



**RANCANG BANGUN SISTEM MONITORING
PENGAWASAN TINGGI AIR SEBAGAI
PENCEGAHAN DINI BANJIR BERBASIS *IOT* PADA
KOMPLEK GRIYA ASRI DEPOK**

Sub Judul:

**RANCANG BANGUN SISTEM PEMANTAUAN *REAL-
TIME* KETINGGIAN AIR DAN DEBIT
MENGUNAKAN *IOT* DENGAN PEMANFAATAN
SENSOR ULTRASONIK *JSN-SR04T* DAN *YF-S201***

SKRIPSI

NATANAEL SIWALETTE

2107421018

**PROGRAM STUDI TEKNIK MULTIMEDIA DAN JARINGAN
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025



**RANCANG BANGUN SISTEM PEMANTAUAN *REAL-TIME* KETINGGIAN AIR DAN DEBIT
MENGUNAKAN *IOT* DENGAN PEMANFAATAN
SENSOR ULTRASONIK *JSN-SR04T* DAN *YF-S201***

SKRIPSI

**Dibuat untuk melengkapi syarat-syarat yang diperlukan untuk
memperoleh diploma empat politeknik**

NATANAEL SIWALETTE

2107421018

**PROGRAM STUDI TEKNIK MULTIMEDIA DAN JARINGAN
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini:

Nama : Natanael Siwalette
NIM : 2107421018
Jurusan / Program Studi : Teknik Informatika dan Komputer / Teknik Multimedia dan Jaringan
Judul Skripsi : Rancang Bangun Sistem Pemantauan *Real-Time* Ketinggian Air Dan Debit Menggunakan *IoT* Dengan Pemanfaatan Sensor Ultrasonik *JSN-SR04T* Dan *YF-S201*

Menyatakan dengan sebenarnya bahwa skripsi ini benar-benar merupakan hasil karya saya sendiri, bebas dari peniruan terhadap karya dari orang lain. Kutipan pendapat dan tulisan orang lain ditunjuk sesuai dengan cara-cara penulisan karya ilmiah yang berlaku.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa dalam skripsi ini terkandung ciri-ciri plagiat dan bentuk-bentuk peniruan lain yang dianggap melanggar peraturan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Depok, 10 Juni 2025

Yang Membuat Pernyataan



Natanael Siwalette
NIM. 2107421018



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

LEMBAR PENGESAHAN

Skripsi diajukan oleh:

Nama : Natanael Siwalette
NIM : 2107421018
Program Studi : Teknik Multimedia dan Jaringan
Judul Skripsi : Rancang Bangun Sistem Pemantauan *Real-Time* Ketinggian Air Dan Debit Menggunakan *IoT* Dengan Pemanfaatan Sensor Ultrasonik *JSN-SR04T* Dan *YF-S201*

Telah diuji oleh tim penguji dalam Sidang Skripsi pada hari *Rabu*,
Tanggal *2*, Bulan *Juli*, Tahun *2025* dan dinyatakan **LULUS**.

Disahkan oleh

Pembimbing I : Dr. Indra Hermawan, S.Kom., M.Kom.

Penguji I : Maria Agustin, S.Kom., M.Kom

Penguji II : Susana Dwi Yulianti, M.Kom

Penguji III : Chandra Wirawan, M.Kom

Mengetahui:

Jurusan Teknik Informatika dan Komputer
Ketua



Dr. Anita Hidayati, S.Kom., M.Kom.

NIP. 197908032003122003

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir berjudul “*Rancang Bangun Sistem Pemantauan Real-Time Ketinggian Air dan Debit Menggunakan IoT dengan Pemanfaatan Sensor Ultrasonik JSN-SR04T dan YF-S201*” dengan baik dan tepat waktu. Selama proses penyusunan tugas akhir ini, penulis mendapatkan banyak bantuan, bimbingan, dan dukungan dari berbagai pihak yang berperan penting dalam kelancaran pelaksanaan penelitian hingga penyusunan laporan ini.

Oleh karena itu, penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Tuhan Yang Maha Esa atas segala rahmat, perlindungan, dan kekuatan yang telah diberikan selama proses penelitian dan penyusunan laporan ini.
2. Kedua orang tua penulis yang selalu memberikan doa, semangat, dan dukungan moral maupun materi dalam setiap tahap kehidupan penulis.
3. Bapak Dr. Indra Hermawan, S.Kom., M.Kom., selaku dosen pembimbing yang telah memberikan bimbingan, arahan, dan motivasi secara konsisten selama proses penyusunan Tugas Akhir ini.
4. Yazmin Nuraini, rekan satu kelompok yang telah bekerja sama dan berkontribusi secara aktif dalam pelaksanaan dan penyusunan Tugas Akhir ini.
5. Rekan-rekan seperjuangan dan teman-teman di Politeknik Negeri Jakarta yang telah menjadi tempat berbagi ilmu, saling mendukung, dan menemani proses perkuliahan hingga penyusunan tugas akhir.

Penulis menyadari laporan ini masih memiliki kekurangan, sehingga kritik dan saran yang membangun sangat diharapkan. Semoga laporan ini bermanfaat dan dapat memberikan kontribusi di bidang pemantauan lingkungan berbasis *IoT*.

Depok, 14 Juni 2025

Natanael Siwalette



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

**SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK
KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Politeknik Negeri Jakarta, saya bertanda tangan dibawah ini:

Nama : Natanael Siwalette
NIM : 2107421018
Jurusan / Program Studi : Teknik Informatika dan Komputer / Teknik Multimedia dan Jaringan

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Politeknik Negeri Jakarta Hak Bebas Royalti Non-Eksklusif atas karya ilmiah saya yang berjudul:

Rancang Bangun Sistem Pemantauan *Real-Time* Ketinggian Air Dan Debit Menggunakan *IoT* Dengan Pemanfaatan Sensor Ultrasonik JSN-SR04T Dan YF-S201

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti NonEksklusif ini Politeknik Negeri Jakarta Berhak menyimpan, mengalihmediakan/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Depok, 14 Juni 2025

Yang Membuat Pernyataan



Natanael Siwalette

NIM. 2107421024



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Rancang Bangun Sistem Pemantauan *Real-Time* Ketinggian Air Dan Debit Menggunakan *IoT* Dengan Pemanfaatan Sensor Ultrasonik *JSN-SR04T* Dan *YF-S201*

ABSTRAK

Banjir merupakan bencana alam yang sering terjadi di kawasan padat penduduk, salah satunya di Komplek Griya Asri, Depok. Faktor utama penyebab banjir di daerah tersebut adalah penyempitan aliran sungai, endapan sedimen yang tidak terkelola, serta kerusakan pada infrastruktur penahan air seperti Situ Studio Alam. Untuk mengatasi hal tersebut, diperlukan sistem pemantauan yang mampu mengukur kondisi air secara *Real-Time* dan memberikan peringatan dini kepada masyarakat maupun pihak berwenang. Penelitian ini bertujuan merancang dan membangun sistem pemantauan ketinggian dan debit air berbasis *Internet of things (IoT)* dengan memanfaatkan sensor *JSN-SR04T* dan *YF-S201*, serta menggunakan modul komunikasi *SIM800L* dengan protokol *MQTT* sebagai media pengiriman data. Sistem yang dibangun mengintegrasikan mikrokontroler *Arduino Nano*, sensor ketinggian dan debit air, serta indikator *LED* yang memberikan informasi kondisi air dalam tiga status: aman, waspada, dan bahaya. Data dikirim ke broker *HiveMQ* dan ditampilkan dalam *dashBoard* pemantauan. Hasil pengujian menunjukkan bahwa sistem mampu bekerja secara akurat dan stabil, dengan nilai *Availability* sebesar 98,21% dan *Durability* sebesar 95,76%. Rata-rata *Delay* sebesar 1156.53 ms menunjukkan bahwa pengiriman data melalui jaringan 2G masih memiliki keterbatasan. Secara keseluruhan, sistem ini merupakan solusi efektif dan terjangkau dalam mendukung mitigasi bencana banjir dan meningkatkan kesiapsiagaan masyarakat.

Kata Kunci: *IoT*, *JSN-SR04T*, *YF-S201*, *SIM800L*, *MQTT*

POLITEKNIK
NEGERI
JAKARTA



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR ISI

SURAT PERNYATAAN BEBAS PLAGIARISME	iii
LEMBAR PENGESAHAN	iv
KATA PENGANTAR.....	v
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS	vi
ABSTRAK	vii
DAFTAR ISI.....	viii
DAFTAR TABEL	x
DAFTAR GAMBAR.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan dan Manfaat.....	3
1.4.1 Tujuan	3
1.4.2 Manfaat	4
1.5 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 Penelitian Terkait.....	6
2.2 Sistem Pengawasan Pencegahan Banjir	7
2.3 <i>Internet of things (IoT)</i>	8
2.4 <i>Arduino Nano</i>	8
2.5 Modul <i>GSM 2G SIM800L</i>	9
2.6 Protokol <i>MQTT</i>	10
2.7 <i>MQTT Broker</i>	10
2.8 Sensor Ultrasonik <i>JSN-SR04T</i>	11
2.9 <i>YF-S201</i>	12
2.10 <i>AWS</i>	13
2.11 <i>Flowchart</i>	13
2.12 <i>Python</i>	16
BAB III METODE PENELITIAN	17
3.1 Rancangan Penelitian	17
3.2 Tahapan Penelitian	17
3.3 Objek Penelitian	20

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB IV HASIL DAN PEMBAHASAN	21
4.1 Analisis Kebutuhan	21
4.2 Perancangan Sistem.....	23
4.2.1 Flowchart Sistem.....	25
4.2.2 Diagram Blok.....	26
4.2.3 Rancangan Skematik Alat.....	27
4.2.3.1 Skematik Catu Daya Microcontroller	27
4.2.3.2 Skematik SIM800L	28
4.2.3.3 Skematik JSN-SR04T	29
4.2.3.4 Skematik YF-S201	30
4.2.3.5 Skematik LED Traffic Light Simulation Module	31
4.2.3.6 Skematik Keseluruhan Alat	32
4.3 Implementasi Sistem	33
4.3.1 Implementasi Perangkat Keras	34
4.3.1.1 Perakitan Komponen Elektronika	34
4.3.1.2 Perancangan Kotak Pelindung dan Piranti Penunjang..	36
4.3.2 Implementasi Perangkat Lunak	37
4.3.2.1 Pemrograman Mikrokontroler dan Pengiriman Data....	37
4.3.2.2 Konfigurasi Broker HiveMQ	40
4.4 Pengujian Sistem	43
4.4.1 Deskripsi Pengujian	43
4.4.2 Prosedur Pengujian	43
4.4.2.1 Prosedur Pengujian Fungsionalitas dan Akurasi.....	43
4.4.2.2 Prosedur Pengujian Availability dan Durability	44
4.4.2.3 Prosedur Pengujian Quality of Service (QoS)	47
4.5 Data Hasil Pengujian	48
4.5.1 Hasil Pengujian Fungsionalitas dan Akurasi	48
4.5.2 Hasil Pengujian Availability dan Durability.....	52
4.5.3 Hasil Pengujian Quality of Service (QoS)	56
BAB V PENUTUP	58
5.1 Kesimpulan.....	58
5.2 Saran	58
DAFTAR PUSTAKA	60
DAFTAR RIWAYAT HIDUP	63
LAMPIRAN.....	64



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR TABEL

Tabel 2. 1 Penelitian Terkait Sebelumnya	6
Tabel 4. 1 Analisis Kebutuhan Perangkat Keras.....	21
Tabel 4. 2 Analisis Kebutuhan Perangkat Lunak.....	23
Tabel 4. 3 Pinout Catu Daya <i>Arduino Nano</i>	28
Tabel 4. 4 Pinout <i>Arduino Nano</i> ke <i>SIM800L</i>	29
Tabel 4. 5 Pinout <i>Arduino Nano</i> dan <i>JSN-SR04T</i>	30
Tabel 4. 6 Pinout <i>Arduino Nano</i> dan <i>YF-S201</i>	31
Tabel 4. 7 Pinout <i>Arduino Nano</i> dan <i>LED Traffic LED Module</i>	32
Tabel 4. 8 Indeks Kategori Nilai <i>Delay</i>	48
Tabel 4. 9 Hasil Pengujian Sensor <i>JSN-SR04T</i>	48
Tabel 4. 10 Hasil Pengujian Sensor <i>YF-S201</i>	50
Tabel 4. 11 Hasil Pengujian <i>SIM800L</i>	52
Tabel 4. 12 Hasil Pengujian <i>Availibility</i>	53
Tabel 4. 13 Hasil Pengujian <i>Durability</i>	54
Tabel 4. 14 Hasil Pengujian <i>Delay</i>	56

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR GAMBAR

Gambar 2. 1 <i>Internet of things</i>	8
Gambar 2. 2 Modul <i>Arduino Nano</i>	9
Gambar 2. 3 Modul <i>SIM800L</i>	9
Gambar 2. 4 Protokol <i>MQTT</i>	10
Gambar 2. 5 Broker <i>HiveMQ</i>	11
Gambar 2. 6 Sensor <i>JSN-SR04T</i>	12
Gambar 2. 7 Sensor <i>YF-S201</i>	12
Gambar 2. 8 <i>Amazon EC2</i>	13
Gambar 2. 9 Simbol Terminator	14
Gambar 2. 10 Simbol Proses	14
Gambar 2. 11 Simbol Decision	15
Gambar 2. 12 Simbol Penghubung	15
Gambar 2. 13 Simbol Arrow	15
Gambar 2. 14 Bahasa Pemrograman <i>Python</i>	16
Gambar 3. 1 Tahapan Penelitian	17
Gambar 4. 1 Arsitektur Sistem Keseluruhan	24
Gambar 4. 2 Diagram Alur Sistem	25
Gambar 4. 3 Blok Diagram Sistem	26
Gambar 4. 4 Skematik Catu Daya	27
Gambar 4. 5 Skematik <i>SIM800L</i>	28
Gambar 4. 6 Skematik <i>JSN-SR04T</i>	29
Gambar 4. 7 Skematik <i>YF-S201</i>	30
Gambar 4. 8 Skematik LED Traffic Light Module	31
Gambar 4. 9 Skematik Keseluruhan Alat	32
Gambar 4. 10 Skematik Alternatif Alat	33
Gambar 4. 11 Hasil Akhir Perangkat Keras	34
Gambar 4. 12 Rangkaian Alat	36
Gambar 4. 13 Kotak Pelindung Alat	37
Gambar 4. 14 Penggunaan <i>Arduino IDE</i> dan library yang digunakan	37
Gambar 4. 15 Konfigurasi Awal Fungsi Setup	38
Gambar 4. 16 Pembacaan Sensor <i>YF-S201</i>	39
Gambar 4. 17 Pembacaan Sensor <i>JSN-SR04T</i> dan LED Indikator	40
Gambar 4. 18 Program Mengirim Data ke Broker	40
Gambar 4. 19 Konfigurasi Broker <i>HiveMQ</i>	41
Gambar 4. 20 Konfigurasi <i>RBAC Extension</i>	42
Gambar 4. 21 Service Broker <i>HiveMQ</i> Yang Sedang Berjalan	42
Gambar 4. 22 Kode <i>Logger.py</i>	46
Gambar 4. 23 Kode <i>Analyze.py</i>	47
Gambar 4. 24 Hasil Analisis dari Data yang disimpan	53
Gambar 4. 25 Grafik Pengujian Availability	54
Gambar 4. 26 Grafik Pengujian Durability	55
Gambar 4. 27 Hasil Pengujian QoS yang Dicatat di Server	56

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB I PENDAHULUAN

1.1 Latar Belakang

Banjir adalah kejadian alam yang terjadi ketika suatu daerah tidak mampu menampung air dalam jumlah besar, baik karena hujan deras, aliran sungai yang meluap, atau sumber air lainnya. Banjir merupakan peristiwa di mana air sungai meluap, yang biasanya disebabkan oleh kerusakan *buffer zone* di wilayah hulu daerah aliran sungai (DAS). Peristiwa ini terjadi ketika volume aliran air di permukaan tanah melebihi kapasitas saluran drainase atau sungai, sehingga menyebabkan genangan yang dapat menimbulkan kerugian bagi manusia dan lingkungan (Eka Putri et al., 2023). Di Komplek Griya Asri, penyempitan alur sungai dan banyaknya endapan menyebabkan air sungai meluap dan menggenangi pemukiman. Hal ini diperburuk dengan kerusakan pada Situ Studio Alam yang terkadang bocor atau rusak, yang membuat banjir menyebar ke wilayah lain, seperti Pondok Sukmajaya, Taman Cipayung, Griya Pandawa, dan Sugutamu. Peningkatan frekuensi banjir dan curah hujan yang tinggi membuat pengawasan kondisi aliran sungai menjadi sangat penting untuk mengurangi kerusakan yang terjadi. Kerusakan yang disebabkan banjir termasuk pagar yang rapuh, kendaraan terendam, jalan tertutup, dan beberapa rumah yang ditinggalkan karena kerusakan parah.

Untuk mengatasi hal ini, teknologi seperti *Internet of things (IoT)* menawarkan solusi yang dapat membantu memantau kondisi lingkungan secara langsung. *Internet of things (IoT)* adalah sebuah konsep yang menghubungkan objek fisik dengan sensor, perangkat lunak, dan teknologi lainnya sehingga memungkinkan pertukaran data dengan perangkat lain melalui internet (Iqbal et al., 2023). Dalam upaya mitigasi bencana, *IoT* dapat berperan dalam memantau kondisi lingkungan secara langsung serta menyampaikan peringatan dini kepada masyarakat. Dengan menggunakan *IoT*, data ketinggian serta debit air dari sensor dapat dikirimkan langsung ke sistem yang dapat dipantau oleh warga dan pihak berwenang. Teknologi ini mempermudah pengawasan dan memungkinkan peringatan dini agar

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

masyarakat bisa lebih siap menghadapi banjir. Selain itu, teknologi ini juga dapat mengumpulkan data yang berguna untuk mempelajari pola banjir di masa depan. Penggunaan *IoT* sangat penting untuk mencegah kerugian besar dan melindungi masyarakat.

