penghubung jaringan.
2. Memiliki tingkatan jaringan.
3. Terhubung dengan *hub* yang digunakan sebagai pusat data dan kendali jaringan.
4. Komunikasi data harus melalui *hub*

Contoh bentuk jaringan topologi *tree* dapat dilihat pada Gambar 2.3.

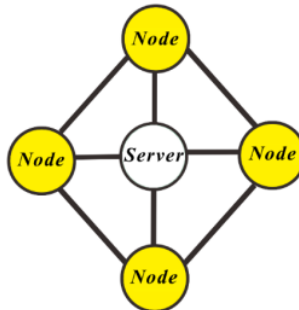


Gambar 2.3 Topologi *Tree*

Gambar 2.3 merupakan bentuk dari arsitektur topologi *tree*. Karakteristik topologi *tree* menyerupai topologi *star* dan *bus*. Penggunaan topologi *tree* dapat memungkinkan komunikasi data yang mencakup jangkauan jaringan yang luas. Arsitektur topologi *tree* merupakan gabungan dari beberapa topologi *star* yang terhubung dengan suatu *bus*.

2.2.3 Topologi *Mesh*

Ciri yang dapat dengan mudah dikenali dari topologi *mesh* adalah saling terhubungnya *node* dengan *node* yang lain. *Node* yang saling terkoneksi secara langsung atau *dedicated link*, proses pengiriman data pada topologi ini menjadi lebih cepat. Arsitektur jaringan topologi *mesh* dapat dilihat pada Gambar 2.4.



Gambar 2.4 Topologi *Mesh*

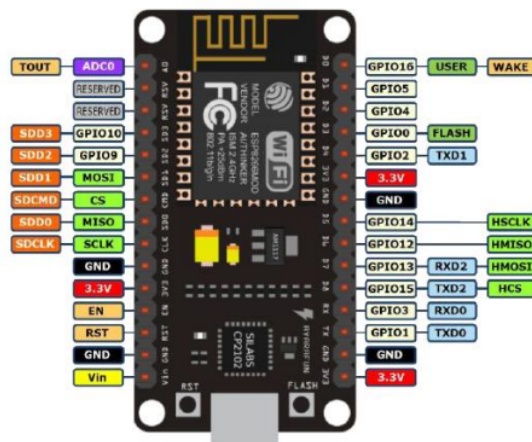
Gambar 2.4 adalah contoh dari bentuk topologi *mesh*. *Node* terhubung dengan *server* dan juga terhubung dengan *node client* yang lain. *Node* yang saling terhubung ini menyebabkan setiap *node* dapat berkomunikasi tidak hanya ke *server*

namun juga dapat berkomunikasi dengan *node* lainnya. Penggunaan topologi ini lebih terpaku terhadap kecepatan pengiriman data, karena setiap *node* terhubung satu dengan yang lain sehingga menyebabkan jangkauan jaringan pada topologi ini tidak terlalu besar. Keuntungan yang diperoleh dari penggunaan topologi ini adalah:

1. Mendeteksi kesalahan
2. Dapat ditingkatkan sesuai kebutuhan
3. Pengiriman data lebih cepat

2.3 NodeMCU

NodeMCU merupakan sebuah *platform* pengembangan *software* dan *hardware* yang bersifat *open source* dan terdiri dari perangkat *System-on-a-Chip* (SoC) yang disebut ESP8266 [20]. Bahasa pemrograman yang digunakan pada NodeMCU adalah bahasa LUA dan bahasa pemrograman C sehingga dapat diprogram menggunakan Arduino IDE. Bentuk dan penjelasan fungsi dari masing-masing *pin board* NodeMCU dapat dilihat pada Gambar 2.5.



Gambar 2.5 NodeMCU [21]

Gambar 2.5 merupakan bentuk dari NodeMCU serta penjelasan fungsi pada setiap *pin* yang terdapat pada NodeMCU. Pemrograman NodeMCU dapat dilakukan dengan mudah sebagaimana menggunakan Arduino. Perbedaan utamanya adalah pada bagian penomor *pin* dari *boardnya*. Terdapat 6 ekstra GPIO *pin*. Semuanya dapat diprogram sebagai *Pulse Width Modulator*, I2C, 1-

Wire, kecuali GPIO16. NodeMCU sendiri menggunakan *file system SPIFFS* untuk menyimpan *file* di *chip flash memory*. NodeMCU memiliki beberapa kelebihan yaitu:

1. Berbiaya rendah
2. Terintegrasi untuk jaringan WiFi
3. Ukuran *board* kecil
4. Konsumsi daya rendah

Sedangkan kelemahan yang dimiliki oleh NodeMCU adalah sebagai berikut:

1. Perlu mempelajari bahasa pemrograman baru dan IDE baru
2. *Pinout* lebih sedikit
3. Dokumentasi yang masih langka, karena penggunaan NodeMCU masih tergolong baru

2.4 *Water Meter*

Water meter atau meteran air merupakan sebuah alat yang digunakan untuk mengukur jumlah aliran air secara konstan melalui sistem kerja yang dilengkapi dengan unit tambahan seperti sensor, unit penghitung, dan indikator penunjuk laju air. Perhitungan pada *water meter* sehingga diperoleh biaya yang harus dikeluarkan untuk pemakaian air menggunakan Persamaan (2.1).

$$\text{Biaya} = \text{Total pembacaan air} \times \text{Golongan} \quad (2.1)$$

Persamaan (2.1) merupakan persamaan untuk mencari total biaya penggunaan air. Biaya adalah total biaya yang harus dibayarkan untuk penggunaan air yang sudah dipakai dan total pembacaan air adalah total air terbaca yang terdapat pada *water meter* dan golongan adalah pembagian kelompok berdasarkan kriteria pengguna.

Golongan dibagi menjadi beberapa bagian dengan biaya yang berbeda setiap daerahnya. Biaya pada daerah Kabupaten Tangerang kelompok pelanggan rumah tangga untuk R1 = Rp. 1875,00 R2 = Rp.2300,00 R3 = Rp.2800,00 R4 = Rp.3350,00 dan R5 = Rp.3900,00. Harga tersebut dapat berubah berdasarkan nilai perolehan air yang dapat dihitung dari harga pokok air dikali dengan faktor nilai air [22]. *Water meter* pada dasarnya dikelompokkan menjadi 5 jenis berdasarkan penggunaannya, jenis-jenis *water meter* yaitu:

1. *Displacement meter*

Jenis meteran air ini juga disebut dengan *volumetric meter*. Penggunaan *displacement meter* diperuntukan untuk aliran air yang kecil sampai sedang. Prinsip kerja jenis meteran air ini adalah dengan melewati air bagian per bagian, setelah memenuhi suatu bagian penampang dalam meter yang diketahui volumenya. Banyaknya aliran diketahui dengan mencatat beberapa kali kontainer tersebut penuh dan kosong. Jenis meter air yang termasuk kedalam kelompok ini adalah jenis *nutating-disk meter*, *rotary* dan *reciprocating*.

2. *Velocity meter*

Velocity meter adalah meteran air yang memanfaatkan kecepatan aliran dengan melewati air melalui suatu penampang yang luasnya sudah diketahui. Jenis meteran air ini biasanya digunakan dalam pengukuran aliran dengan jumlah besar. Jenis meter yang termasuk kelompok ini adalah jenis meter turbin *propeller*, *venturi*, *orifice*, dan *vane-wheel meters*.

3. *Differential pressure flow meter*

Differential pressure flow meter mengukur perbedaan tekanan yang melintasi lubang di mana aliran terkait langsung dengan akar kuadrat dari perbedaan tekanan yang dihasilkan. Terdapat elemen primer dan sekunder dalam pengukur aliran diferensial. Elemen utama menghasilkan perubahan energi kinetik baik menggunakan *flow nozzle*, *pitot tube*, *orifice plate*, atau pengukur aliran venturi. Elemen sekunder mengukur tekanan diferensial dan memberikan sinyal. Pengukur tekanan diferensial mewakili sekitar 1 per 5 dari semua pengukur aliran di seluruh dunia. Mereka biasanya digunakan dalam industri minyak dan gas, bersama dengan minuman, air, farmasi, pertambangan, kertas dan aplikasi kimia.

4. *Mass flow meter*

Mass flow meter lebih efektif dalam proses pengukuran terkait massa karena aliran massa mengukur gaya yang dihasilkan dari percepatan massa. Fluida yang diukur sebagai massa yang bergerak per satuan waktu, bukan volume per satuan waktu. Jenis meteran air yang juga termasuk kedalam *mass flow meter* adalah *Coriolis mass meter* dan pengukur dispersi termal. Penerapan meteran air ini terdapat pada proses pengolahan bahan kimia. Industri kimia, gas, farmasi, listrik,

pertambahan, dan air limbah adalah contoh industri yang menerapkan *mass flow meter*.

5. *Open channel flow meter*

Pengukuran cairan pada saluran terbuka meliputi *v-notch*, bendung dan *flume*. Struktur seperti bendungan ini, atau luapan, memungkinkan aliran bebas cairan terbatas atau terkonsentrasi berdasarkan bentuk dan ukuran unik dari struktur tersebut. Jenis pengukur aliran ini memungkinkan pembacaan debit yang akan dihitung. Aplikasi umum pengukur saluran terbuka mencakup cairan yang mengalir bebas seperti aliran, sungai, saluran irigasi, dan sistem saluran pembuangan air.

2.5 *World Wide Web*

World Wide Web (WWW) atau lebih dikenal dengan sebutan *web* merupakan dokumen yang ditulis dalam format *Hyper Text Markup Language* (HTML), yang hampir selalu bisa diakses melalui http, yaitu protokol yang menyampaikan informasi dari *server* untuk ditampilkan kepada para pemakai melalui sebuah perangkat lunak yang disebut *web browser*. Semua publikasi dari *web* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar yang dapat diakses dengan menggunakan internet.

2.5.1 *Fungsi Web*

Secara garis besar sebuah *web* dapat memiliki fungsi sebagai berikut [23]:

1. *Fungsi komunikasi*

Situs *web* yang mempunyai fungsi komunikasi pada umumnya adalah situs *web* dinamis. Pemrograman *web* dilengkapi fasilitas yang memberikan fungsi-fungsi komunikasi, seperti *web mail*, *form contact*, dan *chatting form*.

2. *Fungsi informasi*

Situs *web* yang memiliki fungsi informasi pada umumnya lebih menekankan pada kualitas bagian kontennya, karena tujuan situs tersebut adalah menyampaikan isisnya. Situs ini sebaiknya berisi teks dan grafik yang dapat di download dengan cepat. Pembatasan penggunaan animasi gambar dan elemen bergerak seperti *shockwave* dan *java* diyakini sebagai langkah yang tepat, diganti

dengan fasilitas yang memberikan fungsi informasi seperti *news*, *profile company*, *library*, dan *reference*.

3. Fungsi *entertainment*

Situs *web* juga dapat memiliki fungsi *entertainment* atau hiburan. Situs *web* yang berfungsi sebagai sarana hiburan maka penggunaan animasi gambar dan elemen bergerak dapat meningkatkan mutu presentasi desainnya, meski tetap harus mempertimbangkan kecepatan *download*nya. Fasilitas yang memberikan fungsi hiburan adalah *game online*, *film online*, dan *music online*.

4. Fungsi transaksi

Situs *web* dapat dijadikan sarana transaksi bisnis seperti barang dan jasa. Situs *web* ini menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik. Pembayaran bisa menggunakan kartu kredit, transfer, atau dengan membayar secara langsung.

2.5.2 Pengertian *Website*

Sebuah situs *web* atau yang lebih dikenal dengan *website* terdiri dari kumpulan halaman-halaman yang di gunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman atau biasa disebut dengan nama *hyperlink* [24]. *Website* terbagi menjadi 2 jenis berdasarkan aspek konten yang disajikan, yaitu [25]:

1. *Website* statis

Website statis adalah *web* yang kontennya konstan. Konten konstan berarti isi dari dokumen *web* tidak dapat diubah dengan cepat dan mudah. Perubahan data pada halaman *web* statis hanya dapat dilakukan dengan cara mengubah secara *manual* melalui perubahan bahasa pemrograman *website* tersebut..

2. *Website* dinamis

Website dinamis adalah *web* yang kontennya dapat diubah-ubah secara langsung oleh *user administrator*. Terdapat program yang berjalan untuk mengatur perubahan data yang ditampilkan dalam *website* dinamis tersebut. Halaman *web* yang dibuat dengan menggunakan bahasa server seperti PHP, Perl, ASP, ASP.NET, JSP, ColdFusion dan bahasa yang lainnya.

Jenis *website* dinamis ini sangat sesuai untuk *website E-Commerce* yang membutuhkan *update* data secara terus menerus. Pemeliharaan *website* dinamis pun lebih mudah daripada *website* statis karena dapat menggunakan *Content Management System (CMS)*.

2.5.3 Unsur-Unsur Website

Sebuah *website* harus memiliki unsur-unsur penunjang untuk memaksimalkan penggunaan dari *web* tersebut. Fungsi dari suatu *website* akan menjadi tidak maksimal bila terdapat unsur yang tidak terpenuhi. Unsur-unsur dari suatu *website* yang harus dipenuhi adalah sebagai berikut:

1. Nama domain

Pengertian Nama *domain* atau biasa disebut dengan *domain name* atau *Uniform Resource Locator (URL)* adalah alamat unik di dunia internet yang digunakan untuk mengidentifikasi sebuah *website*, atau dengan kata lain *domain name* adalah alamat yang digunakan untuk menemukan sebuah *website* pada dunia internet. Nama *domain* diperjualbelikan secara bebas di internet dengan status sewa tahunan.

Nama *domain* sendiri mempunyai identifikasi ekstensi atau akhiran sesuai dengan kepentingan dan lokasi keberadaan *website* tersebut. Contoh nama *domain* berekstensi internasional adalah *com, net, org, info, biz, name*, dan *ws*. Contoh nama *domain* berekstensi lokasi Negara Indonesia adalah *co.id* untuk nama *domain website* perusahaan, *ac.id* untuk nama *domain website* pendidikan, *go.id* untuk nama *domain website* instansi pemerintah, *or.id* untuk nama *domain website* organisasi.

2. Web hosting

Pengertian *web hosting* dapat diartikan sebagai ruangan yang terdapat dalam *hard disk* tempat menyimpan berbagai data, *file-file*, gambar dan lain sebagainya yang akan ditampilkan di *website*. Ukuran data yang bisa dimasukkan tergantung dari besarnya *web hosting* yang dimiliki, semakin besar *web hosting* maka semakin besar data yang dapat dimasukkan dan ditampilkan dalam *website*. *Web Hosting* juga diperoleh dengan menyewa. Besarnya *hosting* ditentukan ruangan *hard disk* dengan ukuran *Mega Byte (MB)* atau *Giga Byte (GB)*. Lama penyewaan *web hosting* rata-rata dihitung per tahun. Penyewaan *hosting* dilakukan dari perusahaan-

perusahaan penyewa *web hosting* yang banyak dijumpai baik di Indonesia maupun luar negeri.

3. Bahasa program

Scripts adalah bahasa yang digunakan untuk menerjemahkan setiap perintah dalam *website* yang pada saat diakses. Jenis bahasa program sangat menentukan statis, dinamis atau interaktifnya sebuah *website*. Ragam bahasa pemrograman yang digunakan maka akan membuat *website* terlihat semakin dinamis, dan interaktif serta terlihat bagus. Jenis-jenis bahasa program yang banyak dipakai para *designer website* antara lain *Hxper Text Markup Language* (HTML), *Active Server Page* (ASP), *Hypertext Preprocessor* (PHP), *Java Server Pages* (JSP), *Java Scripts*, dan *Java applet*. Bahasa dasar yang dipakai setiap situs adalah HTML sedangkan PHP, ASP, JSP dan lainnya merupakan bahasa pendukung yang bertindak sebagai pengatur dinamis, dan interaktifnya situs. Bahasa program ASP, PHP, JSP atau lainnya bisa dibuat sendiri. Bahasa program ini biasanya digunakan untuk membangun portal berita, artikel, forum diskusi, buku tamu, anggota organisasi, *email*, *mailing list* dan lain sebagainya yang memerlukan *update* setiap saat.

4. *Design website*

Desain *website* menentukan kualitas dan keindahan sebuah *website*. Desain sangat berpengaruh kepada penilaian pengunjung akan bagus tidaknya sebuah *website*. Pembuatan *website* biasanya dapat dilakukan sendiri atau menyewa jasa *website designer*. Saat ini sangat banyak jasa *web designer*, terutama di kota-kota besar. Kualitas situs sangat ditentukan oleh kualitas *designer*. *Designer* yang semakin banyak menguasai tentang beragam *software* pendukung pembuatan situs maka akan dihasilkan situs yang semakin berkualitas, demikian pula sebaliknya. Jasa *web designer* ini yang umumnya memerlukan biaya yang tertinggi dari seluruh biaya pembangunan situs dan semuanya itu tergantung kualitas *designer*.

2.6 *Data Base Management System*

Database Management System (DBMS) adalah suatu *software* yang dirancang secara khusus untuk mengelola suatu *database* dan menjalankan operasi terhadap data sesuai dengan jumlah pengguna. DBMS juga dapat digunakan untuk membangun *database* yang rancangannya sudah terkomputasi. DBMS digunakan

untuk memudahkan proses *input, update, delete, restore, view*, dan proses pengambilan informasi terhadap *database*. DBMS memiliki beberapa jenis yaitu:

1. *Operational database*
2. *Relational database*
3. *Distributed database*
4. *External database*

Tujuan utama DBMS adalah menyediakan lingkungan yang mudah dan nyaman bagi *user* untuk menyimpan, mengambil, dan mengolah data dan informasi. *Software* yang tergolong kedalam DBMS adalah MySQL, MongoDB, Oracle, phpMyAdmin, dan lain sebagainya. Fungsi dari DBMS adalah [26]:

1. Memudahkan identifikasi data dengan cara mengelompokkan data.
2. Menghindari data ganda.
3. Mempermudah penggunaan oleh *user*.
4. Penyimpanan dalam bentuk digital.
5. Alternatif penyimpanan ruang dalam suatu aplikasi.

2.7 *Software XAMPP*

XAMPP adalah perangkat lunak *open source*, yang mendukung banyak sistem operasi dan merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri atau *localhost*. Terdiri atas program Apache HTTP *server*, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Contoh gambar dari *software XAMPP* dapat dilihat pada Gambar 2.6.



Gambar 2.6 *Software XAMPP* [27]

Gambar 2.6 merupakan contoh dari *software* XAMPP. Nama XAMPP merupakan singkatan dari X, Apache, MySQL, PHP dan Perl. Program ini tersedia dalam *general public license* dan bersifat gratis. XAMPP merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis. Fungsi XAMPP adalah *server* yang dapat berdiri dengan sendiri atau *localhost*.

2.8 Flow Sensor

Sensor adalah perangkat yang dapat mendeteksi perubahan fisis pada fenomena di lingkungan tertentu. Jenis sensor bermacam-macam, salah satunya adalah *flow sensor*. *Flow sensor* terdiri dari katup, rotor, dan sensor *hall effect*. Sensor ini berfungsi untuk mendeteksi suatu aliran zat fluida. Prinsip kerja dari *flow sensor* memanfaatkan medan magnet dan sensor *hall effect*.

Rotor yang dilewati oleh air maka rotor magnet akan berputar sesuai dengan kecepatan air yang mengalir. Posisi sensor *hall effect* tegak lurus dengan rotor sehingga akan terdapat medan magnet yang disebabkan oleh rotor magnet yang berputar dan akan menghasilkan *output* sinyal pulsa yang berupa tegangan listrik. Sinyal pulsa tersebut kemudian akan dibaca oleh mikrokontroler. *Output* dari pulsa tegangan memiliki tingkat yang sama dengan input yang dimiliki frekuensi laju air, lalu sinyal tersebut dapat diolah menjadi data digital oleh mikrokontroler [28].

