

BAB III

METEDOLOGI PENELITIAN

3.1 Subyek dan Obyek Penelitian

Subyek penelitian merupakan individu yang ikut serta dalam penelitian. Informasi atau data dikumpulkan untuk membantu menjawab pertanyaan yang sedang diteliti. Subyek penelitian dalam penelitian ini yaitu sistem *internet of things* berbasis *website* Node.js pada google cloud platfrom untuk monitoring tanaman cabai dengan mengambil data menggunakan Sensor Capacitive Soil Moisture, DHT22, dan BH1750.

Obyek penelitian adalah himpunan elemen yang akan diteliti. Obyek penelitian dalam penelitian ini yaitu protokol komunikasi MQTT dan HTTP berdasarkan parameter *throughput*, *jitter*, *packet loss*, dan *delay*.

3.2 Alat dan Bahan Penelitian

Adapun kebutuhan *hardware* dan *software* sebagai bagian dari proses penelitian dan *design* sistem meliputi:

3.2.1 Perangkat Keras (*Hardware*)

1. Laptop dengan *processor* Celeron 4205U, 8GB RAM, 512GB SSD, dan sistem operasi Windows 11.
2. NodeMCU ESP8266 12E berfungsi sebagai *Microkontrol*.
3. Sensor Capacitive Soil Moisture berfungsi untuk mengambil data kelembaban tanah.
4. Sensor DHT22 berfungsi untuk mengambil data suhu dan kelembaban udara.
5. Sensor BH1750 berfungsi untuk mengambil data intensitas cahaya.

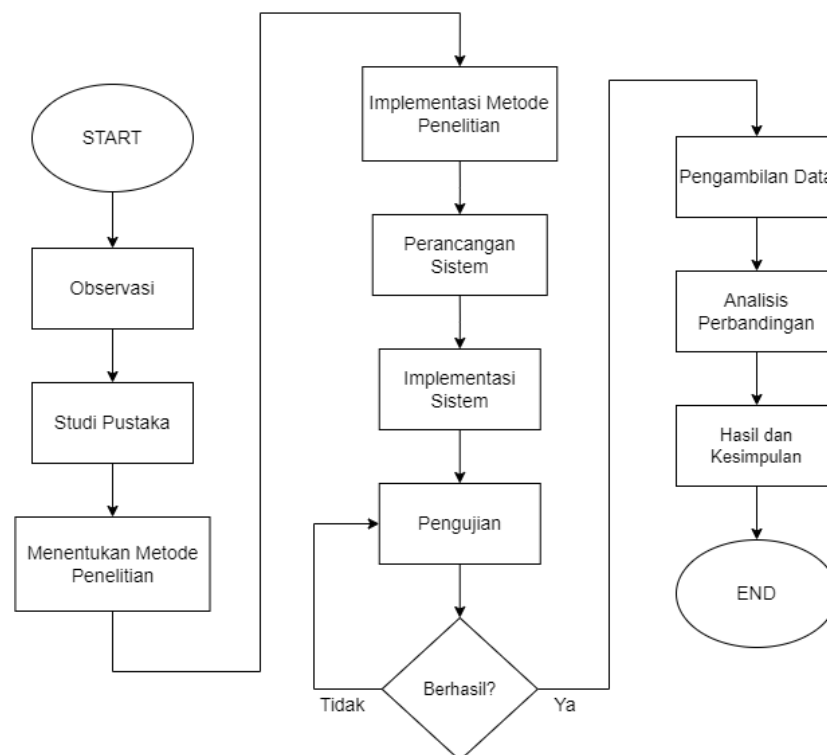
3.2.2 Perangkat Lunak (*Software*)

1. Arduino IDE berfungsi untuk membuat, memverifikasi, dan mengunggah kode program ke nodemcu.
2. Mosquitto Broker berfungsi sebagai perantara antara *publisher* dan *subscriber*.