Untuk memastikan pemantauan yang lebih baik, dibutuhkan sistem pemantauan yang bisa memantau ketinggian air secara *Real-Time* dan mengirimkan data tersebut untuk dianalisis. Sistem Monitoring merupakan sistem yang dirancang untuk secara terus-menerus mengumpulkan, menganalisis, dan menyajikan informasi terkait suatu proses atau aktivitas. Tujuan utama sistem ini adalah memastikan pelaksanaan kegiatan sesuai dengan rencana yang telah ditetapkan, sekaligus mendeteksi potensi masalah yang dapat muncul selama proses berlangsung (Septanto & Hidayatullah, 2022). Dengan adanya sistem pemantauan seperti ini, masyarakat dapat memperoleh informasi tentang potensi banjir lebih cepat dan membantu pengambilan keputusan yang lebih tepat. Sistem ini sangat penting untuk mencegah dampak buruk banjir.

Tujuan dari penelitian ini adalah untuk merancang dan membangun sistem pemantauan ketinggian serta debit air berbasis *IoT* yang dapat memberikan informasi langsung tentang kondisi sungai di Komplek Griya Asri. Diharapkan, sistem ini bisa memberikan peringatan dini dan mengurangi kerusakan akibat banjir. Selain itu, sistem ini juga diharapkan bisa meningkatkan kesadaran masyarakat tentang pentingnya pemantauan lingkungan dan mendukung pengambilan keputusan yang cepat oleh instansi terkait dalam menghadapi banjir.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, permasalahan yang ingin diselesaikan adalah bagaimana mengukur dan memantau kondisi air selama banjir. Adapun rumusan masalah yang dapat diidentifikasi adalah sebagai berikut:

- a. Bagaimana sistem pemantauan ketinggian, dan debit dapat membantu mendeteksi dini potensi banjir di Komplek Griya Asri?

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- b. Seberapa akurat sensor *JSN-SR04T* dan *YF-S201* dalam mengukur ketinggian dan debit air untuk mendukung sistem peringatan dini banjir?
- c. Seberapa andal sistem komunikasi berbasis *SIM800L* dalam mengirimkan data sensor secara berkala ke broker *MQTT*, ditinjau dari aspek koneksi jaringan, keberhasilan pengiriman data, dan kualitas layanan komunikasi?

1.3 Batasan Masalah

Adapun batasan masalah yang akan dibahas dalam penelitian ini adalah sebagai berikut:

- a. Sistem yang dirancang menggunakan mikrokontroler *Arduino Nano* sebagai pengendali utama yang menghubungkan sensor *JSN-SR04T* untuk ketinggian air dan *YF-S201* untuk debit air.
- b. Penelitian ini difokuskan pada pemantauan ketinggian air sungai yang berpotensi meluap di Kali Komplek Griya Asri, khususnya saat kondisi banjir.
- c. Data hasil pembacaan sensor akan dikirim secara berkala melalui modul komunikasi *GSM SIM800L* menggunakan protokol *MQTT* ke server pemantauan.
- d. Sistem ini hanya berfungsi sebagai alat pemantauan (monitoring), tanpa dilengkapi fitur pengendalian otomatis, seperti aktivasi pompa atau alarm fisik berdasarkan kondisi air yang terdeteksi.

Batasan-batasan ini diterapkan agar penelitian tetap terarah dan sesuai dengan waktu serta sumber daya yang tersedia. Penekanan diberikan pada kestabilan sistem pemantauan dan akurasi sensor sebagai tahap awal sebelum pengembangan lebih lanjut ke sistem otomatisasi.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Adapun tujuan dari penelitian ini adalah:

- a. Merancang dan mengembangkan sistem pemantauan kondisi air berbasis *IoT* yang mampu mengukur ketinggian, dan debit air di Kali Komplek Griya Asri sebagai upaya deteksi dini banjir.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- b. Mengimplementasikan mikrokontroler *Arduino Nano* untuk mengintegrasikan sensor JSN-SR04 (ketinggian), *YF-S201* (debit), dalam sistem pemantauan.
- c. Membangun sistem komunikasi data *Real-Time* menggunakan modul *GSM SIM800L* dan protokol *MQTT* untuk memastikan pengiriman data secara efisien.
- d. Menguji fungsi dan akurasi sistem dalam mendeteksi perubahan kondisi air serta mengirimkan data untuk memastikan fungsionalitas alat dalam kondisi nyata..

1.4.2 Manfaat

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

- a. Meningkatkan efektivitas pemantauan kondisi air secara *Real-Time* dengan memperhatikan waktu respon sistem, sehingga dapat memberikan informasi yang cukup cepat untuk membantu mitigasi bencana banjir di kawasan terdampak.
- b. Mengembangkan solusi teknologi berbasis *IoT* yang memungkinkan pengumpulan data lingkungan secara otomatis dan berkelanjutan.
- c. Menyediakan data yang lebih akurat dan dapat diandalkan terkait ketinggian serta debit untuk mendukung pengambilan keputusan dalam manajemen risiko banjir.
- d. Menjadi dasar bagi penelitian lebih lanjut, khususnya dalam pengembangan sistem pemantauan lingkungan berbasis sensor dan *IoT* untuk aplikasi mitigasi bencana.

1.5 Sistematika Penulisan

Sistematika penulisan laporan skripsi ini memberikan gambaran tentang urutan penulisan yang digunakan dalam penelitian. Adapun struktur laporan ini adalah sebagai berikut:

a. BAB I PENDAHULUAN

Bab ini berisi penjelasan mengenai latar belakang masalah yang melatarbelakangi penelitian tentang pemantauan kondisi air selama banjir. Di dalamnya juga akan dibahas perumusan masalah yang ingin

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

dipecahkan, batasan masalah yang diberlakukan dalam penelitian, serta tujuan dan manfaat dari penelitian ini. Di akhir bab ini, akan dijelaskan mengenai gambaran umum dari struktur laporan penelitian.

b. BAB II TINJAUAN PUSTAKA

Bab ini menguraikan berbagai teori yang mendasari penelitian ini, mencakup penjelasan mengenai sensor yang digunakan untuk mengukur ketinggian air, serta debit. Selain itu, bab ini juga akan membahas teknologi mikrokontroler yang digunakan, serta penerapan teknologi komunikasi GSM 2G dalam pengiriman data secara *Real-Time*.

c. BAB III METODE PENELITIAN

Pada bab ini, akan dijelaskan mengenai rancangan penelitian yang mencakup pengembangan sistem pemantauan kondisi air. Di dalamnya dijelaskan tahapan penelitian, objek penelitian, serta metode atau teknik yang digunakan dalam pembuatan sistem. Bagian ini juga menguraikan teknik pengumpulan dan analisis data yang akan dilakukan, bersama dengan jadwal pelaksanaan dan perkiraan anggaran yang dibutuhkan dalam penelitian ini.

d. BAB IV HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil implementasi dari sistem yang telah dirancang dan dikembangkan. Di dalamnya mencakup analisis kebutuhan, perancangan sistem secara menyeluruh (baik perangkat keras maupun perangkat lunak), serta implementasi dari masing-masing bagian sistem. Selain itu, bab ini juga membahas pengujian sistem, termasuk pengujian sensor, pengiriman data, dan evaluasi performa sistem berdasarkan parameter seperti akurasi, *Availability*, *Durability*, dan *Delay*. Hasil dari pengujian tersebut dianalisis untuk menunjukkan efektivitas dan keandalan sistem yang dibangun.

e. BAB V PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian berdasarkan keseluruhan proses mulai dari perancangan hingga pengujian sistem. Kesimpulan

memuat pencapaian terhadap tujuan penelitian, serta keandalan sistem yang dikembangkan. Selain itu, diberikan pula saran sebagai masukan untuk pengembangan sistem lebih lanjut, baik dari sisi teknis maupun dari sisi implementasi di lapangan, agar sistem dapat digunakan secara lebih optimal dalam mendukung mitigasi banjir.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Kajian terhadap penelitian-penelitian sebelumnya menjadi elemen penting dalam membangun dasar teori dan pendekatan yang tepat untuk penelitian. Dalam pengembangan Sistem Monitoring Pengawasan Tinggi Air sebagai Pencegahan Dini Banjir Berbasis *IoT* pada Komplek Griya Asri Depok, penulis merujuk pada berbagai referensi yang relevan untuk mendukung konsep, metode, dan implementasi sistem yang dirancang.

Tabel 2. 1 Penelitian Terkait Sebelumnya

Jurnal	Hasil	Catatan
(Sari et al., 2021)	Tujuan : Memantau ketinggian air sebagai deteksi dini banjir menggunakan aplikasi Telegram. Teknologi : <i>IoT</i>, Sensor Ultrasonik, Aplikasi Telegram. Cara Kerja : Sensor ultrasonik mengukur ketinggian air dan mengirimkan data ke aplikasi Telegram untuk notifikasi <i>Real-Time</i> kepada pengguna	Sistem ini efektif dalam memberikan informasi cepat, tetapi bergantung pada koneksi internet untuk pengiriman data.
(Prabowo et al., 2020)	Tujuan : Merancang prototipe pendeteksi banjir berbasis <i>IoT</i>. Teknologi : NodeMCU, Sensor Ultrasonik, Blynk. Cara Kerja : Sensor ultrasonik mengukur ketinggian air dan mengirimkan data ke aplikasi Blynk untuk monitoring.	Keterbatasan pada penyimpanan data dan ketergantungan pada aplikasi Blynk yang berbayar untuk fitur tertentu.
(Nafis., 2021)	Tujuan : Mengembangkan sistem monitoring untuk mitigasi banjir menggunakan NodeMCU ESP8266. Teknologi : NodeMCU ESP8266, Sensor Ultrasonik, Database. Cara Kerja : Sistem memantau debit dan volume air, serta memberikan notifikasi kepada pengguna melalui aplikasi web.	Sistem ini dapat menyimpan data historis, tetapi perlu pengembangan lebih lanjut untuk integrasi dengan sistem peringatan dini.
(Durrun Nafis, 2021)	Tujuan : Memantau debit, volume, dan jarak permukaan air pada sungai.	Sistem ini efektif untuk monitoring, tetapi memerlukan

Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

	Teknologi : Sensor Ultrasonik, <i>IoT</i> , Database. Cara Kerja : Sensor mengukur parameter air dan mengirimkan data ke server untuk analisis lebih lanjut.	pemeliharaan rutin untuk sensor agar tetap akurat.
(Anorgi et al., 2023)	Tujuan : Rancang bangun prototype monitoring banjir berbasis website. Teknologi : Website, Sensor Ketinggian Air, Database. Cara Kerja : Data dari sensor ketinggian air ditampilkan di website untuk monitoring oleh pengguna.	Penggunaan website memudahkan akses informasi, tetapi perlu perhatian pada keamanan data dan aksesibilitas.

Sebagian besar penelitian sebelumnya masih bergantung pada koneksi WiFi dan menggunakan aplikasi pihak ketiga seperti Telegram atau Blynk untuk menampilkan informasi kondisi air. Pendekatan ini menjadi kurang efektif jika diterapkan di lokasi seperti bantaran sungai yang minim infrastruktur jaringan WiFi. Selain itu, aplikasi Telegram cenderung kurang umum digunakan oleh masyarakat umum, sehingga potensi peringatan dini tidak tersampaikan dengan optimal.

Penelitian ini mengatasi keterbatasan tersebut dengan mengimplementasikan modul *SIM800L* yang memanfaatkan jaringan *GSM (2G)* untuk mengirimkan data sensor ke broker *MQTT* secara berkala. Sistem ini juga mengirimkan peringatan yang lebih familiar dan mudah dijangkau oleh masyarakat. Dengan pendekatan ini, sistem menjadi lebih relevan untuk diterapkan di lingkungan sebenarnya dan mampu memberikan peringatan dini secara efektif berdasarkan pemantauan ketinggian air sebagai parameter utama.

2.2 Sistem Pengawasan Pencegahan Banjir

Sistem pengawasan adalah suatu rangkaian perangkat dan teknologi yang digunakan untuk memantau, merekam, dan menganalisis aktivitas atau kondisi di suatu area tertentu. Sistem ini sering terdiri dari komponen seperti kamera CCTV, sensor gerak, perangkat perekam, dan perangkat lunak analisis yang memungkinkan pengguna untuk mengawasi situasi secara *Real-Time* atau merekam untuk analisis di kemudian hari. (Waskito et al., 2024).

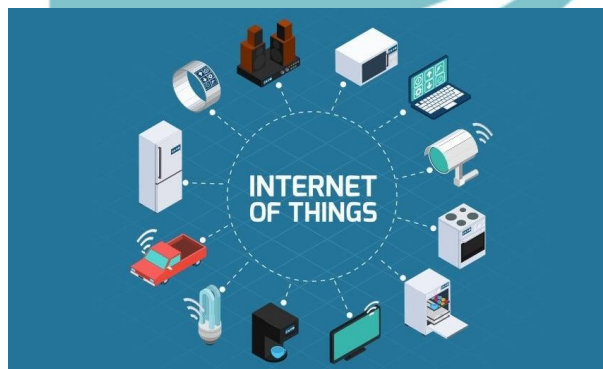
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Dalam konteks penelitian ini, sistem pengawasan diterapkan untuk memantau kondisi air di kali yang berada di Komplek Griya Asri sebagai upaya mitigasi banjir. Dengan memanfaatkan sensor ketinggian air, debit air, dan kualitas air yang terhubung dalam sistem berbasis *Internet of things (IoT)*, data pemantauan dapat diakses secara *Real-Time* melalui platform web. Sistem ini dirancang untuk memberikan peringatan dini terhadap potensi banjir, sehingga pengelola kawasan dapat mengambil langkah pencegahan yang tepat waktu dan berbasis data.

2.3 *Internet of things (IoT)*

Internet of things (IoT) merupakan sebuah konsep yang menghubungkan berbagai perangkat melalui jaringan internet, sehingga perangkat-perangkat tersebut dapat saling berkomunikasi. Dengan adanya *IoT*, perangkat mampu mengumpulkan serta berbagi data, menciptakan sistem yang lebih pintar dan terintegrasi (Istiqomah et al., 2022).



Gambar 2. 1 *Internet of things*

2.4 *Arduino Nano*

Arduino Nano adalah sebuah *Board* mikrokontroler yang berbasis pada *ATmega328*, yang dirancang untuk aplikasi yang memerlukan ukuran kecil dan kemampuan pemrograman yang fleksibel. *Board* ini dilengkapi dengan sejumlah pin *input/output* digital dan analog, yang memungkinkan pengguna untuk menghubungkan berbagai sensor dan perangkat lainnya. *Arduino Nano* sering digunakan dalam proyek-proyek elektronik dan robotika karena kemudahan penggunaannya, dukungan komunitas yang luas, serta kemampuan untuk diprogram menggunakan lingkungan pengembangan

Hak Cipta :

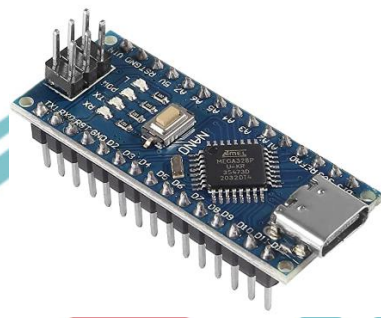
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Arduino yang intuitif (Widhiawan et al., 2021). Pemilihan *Arduino Nano* pada proyek ini karena desainnya yang compact sehingga cocok untuk aplikasi dengan ruang terbatas, serta jumlah pin yang disediakan cukup untuk memenuhi kebutuhan proyek ini.



Gambar 2. 2 Modul *Arduino Nano*

2.5 Modul GSM 2G SIM800L

SIM800L adalah sebuah modul komunikasi *GSM/GPRS* yang dirancang untuk memungkinkan perangkat berkomunikasi melalui jaringan seluler 2G. Menurut (Pandiang, 2021), *SIM800L* memungkinkan komunikasi nirkabel melalui jaringan seluler, mendukung frekuensi quad-band (850/900/1800/1900 MHz). Modul ini dirancang untuk mengirim dan menerima SMS, melakukan panggilan suara, dan terhubung ke internet, sehingga sering digunakan dalam berbagai aplikasi *IoT* (*Internet of things*) dan sistem kendali jarak jauh. Dalam penelitian ini, *SIM800L* dipilih karena mampu mengirim data melalui jaringan seluler di lokasi yang tidak memiliki akses *WiFi*, seperti area sekitar sungai. Meskipun menggunakan jaringan 2G, kemampuannya dinilai sudah memadai untuk pengiriman data sensor secara berkala dalam sistem pemantauan yang dirancang.



Gambar 2. 3 Modul *SIM800L*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.6 Protokol MQTT

MQTT (*Message Queuing Telemetry Transport*) adalah protokol jaringan ringan yang dirancang untuk komunikasi *machine-to-machine* (*M2M*) dan *Internet of things* (*IoT*). Protokol ini menggunakan model *publish/subscribe*, di mana perangkat (*publisher*) mengirimkan pesan ke broker, yang kemudian mendistribusikan pesan tersebut kepada perangkat lain (*subscriber*) yang telah berlangganan topik tertentu. *MQTT* sangat efisien dalam penggunaan bandwidth dan sumber daya, sehingga ideal untuk aplikasi yang memerlukan komunikasi *Real-Time* dengan latensi rendah, seperti pengendalian perangkat dan monitoring data dalam sistem *IoT* (Kurnianto et al., 2022)).

Pada (Hanif & Amri, 2023), dalam perbandingan dengan HTTP (*Hypertext Transfer Protocol*), *MQTT* lebih unggul dalam hal responsivitas dan efisiensi, terutama untuk aplikasi *Real-Time* seperti sistem deteksi banjir. Penelitian menunjukkan bahwa *MQTT* memiliki *Delay* lebih rendah (185.75 ms dibandingkan 361.80 ms pada HTTP) serta *Throughput* lebih tinggi, memungkinkan pengiriman data yang lebih cepat dan efisien. Meskipun HTTP lebih umum digunakan dalam komunikasi berbasis web, *MQTT* lebih cocok untuk aplikasi *IoT* yang membutuhkan kecepatan dan latensi rendah.