2.9 LCD 16x2 I2C

LCD merupakan sebuah alat yang digunakan sebagai indikator untuk menampilkan data yang terbaca dari suatu sistem. Arti dari angka 16 dan 2 pada LCD adalah jumlah kolom dan baris tampilan yang terdapat pada layar LCD. Tampilan kolom pada layar adalah 16 kolom dan tampilan baris pada layar adalah 12 baris. LCD memiliki 16 *pin* dengan fungsinya masing-masing. Contoh LCD 16x2 dapat dilihat pada Gambar 2.7.



Gambar 2.7 LCD 16x2 [29]

Gambar 2.7 merupakan contoh dari LCD 16x2 dengan *background* berwarna hijau. Jumlah penggunaan *pin* pada LCD dapat dipangkas dengan menggunakan sebuah modul yang disebut *Inter Integrated Circuit* (I2C). I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang khusus untuk pengendalian IC. Sistem I2C merupakan sistem komunikasi yang terdiri dari *Serial Clock* (SCL) dan *Serial Data* (SDA) [30]. Penjelasan fungsi *pin* LCD 16x2 dapat dilihat pada Tabel 2.1.

Tabel 2.1 Spesifikasi *Pin* LCD 16x2

No	Pin	Keterangan
1	Vss	Ground
2	Vdd	Power +5V
3	Vo	Contrast adjust
4	RS	Register select signal
5	RW	Data read/write
6	E	Enable signal
7	D0	Data bus line
8	D1	Data bus line
9	D2	Data bus line
10	D3	Data bus line
11	D4	Data bus line
12	D5	Data bus line
13	D6	Data bus line
14	D7	Data bus line
15	A	Power supply for LED B/L (+)
16	K	Power supply for LED B/L (-)

Tabel 2.1 merupakan penjelasan fungsi dari setiap *pin* yang ada pada LCD. *Pin* memiliki fungsi yang berbeda, dapat dilihat *pin* nomor 2 merupakan *pin* yang berfungsi sebagai *power supply* untuk menyalakan LCD, *pin* 7 sampai dengan *pin* 14 merupakan *pin* yang digunakan sebagai identitas untuk menyalakan LED pada layar LCD

2.10 Kajian Pustaka

Penulisan skripsi ini didasarkan beberapa referensi penelitian sebelumnya yang membahas tentang *water meter* digital dan *wireless sensor network*. Penelitian sebelumnya adalah menganalisis tentang penerapan teknologi WSN agar *node*

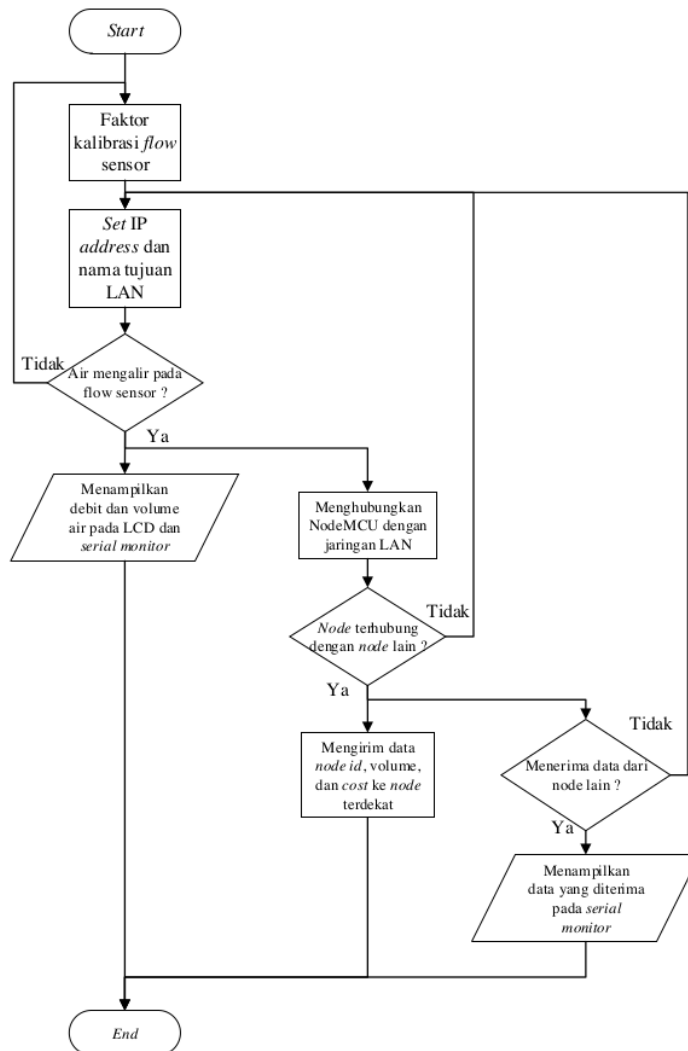
sensor pada teknologi WSN optimal berdasarkan struktur topologi jaringan yang digunakan [10]. Penelitian lainnya adalah menerapkan teknologi WSN untuk *monitoring* lingkungan berdasarkan parameter suhu, kadar gas monoksida dan karbon dioksida, kelembapan, dan curah hujan dan dapat dimonitoring melalui *webservice* [11]. Penelitian lainnya adalah menerapkan *prototype* sistem *monitoring* air dan energi listrik menggunakan IoT pada bangunan tempat tinggal menggunakan aplikasi android [12].

Penelitian lainnya adalah penerapan *water meter* menggunakan jaringan 4G pada *smart city*. Jaringan 4G digunakan sebagai media komunikasi *node* dengan *server* pada penelitian ini [13]. Penelitian lainnya melakukan perancangan dan mengimplementasikan jaringan WSN pada *water meter* menggunakan radio *wireless* nRF24101 dengan menerapkan topologi *mesh* pada jaringan WSN [14]. Penelitian lainnya menerapkan WSN pada *smart agriculture*. *Node sensor* terhubung pada modul ZigBee dan menggunakan mikrokontroler Arduino Nano [15]. Penelitian yang dilakukan sebelumnya masih memiliki beberapa kekurangan seperti masih menggunakan meteran analog dalam pembacaan air dan belum menggunakan *flow sensor*, serta tidak disediakan *website* untuk mempermudah *user* melihat penggunaan air. Berdasarkan hal itu, maka penelitian ini sistem *monitoring* penggunaan air menggunakan *flow sensor* agar pembacaan air lebih akurat dan membuat *website* untuk mempermudah *user* melihat penggunaan air sehingga sistem *monitoring water meter* dapat lebih mudah digunakan dan efisien.

BAB III METODE PENELITIAN

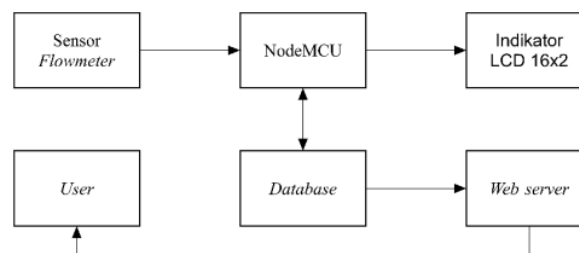
63 3.1 Perancangan Perangkat Keras

Perancangan perangkat keras yang dibuat adalah perancangan sistem alat, perancangan rangkaian listrik, dan perancangan model alat. Perancangan sistem dalam bentuk *flowchart* bertujuan untuk mempermudah melihat cara kerja dari sistem. *Flowchart* dari perangkat keras dapat dilihat pada Gambar 3.1.



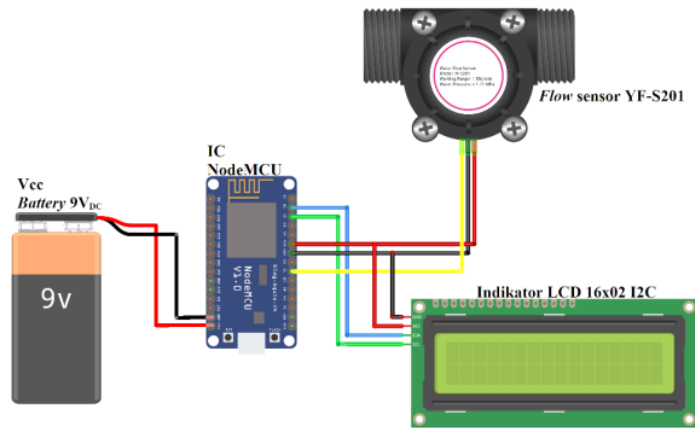
Gambar 3.1 *Flowchart* Perangkat Keras *Water Meter*

Gambar 3.1 menjelaskan proses dari pembacaan air pada *flow* sensor dan pengolahan data pembacaan air sehingga data dapat dikirim menuju *node* lain. Air yang mengalir melalui *flow* sensor akan membuat rotor pada sensor bergerak dan mengirimkan data menuju mikrokontroler. Data tersebut kemudian diolah dengan menggunakan faktor kalibrasi sebesar 5,5 untuk mendapatkan tingkat ketelitian yang akurat. Indikator kemudian akan menampilkan data yang sudah diolah dalam bentuk tampilan volume air dan debit air. Data pembacaan air pada mikrokontroler bukan hanya volume dan debit namun juga *cost* penggunaan air, namun data *cost* penggunaan air tidak di tampilkan pada indikator LCD karena terbatas pada ukuran LCD yang digunakan. Volume, *cost*, serta *id* dari *node* jika *node* sudah terhubung ke jaringan lokal yang dibuat maka *node* akan secara otomatis mengirim data dan menerima data dari *node* lain yang terdekat. *Node* yang tidak terhubung ke jaringan lokal yang dibuat tidak dapat mengirim data ke *node* terdekat sehingga perlu diperiksa kembali apakah nama jaringan yang *diset* pada *node* sudah sesuai dengan nama jaringan, karena jika tidak sesuai satu karakter dapat mengakibatkan *node* tidak dapat terhubung kedalam jaringan lokal. Keseluruhan cara kerja sistem yang dirancang dapat dilihat pada Gambar 3.2



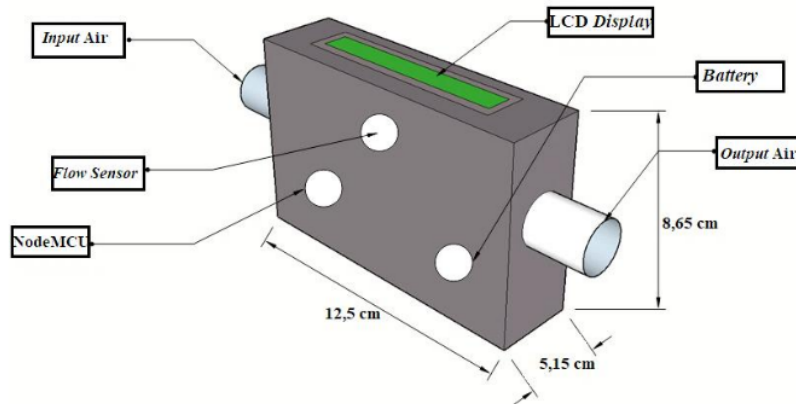
Gambar 3.2 Diagram Blok Sistem

Gambar 3.2 merupakan diagram blok kerja sistem *monitoring water meter*. Air yang mengalir melewati *sensor flowmeter* akan di deteksi oleh sensor sehingga sensor memberikan pulsa ke mikrokontroler. Mikrokontroler kemudian mengirimkan data yang diperoleh dari sensor ke *database* dan menampilkan data yang terukur ke LCD. Data yang sudah disimpan kemudian dapat ditampilkan pada *website* dan dapat diakses oleh *user*. Perancangan perangkat keras juga meliputi rangkaian kelistrikannya. Rangkaian kelistrikan untuk *water meter* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Rangkaian Sistem *Monitoring Water Meter*

Gambar 3.3 merupakan rangkaian listrik dari alat sistem *monitoring water meter*. Komponen utama pada *water meter* terdiri dari NodeMCU, *flow sensor*, LCD 16x2 I2C, dan baterai 9VDC. *Pin SDA LCD* terhubung ke *pin D1* NodeMCU dan *pin SCL LCD* terhubung ke *pin D2* NodeMCU. *Pin Vcc* terhubung ke *pin 3V3* dan *pin ground* terhubung ke *pin GND*. *Flow sensor* terhubung dengan *pin D6* NodeMCU serta 3V3 dan GND. Detail dari *hardware* yang dirancang dapat dilihat pada Gambar 3.4.



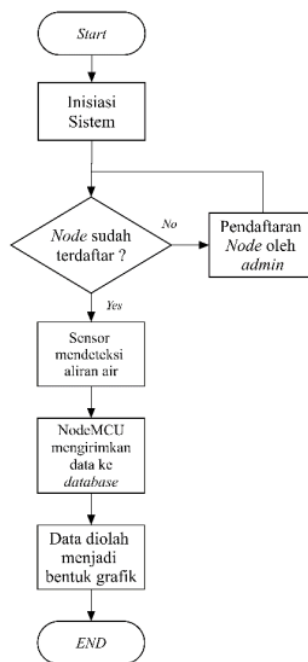
Gambar 3.4 Rancangan Model 3D Alat *Monitoring Water Meter*

Gambar 3.4 merupakan model *hardware* dari alat sistem *monitoring water meter*. *Case water meter* memiliki dimensi panjang 12,5 cm, lebar 5,15cm, dan tinggi

8,65cm. *Water meter* memiliki 2 pipa yang berfungsi sebagai *intake* air menuju *flow sensor* dan *outlet* air dari *flow sensor*. LCD sebagai indikator terdapat pada bagian atas dari *hardware*.

3.2 Perancangan Perangkat Lunak

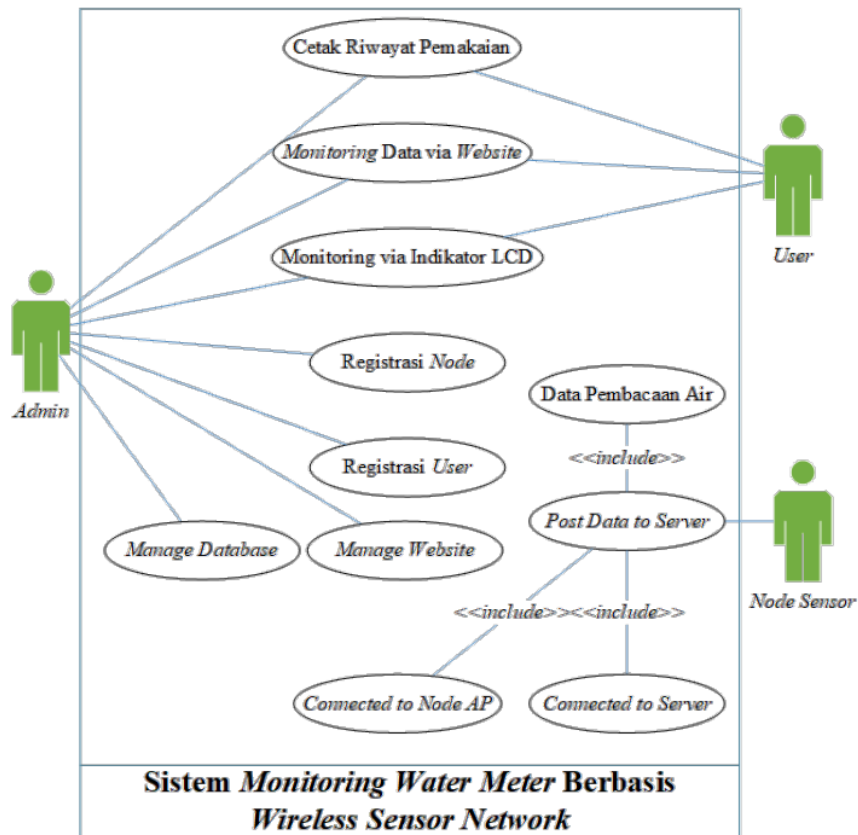
Perangkat lunak pada sistem ini terdiri dari *database*, *website*, dan topologi WSN. Perangkat lunak memiliki fungsi sebagai sistem yang bekerja secara komputasi. Cara kerja sistem perangkat lunak secara garis dapat dilihat pada Gambar 3.4.



Gambar 3.5 *Flowchart* Sistem

Gambar 3.5 merupakan *flowchart* kerja dari sistem yang sudah dirancang. Sistem akan melakukan inisiasi awal dengan menghubungkan sistem ke jaringan WSN, jika *node* sudah memiliki identitas maka *node* dapat memasuki jaringan dan akan terbaca namun jika *node* belum memiliki identitas maka *node* harus didaftarkan terlebih dahulu. *Node* yang sudah memiliki identitas dan sudah masuk kedalam jaringan jika mendeteksi aliran air, maka data akan diolah sehingga

menghasilkan *output* berupa volume penggunaan air, identitas *node*, dan total biaya penggunaan air. Data tersebut kemudian akan dikirim menuju *database*. Data yang sudah berada pada *database* akan diseleksi sesuai dengan identitas *node* dan akan dikirim ke *website* dalam bentuk tabel dan grafik. Data dengan bentuk tabel yang terdapat pada *website* sudah sesuai dengan identitas masing-masing *node* sehingga data yang ditampilkan pada *website* tidak akan memiliki data dari *node* dengan identitas yang berbeda dalam satu tabel. Grafik yang diolah merupakan grafik yang menunjukkan penggunaan air setiap *node* yang diakumulasikan setiap bulannya dalam satu tahun. Detail kerja dari *water meter* dapat dilihat pada *usecase diagram* pada Gambar 3.6.



Gambar 3.6 Use Case Diagram Water Meter

Gambar 3.6 merupakan *use case diagram* dari sistem *monitoring water meter*. Sistem *monitoring water meter* memanfaatkan jaringan lokal dengan

menggunakan komunikasi jaringan yang dimiliki oleh NodeMCU. Data dikirimkan jika *node* sensor terhubung ke *node* AP dan *server*. Data kemudian diolah menjadi bentuk grafik yang dapat dilihat pada *website*. Berdasarkan *use case diagram* dapat dilihat bahwa terdapat 3 aktor pada sistem yang dirancang. Aktor tersebut terdiri dari admin yang memiliki peran untuk melakukan registrasi *node*, *user*, mengelola *website* dan *database*, serta *monitoring* penggunaan air *user*. Aktor selanjutnya yaitu *user*, *user* hanya dapat melakukan *monitoring* penggunaan air pada *website* ataupun secara langsung dengan melihat indikator. *User* juga dapat melakukan cetak riwayat pemakaian penggunaan air. Aktor terakhir yaitu *node* sensor. *Node* sensor berperan dalam pengiriman data penggunaan air, untuk pengiriman data ke *server*, *node* sensor harus menyertakan data pembacaan air, terhubung dengan *node* AP, dan *node* AP sudah terhubung ke *server*.

3.2.1 Perancangan Database

Database berfungsi sebagai gudang penyimpanan data dan juga dapat digunakan untuk mengorganisasi data, menghindari duplikasi data, menghindari hubungan antar data yang tidak jelas, dan *update* yang rumit. Proses *receive* dan *transfer* data terhadap *database* memerlukan perangkat lunak yang disebut dengan DBMS. Banyak sekali DBMS *open source* yang dapat digunakan secara gratis, salah satunya adalah *phpMyAdmin*. Rancangan *database* menggunakan *phpMyAdmin* dapat dilihat pada Gambar 3.7.