3. Node.JS berfungsi sebagai *website* untuk membuat aplikasi *IoT*.
4. Google Cloud Platform berfungsi sebagai *platform cloud* untuk *deploy website* Node.js.
5. Chrome berfungsi sebagai *browser* untuk menampilkan *website* Node.js.
6. Wireshark berfungsi untuk mendapatkan dan menganalisis paket data MQTT dan HTTP.
7. Microsoft Excel berfungsi untuk mengolah data paket MQTT dan HTTP.

3.3 Diagram Alir Penelitian

Metode yang digunakan dalam penelitian ini adalah metode eksperimen. Metode ini digunakan untuk melakukan perancangan sistem monitoring tanaman cabai merah berbasis *website* Node.js. Adapun tahapan penelitian digambarkan melalui diagram dibawah ini:



Gambar 3.1 Diagram Alir

Gambar 3.1 merupakan tahapan penelitian yang akan dilakukan. Tahapan pertama penelitian adalah proses observasi, kemudian studi Pustaka, selanjutnya perancangan sistem, lalu Pengujian dan Pengambilan Data, dilanjut dengan analisis perbandingan, dan terakhir pengambilan hasil dan kesimpulan. Berikut merupakan detail dari setiap tahapan penelitian.

3.3.1 Observasi

Pada tahap ini dilakukan pengamatan pada jenis protokol komunikasi yang digunakan untuk berbagai sistem *internet of things*. Pengamatan juga dilakukan dengan membaca beberapa referensi mengenai kelebihan dan kekurangan dari beberapa protokol komunikasi IoT. Pada tahap observasi didapatkan protokol komunikasi yang umum dan banyak digunakan yaitu MQTT dan HTTP.

3.3.2 Studi Pustaka

Studi Pustaka dilakukan untuk mengetahui hubungan relevan antara penelitian ini dengan penelitian sebelumnya, sehingga dapat dijadikan sebagai referensi dan juga menghindari adanya kesamaan dalam penelitian yang akan dilakukan. Studi pustaka dilakukan dengan membaca dan membandingkan buku, jurnal, makalah tentang protokol komunikasi yang digunakan dalam sistem *internet of things* dan artikel terkait.

3.3.3 Menentukan Metode Penelitian

Metode yang digunakan pada penelitian ini yaitu menggunakan metode R&D. Penelitian ini menggunakan jenis penelitian Research and Development karena memungkinkan untuk memahami validitas suatu produk dalam waktu yang lebih singkat. Selain itu, metode ini juga melakukan eksplorasi mendalam untuk mengatasi kompleksitas dan mengidentifikasi solusi inovatif untuk mengintegrasikan perangkat-perangkat yang digunakan dengan cara yang efisien dan aman. Proses R&D sering melibatkan uji coba dan evaluasi berulang untuk memastikan bahwa sistem berfungsi dengan baik dalam lingkungan nyata dan memenuhi persyaratan yang telah ditentukan.

Metode R&D pada penelitian ini digunakan untuk melakukan eksperimen dengan melakukan perancangan sistem monitoring tanaman cabai merah berbasis website Node.js dan menguji keefektifan dari sistem tersebut, sehingga dapat menghasilkan data penelitian.