Gambar 2. 4 Protokol MQTT

2.7 MQTT Broker

Broker *MQTT* adalah komponen server yang berfungsi sebagai perantara dalam komunikasi antara perangkat yang menggunakan protokol *MQTT*. Broker menerima pesan dari *publisher* dan mendistribusikannya kepada *subscriber* yang telah berlangganan topik tertentu. Dengan mengelola koneksi dan pengiriman pesan, broker memastikan bahwa data dapat dikirim secara

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

efisien dan tepat waktu, serta mengatur kualitas layanan (*Quality of Service*) untuk menjamin pengiriman pesan. Broker *MQTT* juga dapat menyimpan pesan untuk subscriber yang tidak aktif, sehingga mereka dapat menerima pesan yang hilang saat mereka kembali online. Contoh broker *MQTT* yang umum digunakan termasuk Mosquitto, *HiveMQ*, dan Adafruit.IO (Kurnianto et al., 2022).

Dalam penelitian ini, peneliti menggunakan *HiveMQ* sebagai broker *MQTT* karena dapat dijalankan langsung di server tanpa biaya lisensi, memiliki stabilitas tinggi, dan kompatibel dengan modul *SIM800L* yang digunakan dalam sistem. *HiveMQ* mendukung berbagai level *Quality of Service* (QoS), dan sistem ini menggunakan QoS 0 karena paling sesuai dengan karakteristik *SIM800L* yang beroperasi pada jaringan 2G. QoS 0 memungkinkan pengiriman data tanpa perlu konfirmasi dari penerima (at most once), sehingga lebih ringan dan stabil saat digunakan dalam komunikasi data berkala. Penggunaan QoS 0 juga menghindari potensi kegagalan handshake yang kerap terjadi pada level QoS yang lebih tinggi, menjadikannya pilihan paling realistis dan efisien dalam konteks sistem yang dikembangkan.



Gambar 2. 5 Broker *HiveMQ*

2.8 Sensor Ultrasonik *JSN-SR04T*

JSN-SR04 adalah modul sensor ultrasonik yang digunakan untuk mengukur jarak dengan memanfaatkan gelombang ultrasonik. Sensor ini terdiri dari dua komponen utama: pemancar (*transmitter*) dan penerima (*receiver*). Ketika sensor diaktifkan, pemancar mengeluarkan gelombang ultrasonik yang memantul kembali setelah mengenai objek, dan penerima menangkap gelombang yang dipantulkan tersebut. Dengan menghitung waktu yang dibutuhkan gelombang untuk kembali, sensor dapat menentukan jarak antara

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

sensor dan objek dengan akurasi yang tinggi. JSN-SR04 sering digunakan dalam berbagai aplikasi, termasuk robotika, pengukuran jarak, dan sistem deteksi, karena kemudahan penggunaan dan biaya yang relatif rendah (Indra Dharmawan & Imelda Imelda, 2024). Selain itu, tipe *JSN-SR04T* yang digunakan dalam penelitian ini memiliki keunggulan berupa desain tahan air (waterproof) dan kabel bawaan yang cukup panjang, sehingga cocok untuk penggunaan di luar ruangan seperti pemantauan tinggi air pada aliran sungai.



Gambar 2. 6 Sensor JSN-SR04T

2.9 YF-S201

YF-S201 adalah sensor aliran air yang digunakan untuk mengukur laju aliran air dalam sistem pipa atau saluran. Sensor ini bekerja dengan prinsip mekanis, di mana aliran air yang melewati sensor akan memutar rotor di dalamnya, menghasilkan pulsa listrik yang proporsional dengan kecepatan aliran air. Setiap pulsa yang dihasilkan dapat dihitung untuk menentukan volume air yang mengalir. *YF-S201* sering digunakan dalam aplikasi pemantauan penggunaan air, sistem irigasi otomatis, dan proyek-proyek berbasis *Internet of things (IoT)* karena kemudahan penggunaannya, biaya yang terjangkau, dan kemampuan untuk memberikan data *Real-Time* tentang aliran air (Muamaroh & Christanto, 2024).



Gambar 2. 7 Sensor YF-S201

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.10 AWS

Amazon Web Services (AWS) adalah layanan cloud computing yang menyediakan infrastruktur digital berbasis internet, seperti penyimpanan, jaringan, dan komputasi server. Salah satu layanan yang umum digunakan adalah *Amazon Elastic Compute Cloud (EC2)*, yaitu server virtual yang memungkinkan pengguna menjalankan aplikasi atau sistem secara fleksibel dan dapat diakses dari mana saja (Syahrul Mubarak & Izman Herdiansyah, 2023).

Dalam penelitian ini, layanan *EC2* digunakan untuk menjalankan broker *MQTT HiveMQ* sebagai pusat komunikasi data sensor dari perangkat *IoT*. Dengan menggunakan *EC2*, sistem tidak bergantung pada infrastruktur lokal, serta mendukung penerapan plugin seperti *RBAC (Role-Based Access Control)* untuk membatasi akses hanya kepada perangkat atau pengguna yang berwenang. Selain itu, *EC2* juga digunakan dalam proses pencatatan dan pengujian sistem, untuk mengevaluasi performa berdasarkan parameter seperti *Availability*, *Durability*, dan *Delay*.



Gambar 2. 8 Amazon EC2

2.11 Flowchart

Flowchart adalah representasi grafis dari urutan langkah-langkah dalam suatu proses atau program. *Flowchart* membantu programmer dan analis sistem untuk memecah suatu permasalahan ke dalam tahapan-tahapan logis yang lebih kecil, sehingga lebih mudah dianalisis dan diimplementasikan ke dalam bentuk kode program. Dengan menggunakan simbol-simbol standar, *Flowchart* mampu menggambarkan alur kerja sistem secara sistematis dan mudah dipahami (Zalukhu et al., 2023).

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Dalam penelitian ini, *Flowchart* digunakan untuk memodelkan alur proses dari sistem pemantauan ketinggian dan debit air berbasis *IoT*. Proses yang digambarkan meliputi pembacaan data sensor, pengambilan keputusan berdasarkan ambang batas, pengiriman data ke broker *MQTT*, dan pengendalian notifikasi atau indikator.

Berikut adalah beberapa simbol *Flowchart* yang digunakan dalam sistem ini:

- *Terminator*

Simbol berbentuk oval yang digunakan untuk menunjukkan awal (start) dan akhir (end) dari proses.



Gambar 2. 9 Simbol Terminator

- Proses (*Process*)

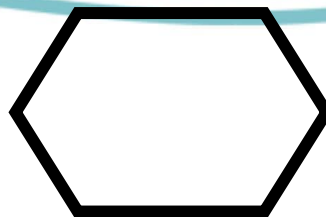
Digambarkan dengan persegi panjang. Menunjukkan aktivitas atau instruksi yang dijalankan oleh sistem, seperti membaca sensor, menginisialisasi perangkat, atau mengirim data.



Gambar 2. 10 Simbol Proses

- Persiapan (*preparation*)

Flowchart persiapan menunjukkan langkah awal sebelum proses utama dimulai. Biasanya berisi inisialisasi variabel, sensor, atau koneksi. Tujuannya untuk memastikan semua komponen siap digunakan.



Gambar 2. 11 Simbol Preparation

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

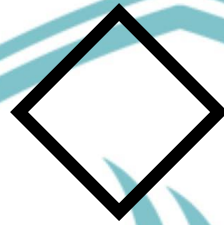
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- Keputusan (*Decision*)

Digambarkan dengan bentuk belah ketupat. Menandakan adanya percabangan logika, misalnya apakah nilai ketinggian air melebihi ambang batas tertentu.



Gambar 2. 12 Simbol Decision

- Connector

Digambarkan dengan lingkaran kecil berisi huruf. Digunakan untuk menghubungkan alur yang terputus agar *Flowchart* tetap ringkas.



Gambar 2. 13 Simbol Penghubung

- Panah (*Arrow*)

Menunjukkan arah alur dari satu proses ke proses lain secara logis dan berurutan, baik dari atas ke bawah maupun dari kiri ke kanan.



Gambar 2. 14 Simbol Arrow

Dalam pembuatannya, *Flowchart* sistem mengikuti aturan dasar seperti urutan proses dari atas ke bawah, penggunaan deskripsi kata kerja dalam setiap langkah, dan penempatan percabangan yang jelas. Struktur *Flowchart* yang dibuat juga mencerminkan prinsip dasar algoritma, yaitu terdiri dari *input*, proses, dan *output*, yang menjadi dasar dalam pemrograman logika system (Zalukhu et al., 2023).

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.12 Python

Python merupakan bahasa pemrograman tingkat tinggi yang dirancang dengan sintaksis sederhana, sehingga mudah dipahami dan digunakan oleh pemula maupun pengembang berpengalaman. *Python* bersifat *open source*, mendukung berbagai paradigma pemrograman, dan telah menjadi salah satu bahasa yang paling populer di bidang komputasi data, pengembangan web, dan sistem reintegrasi (Alfarizi et al., 2023).



Gambar 2. 15 Bahasa Pemrograman Python

Dalam penelitian ini, *Python* digunakan untuk membantu proses pencatatan dan pengujian sistem pemantauan ketinggian dan debit air. Program *Python* yang dibuat berfungsi untuk menerima data dari broker *MQTT*, lalu mencatatnya dalam file log. Selain itu, *Python* juga digunakan untuk melakukan analisis terhadap data tersebut guna mengevaluasi performa sistem, seperti keterlambatan pengiriman data (*Delay*) dan ketersediaan layanan (*Availability*). Dengan menggunakan *Python*, proses pengujian menjadi lebih efisien karena data dapat dicatat dan diproses secara otomatis.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB III

METODE PENELITIAN

3.1 Rancangan Penelitian

Penelitian ini mengembangkan sistem pemantauan banjir berbasis *Internet of things (IoT)* di Komplek Griya Asri dengan memanfaatkan *Arduino Nano*, sensor *JSN-SR04T* untuk mengukur ketinggian air, sensor *YF-S201* untuk debit air,. Sistem ini dirancang untuk mengolah data dari sensor-sensor tersebut secara otomatis, kemudian mengirimkan hasil pengukuran ke server melalui modul *SIM800L* yang menggunakan jaringan *GSM 2G*. Data yang dikirimkan oleh perangkat *IoT* akan disalurkan melalui protokol *MQTT* ke broker yang terhubung dengan server. Setelah sampai di broker, data akan disimpan dalam database yang terkelola dengan baik untuk keperluan pemrosesan dan analisis lebih lanjut. Data yang telah tersimpan di database ini kemudian dapat dipanggil dan ditampilkan secara *Real-Time* pada platform web sebagai sarana pemantauan utama.

3.2 Tahapan Penelitian

Tahapan penelitian yang akan dijalankan melibatkan:



Gambar 3. 1 Tahapan Penelitian

Tahapan penelitian yang mengadopsi metode *Research Development* sangat cocok untuk penelitian yang bertujuan mengembangkan suatu produk baru, dalam hal ini adalah sistem monitoring ketinggian air.

1. Studi Literatur

Tahap studi literatur bertujuan untuk mendapatkan pemahaman mendalam mengenai konsep, metode, dan teknologi yang relevan dengan penelitian ini. Dengan mengumpulkan referensi dari jurnal, artikel

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

ilmiah, dan penelitian terdahulu, penelitian ini dapat mengidentifikasi metode yang telah digunakan, menemukan kelemahan yang masih ada, serta mencari solusi yang lebih optimal. Studi literatur juga membantu dalam menentukan pendekatan yang paling sesuai untuk merancang sistem pemantauan ketinggian air berbasis *IoT*, sehingga penelitian ini memiliki dasar yang kuat dan relevan dengan perkembangan teknologi terkini.

2. Pengumpulan Data

Pengumpulan data dilakukan untuk memahami kondisi nyata di lapangan serta kebutuhan pengguna terhadap sistem yang dikembangkan. Wawancara dengan warga Komplek Griya Asri Depok bertujuan untuk mendapatkan informasi mengenai pengalaman mereka terkait banjir, seberapa sering mereka menghadapi permasalahan tersebut, serta ekspektasi mereka terhadap sistem pemantauan yang akan dibuat. Data ini sangat penting karena akan menjadi dasar dalam menentukan fitur utama sistem, seperti tingkat akurasi yang diharapkan, metode notifikasi yang paling efektif, serta parameter lingkungan yang perlu diperhatikan dalam perancangan sistem.

3. Perancangan Sistem

Pada tahap ini, sistem mulai dirancang berdasarkan hasil studi literatur dan data yang telah dikumpulkan sebelumnya. Perancangan sistem bertujuan untuk memastikan bahwa sistem yang dikembangkan memiliki struktur yang jelas dan dapat berfungsi sesuai dengan kebutuhan yang telah diidentifikasi. Proses ini mencakup pembuatan *Flowchart* untuk menggambarkan alur kerja sistem secara menyeluruh, serta desain perangkat keras dan perangkat lunak yang akan digunakan. Dengan adanya perancangan yang matang, implementasi sistem nantinya dapat dilakukan secara lebih terarah dan efisien.

4. Implementasi

Tahap implementasi bertujuan untuk merealisasikan rancangan sistem menjadi bentuk yang dapat diuji dan digunakan. Proses ini melibatkan penggabungan berbagai komponen perangkat keras seperti sensor JSN-

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SR04 dan YF-S201 dengan mikrokontroler *Arduino Nano*. Selain itu, dilakukan integrasi dengan protokol komunikasi *MQTT* untuk menghubungkan sistem dengan server dan memastikan data dapat dikirim serta diterima secara *Real-Time*. Implementasi ini juga mencakup pengembangan perangkat lunak, baik di sisi firmware untuk mikrokontroler maupun platform pemantauan berbasis web untuk pengguna.

5. Pengujian

Setelah implementasi sistem selesai, dilakukan beberapa tahap pengujian untuk memastikan bahwa seluruh komponen bekerja sesuai dengan spesifikasi. Pengujian yang dilakukan meliputi: (1) pengujian fungsionalitas dan akurasi sensor *JSN-SR04T* dan *YF-S201* dengan membandingkan hasil sensor terhadap pengukuran manual, (2) pengujian keandalan sistem melalui analisis *Availability* dan *Durability* selama sistem berjalan secara kontinu, serta (3) pengujian kualitas layanan (*Quality of Service*) berupa pengukuran *Delay* dan *Throughput* dari proses pengiriman data ke broker *MQTT* menggunakan modul *SIM800L*. Hasil pengujian ini menjadi acuan dalam menilai kelayakan sistem serta menentukan aspek yang perlu disempurnakan.

6. Evaluasi dan Analisis

Setelah pengujian dilakukan, hasilnya dianalisis untuk menilai apakah sistem telah mencapai target penelitian yang telah ditetapkan. Evaluasi ini bertujuan untuk mengidentifikasi kelebihan dan kekurangan sistem, serta mencari solusi jika terdapat aspek yang masih perlu diperbaiki. Data yang dikumpulkan dari pengujian dibandingkan dengan spesifikasi awal sistem dan kebutuhan pengguna yang telah ditentukan sebelumnya. Dari hasil evaluasi ini, dapat ditarik kesimpulan mengenai tingkat keberhasilan sistem dalam memberikan solusi mitigasi banjir berbasis *IoT* di Komplek Griya Asri.

7. Penyusunan Laporan Penelitian

Tahap terakhir adalah penyusunan laporan penelitian, yang bertujuan untuk mendokumentasikan seluruh proses yang telah dilakukan dari awal

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

hingga akhir. Laporan ini harus disusun sesuai dengan pedoman akademik yang berlaku, mencakup latar belakang masalah, metodologi, hasil pengujian, analisis, serta kesimpulan dari penelitian. Laporan ini tidak hanya berfungsi sebagai dokumentasi akademik, tetapi juga sebagai referensi bagi penelitian selanjutnya yang ingin mengembangkan atau menyempurnakan sistem pemantauan ketinggian air berbasis *IoT*.

3.3 Objek Penelitian

Fokus penelitian ini adalah Rancang Bangun Sistem Pengawasan Tinggi Air yang terhubung dengan sensor pendeteksian JSN Ultrasonic, *YF-S201*. Hal ini memungkinkan memberikan kemudahan bagi warga Komplek Griya Asri, Kota Depok untuk melakukan pemantauan ketinggian air sebagai pencegahan dini banjir di sekitar lingkungannya. Sistem akan dirancang dengan komunikasi antar perangkat menggunakan protokol *MQTT* melalui broker *HiveMQ* dilanjutkan dengan menampilkan informasi hasil pengukuran secara akurat dan *Real-Time*, melalui Website.



**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB IV

HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan

Sistem pemantauan ini dirancang untuk mendeteksi kondisi air secara *Real-Time* sebagai bagian dari upaya mitigasi banjir. Oleh karena itu, sistem harus mampu mengukur ketinggian air dan debit air dengan sensor yang sesuai, serta mengirimkan data secara rutin ke broker *MQTT* melalui jaringan seluler. Informasi kondisi air di lapangan ditampilkan secara langsung menggunakan *LED* indikator berwarna hijau, kuning, dan merah, berdasarkan kombinasi nilai ketinggian dan debit air. Sistem juga perlu dirancang agar dapat beroperasi secara mandiri di luar ruangan dan tetap stabil dalam jangka waktu tertentu tanpa perlu pengawasan terus-menerus. Kebutuhan-kebutuhan tersebut menjadi dasar dalam pemilihan komponen, perancangan perangkat, serta metode implementasi dan pengujian.