#	Name	Type
<input type="checkbox"/>	1 id	int(11)
<input type="checkbox"/>	2 node_id	int(11)
<input type="checkbox"/>	3 volume	decimal(10,4)
<input type="checkbox"/>	4 cost	decimal(20,2)
<input type="checkbox"/>	5 created_at	datetime

Field yang digunakan

Tipe Data

Gambar 3.7 Rancangan Database Monitoring Water Meter

Gambar 3.7 merupakan rancangan dari *database*. *Database* terdiri dari tabel yang diberi nama *dashboard*. Tabel *dashboard* berfungsi untuk menyimpan data yang dikirimkan oleh NodeMCU. *Field* *id* berfungsi untuk memberikan

penomoran data yang masuk. *Field* `node_id` berfungsi untuk memberikan identitas bagi setiap *node* yang terdaftar. *Field* `volume` berfungsi sebagai indikasi penggunaan air oleh *user*, *field* `cost` adalah biaya yang harus dibayarkan sesuai dengan pemakaian. *Field* `created_at` adalah waktu penerimaan data terbaru dari NodeMCU ke *database*.

3.2.2 Perancangan Website

Perancangan *website* bertujuan untuk memudahkan akses pengguna dalam *monitoring* penggunaan air. *Website* juga berguna untuk memberikan detail terhadap sistem yang sudah dirancang. Penelitian ini bagian pada *website* dibagi menjadi 3 bagian, yaitu *landing page* yang berisikan informasi tentang *user* dan pemakaiannya selama satu hari, *about us* yang berisikan latar belakang pembuatan *website* dan informasi-informasi tentang ilmu-ilmu terkait, dan *contact* yang berisi kontak *customer service*.

a. Landing page

Tampilan awal atau *landing page* dari *website water meter* berisi *user id*, penggunaan air ketika pengiriman data terakhir serta biaya, dan waktu *update* terakhir atau diterimanya data terbaru dari mikrokontroler. Rancangan *landing page website* dapat dilihat pada Gambar 3.8.

<p style="text-align: center;">LOGO Sistem Monitoring Water Meter</p>				
Dashboard	About Us	Contact		
No.	User ID	Penggunaan	Biaya	Waktu
1				
2				
3				
4				
5				
6				
7				
8				
9				

Gambar 3.8 Rancangan Tampilan *Landing Page*

Gambar 3.8 merupakan rancangan tampilan *landing page* pada *website*. *Landing page* pada sebuah *website* berfungsi sebagai media bagi setiap *user* yang

mengunjungi halaman *web*. Rancangan tampilan *landing page* memiliki pilihan untuk mengunjungi halaman-halaman lain yang tersedia pada *website*, terdapat pilihan halaman *dashboard*, *about us*, dan *contact*. *User* akan mengunjungi halaman *dashboard* secara *default* saat membuka *website*. Halaman *dashboard* dirancang untuk menyediakan informasi berupa *user id*, penggunaan air, biaya yang harus dibayar, serta waktu terakhir data diterima dari *water meter*.

b. Halaman *detail node*

Halaman *detail node* merupakan halaman yang terbuka apabila *node id* pada tabel di *landing page* dipilih. Halaman ini berisikan riwayat dari penggunaan air *node* terpilih. Perancangan halaman *detail node* dapat dilihat pada Gambar 3.9.

LOGO			
Sistem Monitoring Water Meter			
Dashboard	About Us	Contact	
User ID			
Rekap Penggunaan Bulan <input type="text"/> / Tahun <input type="text"/>			
Bulan <input type="text"/>	No	Riwayat (Hari, Tanggal/Bulan/Tahun)	Total Pemakaian
Tahun <input type="text"/>			
<input type="button" value="Pilih"/>			
<input type="button" value="Export PDF"/>			

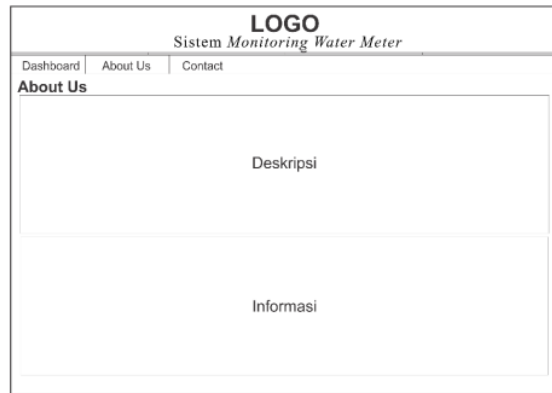
Gambar 3.9 Rancangan Tampilan Halaman *Detail User*

Gambar 3.9 merupakan rancangan tampilan halaman *detail user*. Riwayat penggunaan air dicatat setiap hari. Akumulasi data *node* adalah setiap satu bulan dan selanjutnya dikategorikan pada bulan dan tahun saat perekapan sehingga pencarian pada data-data terdahulu dari *node* tersebut dapat dilakukan dengan mudah. Halaman ini juga menyediakan pilihan *export pdf* yang dapat digunakan untuk melakukan *print out* riwayat pemakaian sehingga dapat diperoleh bentuk fisik dari pencatatan sesuai dengan periode waktu yang diinginkan.

c. Halaman *about us*

Halaman ini merupakan penjelasan tentang *website water meter*. Halaman ini berisi tentang latar belakang pembuatan *website* dan deskripsi pihak pengelola.

Selain itu, halaman ini juga berisi informasi-informasi penting tentang ilmu yang terkait. Rancangan tampilan halaman *about us* dapat dilihat pada Gambar 3.10.

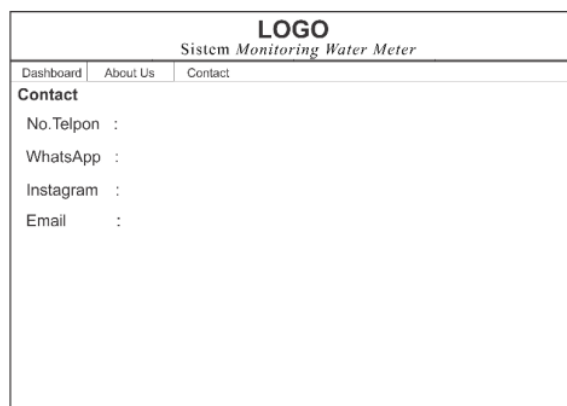


Gambar 3.10 Rancangan Tampilan Halaman *About Us*

Gambar 3.10 merupakan rancangan tampilan halaman *about us*. Halaman *about us* dirancang memiliki 2 bagian utama, yaitu deskripsi dari pihak penyedia air dan informasi bermanfaat terkait pemanfaatan dan pengelolaan air. Halaman ini dirancang dengan sederhana agar *user* yang mengunjungi halaman *website* dapat dengan mudah memahami isi konteks yang dimaksud pada halaman *about us*.

d. Halaman *contact*

Halaman ini memuat tentang daftar nomor telepon, *email*, serta sosial media yang dapat dihubungi oleh *user*. Informasi ini dapat digunakan oleh *user* untuk menghubungi pihak pengelola apabila terjadi masalah pada saluran air yang digunakan. Rancangan tampilan halaman *contact* dapat dilihat pada Gambar 3.11.

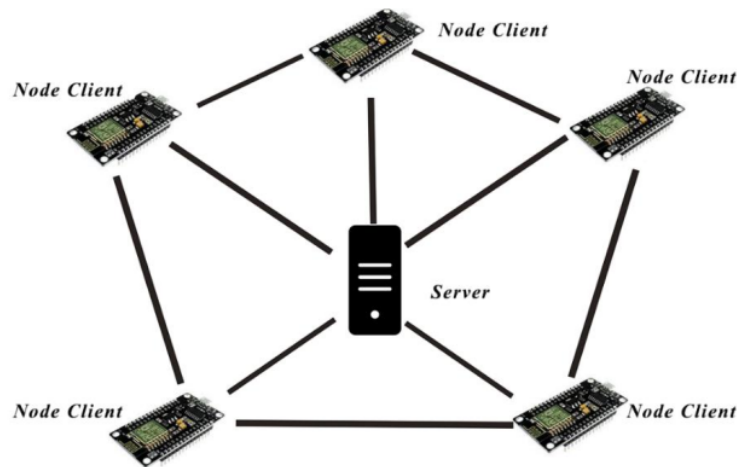


Gambar 3.11 Rancangan Tampilan Halaman *Contact*

Gambar 3.11 merupakan rancangan tampilan halaman *contact*. Informasi berupa nomor telepon, sosial media, dan *email* yang dapat dihubungi disediakan agar memudahkan *user* untuk menghubungi petugas. Informasi tambahan seperti nama dan foto petugas juga akan dicantumkan agar identitas diri dari petugas lebih dikenali oleh *user*.

3.2.3 Perancangan Topologi *Wireless Sensor Network*

Jenis topologi yang digunakan dalam sistem ini adalah topologi *mesh*. Perancangan topologi jaringan sensor nirkabel pada sistem dapat dilihat pada Gambar 3.12.



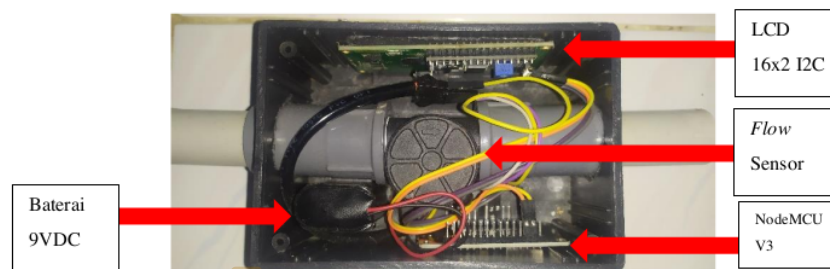
Gambar 3.12 Topologi *Mesh*

Gambar 3.12 merupakan satu jaringan yang terdapat 5 buah *node client* yang langsung terhubung ke *server* dan juga memiliki hubungan dengan *node client* didekatnya. Cakupan luas jangkauan jaringan nirkabel pada topologi ini terbatas antara *node client* dan *server*. *Node client* terhubung secara langsung ke *server* dan setiap *node* juga saling terhubung atau *dedicated link* sehingga memungkinkan komunikasi data yang lebih cepat.

HASIL DAN ANALISIS

4.1 Hasil Perancangan *Hardware*

Komponen utama *hardware* terdiri dari *flow* sensor, NodeMCU, LCD 16x2 I2C, kabel *jumper*, dan baterai 9V. *Hardware* dirancang didalam *case* dengan dimensi 12,5x8x5 cm. Hasil perancangan *hardware* dapat dilihat pada Gambar 4.1.



Gambar 4.1 Tampilan Dalam *Hardware*

Gambar 4.1 merupakan bagian dalam dari *case hardware water meter*. Bagian dalam dilengkapi dengan *sock drat* pipa yang terhubung dengan *flow* sensor. *Shock drat* diperlukan untuk menghindari kebocoran bagian dalam alat. Perlindungan diberikan agar saat sensor menerima air dengan tekanan yang besar air tidak akan keluar dari sambungan pipa dengan sensor.

4.2 Hasil Perancangan *Software*

Perancangan *software* terdiri dari pembuatan *database* dan *website*. *Database* terdiri dari 1 tabel yang bernama *dashboard* dan berisikan kolom *id*, *node_id*, *volume*, *cost*, dan *created_at*. Halaman pada *website* terdiri dari *dashboard*, *about us*, dan *contact*.

4.2.1 Hasil Perancangan *Database*

Perancangan *database* menggunakan *database management system* bernama phpMyAdmin dan hanya menggunakan 1 tabel. Berikut adalah tampilan dari *database* yang sudah dirancang. Hasil perancangan *database* dapat dilihat pada Gambar 4.2.

id	node_id	volume	cost	created_at
1	1	0.0000	0.00	2022-01-04 22:14:03
2	2	0.0000	0.00	2022-01-04 22:14:07
3	3	0.0000	0.00	2022-01-04 22:14:11
4	4	0.0000	0.00	2022-01-04 22:14:16
5	5	0.0000	0.00	2022-01-04 22:14:20
6	1	0.0000	0.00	2022-01-04 22:19:43
7	2	0.0000	0.00	2022-01-04 22:19:47
8	3	0.0000	0.00	2022-01-04 22:19:52
9	4	0.0000	0.00	2022-01-04 22:19:57
10	5	0.0000	0.00	2022-01-04 22:20:00
11	1	0.0000	0.00	2022-01-05 22:27:47
12	2	0.0000	0.00	2022-01-05 22:12:31

Field Tabel

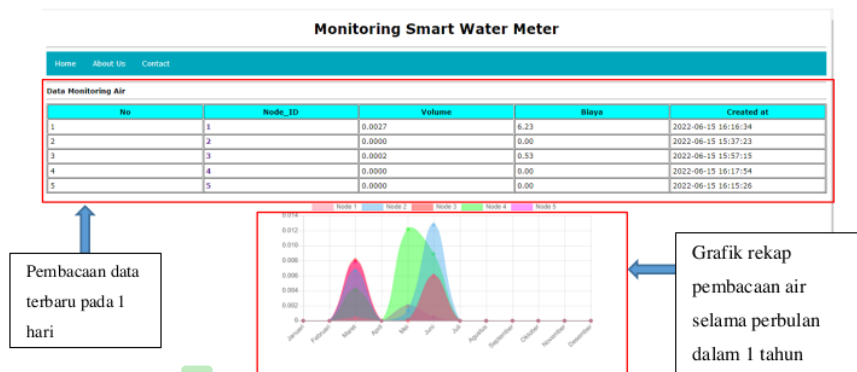
Data Pembacaan Air

42
Gambar 4.2 Tampilan Database Sistem Monitoring

Gambar 4.2 merupakan tampilan dari *database* yang sudah dirancang. Tabel bernama *dashboard* dan berisikan beberapa kolom yaitu *id*, *node_id*, *volume*, *cost*, dan *created_at*. Kolom *id* berfungsi sebagai penomoran pada setiap data yang sudah masuk kedalam *database*. Kolom *node_id* berisikan identitas setiap *node* atau alat, terdapat 5 *node* yang sudah terhubung ke *database*. Kolom *volume* berfungsi untuk menjelaskan jumlah penggunaan air yang sudah terbaca oleh alat. Kolom *cost* berfungsi untuk menjelaskan jumlah biaya yang harus dibayar oleh pengguna air sesuai dengan jumlah penggunaan air yang sudah dipakai. Kolom *created_at* berfungsi sebagai *log* atau waktu data dikirim oleh alat dan diterima di *database*.

4.2.2 Tampilan Halaman Dashboard Web

Halaman awal atau *dashboard* pada *website* berguna sebagai tempat singgah pertama *user* pada *website* dan menjelaskan secara singkat isi dari *website* tersebut. Hasil perancangan halaman *dashboard* dapat dilihat pada Gambar 4.3.

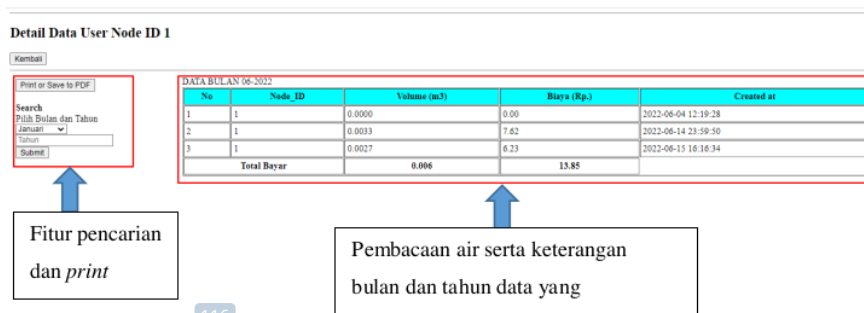


17
Gambar 4.3 Tampilan Halaman *Dashboard Web*

Gambar 4.3 merupakan tampilan halaman *dashboard website* yang sudah dirancang. *Dashboard* menampilkan informasi singkat tentang *monitoring* air. Terdapat tabel yang berisi data paling terbaru yang diterima *web*. Data tersebut kemudian diakumulasi setiap satu bulan dan diolah menjadi grafik. Grafik tersebut menampilkan penggunaan air setiap *node* tiap bulan selama satu tahun dengan indikasi warna berbeda tiap *node* nya.

137 4.2.3 Tampilan Halaman Detail

Halaman detail berfungsi untuk menampilkan riwayat penggunaan air tiap *node* secara rinci. Riwayat ditampilkan dalam bentuk tabel dengan jangka waktu 1 bulan. Hasil perancangan halaman detail *node* dapat dilihat pada Gambar 4.4.



116
Gambar 4.4 Tampilan Halaman Detail *Node 1*

Gambar 4.4 merupakan halaman detail *node* yang berada pada *website*. Tabel pada halaman detail terbagi menjadi 5 kolom yaitu nomor, *node_id*, volume, biaya, dan *created at* atau keterangan waktu data diterima oleh *database*. Data yang diterima diakumulasi setiap hari selama 1 bulan sehingga total penggunaan air dan biaya yang akan dibayarkan dapat langsung dilihat tanpa perlu melakukan perhitungan. Terdapat beberapa fitur pada halaman detail yaitu cetak riwayat dalam bentuk pdf dan pencarian riwayat berdasarkan bulan dan tahun yang terpilih.

4.2.4 Tampilan Halaman *About Us*

Halaman *about us* difungsikan sebagai halaman yang menjelaskan *profile* pihak pengelola air. Halaman ini juga memberikan informasi lain yang bermanfaat yang dapat dibaca oleh *user* saat mengunjungi halaman *web*. Hasil perancangan halaman *about us* pada *website* dapat dilihat pada Gambar 4.5.

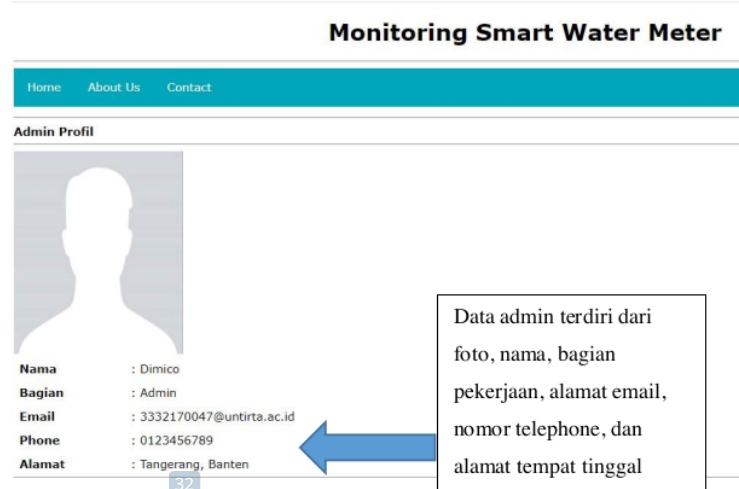


Gambar 4.5 Tampilan Halaman *About Us*

Gambar 4.5 merupakan tampilan dari halaman *about us* yang sudah dirancang. Halaman *about us* berfungsi untuk menjelaskan informasi tentang pihak pengelola air terkait. Halaman ini juga memuat informasi-informasi penting berkaitan dengan manfaat menghemat air, cara menghemat air, dan lain sebagainya.

4.2.5 Tampilan Halaman *Contact*

Halaman *contact* berisikan tentang informasi seputar profil dari admin atau pihak pengelola air terkait yang dapat dihubungi. Halaman ini terdiri dari nama *admin*, bagian pekerjaan, *email*, nomor telepon, dan alamat tempat pengelola air berada. Hasil perancangan halaman *contact* dapat dilihat pada Gambar 4.6.



Gambar 4.6 Tampilan Halaman *Contact*

Gambar 4.6 merupakan hasil perancangan halaman *contact*. Halaman ini memiliki *design* yang sederhana agar dapat dengan mudah dipahami oleh *user*. Informasi yang dapat digunakan oleh *user* untuk menghubungi *admin* dapat diakses dengan mudah jika terdapat gangguan atau membutuhkan bantuan lainnya.

4.3 Pengujian *Hardware*

Pengujian pada *hardware* meliputi pengujian kalibrasi *flow* sensor dan kalibrasi meteran analog. *Flow sensor* yang digunakan memiliki faktor kalibrasi sebesar 5.5. Faktor kalibrasi ini yang akan mempengaruhi tingkat akurasi dari pembacaan penggunaan air *water meter*. Hasil kalibrasi sensor dapat dilihat pada Tabel 4.1.

