

BAB II

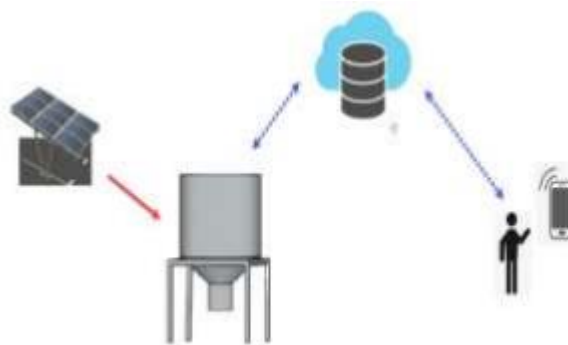
LANDASAN TEORI

2.1 Alat Penebar Pakan Ikan Otomatis

Alat penebar pakan ikan otomatis berbasis IoT merupakan sebuah sistem yang dirancang untuk memberikan pakan ikan secara otomatis sesuai takaran dan jadwal yang sudah ditentukan. (Aris Risnandar et al., 2021) Sistem ini menggunakan sensor ultrasonik untuk mendeteksi ketersediaan pakan dalam wadah pakan dan sensor RTC (Real-Time Clock) untuk mengatur jadwal pemberian pakan. Sistem komunikasi IoT yang digunakan yaitu protokol HTTP sebagai cara untuk mengendalikan alat secara jarak jauh melalui internet.

1. Rancangan Sistem

Terdapat beberapa bagian utama pada rancangan sistem alat penebar pakan ikan otomatis ini, diantaranya, Photovoltaic, alat pakan ikan otomatis, data pada server cloud dan pengguna. Untuk lebih jelasnya bisa dilihat pada gambar 2.1.



Gambar 2. 1 Rancang sistem alat
Sumber: (Aris Risnandar et al., 2021)

2. Desain Alat

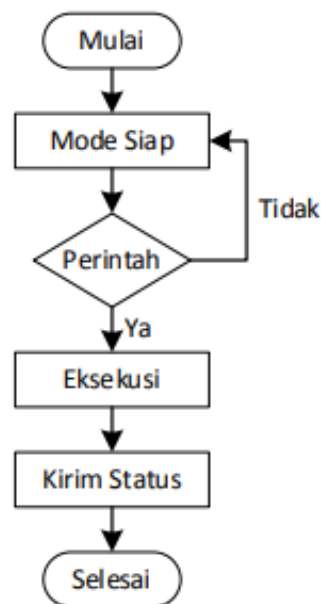
Sumber energi listrik yang digunakan dalam mengoperasikan alat penebar pakan ini menggunakan panel surya, dimana energi yang dihasilkan panel surya ini disimpan melalui baterai dengan pengaturan melalui Solar Charge Controller (SCC). Desain alat lebih jelas bisa dilihat pada gambar 2.2



Gambar 2. 2 Desain alat penebar pakan ikan
Sumber: (Aris Risnandar et al., 2021)

3. Diagram Alir Sistem

Dalam penggunaan sistem tersebut, maka diagram alur sistem yang akan dirancang ditunjukkan oleh :



Gambar 2. 3 Diagram alir sistem
Sumber: (Aris Risnandar et al., 2021)

Cara kerja alat ini secara umum adalah sebagai berikut:

- Sensor RTC mengukur waktu saat ini dan membandingkannya dengan jadwal pemberian pakan yang telah ditentukan.
- Sensor ultrasonik secara berkala mengukur ketinggian pakan dalam wadah.
- Mikrokontroler memproses informasi dari kedua sensor ini.
- Jika waktu sudah tepat dan tinggi pakan mencapai ambang batas, mikrokontroler mengaktifkan motor servo pemberian pakan.
- Mikrokontroler juga dapat mengirimkan laporan tentang pemberian pakan dan kondisi pakan ke server melalui koneksi internet.