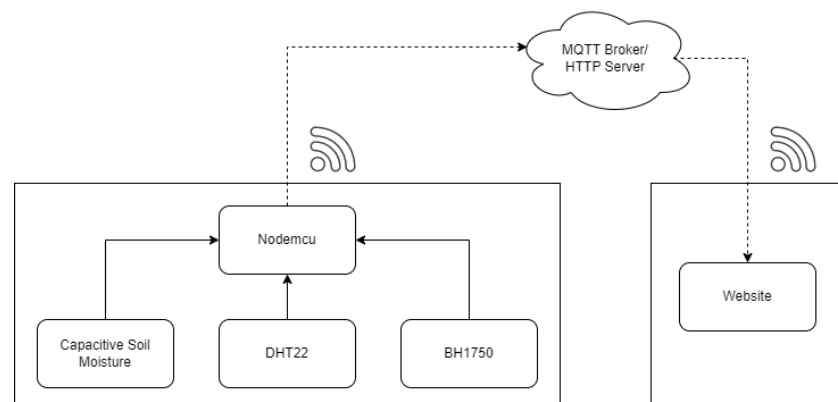
3.3.4 Implementasi Metode Penelitian

Pada tahap ini, peneliti menggunakan Metode R&D untuk melakukan perancangan sistem, implementasi sistem, dan melakukan uji coba pada sistem agar mendapatkan hasil data penelitian. Pada tahap perancangan sistem peneliti melakukan perancangan dengan membuat blok diagram sistem, perancangan hardware, dan perancangan website. Kemudian pada tahap implementasi sistem, peneliti melakukan implementasi protokol komunikasi MQTT dan HTTP terhadap sistem yang sudah dibuat. Selanjutnya pada tahap pengujian, peneliti melakukan uji coba terhadap sistem di beberapa kondisi dan waktu tertentu untuk memastikan sistem yang telah dibuat dapat digunakan untuk proses pengambilan data.

3.3.4.1 Perancangan Sistem

Pada tahap perancangan sistem dilakukan instalasi dan konfigurasi protokol komunikasi MQTT dan HTTP pada sistem monitoring tanaman cabai merah untuk mengambil data menggunakan Sensor Capacitive Soil Moisture, DHT22, dan BH1750.

3.3.4.1.1 Blok Diagram Sistem

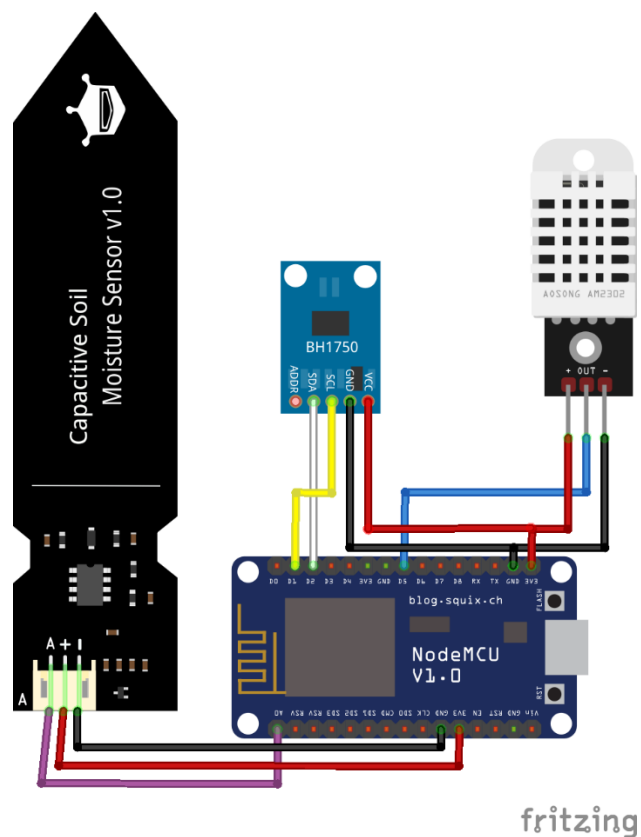


Gambar 3.2 Blok Diagram Sistem

Gambar 3.2 adalah blok diagram sistem atau cara kerja sistem monitoring tanaman cabai menggunakan sensor Capacitive Soil Moisture, DHT22, dan BH1750. Nodemcu mengambil data kelembaban tanah, suhu, dan intensitas cahaya. Selanjutnya data tersebut dikirimkan oleh Nodemcu menggunakan protokol MQTT dan HTTP melalui jaringan internet. Nodemcu juga berfungsi sebagai penghubung antara sensor, *internet*, MQTT Broker, dan HTTP Server. Data kelembaban tanah, suhu, dan intensitas cahaya akan ditampilkan melalui *website* Node.js menggunakan protokol MQTT dan HTTP.