Tabel 4.1 Hasil Kalibrasi *Flow* Sensor

No	Volume air terukur (m ³)	Hasil Pengukuran <i>Flow</i> Sensor (m ³)						%Error
		x1	x2	x3	x4	x5	x6	
1	0,0005	0,0005	0,0005	0,0005	0,0005	0,0005	0,0005	0
2	0,0010	0,0010	0,0010	0,0011	0,0010	0,0010	0,0010	1,67
3	0,0020	0,0020	0,0020	0,0020	0,0020	0,0020	0,0020	0
4	0,0030	0,0030	0,0030	0,0030	0,0030	0,0030	0,0030	0
5	0,0040	0,0040	0,0040	0,0040	0,0040	0,0040	0,0040	0
6	0,0050	0,0050	0,0050	0,0050	0,0051	0,0050	0,0050	1,67%

Tabel 4.1 merupakan hasil dari kalibrasi *flow sensor*. Kalibrasi dilakukan dengan menggunakan metode *self-calibration*. Terdapat 2 tahap kalibrasi pada pengujian kali ini, yaitu kalibrasi *flow sensor* dengan takaran 500mL, 1L, 2L, 3L, 4L, dan 5L, dan kalibrasi meteran analog dengan takaran 500mL, 1L, 2L, 3L, 4L, dan 5L. Pengujian sebanyak 2 tahap ini bertujuan untuk membandingkan hasil pengukuran *flow sensor* dengan meteran analog. Persen error yang terbaca pada *flow sensor* untuk pengukuran volume 500mL menunjukkan hasil yang sama dengan volume air pada wadah takaran air sehingga diperoleh persen *error* sebanyak 0%. Pengukuran pada volume 1L dan 5L memiliki pembacaan *error* sebesar 1,67%. *Error* ini dapat disebabkan oleh jumlah volume air pada wadah air sebelum diukur *flow sensor* lebih dari yang akan diukur. Dapat dilihat pada Tabel 4.1 bahwa rata-rata persentase *error* yang diperoleh *flow sensor* adalah 0,56%. Hasil dari kalibrasi meteran analog dapat dilihat pada Tabel 4.2.

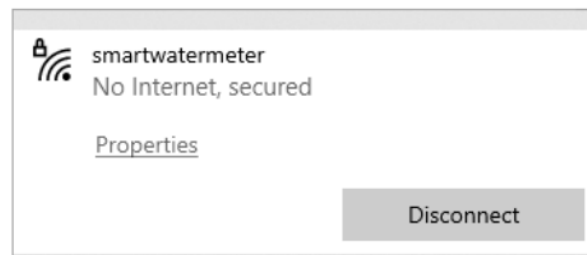
Tabel 4.2 Hasil Kalibrasi Meteran Analog

No	Volume air terukur (m ³)	Hasil Pengukuran Meteran Konvensional (m ³)						%Error
		x1	x2	x3	x4	x5	x6	
1	0,0005	0,0002	0,00025	0,0002	0,0002	0,0002	0,0002	58,34
2	0,0010	0,0005	0,0005	0,0005	0,0005	0,0005	0,0006	48,33
3	0,0020	0,0009	0,0010	0,0009	0,0008	0,0009	0,0010	54,15
4	0,0030	0,0016	0,0012	0,0013	0,0013	0,0011	0,0011	57,77
5	0,0040	0,0017	0,0020	0,0020	0,0020	0,0020	0,0020	51,25
6	0,0050	0,0022	0,0024	0,0020	0,0023	0,0020	0,0022	56,34

Tabel 4.2 merupakan hasil kalibrasi yang diperoleh dari meteran analog. Persen *error* yang diperoleh dari hasil kalibrasi pada volume 500mL, 1L, 2L, 3L, 4L, dan 5L selalu melebihi 48%. Rata-rata *error* yang diperoleh dari hasil kalibrasi meteran analog adalah 54,36%. Hal ini disebabkan oleh kecepatan air saat proses pengujian dilakukan. Pada *flow sensor*, meskipun kecepatan air sangat lambat namun tetap dapat dideteksi oleh *flow sensor*. Kecepatan minimum yang dapat terbaca oleh *flow sensor* adalah 0,17 L/menit.

4.4 Pengujian Software

Pengujian pada *software* meliputi akurasi pengiriman data dari sensor menuju *website*, pengujian ini bertujuan untuk mengetahui apakah terdapat error pada saat pengiriman data dari *node* ke *website*. *Node* yang digunakan menjadi *access point* akan membuat koneksi yang dapat diakses dengan menggunakan jaringan Wifi. Koneksi ini dapat diatur menggunakan program sehingga dapat diberikan fitur pengamanan seperti sandi sehingga koneksi menjadi lebih aman untuk digunakan. Koneksi yang dihasilkan oleh *node access point* dapat dilihat pada Gambar 4.7.



Gambar 4.7 Access Point Node AP

Gambar 4.7 merupakan *access point* yang digunakan dan terdapat pada *Node 1*. Jaringan tersebut kemudian dihubungkan dengan *PC server* dengan menyamakan *IP address*. *Access Point* ini yang akan menjadi penghubung *node AP* dengan *node* lainnya secara nirkabel sehingga pembacaan data dapat diteruskan dari *node* ke *website*. Pengujian pengiriman data untuk menguji kinerja dari jaringan yang digunakan dapat dilihat pada Tabel 4.3.

Tabel 4.3 Pengujian Pengiriman Data Ke Website

Data ke-	Data Pada Alat		Data Pada Website	
	Volume(m ³)	Biaya (Rp.)	Volume(m ³)	Biaya (Rp.)
1	0.0000	0.00	0.0000	0.00
2	0.0003	0.67	0.0003	0.67
3	0.0016	3.68	0.0016	3.68
4	0.0020	4.49	0.0020	4.49
5	0.0029	6.70	0.029	6.70

Tabel 4.3 merupakan hasil dari pengujian pengiriman data dari alat menuju *website* menggunakan jaringan yang sudah dirancang dengan menghubungkan *node client*

ke *node access point*. Data yang terbaca pada *flow sensor* dapat dilihat melalui indikator LCD atau dapat diakses melalui *website*. Tabel 4.3 memperlihatkan bahwa dalam percobaan pengiriman data yang dilakukan ke *website* semua data yang terdapat pada *website* sesuai dengan data penggunaan air yang terbaca oleh *flow sensor*. Tidak terdapat kegagalan saat pengiriman data sehingga semua data dapat diterima oleh *server* dan dapat ditampilkan pada *website*. Kesimpulannya *website* bekerja dengan baik karena dapat menampilkan data yang berada pada alat yang berisikan id *node*, jumlah pemakaian air, dan total biaya yang harus dibayar.

4.5 Pengujian Jaringan

Topologi jaringan yang digunakan pada sistem *monitoring* ini adalah *mesh*. Pengujian pada bagian jaringan meliputi pengujian jarak dan pengujian jalur komunikasi topologi *mesh*. Pengujian ini bertujuan untuk mengetahui tingkat efektifitas dari jaringan yang sudah dirancang dalam menjalankan proses *monitoring water meter*.

4.5.1 Pengujian Jarak Node

Pengujian dilakukan dengan memberikan jarak antar *node* sebesar 1, 5, 10, 15, 20, 25, dan 30 meter. Gambar 4.8 merupakan contoh dari *node* yang terhubung dengan *node access point* atau *node 1*.

```
14:42:52.181 -> CONNECTION: closingTask(): dropping 0 now= 376015084
14:42:52.181 -> CONNECTION: eraseClosedConnections():
14:42:52.181 -> CONNECTION: ~MeshConnection():
14:42:52.181 -> Message received from 1298103367
14:42:52.181 -> Node: 2 Volume: 0.00103
14:42:52.181 -> Biaya: 2.38
```

Pesan dari node asal
terbaca jika node
terhubung

Gambar 4.8 Kondisi Node Terhubung

Gambar 4.8 adalah contoh kondisi saat sebuah *node* terhubung dengan *node* yang lainnya. Syarat *node* terhubung dengan *node* lain dapat dilihat pada serial monitor dengan keterangan pesan yang berada pada *node* asal telah diterima oleh *node* tujuan. *Node* pengirim pada Gambar 4.8 yaitu *node 2* telah mengirimkan pesan berupa volume penggunaan air dan biaya kepada *node* tujuan yaitu *node 1*. Penerapan kondisi tersebut pada pengujian jarak dilakukan dengan memberikan

jarak *node* 1 dengan *node* yang lainnya sehingga dapat diketahui jarak efektif alat untuk saling berkomunikasi. Pengujian jarak ini juga dilakukan tanpa *obstacle* dengan tujuan untuk mengetahui jarak maksimum pada jaringan lokal yang digunakan pada *node water meter*. Hasil pengujian jarak *node client* dengan *node access point* dapat dilihat pada Tabel 4.4.

Tabel 4.4 Pengujian Transmisi Data Antara *Node* Dengan *Access Point*

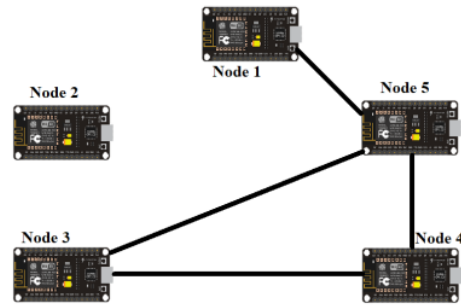
No.	Keterangan	Hasil Pengujian
1.	<i>Access point</i> menerima data dari <i>node client</i> jarak 1 meter	Ok
2.	<i>Access point</i> menerima data dari <i>node client</i> jarak 5 meter	Ok
3.	<i>Access point</i> menerima data dari <i>node client</i> jarak 10 meter	Ok
4.	<i>Access point</i> menerima data dari <i>node client</i> jarak 15 meter	Ok
5.	<i>Access point</i> menerima data dari <i>node client</i> jarak 20 meter	Ok
6.	<i>Access point</i> menerima data dari <i>node client</i> jarak 25 meter	Ok
7.	<i>Access point</i> menerima data dari <i>node client</i> jarak 30 meter	Ok

Tabel 4.4 merupakan hasil pengujian jarak *node client* dengan *node access point* dengan 7 rentang jarak. Hasilnya adalah semua *node* terhubung dengan semua rentang jarak yang ditentukan. Pengujian ini dilakukan untuk mengetahui batas jarak maksimum antar tiap *node* untuk masih dapat saling berkomunikasi. Jarak maksimum yang diperoleh dari pengujian ini adalah 30 meter. Batas maksimum ini diperoleh karena jaringan nirkabel yang digunakan berasal langsung dari *node* itu sendiri bukan menggunakan *router*, sehingga kekuatan sinyal terbatas pada kemampuan pemancar sinyal dan faktor lain seperti bangunan menjadi penghalang sinyal dan dapat mengurangi kekuatan sinyal.

4.5.2 Pengujian Topologi *Mesh*

Pengujian topologi dilakukan dengan menggunakan 5 buah *node*. Dalam pengujian ini diberikan beberapa kondisi, kondisi 1 yaitu *node* 2 dinonaktifkan dan diberikan halangan sehingga *node* 3 harus terhubung dengan *node* 4 atau *node* 5 untuk dapat diteruskan menuju *node* 1, dan kondisi 2 saat *node* 5 dinonaktifkan dan

diberikan halangan sehingga *node 4* harus terhubung dengan *node 2* atau *node 3*.
 Pengujian ini bertujuan untuk mengetahui kemampuan dari topologi jaringan yang sudah dibangun sehingga dapat diterapkan untuk meningkatkan efisiensi pengiriman data tiap *node* jika suatu saat salah satu *node* mengalami gangguan yang tidak diinginkan. Gambar 4.9 merupakan kondisi 1 pengujian topologi *mesh*.



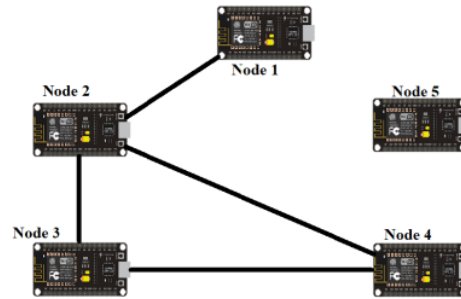
Gambar 4.9 Kondisi 1 *Node 2 Off*

Gambar 4.9 adalah bentuk koneksi *node* dengan *node* lainnya dengan menerapkan kondisi koneksi pada *node 2* dengan *node* yang lainnya dinonaktifkan dengan cara menonaktifkan *node 2*. Hasil yang diperoleh pada kondisi 1 dapat dilihat pada Tabel 4.5.

Tabel 4.5 Hasil Pengujian *Mesh* Kondisi 1 *Node 2 Off*

Node ID	Kondisi Node
3	Ok
4	Ok
5	Ok

Tabel 4.5 merupakan hasil pengujian yang diperoleh dari Pengujian kondisi 1. Pada kondisi 1, *node 3* harus terhubung terlebih dahulu ke *node 4* atau *node 5* agar dapat berkomunikasi dengan *node 1*. Kondisi kedua adalah *node 5* dinonaktifkan. Sehingga *node 4* harus terhubung ke *node 2* atau *node 3* sebelum diteruskan menuju *node 1*. Gambar 4.10 merupakan kondisi 2 pengujian topologi *mesh*.



Gambar 4.10 Kondisi 2 *Node 5 Off*

Gambar 4.10 adalah bentuk topologi pengujian saat kondisi 2 diterapkan. *Node 5* dinonaktifkan sehingga pada Gambar 4.10 garis yang menghubungkan *node 5* dengan *node 1* dan *node 4* hilang. Hasil yang diperoleh pada saat pengujian kondisi 2 dapat dilihat pada Tabel 4.6.

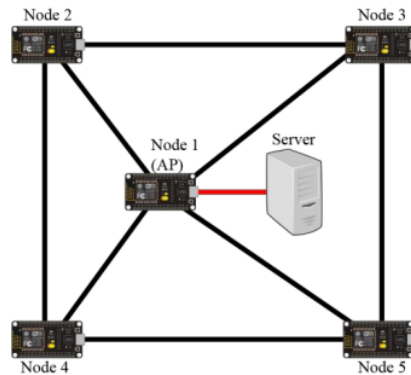
Tabel 4.6 Hasil Pengujian *Mesh* Kondisi 2 *Node 5 Off*

<i>Node ID</i>	<i>Kondisi Node</i>
2	Ok
3	Ok
4	Ok

Tabel 4.5 dan Tabel 4.6 merupakan hasil dari pengujian kondisi 1 dan kondisi 2. Dapat dilihat bahwa komunikasi antara *node* berjalan dengan baik., sedangkan pada kondisi 2 *node 4* harus terhubung ke *node 2* atau *node 3* agar dapat berkomunikasi dengan *node 1*. Kedua *node* tersebut tetap dapat berkomunikasi dengan *node 1* meskipun salah satu jalur komunikasinya diputus.

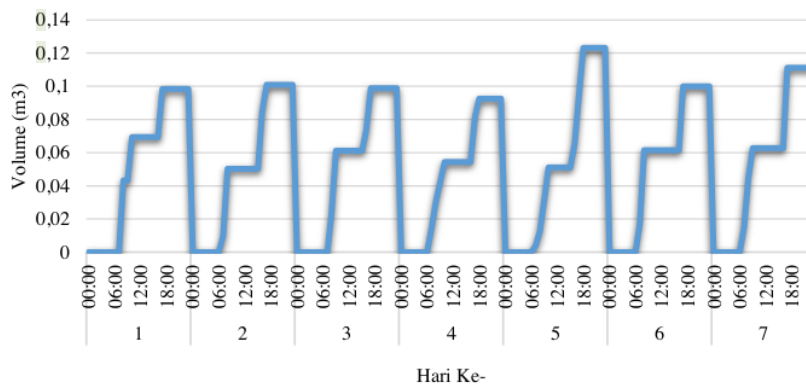
4.6 Pengujian Keseluruhan

Pengujian meliputi pengiriman data oleh *node* lain ke *node 1* atau *node AP*, pengiriman data yang diterima *node 1* ke *database*, dan pembacaan data yang berada didalam *database* oleh *website*. Sistem yang sudah dirancang akan melakukan perekaman data setiap harinya dan melakukan *reset* data jika sudah mencapai 24 jam atau 1 hari. Topologi pengujian keseluruhan dapat dilihat pada Gambar 4.11.



Gambar 4.11 Topologi *Mesh* Pengujian Keseluruhan

Gambar 4.11 adalah bentuk dari jaringan pada proses pengujian keseluruhan dimana 4 *node client* saling terhubung satu sama lain dan juga terhubung dengan *node AP*, kemudian *node AP* terhubung pada *server*. Pengujian dilakukan dengan *node 1* berada pada titik pusat dari setiap *node* untuk memudahkan proses pengiriman data. Jarak *node* lain bervariasi dari 5 meter sampai dengan 12 meter. Grafik pembacaan penggunaan air selama 7 hari dapat dilihat pada Gambar 4.12.



Gambar 4.12 Grafik *Monitoring* Penggunaan Air *Node 2*

Gambar 4.12 merupakan grafik penggunaan air pada *node 2* atau saluran keran air 1 selama 7 hari pemakaian. Lokasi penempatan tiap-tiap *node* berbeda-beda. *Node 1* sebagai AP terhubung langsung ke PC dengan tujuan melihat serial *monitor* pada IDE. *Node 2* berada pada bak mandi 1, *node 3* bak pada bak mandi 2,

node 4 pada keran pencuci piring, dan *node* 5 pada keran taman. Perolehan biaya dapat dilihat pada Persamaan (2.1) dan golongan R2.

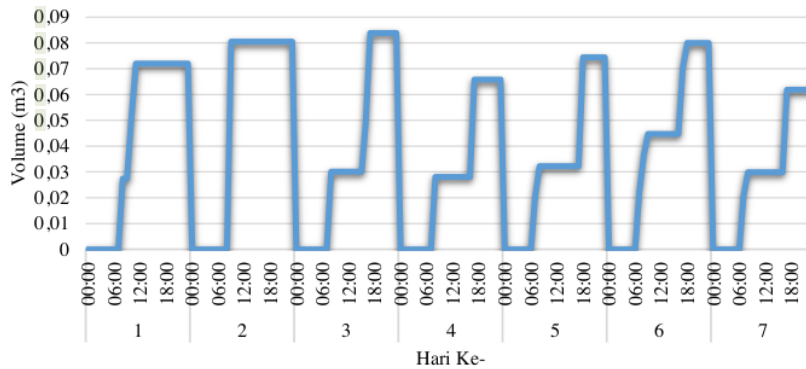
Pembacaan dilakukan mulai dari pukul 00:00 WIB s.d. 00:00 WIB keesokan harinya. *Node* 2 pembacaan baru berjalan ketika keran pada bak mandi 1 digunakan untuk mengisi air pada pukul 08:00 dengan volume terbaca yaitu $0,0432 \text{ m}^3$ dan dapat dilihat bahwa nilai tetap sampai pukul 09:00 karena keran hanya dibuka pada saat rentang waktu pukul 08:00 s.d. 08:59. Nilai pembacaan kemudian berubah kembali pada pukul 10:00 menjadi $0,0693 \text{ m}^3$. Nilai tersebut kemudian tetap sampai pukul 16:00 karena keran pada bak mandi 1 tidak dibuka pada periode waktu tersebut. Keran kembali dibuka pada pukul 17:00 dan air yang terukur menjadi $0,0984 \text{ m}^3$ dan tetap sampai pukul 23:59.