Untuk memenuhi kebutuhan sistem yang dapat bekerja secara optimal dalam memantau kondisi air secara *Real-Time*, maka dilakukan identifikasi terhadap komponen perangkat keras dan perangkat lunak yang digunakan. Identifikasi ini penting untuk memastikan bahwa setiap elemen dalam sistem mampu menjalankan fungsinya dengan baik dan saling terintegrasi.

Berikutnya, Tabel 4.1 menyajikan analisis kebutuhan perangkat keras yang digunakan dalam sistem pemantauan. Tabel ini merinci komponen fisik utama, fungsi masing-masing perangkat, serta spesifikasinya yang mendukung keberhasilan implementasi sistem secara keseluruhan:

Tabel 4. 1 Analisis Kebutuhan Perangkat Keras

No	Perangkat Keras	Fungsi	Spesifikasi
1	Arduino Nano	Mengontrol seluruh sistem dan membaca data dari sensor	Mikrokontroler ATmega328P, 5V, 16 MHz, 22 pin I/O

Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2	<i>SIM800L</i>	Mengirim data sensor ke broker <i>MQTT</i> melalui jaringan seluler	Modul <i>GSM/GPRS</i> , tegangan kerja 3.4–4.4V, mendukung AT Command
3	Sensor <i>JSN-SR04T</i>	Mengukur ketinggian air secara non-kontak	Sensor ultrasonik <i>waterproof</i> , jarak 20–600 cm, tegangan 5V
4	Sensor <i>YF-S201</i>	Mengukur debit air berdasarkan jumlah pulsa air yang mengalir	Tegangan 5V
5	Stepdown (Buck Converter)	Menurunkan tegangan dari power supply ke tegangan yang sesuai modul	Input hingga 24V, output adjustable (4.2V untuk <i>SIM800L</i>)
6	<i>Power Supply</i>	Menyediakan sumber daya ke seluruh sistem	Adaptor DC 5V, 3A
7	<i>LED Traffic Light Mini Module</i>	Menunjukkan status kondisi air dengan indikator visual warna	Tiga <i>LED</i> (merah, kuning, hijau), tegangan kerja 5V
8	<i>Project Box</i>	Melindungi rangkaian dari air dan gangguan luar	Ukuran 14.5 × 9.5 × 5 cm

Setelah perangkat keras ditentukan, keberhasilan sistem juga sangat bergantung pada perangkat lunak yang digunakan untuk memprogram, mengelola komunikasi, serta menganalisis data hasil pemantauan.

Tabel 4.2 berikut ini menyajikan analisis kebutuhan perangkat lunak yang diperlukan dalam proses implementasi sistem. Masing-masing perangkat

lunak memiliki peran penting, mulai dari pemrograman mikrokontroler, pengiriman data, hingga evaluasi performa sistem melalui data log yang dikirimkan oleh perangkat:

Tabel 4. 2 Analisis Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Fungsi
1	<i>Arduino IDE</i>	Menulis dan mengunggah program ke <i>Arduino Nano</i>
2	<i>C++</i>	Bahasa yang digunakan untuk menulis logika pembacaan sensor, kontrol <i>LED</i> , dan komunikasi <i>SIM800L</i>
3	<i>HiveMQ</i>	Broker <i>MQTT</i> yang menerima data sensor, dijalankan di server <i>EC2</i>
4	<i>AWS EC2</i>	Menjadi host server untuk <i>HiveMQ</i> , menyediakan konektivitas <i>MQTT</i> via IP publik
5	<i>Python</i>	Mengambil dan menganalisis data dari broker <i>MQTT</i> untuk mengecek keberhasilan pengiriman serta mengevaluasi <i>Availability</i> dan <i>Durability</i> sistem

4.2 Perancangan Sistem

Perancangan sistem menjelaskan bagaimana penulis merancang sistem pemantauan ketinggian dan debit air berbasis *Internet of things (IoT)*. Perancangan dilakukan untuk memastikan seluruh komponen dapat saling terhubung dan berfungsi sesuai dengan kebutuhan sistem. Pada bagian ini, dijelaskan rancangan blok diagram dan *Flowchart* dari sistem yang telah dibuat. Sebelumnya Gambar dibawah ini memberikan gambaran umum mengenai hubungan antar komponen utama yang membentuk sistem ini.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

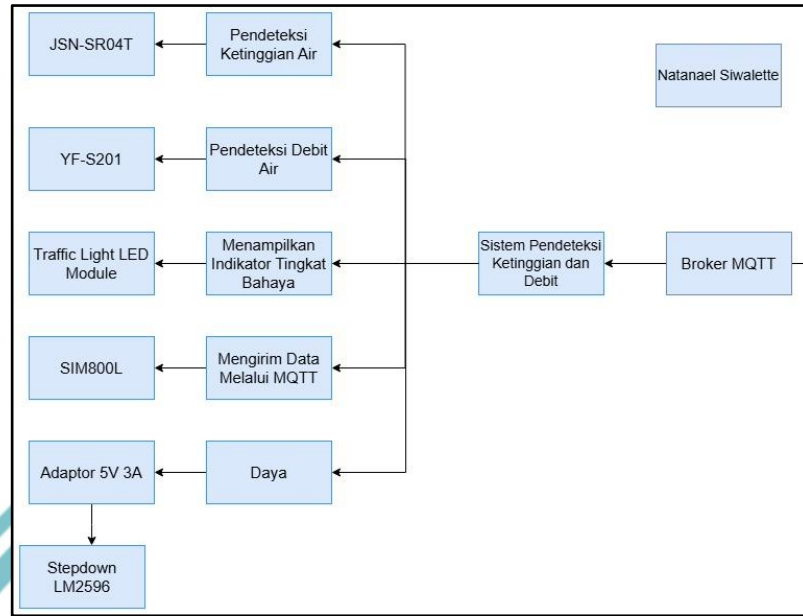
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 1 Arsitektur Sistem Keseluruhan

Sistem pada Gambar 4.1 berfungsi untuk mendeteksi ketinggian dan debit air secara otomatis menggunakan sensor *JSN-SR04T* (untuk ketinggian) dan *YF-S201* (untuk debit). Data dari sensor diproses oleh sistem yang juga menampilkan indikator tingkat bahaya menggunakan *Traffic Light LED Module*. Data yang telah diolah kemudian dikirim melalui modul *SIM800L* menggunakan protokol komunikasi *MQTT*. Untuk mendukung kebutuhan tegangan *SIM800L* yang lebih rendah, digunakan modul step-down *LM2596* khusus untuk menurunkan tegangan dari adaptor 5V 3A.

Hak Cipta :

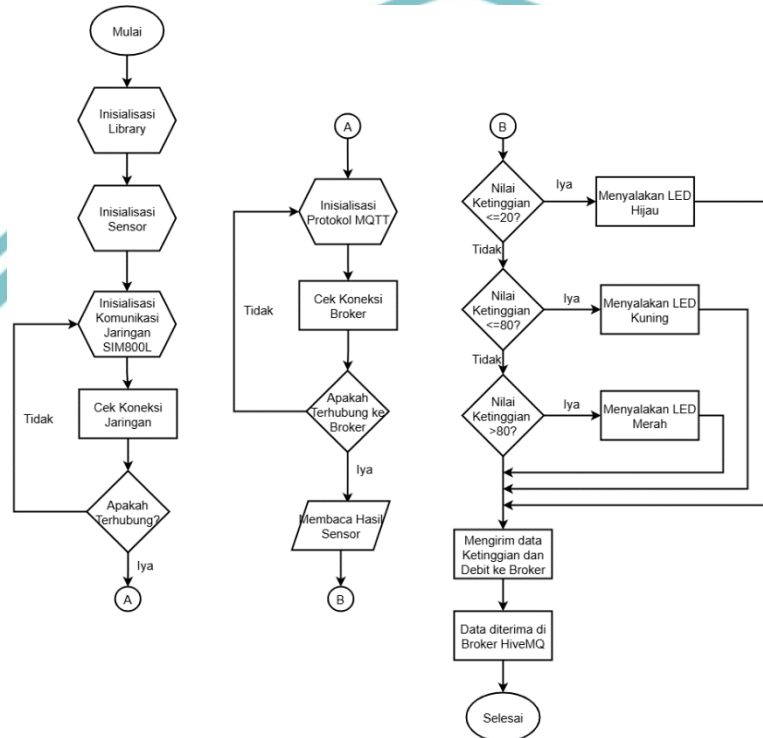
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

4.2.1 Flowchart Sistem

Flowchart berikut menggambarkan alur kerja sistem monitoring ketinggian air secara keseluruhan, mulai dari inialisasi hingga pengiriman data ke broker *MQTT*.



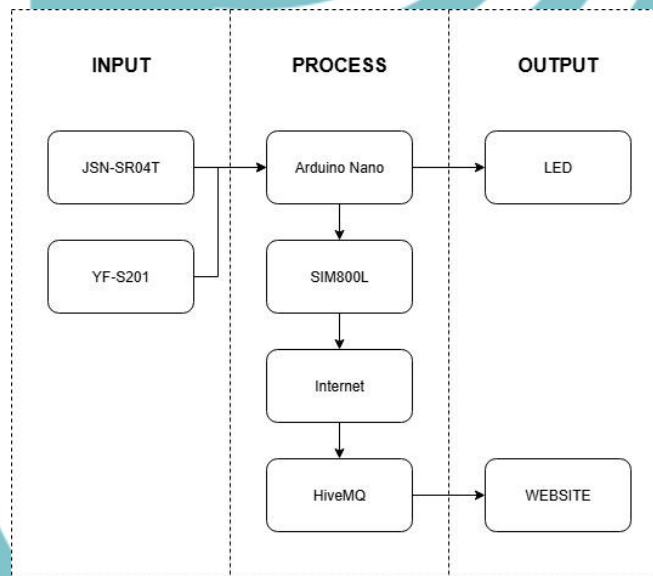
Gambar 4. 2 Diagram Alur Sistem

Flowchart sistem pada Gambar 4.2 menggambarkan alur kerja dari sistem pemantauan ketinggian air berbasis sensor dengan koneksi *MQTT* melalui jaringan *SIM800L*. Proses dimulai dengan inialisasi library, pin, dan sensor, yaitu *JSN-SR04T* untuk pengukuran ketinggian air dan *YF-S201* untuk pengukuran debit air. Setelah itu, sistem menginisialisasi komunikasi jaringan menggunakan modul *SIM800L*. Jika koneksi jaringan berhasil, sistem melanjutkan ke inialisasi protokol *MQTT* dan melakukan pengecekan koneksi ke broker. Bila koneksi ke broker berhasil, sistem mulai membaca data dari sensor. Sensor *JSN-SR04T* mengukur jarak permukaan air dari sensor, dan hasilnya dikonversi menjadi nilai ketinggian air aktual berdasarkan tinggi maksimum wadah atau area pemantauan. Nilai ini kemudian digunakan untuk menentukan status kondisi ketinggian air. Apabila nilai ketinggian yang terdeteksi kurang dari 20 cm, maka

sistem akan menyalakan *LED* hijau yang menunjukkan kondisi aman. Jika ketinggian berada pada rentang antara 20 cm hingga 80 cm, *LED* kuning akan menyala sebagai indikator kondisi siaga. Sedangkan jika ketinggian air melebihi 80 cm, maka *LED* merah akan menyala sebagai peringatan bahwa kondisi sudah memasuki tahap waspada atau bahaya. Selain memberikan indikator visual, sistem juga mengirimkan data ketinggian air dan debit air ke broker *MQTT* secara berkala untuk kemudian ditampilkan di *dashBoard* pemantauan jarak jauh. Setelah seluruh proses selesai, sistem akan mengulangi kembali proses pembacaan sensor, pengiriman data, dan penentuan status secara terus-menerus untuk memastikan pemantauan dilakukan secara *Real-Time*.

4.2.2 Diagram Blok

Berikut merupakan gambar diagram blok dari sistem yang dirancang:



Gambar 4. 3 Blok Diagram Sistem

Pada Diagram Blok Sistem yang ditampilkan pada Gambar 4.3, sistem yang dirancang terbagi menjadi tiga alur utama, yaitu *input*, proses, dan *output*. Pada tahap *input*, sensor-sensor yang digunakan untuk memantau kondisi air kali akan mengumpulkan data. Sensor *JSN-SR04T* mendeteksi objek yang terapung di atas permukaan air, sensor *YF-S201* berfungsi untuk mendeteksi debit air yang mengalir masuk

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun

tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

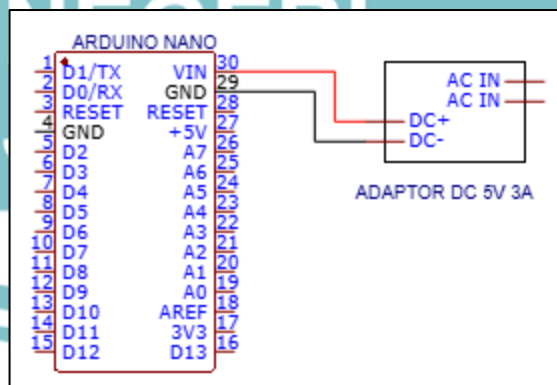
ke Komplek. Selanjutnya, data yang diperoleh dari sensor-sensor ini akan masuk ke alur proses, di mana *Arduino Nano* akan membaca dan mengolah data tersebut. Setelah diproses, data dikirimkan ke server menggunakan protokol *MQTT* melalui *interface SIM800L*. Terakhir, pada alur *output*, hasil pengolahan data dikirimkan ke *HiveMQ* selaku broker *MQTT*.

4.2.3 Rancangan Skematik Alat

Berikut akan dijelaskan skema alat yang digunakan dalam penelitian rancang bangun sistem pemantauan ketinggian dan debit air berbasis *Internet of things (IoT)*. Skema ini menunjukkan susunan dan koneksi antar komponen utama seperti sensor, mikrokontroler, modul komunikasi, indikator *LED*, serta catu daya yang digunakan dalam sistem.

4.2.3.1 Skematik Catu Daya *Microcontroller*

Gambar 4.4 menunjukkan skematik catu daya untuk *Arduino Nano*. Dalam sistem ini, mikrokontroler disuplai menggunakan adaptor DC 5V. Pin *VIN Arduino* terhubung ke kutub positif adaptor (DC +), dan *GND* ke kutub negatif (DC -).



Gambar 4. 4 Skematik Catu Daya

Tabel 4.3 menunjukkan bahwa suplai daya ke *Arduino Nano* berasal dari adaptor DC 5V yang langsung masuk ke pin *VIN* dan *GND*. Konfigurasi ini memastikan *Arduino* mendapatkan sumber tegangan yang sesuai untuk

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

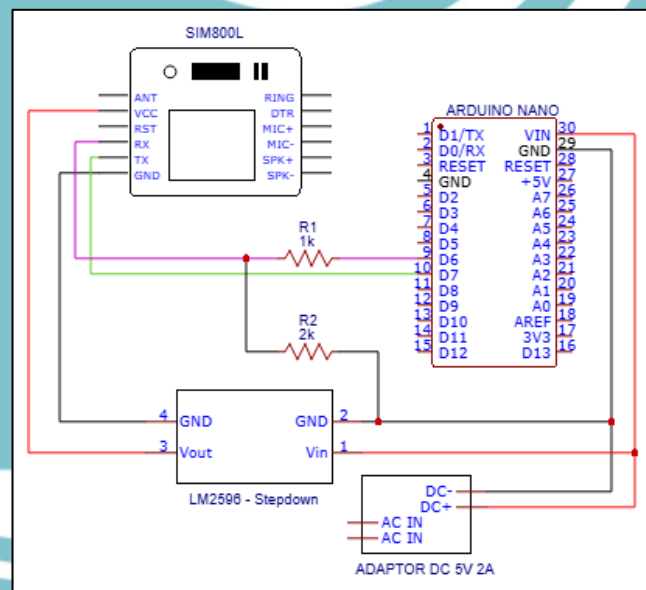
menjalankan seluruh komponen yang terhubung dengannya. Karena sistem bekerja di lingkungan luar ruangan, penyediaan tegangan yang stabil sangat penting agar sistem tetap operasional dalam jangka panjang.

Tabel 4. 3 Pinout Catu Daya Arduino Nano

Arduino Nano	Adaptor 5V
VIN	DC +
GND	DC -

4.2.3.2 Skematik SIM800L

Gambar 4.5 memperlihatkan koneksi antara *Arduino Nano* dan modul *SIM800L*. Modul ini menggunakan komunikasi serial (D6 sebagai RX, D7 sebagai TX). Untuk dayanya, *SIM800L* diberi tegangan 4.2V dari *output* stepdown converter.



Gambar 4. 5 Skematik SIM800L

Dalam tabel 4.4 terlihat bahwa komunikasi data antara *Arduino* dan *SIM800L* dilakukan melalui pin digital D6 (RX) dan D7 (TX). Step-down converter memberikan tegangan sebesar 4.2V ke VCC *SIM800L*, sesuai dengan kebutuhan dayanya yang tidak tahan terhadap tegangan 5V langsung.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

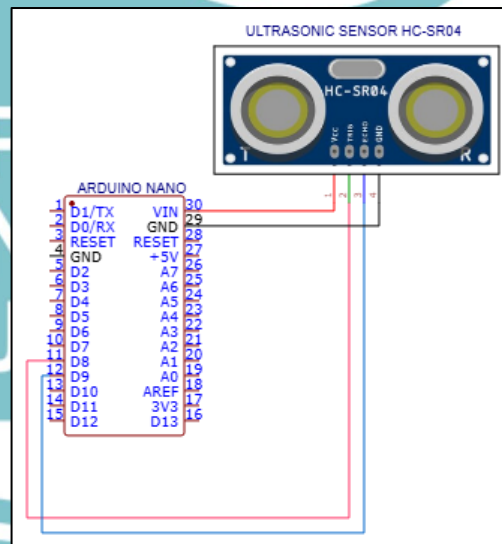
Adanya sambungan ground bersama (GND) antara *Arduino*, *SIM800L*, dan step-down sangat penting untuk menghindari gangguan sinyal dan memastikan referensi tegangan tetap konsisten.