4. Antarmuka Monitoring Alat

Terdapat antarmuka pengguna dari alat pakan ikan otomatis berbasis android. Antarmuka pengguna ini digunakan sebagai monitoring pakan dan pengaturan mode pemberian pakan pada ikan. (Taufiqurrahman et al., 2023). Bisa dilihat pada gambar 2.4.



Gambar 2. 4 Antarmuka monitoring alat
Sumber: (Taufiqurrahman et al., 2023)

Dengan sistem ini dapat mengoptimalkan pemberian pakan kepada ikan dengan mempertimbangkan jadwal yang tepat dan menghindari pemberian pakan berlebihan atau kurang. Selain itu, penggunaan teknologi IoT memungkinkan pengendalian dan pemantauan yang lebih mudah dari jarak jauh, sehingga dapat mengatur alat ini bahkan saat tidak berada di dekatnya.

2.2 *IoT (Internet of Things)*

Internet of Things (IoT) merupakan suatu konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus menerus (Al-Fuqaha et al., 2015). Pada dasarnya IoT (Internet of Things) mengacu

pada benda yang dapat diidentifikasi secara unik sebagai representative virtual dalam struktur berbasis internet. Cara Kerja IoT (Internet of Things) adalah interaksi antara sesama mesin yang terhubung secara otomatis tanpa campur tangan user dan dalam jarak berapa pun. Agar tercapainya cara kerja IoT (Internet of Things) tersebut diatas internet menjadi penghubung di antara kedua interaksi mesin tersebut, sementara user hanya bertugas sebagai pengatur dan pengawas bekerjanya alat tersebut secara langsung. Manfaat yang didapatkan dari konsep IoT (Internet of Things) ialah pekerjaan yang dilakukan bisa menjadi lebih cepat, mudah dan efisien. Sistem dasar dari IoT terdiri dari 3 hal yaitu:

- a. Hardware/fisik (Things)
- b. Koneksi Internet
- c. Cloud Data Center

2.2.1 *IoT Platform*

IoT Platform merupakan software berbasis web server yang berfungsi sebagai penghubung antara sensor, device dan berbagai layanan lainnya agar dapat terhubung ke server. IoT Platform juga memiliki fungsi sebagai alternative yang mampu mendukung komunikasi berbagai perangkat IoT dan menjamin validitas perangkat yang mengirimkan data. Dimana hasil dari request oleh sensor, device, dan layanan lainnya akan di kirimkan ke server dan kemudian server akan merespon kembali sesuai dengan request yang telah diterima. (Windryani et al., n.d.-a)

2.3 *Komunikasi Data*

Data secara bahasa diartikan sebagai keterangan yang benar dan nyata (KBBI, 2019), atau data dapat didefinisikan sebagai hasil observasi langsung

terhadap fenomena alam yang dilengkapi dengan nilai tertentu (Ati *et al.*, 2014). Sedangkan Komunikasi data merupakan suatu proses mengirim dan menerima data dari suatu perangkat pengirim ke perangkat yang menjadi tujuan (*destination*) yang terhubung dalam sebuah jaringan. Untuk terjadinya pertukaran data dalam sebuah sistem komunikasi data terdapat komponen yang harus ada yaitu diantaranya:

- a. *Message*, adalah informasi (data) yang akan dikomunikasikan, dapat berupa teks, angka, gambar, suara, ataupun video.
- b. *Sender* (pengirim), merupakan perangkat yang berperan sebagai pengirim pesan ataupun data, perangkat tersebut dapat berupa komputer, *workstation*, telpon genggam, kamera video dan lain sebagainya.
- c. *Receiver* (penerima), merupakan perangkat yang berperan menerima data atau pesan, perangkat tersebut dapat berupa komputer, *workstation*, telpon genggam, kamera video dan lain sebagainya.
- d. Media Transmisi, merupakan jalur fisik yang menyebabkan pesan dapat bergerak dari pengirim ke penerima, contoh dari media transmisi kawat *twisted-pair*, kabel koaksial, kabel serat optik, dan gelombang radio.