Hari kedua, total pembacaan air di *node* 2 pada hari ke 2 sebesar $0,1009 \text{ m}^3$. Waktu penggunaan air dimulai dari pukul 07:00 s.d. 09:00 dengan volume $0,0503 \text{ m}^3$, selanjutnya air tidak digunakan dan pembacaan tetap. Penggunaan air selanjutnya yang terbaca oleh sensor adalah pada pukul 16:00 s.d. 17:00 dimana volume air bertambah menjadi total pembacaan sebesar $0,1009 \text{ m}^3$. Hari ketiga *node* 2 yang digunakan untuk mengisi bak air 1 mulai membaca penggunaan air dari pukul 09:00 dengan penggunaan air sebesar $0,0612 \text{ m}^3$. Penggunaan air terus meningkat beberapa kali sampai akhirnya pada hari ke 3 penggunaan air pada *node* 2 mencapai $0,0988 \text{ m}^3$.

Hari keempat *node* 2 mulai membaca penggunaan air yang digunakan pada pukul 08:00 sebanyak $0,03 \text{ m}^3$ dan terus meningkat sampai pukul 10:00 menjadi $0,0544 \text{ m}^3$. Pukul 17:00 penggunaan air pada *node* 2 kembali meningkat menjadi $0,0817 \text{ m}^3$ sampai akhirnya penggunaan air yang terdeteksi oleh *node* 2 pada pukul 18:00 pada hari ke 4 sebanyak $0,0925 \text{ m}^3$ dan tetap sampai akhir hari ke 4. Hari kelima penggunaan air pada *node* 2 yang terbaca pada sensor dimulai pada pukul 07:00 s.d. 10:00 dengan volume terbaca sebesar $0,051 \text{ m}^3$. Nilai tetap sampai pada pukul 16:00 dimana sensor kembali mendeteksi penggunaan air. Volume total yang terbaca sampai pukul 23:00 pada hari kelima adalah $0,1232 \text{ m}^3$.

Penggunaan air yang digunakan pada hari keenam terbaca oleh sensor dimulai pada pukul 07:00 dengan jumlah air yang digunakan sebanyak $0,0176 \text{ m}^3$ dan meningkat sampai pukul 08:00 menjadi $0,0615 \text{ m}^3$. Siang hari bak mandi 1 tidak

digunakan karena pada pukul 09:00 s.d. 15:00 tidak ada perubahan dalam penggunaan air yang terbaca. Penggunaan kembali meningkat pada pukul 17:00 menjadi 1m^3 dan tetap sampai akhir hari ke 6. Hari ketujuh volume terbaca pada *node 2* adalah $0,1112\text{m}^3$. Penggunaan air pada *node 3* dapat dilihat pada Gambar 4.13.



Gambar 4.13 Grafik *Monitoring* Penggunaan Air *Node 3*

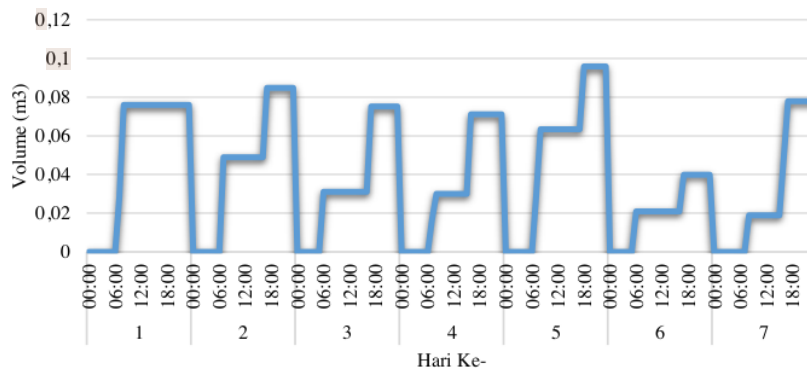
Gambar 4.13 merupakan grafik penggunaan air pada *node 3* atau saluran keran air 2. Pembacaan pada *node 3* hari pertama dimulai pada pukul 08:00 dengan volume terbaca yaitu $0,0274\text{ m}^3$. Nilai tersebut tetap sampai pukul 09:00 karena pada bak mandi 2 air tidak diisi kembali. Air mulai diisi kembali pada pukul 10:00 menjadi $0,0511\text{ m}^3$ dan pada 11:00 meningkat menjadi $0,0720\text{m}^3$. Rentang waktu berikutnya keran pada bak mandi 2 tidak dibuka lagi sehingga nilai pembacaan pada nilai yang sama.

Hari kedua penggunaan air pada *node 3* yang terbaca sensor pada pukul 09:00 dengan volume sebesar $0,0805\text{m}^3$. Nilai tersebut tidak berubah sampai pukul 23:00 karena pada bak mandi 2 air hanya diisi satu kali. Hari ketiga penggunaan air pada *node 3* yaitu sebesar $0,0838\text{m}^3$. Air mulai digunakan pagi hari dan terbaca oleh sensor pada pukul 08:00 dengan volume sebesar $0,0301\text{m}^3$. Nilai tersebut tetap karena air tidak digunakan sampai pada pukul 15:00. Volume kembali meningkat dimulai pada pukul 16:00 s.d. 17:00 sehingga volume terbaca menjadi $0,0838\text{m}^3$.

Hari keempat volume air yang terbaca adalah sebesar $0,0658\text{m}^3$. Data diterima pada pukul 08:00 sebesar $0,0281\text{m}^3$ dan bertambah pada pukul 17:00

menjadi $0,0658\text{m}^3$. Volume air pada hari kelima pada *node* 3 terbaca sebesar $0,0744\text{m}^3$. Data pertama kali masuk pada pukul 07:00 dengan volume penggunaan air sebesar $0,0205\text{m}^3$. Nilai tersebut kemudian bertambah menjadi $0,0322\text{m}^3$ pada pukul 08:00, dan kembali bertambah pada pukul 18:00 menjadi $0,0744\text{m}^3$.

Volume pembacaan air pada hari keenam sebesar $0,08\text{m}^3$ dan pada hari ketujuh sebesar $0,0618\text{m}^3$. Rata-rata waktu penggunaan air pada hari keenam dan ketujuh terjadi pada pukul 07:00 s.d. 08:00 dan 16:00 s.d. 18:00. Waktu ini merupakan saat dimana air digunakan untuk mengisi bak mandi 2. Penggunaan air pada *node* 4 dapat dilihat pada Gambar 4.14.



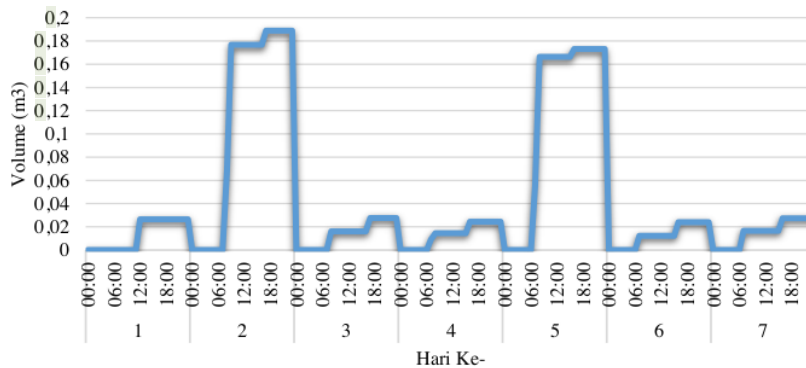
Gambar 4.14 Grafik *Monitoring* Penggunaan Air *Node* 4

Gambar 4.14 merupakan grafik penggunaan air pada *node* 4 atau saluran keran air 3. Volume air yang terbaca pada *node* 4 di hari pertama dimulai pada pukul 07:00, pada waktu tersebut keran pencuci piring sedang digunakan sehingga volume terukur yaitu $0,0281\text{m}^3$. Waktu yang dibutuhkan untuk mencuci piring adalah kurang lebih 1 jam dimulai dari 08:00 s.d. 09:00 dan dalam periode waktu tersebut total pembacaan yaitu $0,0760\text{m}^3$. Pembacaan tidak berubah karena pada hari itu cuci piring hanya dilakukan sekali.

Hari kedua penggunaan pada *node* 4 adalah $0,0847\text{m}^3$. Data masuk pertama pada pukul 07:00 sebesar $0,0489\text{m}^3$, kemudian data tidak berubah sampai pada pukul 17:00 nilai kembali berubah menjadi $0,0847\text{m}^3$. Pembacaan penggunaan air hari ketiga pada *node* 4 sudah mulai membaca penggunaan air sejak pukul 06:00. *Node* 4 yang digunakan untuk mencuci piring membaca penggunaan air sebanyak

0,031m³ dan bernilai tetap sampai pukul 16:00 sebelum akhirnya meningkat kembali pada pukul 17:00 menjadi 0,0752m³.

Hari keempat volume penggunaan air pada *node* 4 adalah 0,0711m³, hari kelima adalah 0,0959m³, hari keenam 0,0399m³, dan hari ketujuh 0,0778m³. Rata-rata waktu keran dibuka untuk mencuci piring pada *node* 4 adalah pagi hari pukul 07:00 dan sore hari pukul 17:00. Penggunaan paling tinggi untuk *node* 4 terjadi pada hari kelima dengan volume air 0,0959m³. Penggunaan air pada *node* 5 dapat dilihat pada Gambar 4.15.



Gambar 4.15 Grafik *Monitoring* Penggunaan Air *Node* 5

Gambar 4.15 merupakan grafik penggunaan air pada *node* 5 atau saluran keran air 4. Pembacaan pada *node* 5 dilakukan pada keran di taman. pembacaan dimulai pada pukul 12:00 untuk mengisi bak air yang berada pada di taman. Volume terbaca adalah 0,0265 m³. Nilai tersebut tidak bertambah karena pengisian bak pada taman hanya dilakukan pada pukul 12:00 s.d. 12:59. Penggunaan air pada *node* 5 di hari ke 2 mengalami kenaikan yang cukup tinggi. Hal ini disebabkan karena pada hari ke 2 *node* 5 selain digunakan untuk menyiram tanaman juga digunakan untuk mencuci baju. Kegiatan mencuci baju dimulai pada pukul 08:00 sampai pukul 09:00. Dapat dilihat pada grafik nilai penggunaan air pada pukul 08:00 hanya 0,0685m³ dan meningkat menjadi 0,1768m³ pada pukul 09:00.

Hari ketiga dan keempat *node* 5 tidak digunakan untuk mencuci piring dan hanya digunakan untuk menyiram tanaman sehingga penggunaan air tidak terlalu besar seperti hari sebelumnya. Volume yang terbaca pada hari ketiga untuk *node* 5

hanya $0,0275\text{m}^3$ dan $0,0245\text{m}^3$ pada hari keempat. Hari kelima *node 5* kembali digunakan untuk mencuci baju. Penggunaan air dimulai pada pukul 07:00 dengan penggunaan air sebanyak $0,0559\text{m}^3$. Penggunaan air terus berlanjut sampai pukul 08:00 sebanyak $0,1664\text{m}^3$. *Node 5* lalu digunakan untuk menyiram tanaman pada pukul 16:00 sehingga pada hari ke 5 total pembacaan air adalah $0,1732\text{m}^3$.

Hari keenam dan ketujuh penggunaan air pada *node 5* hanya $0,024\text{m}^3$ untuk hari keenam dan $0,0273\text{m}^3$ untuk hari ketujuh. Kedua hari tersebut *node 5* hanya digunakan untuk menyiram tanaman. Rata-rata waktu penggunaan air yang terbaca oleh *node 5* adalah pada pukul 07:00 dan 16:00. Volume tertinggi pada *node 5* yaitu $0,189\text{m}^3$ pada hari kedua.

Jumlah pembacaan penggunaan air dari hari ke 1 sampai hari ke 7 dapat dikumpulkan dalam 1 tabel. Biaya yang diperoleh merupakan hasil dari perhitungan jumlah air yang dipakai tiap *node* dikali dengan golongan rumah tangga seperti yang sudah dijelaskan pada Persamaan (2.1). Total pembacaan penggunaan air semua *node* selama satu minggu dapat dilihat pada Tabel 4.7.

Tabel 4.7 Monitoring Pembacaan Air 1 Minggu

Hari Ke-	Node 2		Node 3		Node 4		Node 5	
	Volume (m^3)	Biaya (Rp)	Volume (m^3)	Biaya (Rp)	Volume (m^3)	Biaya (Rp)	Volume (m^3)	Biaya (Rp)
1	0,0984	226,36	0,0720	165,57	0,0760	174,71	0,0265	60,91
2	0,1009	232,14	0,0805	185,15	0,0847	194,81	0,1890	434,70
3	0,0988	227,28	0,0838	192,74	0,0752	173,20	0,0275	63,25
4	0,0952	218,96	0,0658	151,40	0,0711	163,53	0,0245	56,35
5	0,1232	283,36	0,0744	171,17	0,0959	220,66	0,1732	398,36
6	0,1000	230,07	0,0800	184,07	0,0399	91,78	0,0240	55,24
7	0,1112	255,76	0,0618	142,14	0,0788	179,01	0,0273	62,79
Total	0,7277	1673,93	0,5183	1192,24	0,5216	1197,7	0,492	1131,6

Tabel 4.7 adalah total penggunaan air selama 1 minggu beserta biaya yang sudah dihitung secara otomatis oleh sistem. Total volume keseluruhan penggunaan air adalah $2,2596\text{m}^3$ untuk keseluruhan *node* dengan biaya yang harus dibayarkan Rp. 5195,47. Biaya yang dibayarkan dapat berubah tergantung dari golongan dan harga yang ditetapkan oleh pihak PDAM, harga dapat lebih rendah atau lebih tinggi. *Node 2* merupakan *node* dengan penggunaan air terbanyak sebesar $0,7277\text{m}^3$. Hal ini disebabkan karena setiap hari penggunaan air pada bak 1 adalah yang paling sering digunakan untuk kebutuhan sehari-hari seperti mandi, mencuci muka, dan lain-lain.

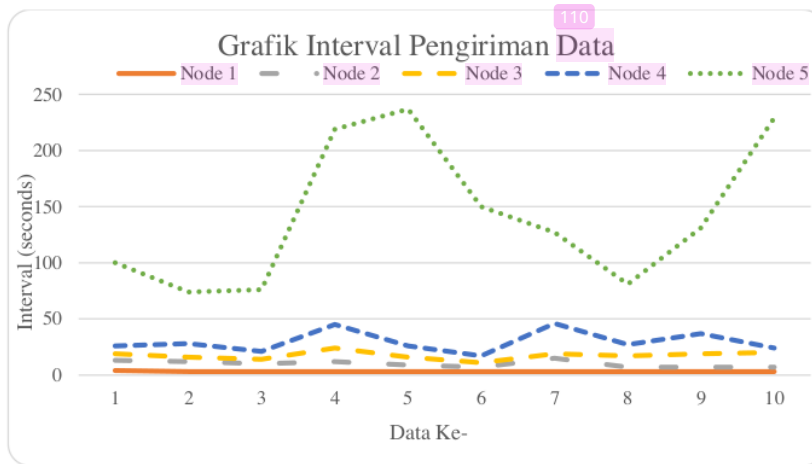
Pengiriman data tiap *node* memiliki interval waktu pengiriman yang berbeda-beda. Hal ini dikarenakan beberapa faktor diantaranya adalah jarak *node client* ke *node AP* dan *obstacle* pada jalur pengiriman data. Interval waktu pengiriman dapat dilihat pada Tabel 4.8.

Tabel 4.8 Interval Pengiriman Data *Node* Menuju *Database*

Node ID	Waktu			
	Running		Idle	
	Minimum	Maksimum	Minimum	Maksimum
1	3 detik	18 detik	3 detik	8 detik
2	4 detik	44 detik	7 detik	123 detik
3	4 detik	184 detik	3 detik	152 detik
4	4 detik	100 detik	3 detik	252 detik
5	6 detik	56 detik	46 detik	211 detik

Tabel 4.8 adalah *Interval delay* setiap *node* saat mengirimkan data menuju *server*. *Interval delay* bervariasi pada setiap *node* tergantung dari jarak dan halangan *node* terhadap *node AP*. Semakin jauh jarak *node client* dan semakin banyak halangannya maka *delay* akan semakin besar. Tabel 4.8 dapat dilihat bahwa terdapat 2 kondisi yaitu kondisi saat alat bekerja membaca *flow* air atau *running*, dan kondisi saat alat tidak membaca *flow* air atau *idle*. Waktu terlama pada saat kondisi *running* yaitu pada *node 3* dan saat kondisi *idle* yaitu pada *node 4*. *Node 1* memiliki interval pengiriman data pada saat *running* tercepat yaitu 3 detik dengan *delay* terlamanya adalah 18 detik, sedangkan pada saat *idle* pengiriman data tercepat adalah 3 detik sedangkan *delay* 8 detik. *Node 2* memiliki interval pengiriman data pada saat *running* tercepat yaitu 4 detik dengan *delay* terlamanya adalah 44 detik, sedangkan pada saat *idle* pengiriman data tercepat adalah 7 detik sedangkan *delay* 123 detik. *Node 3* memiliki interval pengiriman data pada saat *running* tercepat yaitu 4 detik dengan *delay* terlamanya adalah 184 detik, sedangkan pada saat *idle* pengiriman data tercepat adalah 3 detik sedangkan *delay* 152 detik. *Node 4* memiliki interval pengiriman data pada saat *running* tercepat yaitu 4 detik dengan *delay* terlamanya adalah 100 detik, sedangkan pada saat *idle* pengiriman data tercepat adalah 3 detik sedangkan *delay* 252 detik. *Node 5* memiliki interval pengiriman data pada saat *running* tercepat yaitu 6 detik dengan *delay* terlamanya adalah 56 detik, sedangkan pada saat *idle* pengiriman data tercepat adalah 46 detik

sedangkan *delay* 211 detik. Berdasarkan 10 data terakhir pengiriman dapat dibentuk grafik yang dapat dilihat pada Gambar 4.16.



Gambar 4.16 Grafik Interval Pengiriman Data

Gambar 4.16 adalah grafik interval pengiriman data tiap *node client* ke *node AP*. Grafik terdiri dari keterangan waktu pada bagian vertikal dan keterangan jumlah data yang diterima pada sumbu horizontal. Data yang terdapat pada grafik adalah *record* 10 data terakhir dari 5 *node* yang dikirim menuju *server*. Rentang waktu yang berbeda dalam pengiriman data antar *node* dapat disebabkan oleh jarak dan penghalang ke *server*. Faktor lain yang mempengaruhi adalah kualitas daya yang diperoleh dari baterai. Semakin lemah baterai maka performa keseluruhan alat pun menurun, sehingga dapat menyebabkan alat menjadi kurang responsif dan memberikan dampak peningkatan *delay* pada saat mengirim data dari *flow sensor* ke *database*.

PENUTUP**5.1 Kesimpulan**

Kesimpulan yang dapat diambil dari penelitian ini adalah sebagai berikut.

1. Sistem *monitoring water meter* berjalan dengan baik dengan tingkat akurasi yang tinggi dengan *error* hanya 0,56% dan seluruh *node* sudah saling terhubung dan dapat berkomunikasi dengan baik dengan menggunakan topologi *mesh*.
2. Sistem *monitoring water meter* dapat diimplementasikan dalam kehidupan sehari-hari dengan membaca penggunaan air selama 24 jam tanpa henti selama 1 minggu dan hasil pembacaan dapat dilihat melalui *website*. Selama 1 minggu *monitoring* penggunaan air rata-rata digunakan pada pukul 07:00 s.d. 09:00 pagi dan pukul 15:00 s.d. 17:00 sore.

5.2 Saran

Saran yang dapat diberikan untuk penelitian selanjutnya adalah penambahan sistem *login* pada *website* agar pembacaan penggunaan air setiap *user* lebih mudah karena saat memasuki *website user* langsung dapat melihat penggunaan air miliknya tanpa perlu mencari diantara penggunaan air milik *user* lain.