Tabel 4. 4 Pinout *Arduino Nano* ke *SIM800L*

<i>Arduino Nano</i>	<i>SIM800L</i>	Stepdown
D6	RX	-
D7	TX	-
GND	GND	GND
-	VCC	Vout

4.2.3.3 Skematik *JSN-SR04T*

Gambar 4.6 menunjukkan sambungan sensor *JSN-SR04T* dengan *Arduino Nano*. Sensor ini menggunakan dua pin digital (TRIG dan ECHO) untuk mengukur jarak air. Sensor diberi daya dari VIN *Arduino*.



Gambar 4. 6 Skematik *JSN-SR04T*

Tabel 4.5 menjelaskan koneksi sensor ultrasonik yang digunakan untuk mengukur ketinggian air. Sensor ini membutuhkan dua pin digital untuk fungsi trigger dan echo. Pin VIN digunakan sebagai sumber daya karena sensor ini dirancang untuk bekerja pada tegangan 5V, sama seperti

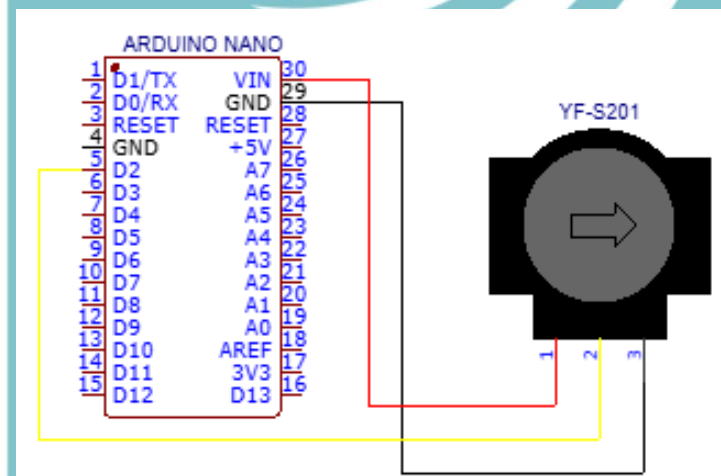
output daya dari adaptor. Koneksi ini memungkinkan pembacaan jarak secara akurat, yang nantinya dikonversi menjadi data ketinggian air.

Tabel 4. 5 Pinout Arduino Nano dan JSN-SR04T

Arduino Nano	JSN-SR04T
VIN	VCC
GND	GND
D8	TRIG
D9	ECHO

4.2.3.4 Skematik YF-S201

Gambar 4.7 menggambarkan koneksi antara *Arduino* dan sensor flowmeter *YF-S201*. Sensor ini mengirimkan pulsa ke pin D2 yang dibaca sebagai sinyal *input* untuk menghitung debit air.



Gambar 4. 7 Skematik YF-S201

YF-S201 mengeluarkan sinyal pulsa sebagai representasi debit air. Dengan menghubungkan pin data ke pin D2 *Arduino*, sistem dapat menghitung laju aliran air berdasarkan jumlah pulsa per satuan waktu. Sensor ini juga mendapat suplai 5V dari pin VIN *Arduino*. Pemilihan pin D2 penting karena mendukung fitur interrupt untuk perhitungan pulsa secara akurat dan efisien.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

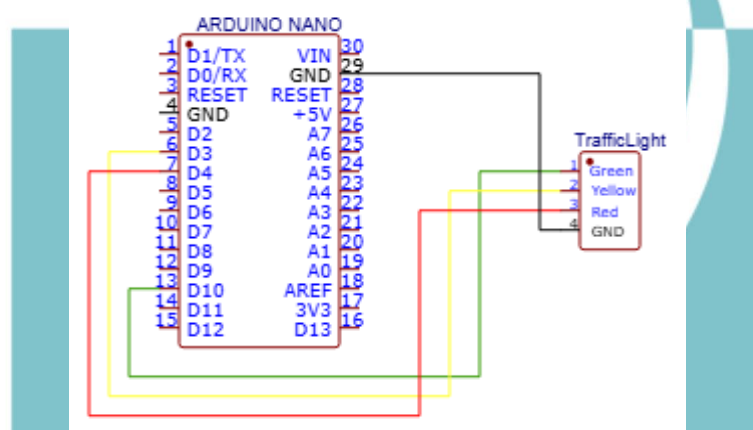
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 4. 6 Pinout Arduino Nano dan YF-S201

Arduino Nano	YF-S201
VIN	VCC
GND	GND
D2	DATA

4.2.3.5 Skematik LED Traffic Light Simulation Module

Gambar 4.8 memperlihatkan koneksi modul LED traffic light ke Arduino. Modul ini berisi tiga LED (merah, kuning, hijau) yang terhubung masing-masing ke pin digital D4, D3, dan D10. GND modul dihubungkan ke GND Arduino.



Gambar 4. 8 Skematik LED Traffic Light Module

Tabel 4.7 memperlihatkan koneksi indikator LED sebagai penanda visual status air. LED merah, kuning, dan hijau masing-masing terhubung ke pin D4, D3, dan D10. Penempatan warna ini mengikuti konvensi umum indikator bahaya: hijau (aman), kuning (waspada), dan merah (bahaya). Koneksi GND bersama juga penting untuk kelistrikan yang stabil. Indikator visual ini memudahkan pengguna atau warga sekitar untuk melihat status kondisi air secara langsung.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun

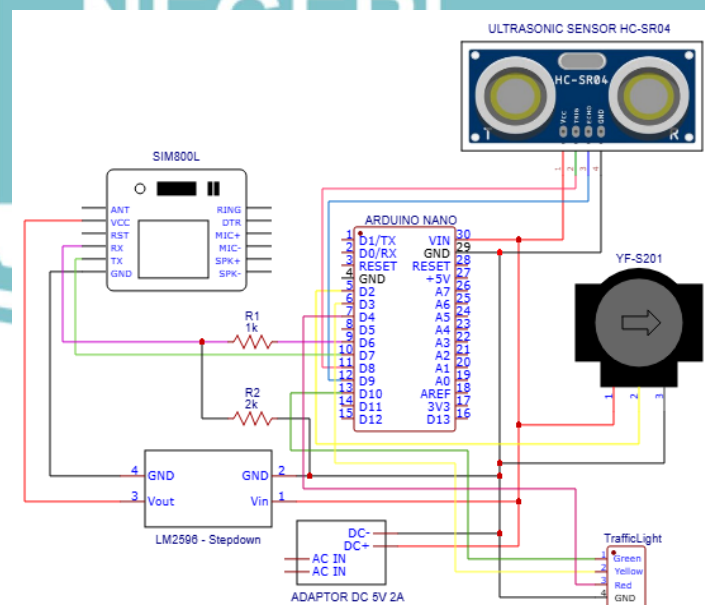
tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 4. 7 Pinout Arduino Nano dan LED Traffic LED Module

Arduino Nano	LED Traffic Light
D4	Red
D3	Yellow
D10	Green
GND	GND

4.2.3.6 Skematik Keseluruhan Alat

Gambar 4.9 menunjukkan skematik lengkap dari rangkaian alat yang digunakan dalam sistem. *Arduino Nano* terhubung dengan sensor *JSN-SR04T* melalui pin D8 dan D9, serta sensor *YF-S201* pada pin D2. Modul *SIM800L* dihubungkan menggunakan komunikasi uart dihubungkan dengan pin D6 (RX) dan D7 (TX), dengan suplai daya dari modul stepdown. *LED* indikator terhubung ke pin D4, D3, dan D10 untuk warna merah, kuning, dan hijau. Seluruh rangkaian mendapatkan catu daya dari adaptor 5V yang masuk ke pin VIN dan GND *Arduino*. Skematik ini merepresentasikan integrasi semua komponen dalam satu sistem yang bekerja secara terpadu.



Gambar 4. 9 Skematik Keseluruhan Alat

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

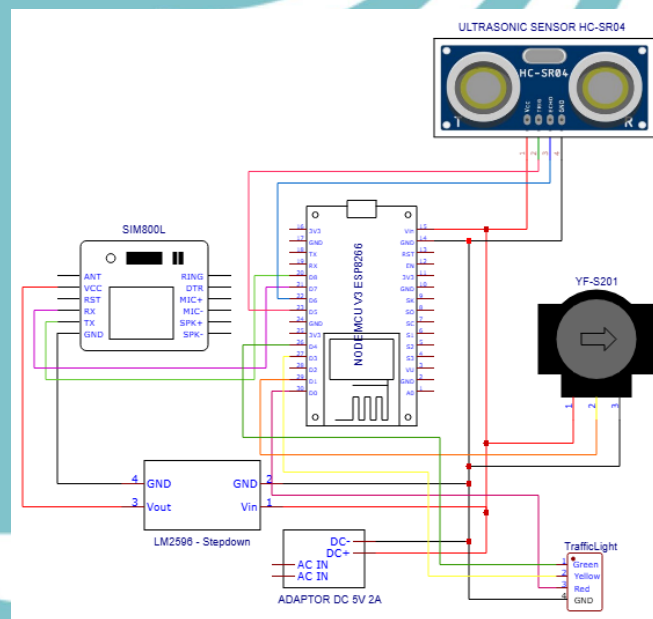
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.10 menunjukkan skematik alternatif dengan menambahkan ESP8266 NodeMCU sebagai pengganti Arduino Nano. Revisi ini memungkinkan pengiriman data sensor melalui koneksi WiFi, sebagai alternatif jika modul SIM800L mengalami gangguan jaringan. NodeMCU membaca data dari sensor HC-SR04 dan YF-S201, lalu mengirimkannya ke broker MQTT menggunakan WiFi.

Modul SIM800L tetap terhubung sebagai cadangan kembali dan disuplai daya melalui step-down LM2596. LED indikator merah, kuning, dan hijau tetap digunakan untuk menampilkan status kondisi air. Dengan kombinasi komunikasi WiFi dan GSM, sistem menjadi lebih fleksibel dan andal dalam menjaga kontinuitas pengiriman data.



Gambar 4. 10 Skematik Alternatif Alat

4.3 Implementasi Sistem

Implementasi sistem dilakukan untuk merealisasikan hasil rancangan menjadi sebuah sistem yang berfungsi secara nyata. Implementasi dibagi menjadi dua bagian utama, yaitu implementasi perangkat keras dan implementasi perangkat lunak. Perangkat keras mencakup seluruh komponen fisik seperti

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

mikrokontroler, sensor, modul komunikasi, indikator *LED*, serta tata letak dan penyusunan rangkaian. Sementara itu, perangkat lunak meliputi pemrograman mikrokontroler dengan *Arduino IDE* dan konfigurasi komunikasi ke broker *MQTT* untuk pengiriman data sensor secara berkala.

4.3.1 Implementasi Perangkat Keras

Pada tahap ini, seluruh komponen fisik dirakit dan disusun sesuai dengan rancangan skematik yang telah dibuat sebelumnya. Proses implementasi perangkat keras mencakup penyambungan kabel antar komponen, pemasangan sensor, modul komunikasi, indikator *LED*, serta penataan komponen ke dalam project box agar aman digunakan di lapangan.



Gambar 4. 11 Hasil Akhir Perangkat Keras

4.3.1.1 Perakitan Komponen Elektronika

Proses perakitan dimulai dengan menyiapkan semua komponen utama yaitu *Arduino Nano*, modul *SIM800L*, sensor *JSN-SR04T*, sensor *YF-S201*, dan modul *LED traffic light*. Komponen-komponen ini dirakit sesuai rancangan skematik untuk memastikan tiap perangkat dapat saling terhubung dan berfungsi dengan baik. Setiap perangkat diposisikan pada papan PCB khusus yang telah disiapkan

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

sebelumnya sebagai media penyolderan dan penyambungan kabel.

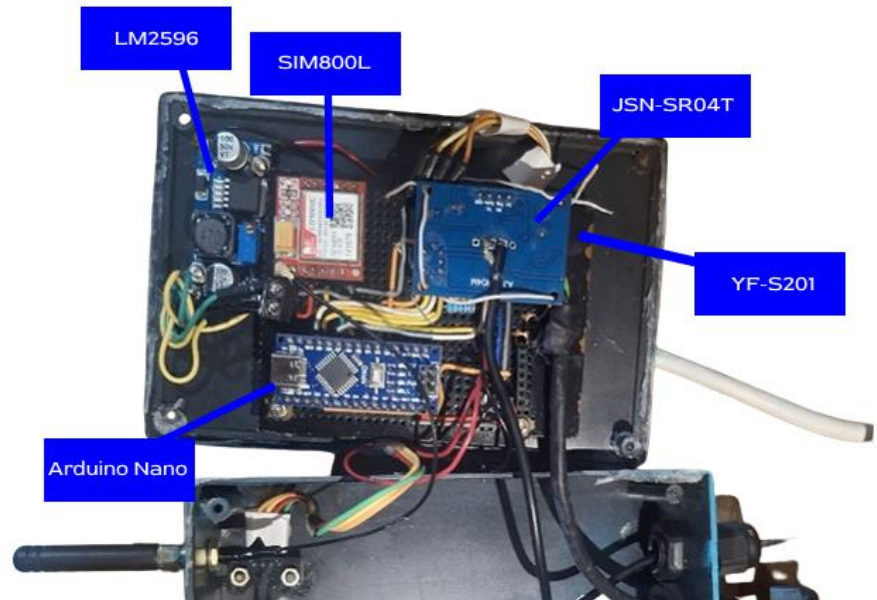
Sambungan antar komponen dilakukan dengan cara penyolderan kabel pada pin-pin yang sudah ditentukan di PCB. Penggunaan PCB memudahkan penataan jalur listrik dan sinyal sehingga lebih rapi dan kuat dibandingkan menggunakan kabel jumper biasa. Selain itu, penyolderan juga memberikan kestabilan mekanik yang penting untuk mencegah gangguan akibat getaran atau guncangan yang mungkin terjadi saat alat digunakan di lapangan.

Untuk catu daya, adaptor 5V dihubungkan ke PCB yang kemudian disambungkan ke modul stepdown (buck converter) untuk menurunkan tegangan menjadi 4,2V yang dibutuhkan oleh modul *SIM800L* agar dapat bekerja secara optimal. Kabel dari modul stepdown disolder dengan rapi ke jalur PCB menuju *SIM800L* dan *Arduino Nano*. Pengaturan tegangan ini sangat penting agar komponen mendapat pasokan daya yang stabil dan tidak rusak.

Sensor *JSN-SR04T* disambungkan ke pin digital D8 (TRIG) dan D9 (ECHO) pada *Arduino Nano* untuk mengukur ketinggian air dengan metode ultrasonik. Sensor *YF-S201* dihubungkan ke pin D2 sebagai *input* pulsa yang merepresentasikan debit air yang mengalir. Sedangkan *LED* traffic light disolder ke pin D3, D4, dan D10 untuk menampilkan indikator warna kuning, merah, dan hijau sesuai dengan status kondisi air yang terdeteksi.

Setelah seluruh komponen terpasang, dilakukan pengecekan menyeluruh menggunakan multimeter untuk memastikan semua sambungan solderan bebas dari hubung singkat dan memiliki kontak listrik yang baik. Pengujian fungsi awal juga

dilakukan dengan menghidupkan sistem untuk memeriksa komunikasi antar modul dan kestabilan suplai daya.



Gambar 4. 12 Rangkaian Alat

4.3.1.2 Perancangan Kotak Pelindung dan Piranti Penunjang

Kotak pelindung pada sistem ini berfungsi sebagai wadah utama untuk menempatkan seluruh rangkaian elektronik agar terlindung dari debu, air, dan gangguan eksternal lainnya saat dipasang di lapangan. Untuk menjaga kerapatan kotak, terutama pada sambungan dan area penutup, digunakan lem selen sebagai sealant utama. Lem selen dipilih karena kemampuannya menutup rapat celah-celah kecil sehingga mencegah masuknya air dan kelembapan yang berpotensi merusak komponen dalam kotak.

Selain itu, pada titik keluaran kabel, kotak dilengkapi dengan gland cable yang berfungsi untuk mengamankan sambungan kabel dan menjaga area tersebut tetap tertutup rapat. Penggunaan gland cable sangat penting untuk mencegah air dan kotoran masuk melalui celah kabel yang biasanya menjadi titik lemah perlindungan. Gland cable juga

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

membantu menjaga posisi kabel agar tidak mudah tergeser atau lepas akibat getaran atau gangguan mekanis lainnya.

Dengan kombinasi penggunaan lem selen yang efektif dalam sealing dan gland cable sebagai perlindungan kabel, kotak pelindung ini mampu memberikan perlindungan maksimal bagi rangkaian elektronik selama operasional di lingkungan luar yang rentan terhadap berbagai gangguan fisik dan cuaca.

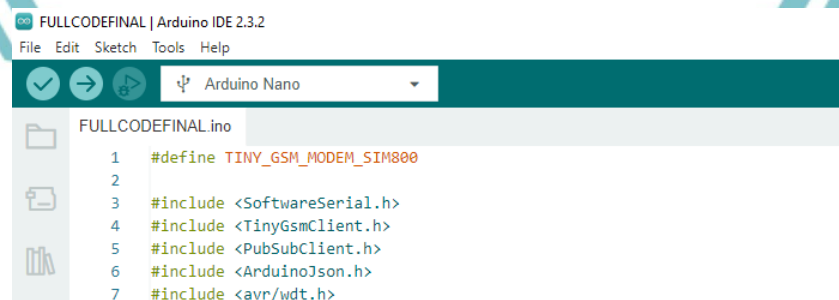


Gambar 4. 13 Kotak Pelindung Alat

4.3.2 Implementasi Perangkat Lunak

4.3.2.1 Pemrograman Mikrokontroler dan Pengiriman Data

Pada sistem ini, mikrokontroler *Arduino Nano* digunakan sebagai pusat pengendali utama untuk membaca data sensor, mengolahnya, dan mengirimkannya ke broker *MQTT* menggunakan modul *SIM800L*. Program ditulis dalam bahasa *C++* menggunakan *Arduino IDE*, dan mengandalkan beberapa library seperti *TinyGSMClient*, *PubSubClient*, dan *ArduinoJSON*.