2.4 Protokol MQTT

MQTT (Message Queue Telemetry Transport) adalah sebuah protokol yang di desain untuk machine to machine, dengan bandwidth rendah sehingga data yang dikirimkan berukuran kecil yaitu hanya sebesar 2 bytes untuk setiap jenis data sehingga sangat ringan. Dengan meminimalkan bandwidth MQTT memberikan jaminan pengiriman data walaupun koneksi tidak stabil (Putra *et al.*, 2018). Pola yang digunakan MQTT dalam mengirimkan data yaitu publish dan subscribe.

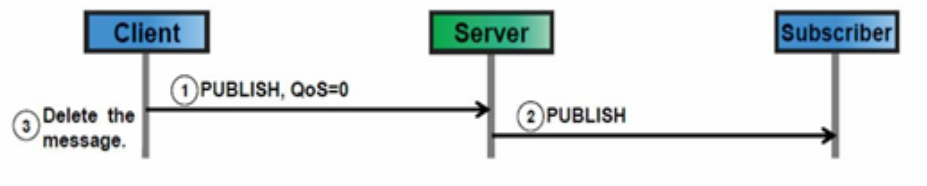
Pengirim data disebut dengan publisher dan penerima data disebut subscriber. Sehingga dalam hal ini memungkinkan pengiriman data dari satu publisher ke banyak subscriber dengan syarat yaitu subscriber tersebut memiliki sebuah topik data yang sama dengan publisher.

Oleh karena data yang ditransmisikan berukuran kecil atau lightweight message maka protokol MQTT ini sangat mendukung untuk teknologi Internet of Things (IoT). MQTT memiliki 3 jenis Quality of Service (QoS 0, QoS 1 & QoS 2). Kita akan dapat jaminan bahwa paket data yang kita kirimkan pasti akan sampai jika menggunakan QoS 1 & QoS 2. QoS merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari suatu servis sehingga pesan diterima (Hasugian, Akbar and Amron, 2018).

2.4.1 MQTT QoS

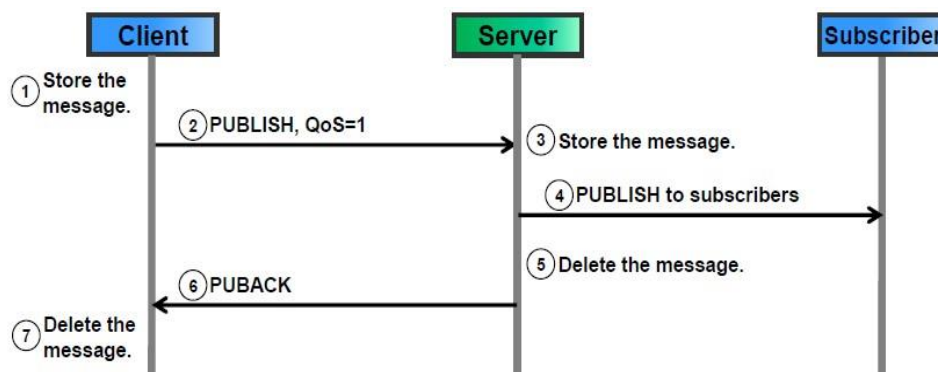
MQTT memiliki beberapa tingkat kualitas pelayanan / *Quality of Service* (QoS), yang menjadi pembeda dengan TCP/IP. Meskipun TCP/IP memiliki pengiriman data yang terjamin, akan tetapi hilang data masih tetap dapat terjadi pada saat koneksi terputus oleh karena itu MQTT menyediakan 3 tingkat kualitas pelayanan (QoS), yaitu :

- a. QoS 0, yaitu diperuntukkan untuk pengiriman data sekali, contoh pengaplikasiannya yaitu pada monitoring sensor suhu yang memperbaharui data secara berkala, meskipun kehilangan satu nilai suhu, sistem masih tetap terintegrasi. Aliran pesan QoS 0 dapat dilihat pada gambar 2.5.