DAFTAR PUSTAKA

- [1] Purba, J. T. and S. Budiono, "Fixed Effects Modeling: Strategi Pembangunan Sumber Daya Manusia dengan Ketercukupan Air Bersih di Indonesia", *Proceeding on: 3rd NCBMA Business Knowledge to be Elevated: Advancing Disruptive Innovation*, 2019.
- [2] Vliet, M. T. H. V., E. R. Jones, M. Florke, W. H. P. Franssen, N. Hanasaki, Y. Wada and J. R. Yearsley, "Global water scarcity including surface water quality and expansions of clean water technologies", *Environmental Research, lett.* 16, 2021.
- [3] Suryani, A. S., "Pembangunan Air Bersih dan Sanitasi saat Pandemi Covid-19", *Aspirasi: Jurnal Masalah-Masalah Sosial*, vol.11, no.2, pp. 199-214, 2020.
- [4] Budiono, S. and J. T. Purba, "MEMERANGIKEMISKINAN MELALUI PARTISIPASI PENDIDIKANDAN AIR BERSIHPADA KABUPATEN DAN KOTA DI INDONESIA", *Media Ekonomi*, vol.28, no. 1, pp. 67-78, 2020.
- [5] Matyakubov, B., I. Begmatov, I. Raimova and G. Teplova, "Factors for the efficient use of water distribution facilities", *IOP Conf. Series: Materials Science and Engineering*, 2020
- [6] Hudiyah, D. B. M. and S. K. Sapmoto, "Analisis Kualitas Air pada Jalur Distribusi Air Bersih di Gedung Baru Fakultas Ekonomi dan Manajemen Institut Pertanian Bogor", *Jurnal Teknik Sipil dan Lingkungan*, vol. 04, no. 01, pp. 14-24, 2019.
- [7] Ramdani, I G.P.W. Wedashwara and A. Zubaidi, "Rancang Bangun *Smart Meter System* untuk Penggunaan Air pada Rumah Tangga Berbasis *Internet of Things*" *J-Cosine*, vol. 4, no. 2, pp. 149-160, 2020.
- [8] Randall, T. and R. Koech, "Smart Water Metering Technology for Water Management in Urban Areas", *Water e-Journal*, vol. 4, no. 1, 2019.
- [9] Sushma, N., H. N. Suresh, J. M. Lakshmi, P. N. Srinivasu and A. K. Bhoi and P. Barsocchi, "A Unified Metering System Deployed for Water and Energy Monitoring in Smart City", *IEEE Access*, vol. 13, pp. 80429-80447, 2023.

- [10] Khalaf, O. I. and B. M. Sabbar, "An overview on wireless sensor networks and finding optimal location of nodes", *Periodical of Engineering and Natural Sciences*, vol. 7, no. 3, pp. 1096-1101, 2019.
- [11] Nurkamid, M. and A. Widodo, "Penerapan *Wireless Sensor Network* Untuk *Monitoring* Lingkungan Menggunakan Modul ESP-WROOM32", *Jurnal IKRA-ITH Informatika*, vol. 5, no. 3, pp. 72-78, 2021.
- [12] Santos, E. L. Q., J. C. D. Bautista, M. W. S. Estanque and C. J. B. Paloa, "Development of an IoT-based water and power monitoring system for residential building", *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 2, pp. 744-751, 2021.
- [13] Hsia, S. C., S. H. Wang and S. W. Hsu, "Smart Water-Meter Wireless Transmission System for Smart Cities", *IEEE Consumer Electronics Magazine*, vol. 10, no.6, pp. 83-88, 2021.
- [14] Sutaya, I W., K. U. Ariawan and D. G. H. Divayana, "Implementasi Jaringan WSN Mesh Berbasis Radio Wireless nRF24101 Pada Sistem Meteran Air Rumah Kos", *Jurnal EECCIS*, vol. 13, no. 1, pp. 11-16, 2019.
- [15] Bouali, E. T., M. R. Abid, E. M. Boufounas, T. Abu and D. Benhaddou, "Renewable Energy Integration into Cloud & IoT-based Smart Agriculture", *IEEE Access*, vol. 10, pp. 1175-1191, 2022.
- [16] Pratiwi, Y. A., R. U. Ginting, H. Situmorang and R. Sitanggang, "Perancangan Sistem Informasi Akademik Berbasis Web Di SMP Rahmat Islamiyah", *Jurnal TEKESNOS*, vol. 2, no. 1, pp. 27-32, 2020.
- [17] Megawaty, D. A., M. Bakri, Setiawansyah and E. Damayanti, "Sistem *Monitoring* Kegiatan Akademik Siswa Menggunakan *Website*", *Jurnal TEKNOKOMPAK*, vol. 14, no. 2, pp. 98-101, 2020.
- [18] Wiryadani, W. and H. Mistialustina, "Aplikasi Penampil Data Hasil *Monitoring* Suhu Dan Kelembaban Ruangan Pada *Wireless Sensor Network*", *Jurnal Infotronik*, vol. 5, no. 1, pp. 24-36, 2020.
- [19] Permana, I M. R., M. Abdurrohman and A. G. Putrada, "Comparative Analysis of Mesh and Star Topologies in Improving Smart Fire Alarms", *2019 Fourth International Conference on Informatics and Computing (ICIC)*, Semarang, 2019.

- [20] Anjari, L. and A. H. S. Budi, “*The Development of Smart Parking System based on NodeMCU 1.0 using the Internet of Things*”, *IOP Conf. Series: Materials Science and Engineering*, 2018.
- [21] Butar-Butar, C. M. G. and Y. T. Samuel, “Perancangan Sistem Kendali Kendaraan Bermotor Jarak Jauh Menggunakan NodeMCU ESP8266”, *TeIka Jurnal Teknologi Informasi dan Komunikasi*, vol. 9, no. 1, pp. 87-97, 2019.
- [22] Aditomo, A. W. P., “Penentuan Harga Pokok Produk Air Pada PDAM Tirta Merapi Kabupaten Klaten”, Skripsi, Fakultas Bisnis dan Ekonomi Universitas Atma Jaya Yogyakarta, 2020.
- [23] Hadi, A. P. and F. A. Rokhman, “Implementasi *Website* Sebagai Media Informasi Dan Promosi Pada Pondok Pesantren Putra-Putri Addainuriyah 2 Semarang”, *Jurnal Ilmiah Komputer Grafis*, vol. 13, no. 1, pp. 39-49, 2020.
- [24] Hasan, S. and N. Muhammad, “Sistem Informasi Pembayaran Biaya Studi Berbasis *Web* Pada Politeknik Sains dan Teknologi Wiratama Maluku Utara”, *Indonesian Journal on Information System*, vol. 5, no. 1, pp. 44-55, 2020.
- [25] Astuti, F. K. and D. S. Agustina, “Membangun *Website* MTS Negeri 01 OKU Timur Menggunakan Php dan Mysql”, *Jurnal Informatika dan Komputer (JIK)*, vol. 13, no. 1, hal. 7-14, 2022.
- [26] Ramadhan, R. F. and R. Mukhaiyar, “Penggunaan *Database* Mysql dengan *Interface* PhpMyAdmin sebagai Pengontrolan *Smarthome* Berbasis Raspberry Pi”, *JTEIN: Jurnal Teknik Elektro Indonesia*, vol. 1, no. 2, pp. 129-134, 2020.
- [27] Siallagan, T. F. P. and D. Wisnu, “Rancang Bangun Sistem Pengidentifikasi *Travel Bag* Pada Kelompok Biro Perjalanan Umroh/Haji Berbasis *Web*”, *Jurnal Teknologi Informasi dan Komunikasi*, vol. 13, no. 1, pp. 26-40, 2020.
- [28] Ramadhan, A. B., S. Sumaryo and R. A. Priramdhi, “Desain dan Implementasi Pengukuran Debit Air Menggunakan Sensor *Water Flow* Berbasis IoT”, *e-Proceeding of Engineering*, vol. 6, no. 2, pp. 2623-2630, 2019.
- [29] Suryantoro, H. and A. Budiyanto, “*Prototype* Sistem *Monitoring Level* Air Berbasis *Labview* & *Arduino* Sebagai Sarana Pendukung Praktikum

Instrumentasi Sistem Kendali”, *Indonesian Journal of Laboratory*, vol. 1, no. 3, pp. 20-32, 2019.

- [30] Setiawan, A. and H. S. Pramono, “Pengembangan Media Pembelajaran Mikrokontroler Antarmuka I²C Di SMK Negeri 2 Pengasih”, *Jurnal Pendidikan Teknik Mekatronika*, vol. 8, no. 6, pp. 479-489, 2018.

LAMPIRAN

LAMPIRAN A DOKUMENTASI PENGUJIAN



Gambar 1 Tampilan LCD Kalibrasi Flowsensor



Gambar 2 Tampilan botol air 500ml Kalibrasi Flowsensor



Gambar 3 Tampilan Kondisi Awal Meteran Konvensional Saat Kalibrasi



Gambar 4 Tampilan Kondisi Akhir Meteran Konvensional Saat Kalibrasi



Gambar 5 Tampilan Botol Air 500mL Kalibrasi Meteran Konvensional

11:39:31.560 -> Flow Rate: 0.17 L/min

Gambar 6 Kecepatan Terbaca Paling Rendah

```
COM3
16:15:34.223 -> 0.0000 L
16:15:34.223 -> Changed connections
16:15:34.223 -> New Connection, nodeId = 1293581979
16:15:34.307 -> Received from 1293581979 msg={"node":3,"vol":0.0008453046903014183}
16:15:34.307 -> Node: 3
16:15:34.307 -> Volume: 0.00
16:15:35.260 -> {"Node":"nodeNumber","Volume":TOTAL}Node: 1
16:15:35.260 -> Volume: 0.00
16:15:37.249 -> VOL. :0.00
16:15:37.249 -> TOTAL:
16:15:37.249 -> 0.0000 L
```

Gambar 7 Pengujian Jaringan Kondisi 1

```
COM3
17:08:08.082 -> Changed connections
17:08:08.082 -> New Connection, NodeId = 1298103367
17:08:11.128 -> Message received from 1298103367
17:08:11.128 -> Node: 2 Volume: 0.00000
17:08:14.127 -> Adjusted time 163184434. Offset = 5184043
17:08:17.173 -> Message received from 1298103367
17:08:17.173 -> Node: 2 Volume: 0.00000
17:08:20.173 -> Message received from 1293581979
17:08:20.220 -> Node: 3 Volume: 0.00000
17:08:26.265 -> Adjusted time 175315887. Offset = 9134
17:08:29.264 -> Message received from 2229147199
17:08:29.311 -> Node: 4 Volume: 0.00000
17:08:32.310 -> Message received from 1298103367
17:08:32.310 -> Node: 2 Volume: 0.00000
17:08:35.357 -> Message received from 2229147199
17:08:35.357 -> Node: 4 Volume: 0.00000
 Autoscroll  Show timestamp
```

Gambar 8 Pengujian Jaringan Kondisi 2

LAMPIRAN B TABEL HASIL PENGUJIAN

Lampiran B.1 Tabel 1 *Monitoring* Penggunaan Air Hari Ke-1

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0	0	0	0	0.0281	64.66	0	0
9	08:00	0.0432	99.45	0.0274	62.91	0.0760	174.71	0	0
10	09:00	0.0432	99.45	0.0274	62.91	0.0760	174.71	0	0
11	10:00	0.0693	159.31	0.0511	117.63	0.0760	174.71	0	0
12	11:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0	0
13	12:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
14	13:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
15	14:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
16	15:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
17	16:00	0.0693	159.31	0.0720	165.57	0.0760	174.71	0.0265	60.91
18	17:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
19	18:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
20	19:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
21	20:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
22	21:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
23	22:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91
24	23:00	0.0984	226.36	0.0720	165.57	0.0760	174.71	0.0265	60.91

Lampiran B.2 Tabel 2 *Monitoring* Penggunaan Air Hari Ke-2

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0099	22.77	0	0	0.0489	112.51	0	0
9	08:00	0.0503	115.71	0	0	0.0489	112.51	0.0685	157.55
10	09:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
11	10:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
12	11:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
13	12:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
14	13:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
15	14:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68

16	15:00	0.0503	115.71	0.0805	185.15	0.0489	112.51	0.1768	406.68
17	16:00	0.0847	194.81	0.0805	185.15	0.0489	112.51	0.1768	406.68
18	17:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
19	18:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
20	19:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
21	20:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
22	21:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
23	22:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70
24	23:00	0.1009	232.14	0.0805	185.15	0.0847	194.81	0.1890	434.70

Lampiran B.3 Tabel 3 *Monitoring Penggunaan Air Hari Ke-3*

No.	Waktu	Node ID							
		43 Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0.031	71.3	0	0
8	07:00	0	0	0	0	0.031	71.3	0	0
9	08:00	0.0235	54.15	0.0301	69.23	0.031	71.3	0.0161	37.03
10	09:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
11	10:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
12	11:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
13	12:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
14	13:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
15	14:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
16	15:00	0.0612	140.76	0.0301	69.23	0.031	71.3	0.0161	37.03
17	16:00	0.0740	170.26	0.0493	113.39	0.031	71.3	0.0161	37.03
18	17:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
19	18:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
20	19:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
21	20:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
22	21:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
23	22:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25
24	23:00	0.0988	227.28	0.0838	192.74	0.0752	173.2	0.0275	63.25

Lampiran B.4 Tabel 4 *Monitoring Penggunaan Air Hari Ke-4*

No.	Waktu	Node ID							
		8 Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0144	33.12	0	0	0.0164	37.72	0.009	20.7

9	08:00	0.03	69.09	0.0281	64.63	0.03	69.09	0.0145	33.35
10	09:00	0.0422	97.1	0.0281	64.63	0.03	69.09	0.0145	33.35
11	10:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
12	11:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
13	12:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
14	13:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
15	14:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
16	15:00	0.0544	127.46	0.0281	64.63	0.03	69.09	0.0145	33.35
17	16:00	0.0544	127.46	0.0281	64.63	0.0711	163.53	0.0245	56.35
18	17:00	0.0817	187.91	0.0658	151.4	0.0711	163.53	0.0245	56.35
19	18:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
20	19:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
21	20:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
22	21:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
23	22:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35
24	23:00	0.0925	212.81	0.0658	151.4	0.0711	163.53	0.0245	56.35

Lampiran B.5 Tabel 5 *Monitoring Penggunaan Air Hari Ke-5*

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0041	9.43	0.0205	47.15	0.0288	66.24	0.0559	128.66
9	08:00	0.0127	29.21	0.0322	74.06	0.0634	145.82	0.1664	382.72
10	09:00	0.0322	74.08	0.0322	74.06	0.0634	145.82	0.1664	382.72
11	10:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
12	11:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
13	12:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
14	13:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
15	14:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
16	15:00	0.051	117.3	0.0322	74.06	0.0634	145.82	0.1664	382.72
17	16:00	0.0657	151.79	0.0322	74.06	0.0634	145.82	0.1732	398.36
18	17:00	0.0961	221.03	0.0322	74.06	0.0634	145.82	0.1732	398.36
19	18:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
20	19:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
21	20:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
22	21:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
23	22:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36
24	23:00	0.1232	283.26	0.0744	171.17	0.0959	220.66	0.1732	398.36

Lampiran B.6 Tabel 6 *Monitoring Penggunaan Air Hari Ke-6*

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0

2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0.0209	48.07	0	0
8	07:00	0.0176	40.48	0.0223	51.29	0.0209	48.07	0.0122	28.15
9	08:00	0.0615	141.45	0.0365	83.95	0.0209	48.07	0.0122	28.15
10	09:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
11	10:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
12	11:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
13	12:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
14	13:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
15	14:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
16	15:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0122	28.15
17	16:00	0.0615	141.45	0.0447	102.94	0.0209	48.07	0.0240	55.24
18	17:00	0.1000	230.07	0.0703	161.69	0.0399	91.78	0.0240	55.24
19	18:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
20	19:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
21	20:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
22	21:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
23	22:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24
24	23:00	0.1000	230.07	0.0800	184.07	0.0399	91.78	0.0240	55.24

Lampiran B.7 Tabel 7 *Monitoring Penggunaan Air Hari Ke-7*

No.	Waktu	Node ID							
		Node 2		Node 3		Node 4		Node 5	
		Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)	Volume (m ³)	Biaya (Rp)
1	00:00	0	0	0	0	0	0	0	0
2	01:00	0	0	0	0	0	0	0	0
3	02:00	0	0	0	0	0	0	0	0
4	03:00	0	0	0	0	0	0	0	0
5	04:00	0	0	0	0	0	0	0	0
6	05:00	0	0	0	0	0	0	0	0
7	06:00	0	0	0	0	0	0	0	0
8	07:00	0.0154	35.42	0.0208	47.84	0	0	0.0166	38.18
9	08:00	0.0444	102.19	0.0299	68.77	0.0189	43.47	0.0166	38.18
10	09:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
11	10:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
12	11:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
13	12:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
14	13:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
15	14:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
16	15:00	0.0628	144.48	0.0299	68.77	0.0189	43.47	0.0166	38.18
17	16:00	0.0628	144.48	0.0299	68.77	0.0473	108.84	0.0273	62.79
18	17:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
19	18:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
20	19:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
21	20:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
22	21:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
23	22:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79
24	23:00	0.1112	255.76	0.0618	142.14	0.0778	179.01	0.0273	62.79

LAMPIRAN C LISTING PROGRAM

1. Listing program Water Meter

a. Listing Program Menghubungkan Node Dengan Node Lainnya

```
#include <painlessMesh.h>
#include <ESPAsyncTCP.h>
#include <Arduino_JSON.h>
#include <ESP8266HTTPClient.h>

// Define untuk koneksi dan port jaringan -----
-----
7 #define MESH_PREFIX "smartwatermeter"
#define MESH_PASSWORD "123456789"
#define MESH_PORT 80

7 //String to send to other nodes with sensor readings
String readings;

Scheduler userScheduler; // to control your personal task
painlessMesh mesh;

// User stub
void sendMessage(); // Prototype so PlatformIO doesn't
complain
String getReadings(); // Prototype for sending sensor
readings

//Create tasks: to send messages and get readings;
Task taskSendMessage(TASK_SECOND * 60 , TASK_FOREVER,
&sendMessage);

String getReadings () {
  JSONVar jsonReadings;
  jsonReadings["node_id"] = nodeNumber;
  jsonReadings["volume"]= TOTAL;
  jsonReadings["cost"]= cost;
4 readings = JSON.stringify(jsonReadings);
return readings;
}
```

```

void sendMessage () {
    String msg = getReadings();
    mesh.sendBroadcast(msg);
}

//Needed for painless library
void receivedCallback( uint32_t from, String &msg ) {
    Serial.printf("Message received from %u\n", from,
msg.c_str());
    String nodemcuData = msg;
    JSONVar myObject = JSON.parse(msg.c_str());
    int node_id = myObject["node_id"];
    double volume = myObject["volume"];
    double cost = myObject["cost"];
    Serial.print("Node: ");
    Serial.print(node_id);
    Serial.print(" Volume: ");
    Serial.println(volume,5);
    Serial.print("Biaya: ");
    Serial.println(cost);
void newConnectionCallback(uint32_t nodeId) {
    Serial.printf("New Connection, NodeId = %u\n", nodeId);
}

void changedConnectionCallback() {
    Serial.printf("Changed connections \n");
}

void nodeTimeAdjustedCallback(int32_t offset) {
    Serial.printf("Adjusted time %u. Offset = %d\n",
mesh.getNodeTime(),offset);
}

void setup(){
    Serial.begin(115200);
    Wire.begin(D1,D2);
    //mesh.setDebugMsgTypes( ERROR | MESH_STATUS | CONNECTION
| SYNC | COMMUNICATION | GENERAL | MSG_TYPES | REMOTE ); //
all types on