Gambar 4. 14 Penggunaan Arduino IDE dan library yang digunakan

Saat perangkat pertama kali dinyalakan, sistem akan menginisialisasi pin digital, memulai komunikasi serial dengan *SIM800L*, dan menghubungkan ke jaringan seluler menggunakan *APN* tertentu. Setelah itu, mikrokontroler akan mencoba koneksi ke broker *MQTT* dengan autentikasi yang telah ditentukan. *LED* indikator digunakan untuk memberi tahu status keberhasilan setup.

```
106 void setup() {
107   pinMode(TRIG_PIN, OUTPUT);
108   pinMode(ECHO_PIN, INPUT);
109   pinMode(FLOW_PIN, INPUT);
110   pinMode(LED_GREEN, OUTPUT);
111   pinMode(LED_YELLOW, OUTPUT);
112   pinMode(LED_RED, OUTPUT);
113   attachInterrupt(digitalPinToInterrupt(FLOW_PIN), increase, RISING);
114   SerialAT.begin(9600);
115   digitalWrite(LED_GREEN, LOW);
116   digitalWrite(LED_YELLOW, LOW);
117   digitalWrite(LED_RED, LOW);
118   blinkYellowSetup();
119
120   modem.restart();
121   modem.gprsDisconnect();
122   while (!modem.gprsConnect(apn, gprsUser, gprsPass)) {
123     delay(500);
124   }
125
126   mqtt.setServer(broker, 1883);
127   while (!mqttConnect()) {
128     blinkRedReconnect();
129     delay(500);
130   }
131   blinkAllSuccess();
}
```

Gambar 4. 15 Konfigurasi Awal Fungsi Setup

Sensor debit air *YF-S201* menghasilkan pulsa listrik sesuai jumlah air yang mengalir. Pulsa ini dihitung menggunakan interrupt yang dijalankan pada pin digital tertentu. Nilai pulsa dikalikan dengan faktor kalibrasi (*calibrationFactor*) untuk memperoleh nilai debit dalam satuan liter per menit (L/min). Nilai kalibrasi ini diperoleh dari pengujian sebelumnya dan bisa disesuaikan tergantung karakteristik masing-masing sensor. Faktor kalibrasi yang digunakan dalam program adalah 0.1253, yang berasal dari hasil pengujian 1 liter air dengan waktu tertentu. Penentuan nilai ini penting karena memengaruhi akurasi pembacaan debit air.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```

134 void loop() {
135     static float flowrate = 0.0;
136     unsigned long now = millis();
137
138     if (!modem.isGprsConnected()) { ...
139     }
140
141
142     if (!mqtt.connected()) { ...
143     }
144
145     mqtt.loop();
146
147     // Pembacaan sensor setiap detik
148     if (now - lastReadTime >= readInterval) {
149         lastReadTime = now;
150
151         // Hitung debit air
152         noInterrupts();
153         unsigned long count = pulse;
154         pulse = 0;
155         interrupts();
156         flowrate = count * calibrationFactor;
157     }
158 }

```

Gambar 4. 16 Pembacaan Sensor YF-S201

Gambar 4.16 sensor JSN-SR04T digunakan untuk mengukur jarak antara sensor dan permukaan air. Pembacaan dilakukan sebanyak tiga kali setiap satu detik, lalu hasil median diambil untuk mengurangi kesalahan akibat noise. Nilai ketinggian air dihitung berdasarkan selisih antara tinggi pemasangan sensor dan nilai jarak yang terukur. Sistem menggunakan tiga buah LED berwarna merah, kuning, dan hijau untuk menunjukkan kondisi ketinggian air. Nilai batasan didasarkan pada persentase dari tinggi maksimal, misalnya: Di bawah 20cm, Aman (LED hijau menyala), 20cm–80cm, waspada (LED kuning menyala), di atas 80cm Bahaya (LED merah menyala).

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
160 float d1 = bacaJarak(); delay(50);
161 float d2 = bacaJarak(); delay(50);
162 float d3 = bacaJarak();
163
164 if (d1 != 0 && d2 != 0 && d3 != 0) {
165     float median = getMedian(d1, d2, d3);
166     lastValidDistance = median;
167
168     float tinggiAir = tinggiSensorCm - median;
169     if (tinggiAir < 0) tinggiAir = 0;
170
171     // Update indikator LED setiap detik
172     digitalWrite(LED_GREEN, LOW);
173     digitalWrite(LED_YELLOW, LOW);
174     digitalWrite(LED_RED, LOW);
175
176     float level = tinggiAir / 100;
177     if (level < 0.20) {
178         digitalWrite(LED_GREEN, HIGH);
179     } else if (level <= 0.80) {
180         digitalWrite(LED_YELLOW, HIGH);
181     } else {
182         digitalWrite(LED_RED, HIGH);
183     }
184 }
185 }
```

Gambar 4. 17 Pembacaan Sensor JSN-SR04T dan LED Indikator

Gambar 4.17 menunjukkan setiap satu menit, data ketinggian air dan debit air dikemas dalam format *JSON* dan dikirimkan ke broker *MQTT* pada topik sensor/data. Pengiriman dilakukan dengan opsi *retain = true* agar data terakhir selalu tersedia bagi subscriber, bahkan saat subscriber baru terhubung. Sistem secara otomatis akan mencoba menyambung ulang apabila koneksi *MQTT* terputus. Jika koneksi *GPRS* terdeteksi tidak aktif, maka sistem akan melakukan inisialisasi ulang modem untuk memastikan komunikasi data tetap berjalan.

```
188 if (now - lastPublishTime >= publishInterval) {
189     lastPublishTime = now;
190     if (lastValidDistance >= 0) {
191         float tinggiAir = tinggiSensorCm - lastValidDistance;
192         if (tinggiAir < 0) tinggiAir = 0;
193
194         StaticJsonDocument<128> doc;
195         doc["tinggi_air"] = tinggiAir;
196         doc["debit"] = flowrate;
197
198         char buffer[128];
199         serializeJson(doc, buffer);
200         mqtt.publish(mqttTopic, buffer, true);
201     }
202 }
203
204 }
```

Gambar 4. 18 Program Mengirim Data ke Broker

4.3.2.2 Konfigurasi Broker *HiveMQ*

Broker *MQTT* yang digunakan pada sistem ini adalah *HiveMQ* yang berjalan pada sistem operasi Debian Linux. Broker ini dipilih karena mendukung komunikasi *MQTT*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

secara stabil dan dapat diperluas dengan ekstensi sesuai kebutuhan. Instalasi *HiveMQ* dilakukan dengan mengunduh paket dari situs resmi *HiveMQ*, kemudian mengekstraknya ke dalam direktori proyek. Berikut merupakan bagian konfigurasi port di file `conf/HiveMQ.conf`. Secara default, port *MQTT* non-TLS adalah 1883. Konfigurasi ini menunjukkan bahwa broker akan menerima koneksi klien *MQTT* pada port tersebut

```
<hivemq xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="config.xsd">

  <listeners>
    <tcp-listener>
      <port>1883</port>
      <bind-address>0.0.0.0</bind-address>
    </tcp-listener>

    <websocket-listener>
      <port>8000</port> <!-- Port WebSocket -->
      <bind-address>0.0.0.0</bind-address>
      <path>/mqtt</path> <!-- Path untuk koneksi WebSocket -->
      <subprotocols>
        <subprotocol>mqtt</subprotocol>
      </subprotocols>
    </websocket-listener>
  </listeners>

  <anonymous-usage-statistics>
    <enabled>true</enabled>
  </anonymous-usage-statistics>
  <control-center>
    <enabled>true</enabled>
  </control-center>
</hivemq>
```

Gambar 4. 19 Konfigurasi Broker HiveMQ

Role-Based Access Control (RBAC) merupakan sistem pengelolaan hak akses berbasis peran yang diterapkan pada broker *HiveMQ* untuk meningkatkan keamanan dan pengendalian terhadap aktivitas klien *MQTT*. Dalam sistem ini, setiap pengguna yang ingin terhubung ke broker harus memiliki kredensial tertentu (username dan password), serta ditetapkan peran (role) yang menentukan izin akses pengguna terhadap topik *MQTT*, seperti hak untuk mempublikasikan atau membaca data.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
hivemq-ce-2024.2/extensions/hivemq-file-rbac-extension/conf/credentials.xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<file-rbac>
  <users>
    <user>
      <name>IOT</name>
      <password>iot123</password>
      <roles>
        <id>iot-user</id>
      </roles>
    </user>
    <user>
      <name>Website</name>
      <password>website123</password>
      <roles>
        <id>iot-user</id>
      </roles>
    </user>
  </users>
  <roles>
    <role>
      <id>iot-user</id>
      <permissions>
        <permission>
          <topic>#</topic>
          <activity>PUBLISH</activity>
        </permission>
        <permission>
          <topic>#</topic>
          <activity>SUBSCRIBE</activity>
        </permission>
      </permissions>
    </role>
  </roles>
</file-rbac>
```

Gambar 4. 20 Konfigurasi RBAC Extension

Konfigurasi RBAC pada HiveMQ dilakukan melalui berkas RBAC.xml yang berada di dalam direktori ekstensi RBAC. Berkas ini terdiri dari dua bagian utama, yaitu users dan roles. Bagian users berisi daftar pengguna yang diperbolehkan mengakses broker, lengkap dengan nama pengguna, kata sandi, dan peran yang dimiliki. Sementara itu, bagian roles menjelaskan izin spesifik yang dimiliki oleh setiap peran, seperti hak untuk melakukan publish atau subscribe pada topik tertentu.

```
hivemq.service - HiveMQ MQTT Broker
Loaded: loaded (/etc/systemd/system/hivemq.service; enabled; preset: enabled)
Active: active (running) since Mon 2025-05-05 05:42:25 UTC; 1 month 6 days ago
Main PID: 32558 (java)
Tasks: 45 (limit: 1137)
Memory: 289.8M
CPU: 2h 1min 25.78s
CGroup: /system.slice/hivemq.service
└─32558 java -Dhivemq.home=/home/admin/hivemq-ce-2024.2 -XX:HeapDumpPath=/home/admin/hivemq

May 05 05:42:27 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:27,401 INFO - This HiveMQ ID is PIT6N
May 05 05:42:30 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:30,092 INFO - Starting with file persi
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,055 INFO - Starting HiveMQ extensio
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,146 INFO - Starting File RBAC extens
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,440 INFO - Extension "HiveMQ File RB
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,480 INFO - Starting TCP listener on
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,515 INFO - Starting Websocket liste
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,521 INFO - Started TCP Listener on
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,521 INFO - Started Websocket Listene
May 05 05:42:31 ip-172-31-25-220 run.sh[32558]: 2025-05-05 05:42:31,521 INFO - Started HiveMQ in 4661ms
```

Gambar 4. 21 Service Broker HiveMQ Yang Sedang Berjalan

4.4 Pengujian Sistem

4.4.1 Deskripsi Pengujian

Pengujian sistem dilakukan untuk memastikan bahwa seluruh komponen dan perangkat lunak yang digunakan dalam sistem pemantauan ketinggian dan debit air berfungsi sesuai dengan yang dirancang. Pengujian dilakukan secara bertahap, mulai dari pengujian fungsionalitas sensor, pengujian konektivitas dan pengiriman data melalui *SIM800L*, hingga pengujian performa sistem dalam jangka waktu tertentu. Selain itu, dilakukan pula pengujian kualitas layanan (*Quality of Service*) yang mencakup pengukuran *Delay* dan *Throughput* dari pengiriman data ke broker *MQTT*.

4.4.2 Prosedur Pengujian

4.4.2.1 Prosedur Pengujian Fungsionalitas dan Akurasi

Pengujian dilakukan secara terpisah untuk masing-masing perangkat utama, yaitu sensor *JSN-SR04T*, sensor *YF-S201*, *LED* indikator, dan modul *SIM800L*, guna memastikan bahwa semua komponen bekerja sesuai fungsi dan memiliki akurasi yang baik terhadap nilai referensinya.

a. Sensor *JSN-SR04T*

Sensor ini diuji dengan mengisi ember air setinggi 90 cm. Sensor dipasang pada ketinggian tetap sebesar 110 cm dari dasar ember. Nilai ketinggian air dihitung berdasarkan selisih antara tinggi sensor dan hasil pembacaan jarak sensor terhadap permukaan air, menggunakan rumus:

$$\text{Tinggi Air (cm)} = \text{Tinggi Sensor} - \text{Jarak Sensor ke Air}$$

Sumber: (Ramadhani et al., 2022)

Hasil sensor dibandingkan dengan pengukuran manual menggunakan meteran untuk menghitung selisih (error).

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

b. Sensor YF-S201

Sensor ini diuji dengan mengalirkan volume air tertentu dari wadah tertutup dan mencatat waktu aliran. Sensor menghasilkan pulsa yang dihitung menggunakan interrupt. Nilai debit dihitung menggunakan rumus:

$$\text{Debit Air (L/min)} = \text{Pulsa} \times \text{Faktor Kalibrasi}$$

Sumber: (Ghaniyyah & Eka Putri, 2023)

(Dengan faktor kalibrasi hasil pengujian: 0.1253)

Debit aktual dihitung dengan membagi volume air dengan waktu aliran, dan hasilnya dibandingkan dengan *output* sensor.

c. SIM800L

Pengujian dilakukan dengan memastikan bahwa modul *SIM800L* berhasil menghubungkan perangkat ke jaringan *GPRS* dengan *APN* tertentu dan dapat mengirim data sensor ke broker *MQTT*. Keberhasilan pengiriman diverifikasi melalui log subscriber pada broker. Bila data berhasil terkirim dan diterima setiap menit sesuai jadwal, maka pengujian dinyatakan berhasil (Ristian et al., 2018).

4.4.2.2 Prosedur Pengujian Availability dan Durability

Pengujian ini dilakukan untuk mengetahui seberapa baik sistem dapat beroperasi secara terus menerus (*Availability*) dan seberapa konsisten sistem mengirimkan data selama waktu aktifnya (*Durability*). Pengujian dilakukan dengan membiarkan alat berjalan selama beberapa hari tanpa gangguan, dan data yang masuk ke broker dicatat setiap jam. Suatu jam dianggap berhasil jika menerima minimal 30 data. *Availability* dihitung dari rasio jumlah jam berhasil terhadap total jam operasional. *Durability* dihitung dari rasio jam berhasil terhadap total jam uptime (waktu alat menyala dan

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

terhubung). Hasil pengujian ini mencerminkan keandalan sistem dalam pengoperasian jangka panjang.

Berikut rumus perhitungan *Availability*:

$$Availability (\%) = \frac{Total\ Uptime}{Total\ Uptime + Total\ Downtime} \times 100\%$$

Sumber: (Aryatama, 2022)

Serta Rumus perhitungan *Durability*:

$$Durability (\%) = \frac{Total\ Keberhasilan}{Total\ Uptime} \times 100\%$$

Sumber: (Aryatama, 2022)

Dalam proses pengujian ini, data dikumpulkan menggunakan program *Python* logger.py, yang berfungsi sebagai subscriber pada topik sensor/data di broker *MQTT*. Ketika data baru diterima, timestamp kedatangan dicatat ke dalam file *data_log.JSON* menggunakan format waktu lokal Asia/Jakarta. Untuk menghindari pencatatan data yang sudah tersimpan sebagai retained message, sistem akan mengabaikan satu pesan retained pertama setelah setiap koneksi ulang.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
GNU nano 7.2
import json
import datetime
from zoneinfo import ZoneInfo
import paho.mqtt.client as mqtt

DATA_FILE = "data_log.json"
TOPIC = "sensor/data"
MQTT_BROKER = "18.142.250.134"
MQTT_PORT = 1883
USERNAME = "IoT1"
PASSWORD = "iot123"

first_retained_skipped = False # flag global

# Inisialisasi file jika belum ada
def init_data_file():
    try:
        with open(DATA_FILE, "r") as f:
            pass
    except FileNotFoundError:
        with open(DATA_FILE, "w") as f:
            json.dump([], f)

def save_data(timestamp):
    with open(DATA_FILE, "r") as f:
        data = json.load(f)
        data.append(timestamp)
    with open(DATA_FILE, "w") as f:
        json.dump(data, f, indent=2)

def on_connect(client, userdata, flags, rc):
    global first_retained_skipped
    print("Connected:", rc)
    client.subscribe(TOPIC)
    first_retained_skipped = False # reset flag setiap reconnect

def on_message(client, userdata, msg):
    global first_retained_skipped

    # skip retained pertama tiap koneksi
    if msg.retain and not first_retained_skipped:
        print("Retained pertama diabaikan")
        first_retained_skipped = True
    return
```

Gambar 4. 22 Kode Logger.py

Selanjutnya, file log yang berisi deretan timestamp tersebut dianalisis menggunakan program *analyze.py*. Program ini akan mengelompokkan data berdasarkan hari dan jam, lalu menghitung jumlah data per jam. Jika dalam satu jam terdapat minimal 30 data, maka jam tersebut dianggap berhasil. Dari pengelompokan ini, sistem menghitung *Availability* berdasarkan jumlah jam alat aktif dibandingkan total jam pengujian, dan *Durability* berdasarkan jumlah jam yang berhasil mengirim data sesuai ambang batas dibandingkan total jam uptime. *Output* dari program ini menampilkan hasil rekap harian serta persentase performa sistem secara keseluruhan.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```

GNU nano 7.2
import json
import datetime
from zoneinfo import ZoneInfo
from collections import defaultdict