Gambar 2. 5 Message flow QoS 0

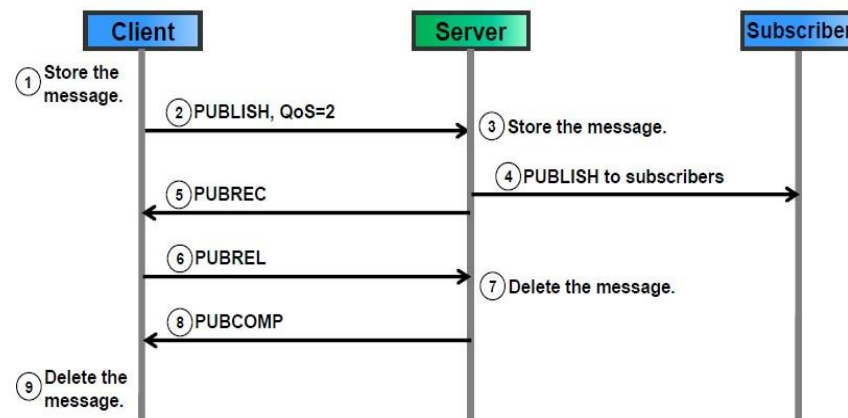
- b. QoS 1, yaitu diperuntukkan untuk pesan yang dijamin sampai untuk sekali pengiriman, namun mungkin akan ada pesan duplikat. pengirim akan menyimpan pesan sampai menerima paket PUBACK dari penerima pesan. Contoh pengaplikasiannya yaitu pada keadaan sensor gerbang/ pintu, karena penting untuk mengetahui keadaan pintu terbuka atau tertutup. Keadaan sensor pintu akan di *publish* tanpa kehilangan data pada penerima. Aliran pesan QoS 1 dapat dilihat pada gambar 2.6.



Gambar 2. 6 Message flow QoS 1

- c. QoS 2, yaitu diperuntukkan untuk pesan dengan jaminan pengiriman paling tinggi dengan setidaknya dua aliran permintaan/ *respons* antara pengirim dan penerima. Pengirim dan penerima akan menggunakan pengenal paket dari pesan *original publish* untuk mengkoordinasikan

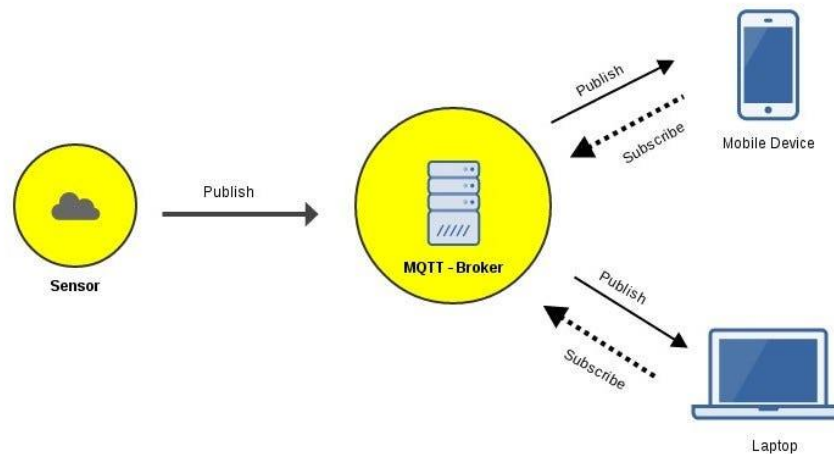
pengiriman pesan (Egli, 2016). Aliran pesan QoS 2 dapat dilihat pada gambar 2.7.



Gambar 2. 7 Message flow QoS 2

2.4.2 MQTT Broker

MQTT broker merupakan fasilitator yang menghubungkan antara *publisher* dan *subscriber*. Pada protokol MQTT, *broker* memegang peranan penting pada keberhasilan proses komunikasi. Karena komunikasi antara *subscriber* dan *publisher* bersifat asinkron yang artinya harus melalui sebuah *broker* sehingga antara *publisher* dan *subscriber* tidak perlu saling mengetahui dan berhubungan (Hayun and Wibisono, 2017). Secara sederhana Arsitektur protokol komunikasi dapat dijelaskan dengan gambar 2.8.