```

```

    mesh.setDebugMsgTypes( ERROR | STARTUP | CONNECTION ); //
    set before init() so that you can see startup messages

    // Channel set to 6. Make sure to use the same channel for
    your mesh and for you other
    // network (STATION_SSID)
    mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler,
    MESH_PORT);
    mesh.onReceive(&receivedCallback);
    mesh.onNewConnection(&newConnectionCallback);
    mesh.onChangedConnections(&changedConnectionCallback);
    mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);

    userScheduler.addTask(taskSendMessage);
    taskSendMessage.enable();

    pinMode(input, INPUT);
    lcd.begin();
    lcd.backlight();
}

void loop(){
    mesh.update();
    delay(1000)
}

```

b. Listing Program Pembacaan Debit Air Dan Menampilkan Data Pada LCD I2C 16x2

```

#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <EEPROM.h>
int nodeNumber = 1;

LiquidCrystal_I2C lcd(0x27, 16, 2); //biru pakai 27 hijau 3f

const int input = D6;
int addr = 0;
byte sensorInterrupt = 0;

```

```

float calibrationFactor = 5.5;
volatile byte pulseCount;

float flowRate, liter, TOTAL, cost;

unsigned long oldTime;
unsigned long prevTime= 0;
unsigned long seconds=0;

//biaya tergantung kelompok rumah tangga (berdasarkan daerah
tangerang)
//R1 = 1875; R2 = 2300; R3 = 2800; R4 = 3350;
int R2 = 2300;

void setup(){
  Serial.begin(115200);
  Wire.begin(D1, D2);

  pinMode(input, INPUT);
  pulseCount = 0;
  flowRate = 0;
  liter = 0;
  TOTAL = 0;
  oldTime = 0;
  cost = 0;

  lcd.begin();
  lcd.backlight();

  digitalWrite(input, HIGH);

  attachInterrupt (digitalPinToInterrupt (input), pulseCounter, RISING);
}

void loop()
{
  //Reset Data Tiap Hari (24jam)
  unsigned long currentTime = millis();

```

```

if ((currentTime-prevTime) >= 1000){
    seconds++;
    Serial.print("Waktu: ");
    Serial.println(seconds);
    prevTime = currentTime;
}
1 if (millis()-oldTime>1000){
    detachInterrupt(sensorInterrupt);

    flowRate = ((1000.0/(millis()-
oldTime))*pulseCount)/calibrationFactor;

    oldTime = millis();

    liter += flowRate/60;
    TOTAL = liter/1000;
    cost = R2*TOTAL;

    unsigned int frac;

    Serial.print("Node Id: ");
    Serial.println(nodeNumber);

    Serial.print("Flow Rate: ");
    Serial.print(int(flowRate));
    Serial.print(".");

    frac = (flowRate - int(flowRate))*100;
    Serial.print(frac,DEC);
    Serial.println(" L/min");

    Serial.print("Output Total Quantity: ");
    Serial.print(TOTAL,4);
    Serial.println(" m^3" );

    Serial.print("Biaya: Rp.");
    Serial.println(cost);
    Serial.print("\n");

```

```

lcd.setCursor(0,0);
lcd.print("Speed: ");
lcd.print(flowRate,2);
lcd.setCursor(13,0);
lcd.print("L/M");
lcd.setCursor(0,1);
lcd.print("Total: ");
lcd.setCursor(6,1);
lcd.print(TOTAL,4);
lcd.setCursor(13,1);
lcd.print("m^3");

pulseCount = 0;

attachInterrupt(sensorInterrupt, pulseCounter,
FALLING);
}
if (seconds >= 86400){
    seconds = 0;
    TOTAL = 0;
    cost = 0;
    lcd.clear();
}
}

void pulseCounter(){
    pulseCount++;
}

```

c. Listing Program Pengiriman Data Pembacaan Ke Server

```

#include <ESP8266HTTPClient.h>
String url =
"http://10.249.193.100/monitoring/nodemcu_log/webapi/api/cre
ate.php";

StaticJsonDocument<1024> doc;

```

```

String nodemcuData;
int node_id = nodeNumber;
double volume = TOTAL;
double costs = cost;

doc["node_id"] = String(node_id);
doc["volume"] = String(volume,4);
doc["cost"] = String(costs);

2
HTTPClient http;
http.begin(url);
http.addHeader("Content-Type", "application/json");

serializeJson(doc, nodemcuData);
Serial.print("POST data >> ");
Serial.println(nodemcuData);

int httpCode = http.POST(nodemcuData); //Send the
request
String payload;
if (httpCode > 0) { //Check the returning code
    payload = http.getString(); //Get the request
response payload
    payload.trim();
    if( payload.length() > 0 ){
        Serial.println(payload + "\n");
    }
}
http.end();
}

```

d. Listing Program Pengujian Keseluruhan

```

#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <EEPROM.h>
#include <painlessMesh.h>
#include <ESPAsyncTCP.h>
#include <Arduino_JSON.h>
#include <ESP8266HTTPClient.h>

```

```

// Define untuk koneksi dan port jaringan -----
-----
#define MESH_PREFIX "smartwatermeter"
#define MESH_PASSWORD "123456789"
#define MESH_PORT 80

//#define STATION_SSID "Redmin7"
//#define STATION_PASSWORD "12345567"
//#define STATION_PORT 80
//uint8_t station_ip[4] = {192,168,43,1}; //ip server

//node 1=10.249.193.100
//node 2=10.124.71.100
//node 3=10.126.155.100
//node 4=10.22.63.100
//node 5=10.104.253.100
String url =
"http://10.249.193.100/monitoring/nodemcu_log/webapi/api/create.php";
//Number for this node
int nodeNumber = 1;

LiquidCrystal_I2C lcd(0x27, 16, 2); //biru pakai 27 hijau 3f

const int input = D6;
int addr = 0;
byte sensorInterrupt = 0;

float calibrationFactor = 5.5;
volatile byte pulseCount;

float flowRate, liter, TOTAL, cost;
unsigned int flowMilliLitres;
unsigned long totalMilliLitres;

unsigned long oldTime;
unsigned long prevTime= 0;
unsigned long seconds=0;

```



```

//biaya tergantung kelompok rumah tangga (berdasarkan daerah
tangerang)
//R1 = 1875; R2 = 2300; R3 = 2800; R4 = 3350;
int R2 = 2300;

7
//String to send to other nodes with sensor readings
String readings;

Scheduler userScheduler; // to control your personal task
painlessMesh mesh;

// User stub
void sendMessage(); // Prototype so PlatformIO doesn't
complain
String getReadings(); // Prototype for sending sensor
readings

//Create tasks: to send messages and get readings;
Task taskSendMessage(TASK_SECOND * 5, TASK_FOREVER,
&sendMessage);

String getReadings () {
  JSONVar jsonReadings;
  jsonReadings["node_id"] = nodeNumber;
  jsonReadings["volume"]= TOTAL;
  jsonReadings["cost"]= cost;
7
  readings = JSON.stringify(jsonReadings);
  return readings;
}

void sendMessage () {
  String msg = getReadings();
  mesh.sendBroadcast(msg);
}

```

```

47 //Needed for painless library
void receivedCallback( uint32_t from, String &msg ) {
    Serial.printf("Message received from %u\n", from,
msg.c_str());
    String nodemcuData1 = msg;
    JSONVar myObject = JSON.parse(msg.c_str());
    int node_id = myObject["node_id"];
    double volume = myObject["volume"];
    double cost = myObject["cost"];
    Serial.print("Node: ");
    Serial.print(node_id);
    Serial.print(" Volume: ");
    Serial.println(volume,5);
    Serial.print("Biaya: ");
    Serial.println(cost);

2 HTTPClient http;
http.begin(url);
http.addHeader("Content-Type", "application/json");

Serial.print("POST data >> ");
Serial.println(nodemcuData1);

int httpCode1 = http.POST(nodemcuData1);//Send the request
String payload1;
if (httpCode1 > 0) { //Check the returning code
    payload1 = http.getString(); //Get the request
response payload
    payload1.trim();
    if( payload1.length() > 0 ){
        Serial.println(payload1 + "\n");
    }
}
}

51 void newConnectionCallback(uint32_t nodeId) {
    Serial.printf("New Connection, NodeId = %u\n", nodeId);
}

```

```

void changedConnectionCallback() {
    Serial.printf("Changed connections \n");
}

void nodeTimeAdjustedCallback(int32_t offset) {
    Serial.printf("Adjusted time %u. Offset = %d\n",
mesh.getNodeTime(),offset);
}

void setup(){
    Serial.begin(115200);
    Wire.begin(D1,D2);
    //mesh.setDebugMsgTypes( ERROR | MESH_STATUS | CONNECTION
| SYNC | COMMUNICATION | GENERAL | MSG_TYPES | REMOTE ); //
all types on
    mesh.setDebugMsgTypes( ERROR | STARTUP | CONNECTION ); //
set before init() so that you can see startup messages

    // Channel set to 6. Make sure to use the same channel for
your mesh and for you other
    // network (STATION_SSID)
    mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler,
MESH_PORT);
    mesh.onReceive(&receivedCallback);
    mesh.onNewConnection(&newConnectionCallback);
    mesh.onChangedConnections(&changedConnectionCallback);
    mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);

    //khusus untuk node bridge
    //mesh.initOTAReceive("bridge");
    //mesh.stationManual(STATION_SSID, STATION_PASSWORD,
STATION_PORT, station_ip);
    //mesh.setRoot(true);
    //mesh.setContainsRoot(true);

    userScheduler.addTask(taskSendMessage);
    taskSendMessage.enable();
    delay(10);
}

```

```

pinMode(input, INPUT);
101 pulseCount = 0;
flowRate = 0;
flowMilliLitres = 0;
totalMilliLitres = 0;
liter = 0;
TOTAL = 0;
oldTime = 0;
cost = 0;

lcd.begin();
lcd.backlight();

digitalWrite(input, HIGH);

attachInterrupt (digitalPinToInterrupt (input), pulseCounter, RISING);
}

4 void loop()
{
  unsigned long currentTime = millis();
  if ((currentTime-prevTime) >= 1000) {
    seconds++;
    Serial.print("Waktu: ");
    Serial.println(seconds);
    prevTime = currentTime;
  }

1  if (millis()-oldTime>1000){
    detachInterrupt (sensorInterrupt);

    flowRate = ((1000.0/(millis()-oldTime))*pulseCount)/calibrationFactor;

    oldTime = millis();

    liter += flowRate/60;
    TOTAL = liter/1000;
  }
}

```

```

cost = R2*TOTAL;

85 unsigned int frac;

Serial.print("Node Id: ");
Serial.println(nodeNumber);

4 Serial.print("Flow Rate: ");
Serial.print(int(flowRate));
Serial.print(".");

frac = (flowRate - int(flowRate))*100;
Serial.print(frac,DEC);
Serial.println(" L/min");

Serial.print("Output Total Quantity: ");
Serial.print(TOTAL,4);
Serial.println(" m^3");

Serial.print("Biaya: Rp.");
Serial.println(cost);
Serial.print("\n");

1 lcd.setCursor(0,0);
lcd.print("Speed: ");
lcd.print(flowRate,2);
lcd.setCursor(13,0);
lcd.print("L/M");
lcd.setCursor(0,1);
lcd.print("Total: ");
lcd.setCursor(6,1);
lcd.print(TOTAL,4);
lcd.setCursor(13,1);
lcd.print("m^3");

pulseCount = 0;

attachInterrupt(sensorInterrupt, pulseCounter, FALLING);
}

```

```

//Gunakan listing program dibawah jika node AP juga akan
digunakan untk pembacaan air (node ap akan mengirimkan data
pembacaan air miliknya sendiri ke server)//
//StaticJsonDocument<1024> doc;
//String nodemcuData;
//int node_id = nodeNumber;
//double volume = TOTAL;
//double costs = cost;

//doc["node_id"] = String(node_id);
//doc["volume"] = String(volume,4);
//doc["cost"] = String(costs);

2 //HTTPClient http;
//http.begin(url);
//http.addHeader("Content-Type", "application/json");

//serializeJson(doc, nodemcuData);
//Serial.print("POST data >> ");
//Serial.println(nodemcuData);

//int httpCode = http.POST(nodemcuData);//Send the request
//String payload;
//if (httpCode > 0) { //Check the returning code
// payload = http.getString(); //Get the request response
payload
// payload.trim();
// if( payload.length() > 0 ){
//   Serial.println(payload + "\n");
// }
// }
//http.end();

mesh.update();
delay(1000);
if (seconds >= 86400){
seconds = 0;
35 pulseCount = 0;
}

```

```

        flowRate = 0;
        liter = 0;
        TOTAL = 0;
        cost = 0;
        lcd.clear();
    }
}

```

```

void pulseCounter(){
    pulseCount++;
}

```

2. Listing program website

a. Halaman Dashboard

```

<?php
    include "config.php";
    $label=
["Januari", "Februari", "Maret", "April", "Mei", "Juni", "Juli", "Agustus",
"September", "Oktober", "November", "Desember"];
?>

<html>
<head>
    <script type="text/javascript" src="chartjs/Chart.js"></script>
</head>
<hr>
<b>Data Monitoring Air</b>
<hr>

<body>
    <table width= 100% border="1" cellpadding="4">
        <tr bgcolor="00ffffff">
            <th>No</th>
            <th>Node_ID</th>
            <th>Volume</th>
            <th>Biaya</td>
            <th>Created at</th>
        </tr>

```

```

<!-- Input data ke tabel dari Database-->
<?php
    $no=0;
    $query= mysqli_query($connection, "SELECT m1.* FROM
dashboard m1 LEFT JOIN dashboard m2 ON (m1.node_id = m2.node_id AND
m1.id < m2.id) WHERE m2.id IS NULL ORDER BY node_id;");
    while($result= mysqli_fetch_array($query)){
        $no++
        ?>

        <tr>
            <td><?php echo $no?></td>
                <td><b><a style="text-decoration:none"
href="page/detail-data1.php?node_id=?=$result['node_id']?"><?php
echo $result['node_id']?></a></b></td>
                    <td><?php echo $result['volume']?></td>
                        <td><?php echo $result['cost']?></td>
                            <td><?php echo $result['created_at']?></td>
                                </tr>
                                    <?php
                                        }
                                            ?>
                                                </table>

                                                    <div style="width:600px; margin: 10px auto">
                                                        <canvas id="myChart"></canvas>
                                                            </div>

                                                                <script>
                                                                    var ctx=
document.getElementById('myChart').getContext('2d');
                                                                    var myChart = new Chart(ctx, {
                                                                        type:'line',
                                                                        data: {
                                                                            labels: <?php echo json_encode($label); ?>,
                                                                            datasets: [
                                                                                {

```



```

        label: 'Node 1',
        data: [
            <?php
                $year=date('Y');
                for ($i=1; $i<=12 ; $i++) {
                    $data_node_1=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=1 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
                    $data1="";
                    while($data_n1=mysqli_fetch_assoc($data_no
de_1)){
                        $dn1=$data_n1['volume'];
                        $data1 .= "$dn1".".",";
                        echo $data1;
                    }
                }
            ?>

            ],
            backgroundColor:[ 'rgba(255, 99, 132, 0.4)',
            borderColor:['rgba(255, 99, 132, 0.4)',
            borderWidth: 1
        ],
        {
            label: 'Node 2',
            data: [
                <?php
                    for ($i=1; $i<=12 ; $i++) {
                        $data_node_2=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=2 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
                        $data2="";
                        while($data_n2=mysqli_fetch_assoc($data_no
de_2)){

```

```

        $dn2=$data_n2['volume'];
        $data2 .= "'$dn2'\".\"";
        echo $data2;
    }
}
?>

],
32 backgroundColor:[ 'rgba(54, 162, 235, 0.4)',
borderColor:[ 'rgba(54, 162, 235, 0.4)',
borderWidth: 1
],
{
{
label: 'Node 3',
data: [
    <?php
        for ($i=1; $i<=12 ; $i++) {
            $data_node_3=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=3 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
            $data3="";
            while($data_n3=mysqli_fetch_assoc($data_no
de_3)){
                $dn3=$data_n3['volume'];
                $data3 .= "'$dn3'\".\"";
                echo $data3;
            }
        }
    ?>
],
49 backgroundColor:[ 'rgba(255, 0, 0, 0.4)',
borderColor:[ 'rgba(255, 0, 0, 0.4)',
borderWidth: 1
],
{
{
label: 'Node 4',
data: [

```

```

<?php
    for ($i=1; $i<=12 ; $i++) {
        $data_node_4=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=4 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
        $data4="";
        while($data_n4=mysqli_fetch_assoc($data_no
de_4)){
            $dn4=$data_n4['volume'];
            $data4 .= "'$dn4'","";
            echo $data4;
        }
    }
    ?>

],
backgroundColor:[ 'rgba(0, 255, 0, 0.4)',
borderColor:['rgba(0, 255, 0, 0.4)',
borderWidth: 1
],
{
label: 'Node 5',
data: [
    <?php
        for ($i=1; $i<=12 ; $i++) {
            $data_node_5=mysqli_query($connection,"SELECT
sum(volume) as volume FROM dashboard t1 JOIN (SELECT
DATE(created_at) date_date, node_id, MAX(created_at) max_date FROM
dashboard GROUP BY date_date, node_id ) t2 ON t1.created_at =
t2.max_date WHERE t2.node_id=5 AND YEAR(t1.created_at)=$year AND
MONTH(t1.created_at)=$i");
            $data5="";
            while($data_n5=mysqli_fetch_assoc($data_no
de_5)){
                $dn5=$data_n5['volume'];
                $data5 .= "'$dn5'","";
                echo $data5;

```

```

    }
  }
  ?>
  ],
  backgroundColor: [ 'rgba(255, 0, 255, 0.4)',
  borderColor: [ 'rgba(255, 0, 255, 0.4)',
  borderWidth: 1
  ]
  ],
  },
  options: {
    scales: {
      yAxes: [ {
        ticks: {
          beginAtZero: true
        }
      } ]
    }
  }
  });
</script>
</body>
</html>

```

b. Listing program halaman detail data

```

<html>
<body>
<hr>
<h2><b>Detail Data User Node ID <?php echo
$_GET['node_id'];?></b></h2>
<style>
  @media print{
    #art-sheet #sidebar{
      width:0;
    }
    #art-sheet #article{
      width: 100%;
    }
    .back-page, .print-page, .search-date, .label-pencarian,
    .tahun-pencarian, .submit-pencarian, .bulan-pencarian{
      display: none;
    }
  }

```

```

    }

}
#art-sheet{
border-top: 1px solid #FEFEFE;
clear: both;
width: 100%;
margin: 1px auto 0;
background: white;
color: #000000;
text-align:left;
}

#sidebar{
float: left;
width: 15%;
margin: 0 5px auto;
padding: 5px;
position:fixed;
}

#article{
padding-left:40px;
float: right;
width: 80%;
}

.article-header{
float: left;
width: 100%;
height: auto;
line-height:40px;
margin: 10px 0 2px 0;
padding: 0;
}

.article-content{
padding: 10px;
float: left;
width: 100%;
overflow:scroll;
}

</style>
<button onclick="location.href='../';" class="back-
page">Kembali</button>
<hr>