DATA_FILE = "data_log.json"
MIN_VALID_PER_HOUR = 30

def load_data():
    with open(DATA_FILE, "r") as f:
        timestamps = json.load(f)
    return [datetime.datetime.fromisoformat(ts) for ts in timestamps]

def group_data_by_day_and_hour(timestamps):
    start_time = min(timestamps)
    daily_data = defaultdict(lambda: defaultdict(list))
    for ts in timestamps:
        day_idx = int((ts - start_time).total_seconds() // 86400)
        hour_idx = int((ts - start_time).total_seconds() // 3600)
        daily_data[day_idx][hour_idx] += [ts]
    return daily_data

def analyze(timestamps):
    grouped = group_data_by_day_and_hour(timestamps)

    availability = []
    durability = []

    for day, hours in grouped.items():
        uptime = 0
        downtime = 0
        berhasil = 0
        for _hour, ts_list in hours.items():
            if len(ts_list) >= MIN_VALID_PER_HOUR:
                uptime += 1
                berhasil += 1
            else:
                uptime += 1
                # jam dianggap uptime tapi gagal
        total_jam = uptime + downtime
        availability.append((day + 1, uptime, 24 - uptime))
        durability.append((day + 1, uptime, berhasil))

    return availability, durability
  
```

Gambar 4. 23 Kode Analyze.py

4.4.2.3 Prosedur Pengujian Quality of Service (QoS)

Pengujian *Quality of Service* dilakukan untuk mengevaluasi performa sistem dalam proses pengiriman data sensor ke broker *MQTT* melalui jaringan seluler. Dua parameter utama yang diuji adalah *Delay* dan *throughout*.

Delay merupakan waktu tunda yang dihitung sebagai selisih antara waktu saat data dikirim dari perangkat (timestamp lokal alat) dan waktu saat data diterima oleh broker *MQTT*. Nilai rata-rata *Delay* dihitung menggunakan rumus:

$$\text{Rata rata Delay} = \frac{\text{Total Delay}}{\text{Total paket yang diterima}}$$

Penilaian terhadap nilai *Delay* dilakukan berdasarkan indeks kategori seperti pada tabel berikut:

Tabel 4. 8 Indeks Kategori Nilai Delay

Rata-rata Delay	Kategori
<150 ms	Sangat Bagus
150-300 ms	Bagus
300 – 450 ms	Sedang
>450ms	Kurang

Sumber: (Prakoso, 2020)

Throughput dihitung untuk mengetahui jumlah data (dalam byte) yang berhasil diterima dalam suatu periode waktu. Nilainya diperoleh menggunakan rumus:

$$\text{Throughput} = \frac{\text{Jumlah Data yang Dikirim}}{\text{Waktu Pengiriman Data}}$$

Sumber: (Prakoso, 2020)

4.5 Data Hasil Pengujian

4.5.1 Hasil Pengujian Fungsionalitas dan Akurasi

a. Sensor JSN-SR04T

Untuk menguji keakuratan sensor JSN-SR04T dalam mengukur ketinggian air, dilakukan pengujian dengan membandingkan hasil pembacaan sensor terhadap nilai pengukuran manual (sebagai data pembanding). Selisih antara kedua nilai tersebut dihitung sebagai error. Hasil pengujian disajikan pada Tabel 4.9 berikut.

Tabel 4. 9 Hasil Pengujian Sensor JSN-SR04T

No	Hasil Pembacaan Sensor (cm)	Hasil Perhitungan pembanding (cm)	Error (cm)
1	34,17	38	3,83
2	34,76	39	4,24
3	35,31	39	3,69

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4	40,74	44	3,26
5	51,8	50	1,8
6	50,95	51	0,05
7	52,12	52	0,12
8	52,91	53	0,09
9	54,92	55	0,08
10	56,01	56	0,01
11	58,2	58	0,2
12	61,25	61	0,25
13	67,13	67	0,13
14	67,18	67	0,18
15	66,56	67	0,44
16	68,93	68	0,93
17	68,99	69	0,01
18	68,99	69	0,01
19	70,03	70	0,03
20	70,4	70	0,4
21	74,6	75	0,4
22	77,9	78	0,1
23	78,96	79	0,04
24	79,98	80	0,02
25	79,27	80	0,73
26	85,99	86	0,01
27	88,45	88	0,45
28	88,99	89	0,01
29	89,11	89	0,11
30	89,99	90	0,01
Mean Error (cm)			0,72

Berdasarkan data pada Tabel 4.9, terlihat bahwa nilai error tertinggi terjadi pada pengukuran awal (baris ke-2) yaitu sebesar 4,24 cm, sementara error terkecil adalah 0,01 cm, yang muncul pada beberapa pengukuran akhir. Hasil ini menunjukkan bahwa

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

semakin dekat jarak air ke sensor, maka akurasi pengukuran cenderung meningkat, karena nilai error menjadi lebih kecil dan stabil.

Dari seluruh 30 data yang diuji, diperoleh rata-rata error (mean error) sebesar 0,72 cm, yang menunjukkan bahwa pembacaan sensor relatif akurat untuk digunakan dalam sistem pemantauan ketinggian air. Nilai error di bawah 1 cm tergolong kecil dan masih dapat diterima dalam konteks aplikasi monitoring lingkungan. Selain itu, fluktuasi error yang rendah secara keseluruhan mencerminkan kestabilan hasil pengukuran. Meskipun terjadi deviasi pada beberapa pengukuran, pola error tetap konsisten, sehingga sistem dapat dianggap cukup presisi dan andal untuk diterapkan dalam pemantauan *Real-Time*.

b. Sensor YF-S201

Sensor *YF-S201* bekerja dengan menghasilkan pulsa yang sebanding dengan laju aliran air. Untuk memperoleh hasil yang akurat, sensor ini memerlukan proses kalibrasi terlebih dahulu guna menentukan faktor konversi pulsa ke debit air. Kalibrasi dilakukan dengan mengalirkan volume air tertentu dan mencatat waktu aliran, kemudian membandingkan hasil pembacaan pulsa sensor terhadap debit aktual. Berdasarkan hasil pengujian sebelumnya, diperoleh faktor kalibrasi empiris sebesar 0.1253, yang digunakan dalam semua pengukuran debit selama pengujian.

Tabel 4. 10 Hasil Pengujian Sensor YF-S201

No	Hasil Pembacaan Sensor (L/Min)	Hasil Perhitungan pembanding (L/Min)	Error (L/min)
1	0,46	0,44	0,02
2	0,79	0,59	0,2
3	0,86	0,76	0,1
4	0,75	0,88	0,13

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

5	0,85	0,95	0,1
6	1,11	1,16	0,05
7	1,45	1,2	0,25
8	1,36	1,2	0,16
9	1,17	1,22	0,05
10	1,61	1,32	0,29
11	1,58	1,33	0,25
12	1,75	1,46	0,29
13	1,51	1,6	0,09
14	2,24	1,81	0,43
15	2,26	1,87	0,39
16	2,67	2,19	0,48
17	2,22	2,2	0,02
18	2,95	2,44	0,51
19	3,19	2,86	0,33
20	3,33	2,93	0,4
21	3,6	3,19	0,41
22	3,69	3,21	0,48
23	3,49	3,68	0,19
24	4,33	3,69	0,64
25	4,65	3,79	0,86
26	4,11	3,93	0,18
27	4,91	4,25	0,66
28	5,46	4,4	1,06
29	5,51	4,78	0,73
30	5,61	4,86	0,75
Mean Error (L/Min)			0.35

Tabel 4.10 menunjukkan bahwa sensor *YF-S201* memiliki rata-rata error sebesar 0.35 L/menit jika dibandingkan dengan debit aktual yang dihitung manual. Nilai ini tergolong kecil dan dapat diterima untuk sistem pemantauan debit berbasis mikrokontroler. Akurasi ini dicapai berkat proses kalibrasi yang

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

tepat dan pengambilan data secara periodik menggunakan interrupt. Secara keseluruhan, sensor ini menunjukkan performa yang cukup andal dalam membaca variasi aliran air dengan ketelitian yang memadai.

c. SIM800L

Pengujian terhadap modul *SIM800L* menunjukkan keberhasilan penuh dalam menjalin koneksi ke jaringan *GPRS* dan broker *MQTT*.

Tabel 4. 11 Hasil Pengujian SIM800L

No	Status koneksi	Status Pengiriman data	Keterangan
1	Terhubung	Berhasil	Sukses
2	Terhubung	Berhasil	Sukses
3	Terhubung	Berhasil	Sukses
4	Terhubung	Berhasil	Sukses
5	Terhubung	Berhasil	Sukses

Seluruh sesi pengujian memperlihatkan status koneksi yang stabil dan data berhasil dikirim serta diterima oleh subscriber. Hal ini membuktikan bahwa modul *SIM800L* dapat digunakan secara efektif untuk komunikasi data berbasis seluler dalam sistem *IoT*.

4.5.2 Hasil Pengujian Availability dan Durability

Pengujian *Availability* dan *Durability* dilakukan untuk menilai keandalan sistem dalam menjaga konektivitas dan konsistensi pengiriman data secara berkelanjutan. Pengujian dilakukan selama tujuh hari berturut-turut, di mana perangkat dibiarkan menyala tanpa intervensi manual dan diletakkan pada lokasi semi-terbuka di luar ruangan.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Setiap data yang berhasil dikirim oleh perangkat tercatat di broker *MQTT* melalui topik sensor/data. Server yang terhubung sebagai subscriber mencatat waktu kedatangan setiap pesan dan menyimpannya ke dalam file log *data_log.JSON*. Setiap entri dalam log ini berisi timestamp data masuk, yang selanjutnya digunakan sebagai dasar perhitungan jumlah data yang diterima per jam. Proses evaluasi dilakukan secara otomatis menggunakan skrip *Python* untuk mengelompokkan data berdasarkan hari dan jam, menghitung total jam aktif (*uptime*), *downtime*, serta keberhasilan dalam pengiriman data sesuai ambang batas minimal (≥ 30 data per jam).

```

admin@ip-172-31-25-220: ~/uji
(venv) admin@ip-172-31-25-220:~/uji$ ls
analyze.py data_log.json delay.py logger.py
(venv) admin@ip-172-31-25-220:~/uji$ python analyze.py

Availability:
Hari ke-1: Uptime = 24 jam, Downtime = 0 jam
Hari ke-2: Uptime = 24 jam, Downtime = 0 jam
Hari ke-3: Uptime = 21 jam, Downtime = 3 jam
Hari ke-4: Uptime = 24 jam, Downtime = 0 jam
Hari ke-5: Uptime = 24 jam, Downtime = 0 jam
Hari ke-6: Uptime = 24 jam, Downtime = 0 jam
Hari ke-7: Uptime = 24 jam, Downtime = 0 jam
Total Uptime: 165 jam, Downtime: 3 jam
Persentase Availability: 98.21%

Durability:
Hari ke-1: Uptime = 24 jam, Keberhasilan = 24 jam
Hari ke-2: Uptime = 24 jam, Keberhasilan = 24 jam
Hari ke-3: Uptime = 21 jam, Keberhasilan = 18 jam
Hari ke-4: Uptime = 24 jam, Keberhasilan = 23 jam
Hari ke-5: Uptime = 24 jam, Keberhasilan = 23 jam
Hari ke-6: Uptime = 24 jam, Keberhasilan = 22 jam
Hari ke-7: Uptime = 24 jam, Keberhasilan = 24 jam
Total Keberhasilan: 158 jam dari 165 jam
Persentase Durability: 95.76%
(venv) admin@ip-172-31-25-220:~/uji$
  
```

Gambar 4. 24 Hasil Analisis dari Data yang disimpan

Availability menunjukkan lamanya perangkat tetap aktif selama pengujian. Setiap jam dihitung sebagai *uptime* jika perangkat menyala, sementara jam tanpa data sama sekali akibat mati atau putus koneksi dihitung sebagai *downtime*.

Tabel 4. 12 Hasil Pengujian Availability

Hari Ke	Uptime (jam)	Downtime (jam)
1	24	0
2	24	0

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

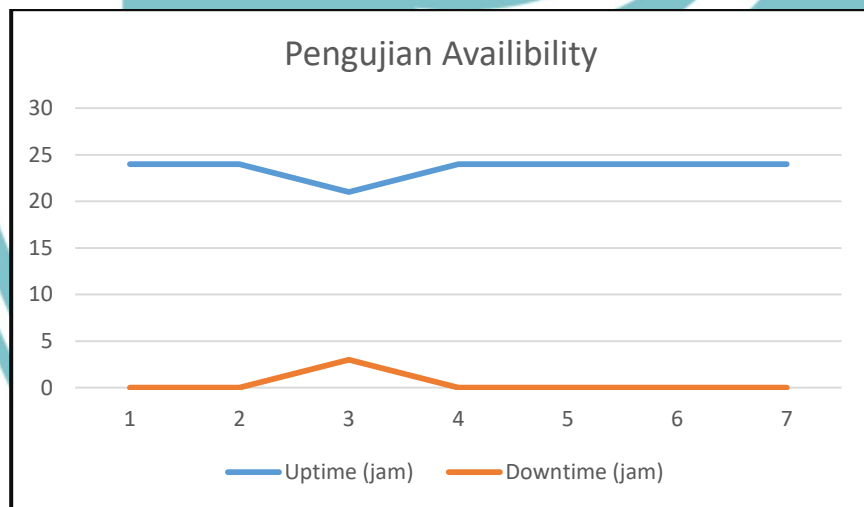
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3	21	3
4	24	0
5	24	0
6	24	0
7	24	0
Total	165	3
Presentase Availability		98.21 %

Nilai *Availability* sebesar 98.21% menunjukkan bahwa perangkat mampu menjaga status aktif hampir sepanjang waktu pengujian. Hal ini mencerminkan bahwa sistem cukup stabil dalam hal daya dan konektivitas, dengan gangguan yang sangat minimal hanya terjadi pada hari ketiga. Nilai ini tergolong sangat baik untuk sistem pemantauan lingkungan berbasis *IoT* yang beroperasi di kondisi luar ruangan tanpa pemeliharaan langsung.



Gambar 4. 25 Grafik Pengujian Availability

Berikutnya, *Durability* mengukur konsistensi sistem dalam mengirimkan data selama alat aktif. Satu jam dianggap berhasil jika terdapat minimal 30 data terkirim.

Tabel 4. 13 Hasil Pengujian Durability

Hari Ke	Uptime (jam)	Keberhasilan (jam)
---------	--------------	--------------------

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

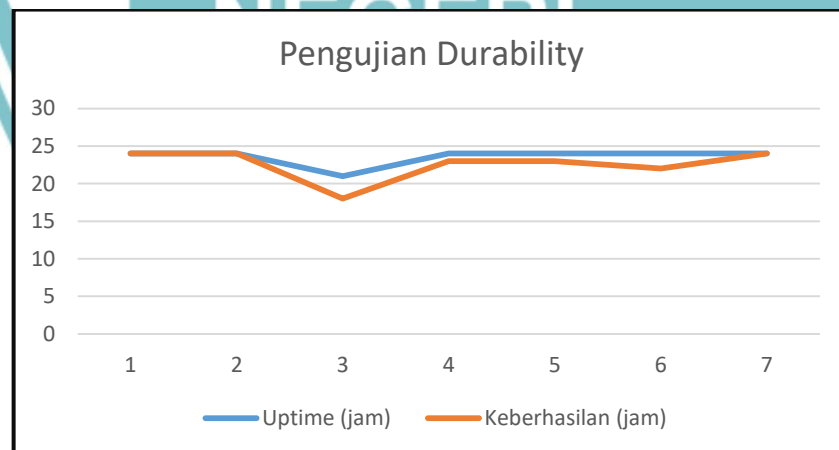
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

1	24	24
2	24	24
3	21	18
4	24	23
5	24	23
6	24	22
7	24	24
Total	165	158
Presentase <i>Durability</i>		95.76%

Dengan nilai *Durability* sebesar 95.76%, sistem terbukti tidak hanya aktif, tetapi juga konsisten dalam menjalankan fungsinya, yakni mengirimkan data secara rutin. Mayoritas jam uptime berhasil memenuhi ambang minimal data, menunjukkan bahwa sistem tidak mengalami hambatan signifikan dalam proses pengiriman data. Ini mengindikasikan bahwa perangkat layak diandalkan untuk aplikasi pemantauan *Real-Time* dengan ritme pengiriman data tinggi secara terus-menerus.



Gambar 4. 26 Grafik Pengujian Durability

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.5.3 Hasil Pengujian *Quality of Service (QoS)*

Pengujian *Quality of Service (QoS)* dilakukan untuk mengukur performa sistem dalam pengiriman data melalui jaringan seluler menggunakan modul *SIM800L*. Dua parameter utama yang diuji adalah *Delay* dan *Throughput*.

Delay mengukur waktu tunda dari saat data dikirim oleh perangkat hingga diterima oleh broker *MQTT*. Nilai *Delay* diperoleh dari selisih antara timestamp yang dikirim oleh perangkat dan waktu penerimaan data yang dicatat di sisi server. *Throughput* mengukur jumlah data (dalam *byte*) yang berhasil diterima oleh server per satuan waktu. Nilai ini mencerminkan efisiensi komunikasi data antar perangkat dan broker *MQTT* selama sistem berjalan.