Gambar 2. 8 Arsitektur protokol komunikasi MQTT
Sumber: (medium.com)

2.5 Protokol HTTP

Hyper Text Transfer Protocol (HTTP) adalah protokol yang berjalan di atas TCP/IP dan menganut prinsip request/response pada client dan server, juga merupakan protokol yang mendasari World Wide Web. Semenjak HTTP beroperasi di atas TCP/IP maka tersedia komunikasi yang reliable, namun koneksi yang dijalankan oleh TCP terputus pada setiap akses karena akses data berbasis IP address dan URL berubah secara dinamis, oleh karena itu komunikasi untuk IoT menjadi masalah yang serius pada overhead yang tinggi selama berkomunikasi (Luthfi et al., 2018). HTTP traffic saat ini diakui sebagai internet traffic yang paling sering digunakan, terdiri dari HTTP request dan response message. Setiap transfer data terdiri dari request data dari client dan response dari server dengan data yang diminta, keduanya request maupun response mengandung identification (ID) dan control information pada headernya. HTTP request/response adalah pesan yang

mengandung request line, status line, header fields, empty line yang terdapat pada akhir header, dan mengandung message body sebagai pilihan (Shaposhnikov et al., n.d.)

METHOD	SP	URI	SP	HTTP-version	CRLF
---------------	-----------	------------	-----------	---------------------	-------------

Gambar 2. 9 Format *request line*
Sumber: (Shaposhnikov et al., n.d.)

HTTP-version	SP	Status code	SP	Reason message	CRLF
---------------------	-----------	--------------------	-----------	-----------------------	-------------

Gambar 2. 10 Format *status line*
Sumber: (Shaposhnikov et al., n.d.)

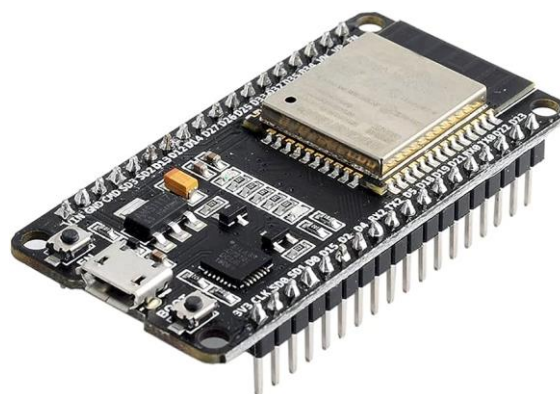
Pada Gambar 2.9 dan Gambar 2.10 menjelaskan mengenai *request line* dan *status line* dimana pada *request line* dari HTTP request di dalamnya terdapat starting line yang direpresentasikan pada kolom METHOD, URI sebagai resource indicator dan versi HTTP. Semua pesan HTTP diawali dengan starting line kemudian metode starting line memberi perintah apa yang harus dilakukan oleh server. Starting line yang sering digunakan diantaranya GET, HEAD, POST, adapun perintah GET digunakan untuk meminta server mengirim resource tanpa entity header, HEAD mirip dengan GET namun hanya mendapatkan bagian pertama dari dokumen yang dikirim oleh server dan tidak termasuk entity header, POST digunakan untuk mengirim data untuk diproses pada server dan biasanya digunakan untuk mendukung format HTML.

Status line dari HTTP response di dalamnya terdapat versi HTTP, status code, dan reason message. Balasan dari setiap request mengandung reason message

yang direpresentasikan juga dengan status code. Reason message dan status code ini menentukan kondisi pesan yang diterima oleh server atau yang dikirim oleh client. Status code digolongkan berdasarkan digit pertama. Digit 1xx merupakan representasi dari informational responses, 2xx merupakan representasi dari success, 3xx merupakan representasi dari redirection, 4xx merupakan representasi dari client errors, dan 5xx merupakan representasi dari server errors.