```

```

<div id=art-sheet>
  <div id=sidebar>
    <button onclick="window.print()" class="print-page">Print
or Save to PDF</button>
    <br>
    <br>
    <b class="search-date">Search</b>
    <form action="" method= "POST">
    <label class="label-pencarian" for="pencarian">Pilih Bulan
dan Tahun</label>
    <select class="bulan-pencarian" name="bulan" id="cari">
    <option value="01">Januari</option>
    <option value="02">Februari</option>
    <option value="03">Maret</option>
    <option value="04">April</option>
    <option value="05">Mei</option>
    <option value="06">Juni</option>
    <option value="07">Juli</option>
    <option value="08">Agustus</option>
    <option value="09">September</option>
    <option value="10">Oktober</option>
    <option value="11">November</option>
    <option value="12">Desember</option>
    <input class="tahun-pencarian" type="text" name="tahun"
placeholder="Tahun">
    </select>
    <button class="submit-pencarian" type="submit"
name="submit" >Submit</button>
  </form>
</div>
<div id=article>
  <div class=aricle-content>
    DATA BULAN <?php
      if(!isset($_POST['bulan'])){
        $_POST['bulan']=date('m');
      }
      if(!isset($_POST['tahun'])){
        $_POST['tahun']=date('Y');
      }
      echo $_POST['bulan'],- $_POST['tahun'];?>
    <table width= 100% border="1" cellpadding="4">
      <tr bgcolor="00ffffff">
        <th>No</th>
        <th>Node_ID</th>
        <th>Volume (m3)</th>
        <th>Biaya (Rp.)</th>
        <th>Created at</th>

```

```

        </tr>
<?php
    if(!isset($_POST['bulan'])){
        $bulan = date('m');
    }
    else{
        $bulan = $_POST['bulan'];
    }
    if(!isset($_POST['tahun'])){
        $tahun = date('Y');
    }
    else{
        $tahun= $_POST['tahun'];
    }
?>
14
<?php
    if(isset($_GET['node_id'])){
        $node_id = $_GET['node_id'];
    }
    else {
        die ("Error, No ID Selected!");
    }
?>
<!--Tambah data-->
<?php
    include "config.php";
    $node_id = $_GET['node_id'];
    $no=0;
    $total=0;
    $tot_bayar=0;
    $tot_volume=0;
    $tot_vol=0 ;
    $query= mysqli_query($connection, "SELECT t1.* FROM
dashboard t1 JOIN (SELECT DATE(created_at) date_date, node_id,
MAX(created_at) max_date FROM dashboard GROUP BY date_date,
node_id ) t2 ON t1.created_at = t2.max_date WHERE
t2.node_id=$node_id AND YEAR(t1.created_at)=$tahun' AND
MONTH(t1.created_at)=$bulan");
    while($result= mysqli_fetch_array($query)){
        $total= $result['cost'];
        $tot_vol= $result['volume'];
        $tot_volume += $tot_vol;
        $tot_bayar += $total;
        $no++
?>
50
<tr>

```

```

        <td><?php echo $no?></td>
        <td><?php echo $result['node_id']?></td>
        <td><?php echo $result['volume']?></td>
        <td><?php echo $result['cost']?></td>
        <td><?php echo $result['created_at']?></td>
    </tr>

    <?php
    }
    ?>

    <tr>
        <th colspan="2">Total Bayar</th>
        <th><?php echo $tot_volume ?></th>
        <th><?php echo $tot_bayar ?></th>
    </tr>
</div>
</div>
</table>

</body>
</html>

```

c. Listing program halaman *about us*

```

<hr>
<b>ABOUT US</b>
<hr>

<div class="news">
    <div class="judul">
        Profile
    </div>
    <div class="sinopsis">
        Web yang bertujuan untuk memudahkan monitoring penggunaan air
        di setiap rumah
    </div>
</div>
<div class="news">
    <div class="judul">
        Information
    </div>
    <div class="sinopsis">
55 Air membawa nutrisi yang dibutuhkan oleh tubuh. Air sebagai
sumber pemenuhan kebutuhan hidup, seperti irigasi persawahan,

```


kehidupan binatang, nutrisi tumbuhan, dan lainnya. Penggunaan air yang bersih bisa mencegah berbagai macam penyakit, seperti diare, kolera, dan sebagainya.

Hari Air Sedunia dirayakan setiap tanggal 22 Maret.

Penghematan air dapat dilakukan dengan cara berikut.
1. Matikan keran air jika tidak digunakan.
2. Menggunakan air seperlunya.
3. Merawat peralatan pipa air, keran, dan penampungan air.
4. Pantau tagihan air secara berkala.

</div>
</div>

d. Listing program halaman contact

```
11
<hr>
<b>Admin Profil</b>
<hr>

<div style="width: 200px; float: left;">

</div>
<div style="float: left;">
<table width="100%" cellpadding="5">
  <tr>
    <th width="120" align="left">Nama</th>
    <td> Dimico</td>
  </tr>
  <tr>
    <th width="120" align="left">Bagian</th>
    <td> Admin</td>
  </tr>
  <tr>
    <th width="120" align="left">Email</th>
    <td> 3332170047@untirta.ac.id</td>
  </tr>
  <tr>
    <th width="120" align="left">Phone</th>
    <td> 0123456789</td>
  </tr>
  <tr>
    <th width="120" align="left">Alamat</th>
    <td> Tangerang, Banten</td>
  </tr>
</table>
</div>

<hr style="clear: both">
```

e. Listing program config.php

```
80
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "monitoring";

$conn = mysqli_connect($servername, $username, $password,
$dbname);
// if (!$conn){
//     die("Connection Failed:".mysqli_connect_error());
// }

function query($query){
    global $conn;
    $result = mysqli_query($conn,$query);
    $row = [];
    while ($row = mysqli_fetch_assoc($result)){
        $row[] = $row;
    }
    return $row;
}
?>
```

f. Listing program database.php

```
2
<?php
class Database {
    private $host = "localhost";
    private $database_name = "monitoring";
    private $username = "root";
    private $password = "";

    public $conn;

    public function getConnection(){
        $this->conn = null;
        try{
            $this->conn = new PDO("mysql:host=" . $this->host
            . ";dbname=" . $this->database_name, $this->username, $this->
            >password);
            $this->conn->exec("set names utf8");
        }catch(PDOException $exception){
            echo "Database could not be connected: " .
            $exception->getMessage();
        }
    }
}
```

```

        return $this->conn;
    }
}
?>

```

g. Listing program create.php

```

<?php
    header("Access-Control-Allow-Origin: *");
    header("Content-Type: application/json; charset=UTF-8");

    include_once '../config/database.php';
    include_once '../class/nodemcu_log.php';

    $database = new Database();
    $db = $database->getConnection();

    $item = new monitoring($db);

    if ($_SERVER['REQUEST_METHOD'] === 'POST') {
        // The request is using the POST method
        $data = json_decode(file_get_contents("php://input"));
        $item->node_id = $data->node_id;
        $item->volume = $data->volume;
        $item->cost = $data->cost;
    }

    if($item->createLogData()){
        echo 'Data created successfully.';
    } else{
        echo 'Data could not be created.';
    }
}
?>

```

h. Listing program nodemcu_log.php

```

<?php
class monitoring{

    // Connection
    private $conn;

    // Table
    private $db_table = "dashboard";

    // Columns
    public $id;
}

```

```

public $node_id;
public $volume;
public $cost;
public $created_at;

// Db connection
public function __construct($db){
    $this->conn = $db;
}

// CREATE
public function createLogData(){
    $sqlQuery = "INSERT INTO
        ". $this->db_table ."
        SET
            node_id = :node_id,
            volume = :volume,
            44 cost = :cost";
    $stmt = $this->conn->prepare($sqlQuery);

    // sanitize
    $this->node_id=htmlspecialchars(strip_tags($this-
>node_id));
    $this->volume=htmlspecialchars(strip_tags($this-
>volume));
    $this->cost=htmlspecialchars(strip_tags($this->cost));

    // bind data
    $stmt->bindParam(":node_id", $this->node_id);
    $stmt->bindParam(":volume", $this->volume);
    $stmt->bindParam(":cost", $this->cost);
    if($stmt->execute()){
        return true;
    }
    return false;
}
}
?>

```

i. Listing program style css

```

body {
    font-family: verdana;
    font-size: 0.85em;
}

.menu {

```

```
display: block;
background-color:#00A6BB;
height:46px;
}

.content {
clear: both;
padding:5px 0;
min-height:200px;
}

.footer {
clear: both;
margin-top:10px;
background-color: #eee;
padding:10px 8px;
}

#navigasi {
position:relative;
line-height:30px;
margin:0;
padding:0;
list-style-type:none;
list-style-position:outside;
}

#navigasi a {
display:block;
padding:8px 16px;
background-color:#00A6BB;
color:#fff;
text-decoration:none;
}

#navigasi a:hover {
background-color:#00BCD4;
color:#fff;
}

#navigasi li {
position:relative;
float:left;
}

#navigasi ul {
```

```

    position:absolute;
    display:none;
    margin:0;
    padding:0;
    list-style-type:none;
    list-style-position:outside;
}

#navigasi li ul a{
    width:12em;
    height:auto;
    float:left;
}

#navigasi li:hover ul{
    display:block;
}

#navigasi li:hover ul ul{
    display:none;
}

.news {
    background: #ddd;
}

.judul {
    padding: 4px;
    border-top: 1px solid #ccc;
    border-bottom: 1px solid #ccc;
    font-weight: bold;
}

.sinopsis {
    padding: 4px;
    margin-bottom: 5px;
}

151
table {
    table-layout:fixed;
}

td {
    overflow: hidden;
    text-overflow: ellipsis;
    word-wrap: break-word;
}

```

RANCANG BANGUN SISTEM MONITORING WATER METER BERBASIS WIRELESS SENSOR NETWORK

ORIGINALITY REPORT

27%

SIMILARITY INDEX

26%

INTERNET SOURCES

7%

PUBLICATIONS

17%

STUDENT PAPERS

PRIMARY SOURCES

1	123dok.com Internet Source	2%
2	www.arducoding.com Internet Source	1%
3	eprints.untirta.ac.id Internet Source	1%
4	forum.arduino.cc Internet Source	1%
5	mbkm.unila.ac.id Internet Source	1%
6	repository.uin-suska.ac.id Internet Source	1%
7	randomnerdtutorials.com Internet Source	1%
8	repository.its.ac.id Internet Source	1%
9	repositori.usu.ac.id Internet Source	1%

10	www.hackster.io Internet Source	1 %
11	www.belajarwebpedia.com Internet Source	1 %
12	repository.usd.ac.id Internet Source	1 %
13	agustinusvendetta.blogspot.com Internet Source	<1 %
14	www.rajaputramedia.com Internet Source	<1 %
15	konsultasiskripsi.com Internet Source	<1 %
16	docplayer.info Internet Source	<1 %
17	Submitted to STT PLN Student Paper	<1 %
18	www.coursehero.com Internet Source	<1 %
19	mail.tutorkeren.com Internet Source	<1 %
20	eprints.akakom.ac.id Internet Source	<1 %
21	Submitted to Universitas Muria Kudus Student Paper	<1 %

22

[Submitted to Universitas Brawijaya](#)

Student Paper

<1 %

23

www.scribd.com

Internet Source

<1 %

24

www.slideshare.net

Internet Source

<1 %

25

[Submitted to Universitas Pamulang](#)

Student Paper

<1 %

26

id.scribd.com

Internet Source

<1 %

27

www.trijurnal.lemlit.trisakti.ac.id

Internet Source

<1 %

28

repository.ub.ac.id

Internet Source

<1 %

29

ijiswiratama.org

Internet Source

<1 %

30

www.warungbelajar.com

Internet Source

<1 %

31

dspace.uui.ac.id

Internet Source

<1 %

32

[Submitted to Universitas Trunojoyo](#)

Student Paper

<1 %

33

www.repository.umuslim.ac.id

Internet Source

<1 %

34	digilib.unila.ac.id Internet Source	<1 %
35	repo.darmajaya.ac.id Internet Source	<1 %
36	etheses.uin-malang.ac.id Internet Source	<1 %
37	Submitted to itera Student Paper	<1 %
38	repository.dinamika.ac.id Internet Source	<1 %
39	repository.unhas.ac.id Internet Source	<1 %
40	Submitted to Universitas Putera Batam Student Paper	<1 %
41	elektro.untirta.ac.id Internet Source	<1 %
42	text-id.123dok.com Internet Source	<1 %
43	vdocuments.net Internet Source	<1 %
44	Submitted to RDI Distance Learning Student Paper	<1 %
45	docobook.com Internet Source	<1 %

46	edoc.pub Internet Source	<1 %
47	espitek.com Internet Source	<1 %
48	id.123dok.com Internet Source	<1 %
49	deepcred.mpi-inf.mpg.de Internet Source	<1 %
50	Submitted to University of Greenwich Student Paper	<1 %
51	microcontrollerslab.com Internet Source	<1 %
52	www.pustaka.ut.ac.id Internet Source	<1 %
53	theses.iainkediri.ac.id Internet Source	<1 %
54	repository.upi.edu Internet Source	<1 %
55	www.kompasiana.com Internet Source	<1 %
56	Submitted to Sriwijaya University Student Paper	<1 %
57	repo.palcomtech.ac.id Internet Source	<1 %

58	retii.sttnas.ac.id Internet Source	<1 %
59	perpustakaan.jak-stik.ac.id Internet Source	<1 %
60	repository.unib.ac.id Internet Source	<1 %
61	Submitted to Quest International University Perak Student Paper	<1 %
62	Submitted to Universitas Sebelas Maret Student Paper	<1 %
63	adoc.pub Internet Source	<1 %
64	openlibrarypublications.telkomuniversity.ac.id Internet Source	<1 %
65	repository.itelkom-pwt.ac.id Internet Source	<1 %
66	Submitted to LL DIKTI IX Turnitin Consortium Part II Student Paper	<1 %
67	jurnal.usbypkp.ac.id Internet Source	<1 %
68	Submitted to Abington Friends School Student Paper	<1 %

69	github.com Internet Source	<1 %
70	pt.scribd.com Internet Source	<1 %
71	eprints.poltektegal.ac.id Internet Source	<1 %
72	repository.unsri.ac.id Internet Source	<1 %
73	Submitted to Institut Teknologi Nasional Malang Student Paper	<1 %
74	Submitted to Lambung Mangkurat University Student Paper	<1 %
75	Submitted to Universitas Muhammadiyah Sidoarjo Student Paper	<1 %
76	Submitted to Universitas Siliwangi Student Paper	<1 %
77	Submitted to University of Hertfordshire Student Paper	<1 %
78	eprints.radenfatah.ac.id Internet Source	<1 %
79	ft.untirta.ac.id Internet Source	<1 %

80	noc-id.blogspot.com Internet Source	<1 %
81	Submitted to STIE Perbanas Surabaya Student Paper	<1 %
82	eprints.umm.ac.id Internet Source	<1 %
83	journal.iainkudus.ac.id Internet Source	<1 %
84	repository.pnj.ac.id Internet Source	<1 %
85	kursuselektronikaku.blogspot.com Internet Source	<1 %
86	ojs.uajy.ac.id Internet Source	<1 %
87	repository.unwira.ac.id Internet Source	<1 %
88	vaskoedo.wordpress.com Internet Source	<1 %
89	zpreneur.org Internet Source	<1 %
90	Submitted to Universiti Malaysia Pahang Student Paper	<1 %
91	repository.bsi.ac.id Internet Source	<1 %

92	suruhbelajar.blogspot.com Internet Source	<1 %
93	Submitted to Leeds Metropolitan University Student Paper	<1 %
94	Submitted to Syiah Kuala University Student Paper	<1 %
95	Submitted to Universidad de San Carlos de Guatemala Student Paper	<1 %
96	Submitted to Universitas Bengkulu Student Paper	<1 %
97	e-journal.uajy.ac.id Internet Source	<1 %
98	repository.umy.ac.id Internet Source	<1 %
99	repository.wima.ac.id Internet Source	<1 %
100	Andree Fajar Pratama, Syamsyarief Baqaruzi, Ali Muhtar. "Quality of Service Packet Loss Pada Sistem Home Monitoring Water Flow Berbasis Internet of Things", ELECTRON : Jurnal Ilmiah Teknik Elektro, 2021 Publication	<1 %
101	Submitted to University of Hull Student Paper	<1 %

102	blogpunyahari.blogspot.com Internet Source	<1 %
103	jurnal.poliupg.ac.id Internet Source	<1 %
104	repository.atmaluhur.ac.id Internet Source	<1 %
105	repository.lppm.unila.ac.id Internet Source	<1 %
106	Mirza Fissabila, Priyambada Cahya Nugraha, Muhammad Ridha Mak'ruf. "Pengembangan Pusat Pemantauan Central SpO2 untuk Ruang Neonate denga Sistem Wireless", Jurnal Teknokes, 2020 Publication	<1 %
107	fdocuments.net Internet Source	<1 %
108	jmiki.apfirmik.or.id Internet Source	<1 %
109	www.dewaweb.com Internet Source	<1 %
110	www.mbdyn.org Internet Source	<1 %
111	www.researchgate.net Internet Source	<1 %

112	agungbudip94.blogspot.com Internet Source	<1 %
113	es.scribd.com Internet Source	<1 %
114	lib.unnes.ac.id Internet Source	<1 %
115	pt.slideshare.net Internet Source	<1 %
116	Mochamad Fandi. "Aplikasi Identifikasi Jenis Buah Kurma Dengan Metode GLCM Berbasis Android", Jurnal Pengembangan Rekayasa dan Teknologi, 2020 Publication	<1 %
117	N. Sushma, H. N. Suresh, J Mohana Lakshmi, Parvathaneni Naga Srinivasu, Akash Kumar Bhoi, Paolo Barsocchi. "A unified metering system deployed for water and energy monitoring in smart city", IEEE Access, 2023 Publication	<1 %
118	Rizal Justian Setiawan, Ageng Widi Atmoko, Imam Fauzi. "IoT-Based Electric Vampire Remover to Overcome Electric Vampire On Electronic Equipment", JTECS : Jurnal Sistem Telekomunikasi Elektronika Sistem Kontrol Power Sistem dan Komputer, 2021 Publication	<1 %

119	bmath.org Internet Source	<1 %
120	elshintacom.com Internet Source	<1 %
121	eprints.ums.ac.id Internet Source	<1 %
122	eprints.uny.ac.id Internet Source	<1 %
123	issuu.com Internet Source	<1 %
124	journals.upi-yai.ac.id Internet Source	<1 %
125	karyailmiah.unisba.ac.id Internet Source	<1 %
126	kiec.co.id Internet Source	<1 %
127	repository.mercubuana.ac.id Internet Source	<1 %
128	repository.radenintan.ac.id Internet Source	<1 %
129	syaifudininfo.blogspot.com Internet Source	<1 %
130	www.heart-to-heart-world.org Internet Source	<1 %

131	akamigas.esdm.go.id Internet Source	<1 %
132	bappeda.bantulkab.go.id Internet Source	<1 %
133	dokumen.tips Internet Source	<1 %
134	fr.scribd.com Internet Source	<1 %
135	jamalludinsitepu.blogspot.com Internet Source	<1 %
136	jurnal.unmuhjember.ac.id Internet Source	<1 %
137	jurnal.usahidsolo.ac.id Internet Source	<1 %
138	markbro.com Internet Source	<1 %
139	repositori.umsu.ac.id Internet Source	<1 %
140	repository.ipb.ac.id Internet Source	<1 %
141	repository.unj.ac.id Internet Source	<1 %
142	rifqimulyawan.com Internet Source	<1 %

143	supriani.com Internet Source	<1 %
144	www.batamnews.co.id Internet Source	<1 %
145	www.bromindo.com Internet Source	<1 %
146	zombiedoc.com Internet Source	<1 %
147	Imam Syafi'i, Mohammad Iqbal, Noor Yulita Dwi Setyaningsih. "RANCANG BANGUN SISTEM ABSENSI KARYAWAN INDUSTRI RUMAHAN MENGGUNAKAN E-KTP BERBASIS WEB", E-Link: Jurnal Teknik Elektro dan Informatika, 2021 Publication	<1 %
148	doku.pub Internet Source	<1 %
149	repositori.uin-alauddin.ac.id Internet Source	<1 %
150	www.maritimeprofessional.com Internet Source	<1 %
151	social.msdn.microsoft.com Internet Source	<1 %

Exclude quotes On

Exclude matches < 3 words

Exclude bibliography On