```
(venv) admin@ip-172-31-25-220:~/uji$ python delay.py
/home/admin/uji/delay.py:58: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
client = mqtt.Client()
Connected to MQTT Broker!
[Delay] 725 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 17:59:34 | Server Time: 2025-06-11 17:59:34.725758
[Delay] 1122 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 17:59:45 | Server Time: 2025-06-11 17:59:46.122591
[Delay] 1223 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 17:59:56 | Server Time: 2025-06-11 17:59:57.223444
[Delay] 1241 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:00:07 | Server Time: 2025-06-11 18:00:08.241394
[Delay] 1462 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:00:18 | Server Time: 2025-06-11 18:00:19.462156
[Throughput] 350 bytes / 60 s = 5.83 byte/s

[Delay] 1563 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:00:29 | Server Time: 2025-06-11 18:00:30.563353
[Delay] 479 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:00:41 | Server Time: 2025-06-11 18:00:41.479435
[Delay] 821 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:00:52 | Server Time: 2025-06-11 18:00:52.821044
[Delay] 920 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:01:03 | Server Time: 2025-06-11 18:01:03.920182
[Delay] 804 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:01:14 | Server Time: 2025-06-11 18:01:14.804578
[Delay] 1282 ms | Tinggi Air: 45.2 cm | Debit: 1.67 L/min
Device Time: 2025-06-11 18:01:25 | Server Time: 2025-06-11 18:01:26.282592
[Throughput] 420 bytes / 60 s = 7.00 byte/s
```

Gambar 4. 27 Hasil Pengujian QoS yang Dicatat di Server

Tabel 4.14 menunjukkan hasil pengukuran *Delay* dari 15 data pengiriman sensor yang dicatat dalam satuan milidetik (ms):

Tabel 4. 14 Hasil Pengujian *Delay*

No	<i>Delay</i> (ms)
1	725
2	1122
3	1223
4	1241
5	1462

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

6	1563
7	479
8	821
9	920
10	804
11	1282
12	1280
13	1121
14	1642
15	1620

Rata – Rata <i>Delay</i> (ms)	1156.53
-------------------------------	---------

Berdasarkan kategori *Delay* yang telah ditentukan, nilai rata-rata *Delay* sebesar 1156.53 ms ini masuk ke dalam kategori Kurang, yang menunjukkan bahwa waktu tunda pengiriman data relatif tinggi dan dapat berdampak pada kecepatan respons sistem.

Berikutnya, *Throughput* diukur dengan menghitung jumlah *byte* data yang berhasil diterima selama 10 detik. Contoh hasil pengukuran *Throughput* dari pengujian adalah sebagai berikut:

- 510 *byte* / 60 s = 8.50 *byte/s*
- 540 *byte* / 60 s = 9.00 *byte/s*
- 495 *byte* / 60 s = 8.25 *byte/s*

Nilai *Throughput* ini dihitung berdasarkan total data *JSON* yang berhasil dikirim dan diterima broker *MQTT* selama periode 60 detik. Karena sistem mengirim data setiap 10 detik, maka dalam satu menit ada sekitar 6 kiriman, sehingga total *byte* yang diterima berkisar 480–540 *byte*. Dengan rata-rata *Throughput* sekitar 8.5 *byte/s*, sistem menunjukkan performa pengiriman data yang cukup stabil dan efisien untuk kebutuhan pemantauan sensor secara periodik berbasis jaringan seluler.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem pemantauan ketinggian dan debit air berbasis *IoT* yang telah dilakukan, maka diperoleh kesimpulan sebagai berikut:

1. Sistem pemantauan ketinggian dan debit air yang dikembangkan mampu mendeteksi potensi banjir secara dini di Komplek Griya Asri. Hal ini dibuktikan dengan kemampuan sistem membaca kondisi air secara *Real-Time* dan mengirimkannya ke server untuk ditampilkan dalam bentuk notifikasi dan grafik.
2. Sensor *JSN-SR04T* dan *YF-S201* memberikan tingkat akurasi yang cukup baik dalam mengukur ketinggian dan debit air. Berdasarkan hasil pengujian, rata-rata error sensor ketinggian adalah 0,72 cm, dan sensor debit menghasilkan pembacaan yang stabil serta sesuai perhitungan manual.
3. Modul komunikasi *SIM800L* terbukti andal dalam mengirimkan data sensor secara berkala ke broker *MQTT*. Selama pengujian, sistem menunjukkan tingkat keberhasilan pengiriman yang tinggi, meskipun terdapat *Delay* akibat keterbatasan jaringan seluler. Secara keseluruhan, komunikasi berjalan stabil dan mendukung pengiriman data *Real-Time*.
4. Dari sisi keandalan sistem, pengujian *Availability* dan *Durability* menunjukkan performa yang baik. Sistem dapat beroperasi secara kontinyu selama beberapa hari, dengan uptime yang konsisten dan tingkat keberhasilan pengiriman data yang tinggi.
5. Meskipun terdapat kendala berupa *Delay* pada pengiriman data, sistem secara keseluruhan dinilai efektif dan layak digunakan untuk pemantauan kondisi air sebagai bagian dari strategi mitigasi banjir.

5.2 Saran

1. Gunakan modul komunikasi *4G* (seperti *SIM7600*) untuk mengurangi *Delay* dan meningkatkan kecepatan pengiriman data.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2. Tambahkan integrasi AI untuk analisis data historis dan prediksi potensi banjir secara otomatis.
3. Tempatkan alat/sensor pemantauan di lokasi yang berada agak jauh sebelum area rawan banjir, sehingga sistem dapat memberikan peringatan lebih dini dan meningkatkan efektivitas mitigasi banjir.
4. Lakukan edukasi masyarakat agar lebih siap merespons peringatan dari sistem pemantauan.
5. Pemerintah perlu melakukan perbaikan serius terhadap kondisi aliran sungai di Depok, termasuk normalisasi sungai, pengangkatan endapan, serta perawatan Situ Studio Alam yang sering bocor. Tanpa dukungan infrastruktur yang baik, sistem monitoring tidak akan efektif dalam pencegahan banjir.

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR PUSTAKA

- Alfarizi, M. R. S., Al-farish, M. Z., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2023). Penggunaan *Python* Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning. *Karya Ilmiah Mahasiswa Bertauhid (KARIMAH TAUHID)*, 2(1), 1–6.
- Anorgi, A., Saragih, A. S., Noor, N., & Sari, K. (2023). Rancang Bangun Prototype Monitoring Banjir Berbasis Website. *JOINTECOMS (Journal of Information Technology and Computer Science)*, 3(2), 2798–3862.
- Aryatama, A. R. (2022). *Rancang Bangun Prototype Sistem Monitoring Kualitas Air Toren Berbasis IoT*. 6–74.
- Eka Putri, S., Corp, A. F., Rembrandt, Dasman Lanin, Genius Umar, & Mulya Gusman. (2023). Kota Padang : Identifikasi Potensi Bencana Banjir Dan Upaya Mitigasi. *Jurnal Ilmiah Multidisiplin Nusantara (JIMNU)*, 1(3), 116–122. <https://doi.org/10.59435/jimnu.v1i3.56>
- Ghaniyyah, F., & Eka Putri, R. (2023). Sistem Monitoring Penggunaan Air Kamar Kos. *Chipset*, 4(01), 80–87. <https://doi.org/10.25077/chipset.4.01.80-87.2023>
- Hanif, A., & Amri, R. (2023). Implementasi *Internet of things* Pada Protokol MQTT Dan HTTP Dalam Sistem Pendeteksi Banjir. *INOVTEK Polbeng - Seri Informatika*, 8(2), 498. <https://doi.org/10.35314/isi.v8i2.3767>
- Iqbal, M., Rosadi, A., & Kresna Andana, E. (2023). Perancangan Deteksi Dini Banjir Berbasis IoT Dan Water Level Dengan Notifikasi Blynk Dan Alarm. *Seminar Nasional Teknologi Informasi & Ilmu Komputer*, x, 6–13.
- Istiqomah, G., Yuliatmojo, P., & Yusro, M. (2022). *Sistem Monitoring Ketinggian Air, Debit Air, Curah Hujan Pada Sungai Berbasis IoT*. 5(2), 60–64. [http://repository.unj.ac.id/id/eprint/33432%0Ahttp://repository.unj.ac.id/33432/7/DAFTAR PUSTAKA.pdf](http://repository.unj.ac.id/id/eprint/33432%0Ahttp://repository.unj.ac.id/33432/7/DAFTAR%20PUSTAKA.pdf)

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- Kurnianto, A., Dedy Irawan, J., Ariwibisono, F. X., & Wardhana, A. (2022). Penerapan *IoT (Internet of things)* Untuk Controlling Lampu Menggunakan Protokol *MQTT* Berbasis Web. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 6(2), 1153. <https://www.embedded.com/>
- Muamaroh, N., & Christanto, F. W. (2024). Pengukur Penggunaan Air Otomatis Menggunakan Water Flow Sensor *YF-S201* dan NodeMCU ESP8266 Berbasis *IoT*. *JIKO (Jurnal Informatika Dan Komputer)*, 8(1), 88. <https://doi.org/10.26798/jiko.v8i1.1104>
- Pandiang, D. T. (2021). Perancangan Sistem Alat Kontrol Lampu menggunakan Perintah SMS dengan Modul GSM SIM 800l berbasis Metode *Arduino*. *JUKI : Jurnal Komputer Dan Informatika*, 3(2), 52–58. <https://doi.org/10.53842/juki.v3i2.61>
- Prakoso, A. D. (2020). Analisis Perbandingan Kualitas Layanan Sistem Antara Protokol HTTP dan *MQTT* Pada Monitoring Kelembaban Tanah. *PRotek : Jurnal Ilmiah Teknik Elektro*, 7(2), 55–59. <https://doi.org/10.33387/protk.v7i2.1658>
- Ramadhani, D. A., Hidayat, E. P., & Nugraha, A. T. (2022). Pemanfaatan Sensor Ultrasonik sebagai Purwarupa Pengukur Ketinggian Air pada Tangki Pembuangan Air Kotor di Kapal. *Elektriese: Jurnal Sains Dan Teknologi Elektro*, 12(02), 109–116. <https://doi.org/10.47709/elektriese.v12i02.1871>
- Ristian, U., Pangestu, D., & Muid, A. (2018). Purwarupa Sistem Informasi Titik Lokasi Dan Intensitas Curah Hujan Di Kota Pontianak Berbasis Website. *Coding Jurnal Komputer Dan Aplikasi*, 6(3), 247–254. <https://doi.org/10.26418/coding.v6i3.29059>
- Septanto, H., & Hidayatullah, A. (2022). Perancangan Sistem Informasi Monitoring Proyek Berbasis Web Untuk Mendukung Implementasi Paperless Office. *Jurnal Tera*, 2(2), 34–43. <http://jurnal.undira.ac.id/index.php/jurnaltera/article/view/130>

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Syahrul Mubarak, M., & Izman Herdiansyah, M. (2023). Implementasi Cloud Computing Amazon Web Services (AWS) Pada Web Reservasi Kamar Hotel. *Kajian Ilmiah Informatika Dan Komputer*, 4(2), 698–708. <https://doi.org/10.30865/klik.v4i2.1212>

Waskito, A. M. W., Rochim, O. A. F., Asmid, F., & ... (2024). Rancang Bangun Sistem Pengawasan pada Ruangan Berbasis *Internet of things*. *Mars: Jurnal Teknik* ..., 2(4). <https://journal.artei.or.id/index.php/Mars/article/view/252%0Ahttps://journal.artei.or.id/index.php/Mars/article/download/252/422>

Widhiawan, B. A. W., Handoko, S., & Darjat, D. (2021). Perancangan Marindani, E. D., Sanjaya, B. W., & Gusmanto. (2016). Rancang Bangun Sistem Peringatan Dini Dan Pelacakan Pada Kendaraan Sepeda Motor Dengan Menggunakan Mikrokontroler *Arduino Nano*. *Jurnal Elektro*, 1–11. <https://www.tokopedia.com/microlife/gps-ubl>. *Transient: Jurnal Ilmiah Teknik Elektro*, 10(1), 17–25. <https://doi.org/10.14710/transient.v10i1.17-25>

Zalukhu, A., Singly, P., & Darma, D. (2023). Perangkat Lunak Aplikasi Pembelajaran *Flowchart*. *Jurnal Teknologi, Informasi Dan Industri*, 4(1), 61–70. <https://ejurnal.istp.ac.id/index.php/jtii/article/view/351>

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR RIWAYAT HIDUP



Natanael Siwalette

Lahir di Depok pada tanggal 24 Desember 2003. Penulis merupakan putra dari pasangan Simon Siwalette dan Sriyanti. Penulis memulai pendidikan formal di SDN Mekarjaya 4 Depok pada tahun 2009. Kemudian melanjutkan pendidikan menengah pertama di SMP PGRI Depok 2 Tengah pada tahun 2015. Setelah itu, penulis melanjutkan pendidikan menengah kejuruan di SMKN 3 Depok dengan jurusan Teknik

Komputer dan Jaringan pada tahun 2018. Pada tahun 2021, penulis melanjutkan pendidikan tinggi di Politeknik Negeri Jakarta, Jurusan Teknik Informatika dan Komputer, Program Studi Teknik Multimedia dan Jaringan.

**POLITEKNIK
NEGERI
JAKARTA**

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

LAMPIRAN

Lampiran 1 Dokumentasi





© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 2 Source Code *Arduino Nano*

```
#define TINY_GSM_MODEM_SIM800

#include <SoftwareSerial.h>
#include <TinyGsmClient.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#include <avr/wdt.h>

#define TRIG_PIN 8
#define ECHO_PIN 9
#define FLOW_PIN 2
#define LED_GREEN 10
#define LED_YELLOW 3
#define LED_RED 4
#define MODEM_RX 7
#define MODEM_TX 6

float tinggiSensorCm = 110.0;
float lastValidDistance = -1;

volatile unsigned long pulse = 0;
float calibrationFactor = 0.1253;

unsigned long lastReadTime = 0;
unsigned long lastPublishTime = 0;
const unsigned long readInterval = 1000;
const unsigned long publishInterval = 60000;

SoftwareSerial SerialAT(MODEM_RX, MODEM_TX);

const char apn[] = "M2MAUTOTRONIC";
const char gprsUser[] = "";
const char gprsPass[] = "";

const char* broker = "18.142.250.134";
const char* mqttUsername = "IOT";
const char* mqttPassword = "iot123";
const char* mqttTopic = "sensor/data";

TinyGsm modem(SerialAT);
TinyGsmClient client(modem);
PubSubClient mqtt(client);
```


© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta:

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
void increase() {
    pulse++;
}

bool mqttConnect() {
    return mqtt.connect("GsmClientN", mqttUsername, mqttPassword);
}

float bacaJarak() {
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    long durasi = pulseIn(ECHO_PIN, HIGH, 30000);
    if (durasi == 0) return 0.0;
    return (durasi * 0.0343) / 2.0;
}

float getMedian(float a, float b, float c) {
    if ((a >= b && a <= c) || (a >= c && a <= b)) return a;
    if ((b >= a && b <= c) || (b >= c && b <= a)) return b;
    return c;
}

void blinkYellowSetup() {
    for (int i = 0; i < 6; i++) {
        digitalWrite(LED_YELLOW, HIGH);
        delay(300);
        digitalWrite(LED_YELLOW, LOW);
        delay(300);
    }
}

void blinkAllSuccess() {
    for (int i = 0; i < 3; i++) {
        digitalWrite(LED_GREEN, HIGH);
        digitalWrite(LED_YELLOW, HIGH);
        digitalWrite(LED_RED, HIGH);
        delay(200);
        digitalWrite(LED_GREEN, LOW);
        digitalWrite(LED_YELLOW, LOW);
        digitalWrite(LED_RED, LOW);
        delay(200);
    }
}

void blinkRedReconnect() {
    for (int i = 0; i < 5; i++) {
        digitalWrite(LED_RED, HIGH);
        delay(300);
        digitalWrite(LED_RED, LOW);
        delay(300);
    }
}
```

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
void resetModem() {
    modem.restart();
    modem.gprsDisconnect();
    modem.gprsConnect(apn, gprsUser, gprsPass);
}

void setup() {
    pinMode(TRIG_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);
    pinMode(FLOW_PIN, INPUT);
    pinMode(LED_GREEN, OUTPUT);
    pinMode(LED_YELLOW, OUTPUT);
    pinMode(LED_RED, OUTPUT);
    attachInterrupt(digitalPinToInterrupt(FLOW_PIN), increase, RISING);
    SerialAT.begin(9600);
    digitalWrite(LED_GREEN, LOW);
    digitalWrite(LED_YELLOW, LOW);
    digitalWrite(LED_RED, LOW);
    blinkYellowSetup();

    modem.restart();
    modem.gprsDisconnect();
    while (!modem.gprsConnect(apn, gprsUser, gprsPass)) {
        delay(500);
    }

    mqtt.setServer(broker, 1883);
    while (!mqtt.connect()) {
        blinkRedReconnect();
        delay(500);
    }
    blinkAllSuccess();
}

void loop() {
    static float flowrate = 0.0;
    unsigned long now = millis();

    if (!modem.isGprsConnected()) {
        resetModem();
    }

    if (!mqtt.connected()) {
        blinkRedReconnect();
        mqttConnect();
    }

    mqtt.loop();
}
```


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
if (now - lastReadTime >= readInterval) {
    lastReadTime = now;

    // Hitung debit air
    noInterrupts();
    unsigned long count = pulse;
    pulse = 0;
    interrupts();
    flowrate = count * calibrationFactor;

    // Baca 3x jarak dan ambil median
    float d1 = bacaJarak(); delay(50);
    float d2 = bacaJarak(); delay(50);
    float d3 = bacaJarak();

    if (d1 != 0 && d2 != 0 && d3 != 0) {
        float median = getMedian(d1, d2, d3);
        lastValidDistance = median;

        float tinggiAir = tinggiSensorCm - median;
        if (tinggiAir < 0) tinggiAir = 0;

        // Update indikator LED setiap detik
        digitalWrite(LED_GREEN, LOW);
        digitalWrite(LED_YELLOW, LOW);
        digitalWrite(LED_RED, LOW);

        float level = tinggiAir / 100;
        if (level < 0.20) {
            digitalWrite(LED_GREEN, HIGH);
        } else if (level <= 0.80) {
            digitalWrite(LED_YELLOW, HIGH);
        } else {
            digitalWrite(LED_RED, HIGH);
        }
    }
}

// Kirim data ke MQTT setiap 1 menit
if (now - lastPublishTime >= publishInterval) {
    lastPublishTime = now;
    if (lastValidDistance >= 0) {
        float tinggiAir = tinggiSensorCm - lastValidDistance;
        if (tinggiAir < 0) tinggiAir = 0;

        StaticJsonDocument<128> doc;
        doc["tinggi_air"] = tinggiAir;
        doc["debit"] = flowrate;

        char buffer[128];
        serializeJson(doc, buffer);
        mqtt.publish(mqttTopic, buffer, true);
    }
}
```