2.6 Mikrokontroler ESP32

ESP32 merupakan Smart on Chip (SoC) Wi-Fi dan kemampuan Bluetooth dua mode yang didesain berukuran minimalis dan hanya menggunakan sedikit rangkaian eksternal. Memiliki Frekuensi wifi 802.11 b/g/n, Prosesor 32-bit, 12-bit ADC, TCP/IP protokol stack, TR switch, LNA, SRAM, Bluetooth power amplifier dan jaringan. Chip tersebut dapat berkomunikasi melalui infrastruktur wifi menggunakan protokol IPv4, TCP/IP, dan HTTP (Aditya Firmansah & Eko Susilo, n.d.). Pada Gambar 2.11 menunjukkan Gambar Mikrokontroler ESP32.



Gambar 2. 11 Mikrokontroler ESP32

2.7 Sensor Suhu DS18B20

Sensor DS18B20 adalah sensor yang digunakan untuk mendeteksi suhu tubuh seseorang dan tahan air (waterproof). Output dari sensor DS18B20 berupa data digital. Karakteristik dari sensor ini antara lain, digunakan pada tegangan 3-5V, tingkat akurasi kesalahan yaitu $\pm 0,5^{\circ}\text{C}$ dengan kisaran suhu antara -10°C sampai 85°C , kabel merah pada sensor DS18B20 untuk VCC, kabel hitam pada sensor DS18B20 untuk GND, kabel kuning pada sensor DS18B20 untuk data, diameter kabel yaitu 4mm dengan Panjang 90cm (Aritonang, Abdi Bangsa and Rahmadewi,). Pada Gambar 2.12 menunjukkan Gambar sensor DS18B20.



Gambar 2. 12 Sensor suhu DS18B20

2.10 Sensor pH

Sensor *pH* berfungsi mengukur *pH* larutan dengan memperkirakan perbedaan potensial listrik antara dua elektroda. Sensor *pH* kombinasi adalah jenis sensor *pH* elektrokimia yang mencakup elektroda pengukur dan elektroda referensi. Elektroda pengukur merasakan perubahan nilai *pH* sementara referensi memberikan sinyal yang stabil untuk korelasi. Perangkat impedansi tinggi

digunakan untuk menunjukkan sinyal millivolt dalam satuan *pH*. Meteran kemudian berubah dari potensi ini menjadi pembacaan *pH*. Pada Gambar 2.13 menunjukkan Gambar sensor *pH Dfrobot*.



Gambar 2. 13 Sensor *pH Dfrobot*

2.11 Pengukuran Kinerja

2.11.1 *Wireshark*

Wireshark merupakan sebuah software sniffer freeware yang dapat didownload dengan mudah di www.wireshark.org. Program sniffer adalah program yang dapat digunakan apabila kita ingin ‘mengintip / mengendus / sniff’ sebuah jaringan, baik Ethernet maupun non-ethernet (Wulandari, 2016).

Wireshark adalah packet analyzer gratis dan open - Source. Tools ini seringkali digunakan untuk menemukan masalah pada jaringan, pengembangan perangkat lunak dan protokol komunikasi, dan pendidikan. *Wireshark* bersifat cross – platform dan menggunakan pcap untuk meng-capture paket jaringan. *Wireshark* dapat berjalan pada hampir semua sistem operasi yang tersedia dan mampu menangkap paket-paket data yang ada pada jaringan tersebut. Semua jenis paket informasi dalam berbagai format protokol pun akan dengan mudah ditangkap dan dianalisa (Tri Novita et al., 2021b).

Hal-hal yang dapat dilakukan wireshark:

- Network Administrator menggunakan wireshark untuk *troubleshoot* masalah jaringan.
- *Network Security* menggunakan wireshark untuk memecahkan masalah keamanan jaringan.
- Pengembang menggunakan untuk *debug* implementasi protokol.
- Pengguna menggunakannya untuk belajar protokol jaringan internalnya.
- Mendiagnosa permasalahan.
- *Mengcapture* informasi jaringan.
- Melakukan *decode* pada frame.
- Melakukan *filtering* pada *trace file*.