3.3.4.1.2 Skema *Hardware*

Berikut merupakan skema *hardware* untuk sistem monitoring tanaman cabai merah.

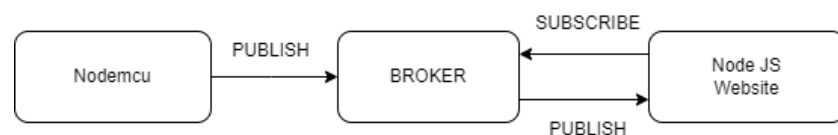


Gambar 3.3 Skema *Hardware*

Perancangan perangkat keras dalam pembuatan sistem monitoring tanaman cabai terdiri dari Nodemcu sebagai Mikrokontrol dan juga penghubung ke *internet*. Terdapat beberapa komponen yang terintegrasi pada Nodemcu diantaranya Sensor Capacitive Soil Moisture untuk mengambil data kelembaban tanah, Sensor DHT22 untuk mengambil data suhu, dan BH1750 untuk mengambil data intensitas cahaya.

3.3.4.1.3 Implementasi Protokol MQTT dan HTTP

Dalam mengimplementasikan protokol MQTT, dibutuhkan *library AsyncMqttClient* untuk menggunakan MQTT dengan ESP8266. Kemudian menginisiasi *host* atau *broker* MQTT, *port*, dan topik MQTT yang akan digunakan, *broker* yang digunakan dalam penelitian ini yaitu *Mosquitto* dengan *port* 1883. *Mosquitto* merupakan sebuah *broker* yang menggunakan protokol MQTT agar perangkat dapat berkomunikasi dengan mengirim dan menerima pesan. *Mosquitto* terdiri dari komponen yang terhubung ke *server* pengiriman pesan, *publish*, dan *subscrбие* suatu topik.



Gambar 3.4 Alur Kerja MQTT

Pertama menghubungkan nodemcu dengan *broker mosquitto* (*test.mosquitto.org*). Jika sudah terhubung selanjutnya *nodemcu* akan membaca sensor dan melakukan *publish* data sensor tersebut kedalam suatu topik pada *broker mosquitto*. Node.js berperan sebagai *subscriber* terhadap suatu topik yang digunakan untuk mendapatkan dan menampilkan data-data sensor kedalam *website*.

Sedangkan untuk protokol HTTP menggunakan metode permintaan HTTP *POST* dengan jenis permintaan *url encoded*. Dalam mengimplementasikan protokol HTTP, dibutuhkan *library WiFiClientSecure* untuk menjadikan nodemcu sebagai *client*.

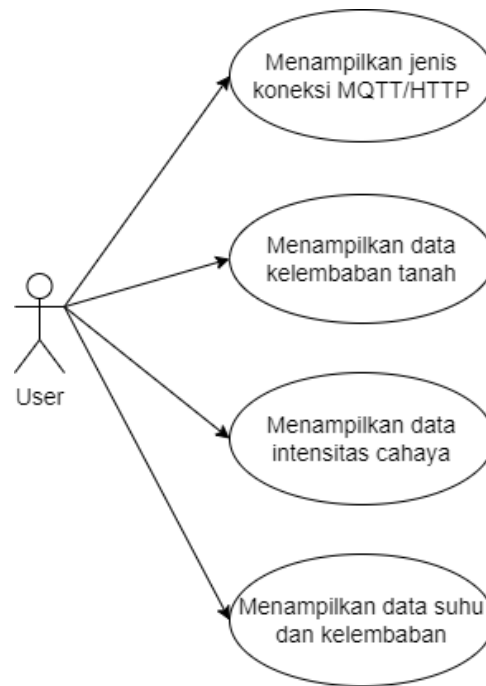


Gambar 3.5 Alur Kerja HTTP

Pertama menghubungkan nodemcu dengan *host http server* atau *website* yang digunakan. Selanjutnya nodemcu akan membaca sensor dan mengirimkan data sensor tersebut menggunakan metode HTTP *post*. Setelah itu, Node.js akan melakukan HTTP *get* untuk mendapatkan dan menampilkan data sensor kedalam *website*.

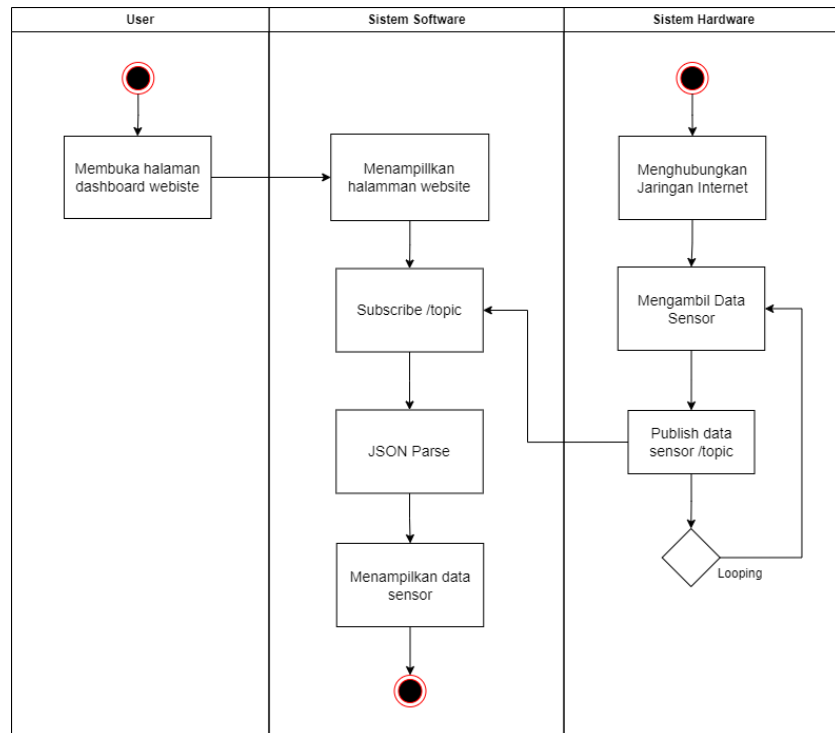
3.3.4.1.4 Perancangan Website

Berikut merupakan *use case diagram* dari website Node.js untuk sistem monitoring tanaman cabai merah.