2.11.2 *Quality of Service (QoS)*

Quality of Service (QoS) merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat pada satu servis. Pada QoS menggunakan teknik untuk mengelola *throughput*, *delay*, *packet loss*, dan *jitter* dalam jaringan. Kemampuan suatu jaringan untuk menyediakan layanan yang baik dengan menyediakan bandwidth, mengatasi jitter dan delay. Parameter QoS adalah *throughput*, *delay*, dan *packet loss* (Rivai et al., 2018).

1. *Throughput*

Throughput yaitu kecepatan (rate) transfer data efektif, yang diukur dalam bps (bit per second). *Throughput* merupakan jumlah kedatangan

paket yang dapat teramati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut (Wulandari, 2016).

Kategori *throughput* dapat dilihat pada Tabel 2.1 dibawah ini:

Tabel 2. 1 Kategori *throughput*
Sumber: (THIPON)

Kategori <i>Throughput</i>	<i>Throughput (bps)</i>	<i>Indeks</i>
Sangat Bagus	100	4
Bagus	75	3
Sedang	50	2
Jelek	<25	1

Adapun persamaan untuk menghitung *throughput*:

$$\textit{Throughput} = \frac{\text{Jumlah data yang dikirim}}{\text{Waktu pengiriman data}}$$

2. *Delay*

Delay merupakan waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media fisik, congesti atau juga waktu proses yang lama. *Delay* diperoleh dari selisih waktu pengiriman antara satu paket dengan paket yang lainnya dalam satuan detik (Wulandari, 2016).

Kategori *delay* dapat dilihat pada tabel 2.2 dibawah ini:

Tabel 2. 2 Kategori *delay*
Sumber: (THIPON)

Kategori <i>Delay</i>	Besar <i>Delay</i> (ms)	<i>Indeks</i>
Sangat Bagus	< 150 ms	4
Bagus	150 ms s/d 300 ms	3
Sedang	300 ms s/d 450 ms	2
Jelek	> 450 ms	1

Adapun persamaan untuk menghitung delay:

$$Delay = \frac{Timespan}{Packet}$$

3. *Packet Loss*

Packet loss merupakan banyaknya paket yang hilang pada suatu jaringan paket yang disebabkan oleh tabrakan (collision), penuhnya kapasitas jaringan, dan penurunan paket yang disebabkan oleh habisnya TTL (Time To Live) paket. Kategori *packet loss* dapat dilihat pada tabel 2.3 dibawah :

Tabel 2. 3 Kategori *packet loss*
Sumber: (THIPON)

Kategori <i>Degradasi</i>	<i>Packet Loss</i> (%)	<i>Indeks</i>
Sangat Bagus	0	4
Bagus	3	3
Sedang	15	2
Jelek	25	1

Adapun persamaan untuk menghitung *packet loss*:

$$Packet Loss = \frac{(\text{Paket data yang dikirim} - \text{Paket data yang diterima})}{\text{Paket data yang dikirim}} \times 100\%$$

2.12 Kualitas Air

Salah satu faktor penting yang berpengaruh terhadap kesuksesan perikanan budidaya adalah aspek kualitas air kolam yang tergambar pada beberapa parameter fisik antara lain suhu, derajat keasaman (*pH*), oksigen terlarut maupun kekeruhan air. Sebagai tempat hidup ikan, perubahan parameter fisik tersebut dapat berpengaruh secara langsung terhadap pertumbuhan dan daya tahan ikan budidaya (Bhawiyuga & Yahya, 2019). Parameter kualitas air dapat dilihat seperti pada tabel 2.4 dibawah ini:

Tabel 2. 4 Parameter kualitas air kolam ikan
Sumber: (Hidayatullah et al., 2018).

No	Parameter	Nilai kisaran untuk budidaya ikan	Peralatan Pengukuran
1	Suhu	20 – 30 °C	Termometer
2	Kekeruhan	25 – 400 JTU	Turbiditymeter
3	<i>pH</i>	6,5 - 8	<i>pH</i> meter/Kertas lakmus

2.13 Penelitian Terkait

Berdasarkan studi tentang analisis perbandingan komunikasi data protokol HTTP dan MQTT sudah banyak dilakukan. Beberapa penelitian yang signifikan dapat dilihat pada Tabel 2.5.

Tabel 2. 5 Tabel penelitian terkait

NO	Judul Jurnal	Nama Peneliti	Tempat dan Tahun Penelitian	Pembahasan Jurnal
1	ANALISA PERBANDINGAN PROTOKOL MQTT DENGAN HTTP PADA IOT PLATFORM PATRIOT	Nindithia Putri Windryan	Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom, 2019	Penelitian ini bertujuan untuk dapat meminimalisir penggunaan Bandwidth jaringan dan kebutuhan sumber daya pada perangkat sehingga sangat solutif dalam pengembangan IoT Platform
2	PERBANDINGAN PERFORMANSI LATENCY PROTOKOL KOMUNIKASI HTTP DAN MQTT PADA INTERNET OF THINGS	Susetyo Bagas Bhaskoro	Politeknik Manufaktur Bandung, 2022	Penelitian ini bertujuan untuk melakukan perbandingan performansi parameter latency pada protokol HTTP dan MQTT. Penelitian ini melakukan analisis perbandingan protokol komunikasi data antara MQTT dan HTTP pada sistem IoT. Proses sensing yang dilakukan pada penelitian ini dengan mengambil data dari sebuah sensor DHT22 dan monitoring menggunakan media web.

NO	Judul Jurnal	Nama Peneliti	Tempat dan Tahun Penelitian	Pembahasan Jurnal
3	ANALISIS PERBANDINGAN PERFORMANSI MQTT DAN HTTP PADA PLATFORM IOT NODE RED	Elza Fitria Dwi Permatasari	Fakultas Informatika, Universitas Telkom, Bandung, 2019	Penelitian ini bertujuan untuk menganalisis sistem monitoring suhu dan kelembaban udara yang berbasis MQTT dan HTTP yang terkoneksi dengan NodeMCU, dan sensor suhu DHT22 yang berfokus pada perbandingan performansi kedua protokol yang digunakan.
4	ANALISIS PERBANDINGAN KUALITAS LAYANAN SISTEM ANTARA PROTOKOL HTTP DAN MQTT PADA MONITORING KELEMBABAN TANAH	Anang Dwi Prakoso, Fikra Titan Syifa, Danny Kurnianto.	Program Studi Teknik Elektro Telekomunikasi Fakultas Teknik Telekomunikasi dan Institut Teknologi Telkom Purwokerto, 2020	Penelitian ini bertujuan untuk mengetahui perbandingan kualitas layanan pada protokol HTTP dan MQTT terhadap perangkat sensor kelembaban tanah
5	ANALISIS DAN IMPLEMENTASI PROTOTIPE PENGATUR KELEMBABAN BERBASIS INTERNET OF THINGS (IoT) PADA PENYIMPANAN SAYUR	Fadhil Riardy Rivai	Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom, Bandung, 2018	Penelitian ini bertujuan untuk merancang sebuah prototipe pengatur kelembaban pada penyimpanan sayur berbasis arduino dan Monitoring data dengan konsep internet of things (IOT). pengujian dan analisis Quality of Service

				(QoS) yang dilakukan pada prototipe dengan metode protokol MQTT dan HTTP
--	--	--	--	--

Berdasarkan studi tentang analisis perbandingan protokol HTTP dan MQTT dalam sistem komunikasi data perangkat IoT yang membedakan penelitian ini dengan penelitian terkait adalah sensor, mikrokontroler, protokol MQTT dan perangkat IoT. Pada penelitian ini menggunakan sensor Suhu dan sensor *pH*. Mikrokontroler yang digunakan pada penelitian ini adalah ESP32 sedangkan pada penelitian sebelumnya menggunakan mikrokontroler Arduino uno, atmega328 dan Raspberry PI 3. Sistem komunikasi data yang digunakan adalah protokol HTTP dan MQTT yang merupakan unit terpenting disini sebagai jembatan atau pengantar komunikasi data dari sistem atau unit ke internet.