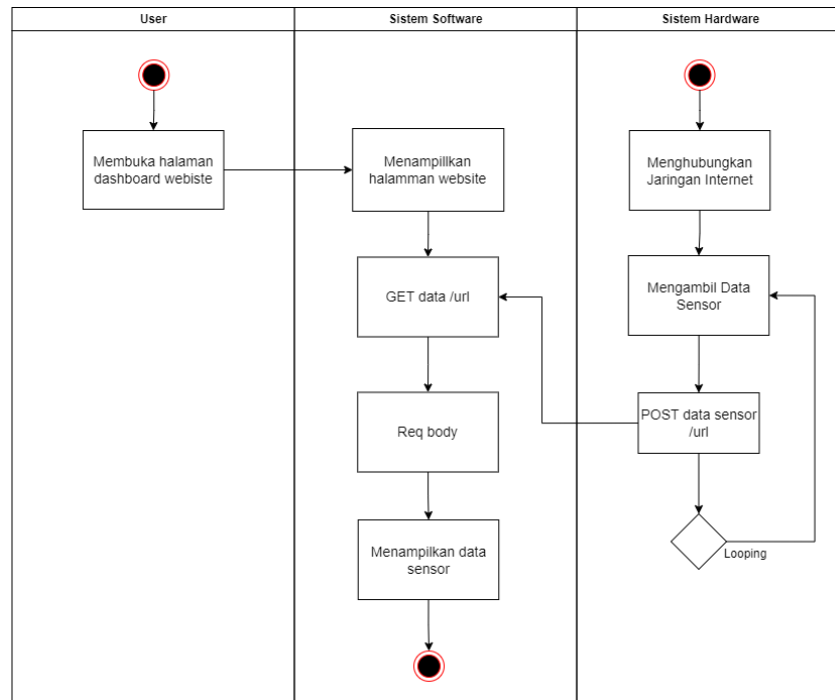
Gambar 3.6 *Use Case Diagram Website*

Pada gambar 3.6 adalah *use case diagram* dari *website*, *user* merupakan orang yang mengakses *website* atau melakukan monitoring terhadap tanaman cabai merah, *user* dapat mengakses halaman *dashboard website* dan menampilkan informasi seperti jenis koneksi yang digunakan, data kelembaban tanah, intensitas cahaya, suhu dan kelembaban.



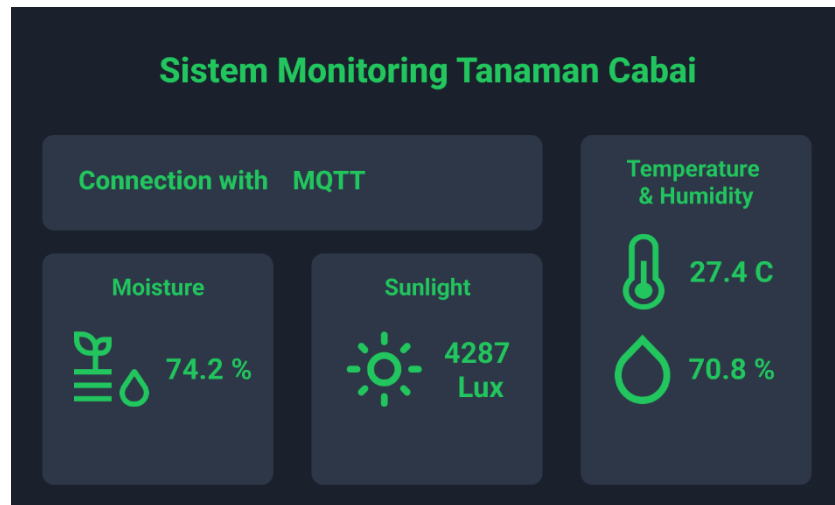
Gambar 3.7 Activity Diagram Website menggunakan MQTT

Pada gambar 3.7 adalah *activity diagram website* Node.js menggunakan protokol komunikasi MQTT untuk melakukan komunikasi data. Pertama *user* membuka halaman *dashboard website*, kemudian *nodemcu* akan menghubungkan ke jaringan internet, melakukan pengambilan data sensor dan *publish* data sensor pada suatu topik, *website* akan melakukan *subscribe* terhadap suatu topik tersebut, selanjutnya melakukan *json parse* untuk mengambil dan menampilkan data sensor kedalam *website*.



Gambar 3.8 Activity Diagram Website menggunakan HTTP

Pada gambar 3.8 adalah *activity diagram website* Node.js menggunakan protokol komunikasi HTTP untuk melakukan komunikasi data. Pertama *user* membuka halaman *dashboard website*, kemudian *nodemcu* akan menghubungkan ke jaringan internet, melakukan pengambilan data sensor dan *post* data sensor pada sebuah *url*, website akan melakukan *get* terhadap *url* tersebut, selanjutnya melakukan *request body* untuk mengambil dan menampilkan data sensor kedalam *website*.



Gambar 3.9 *Dashboard Website*

Pada Gambar 3.6 merupakan halaman *dashboard* dari *website* yang akan dibuat. Pada halaman *dashboard* terdapat beberapa informasi yang ditampilkan diantaranya jenis protokol komunikasi yang digunakan, yaitu MQTT atau HTTP. Kemudian terdapat beberapa informasi data-data sensor yang digunakan, diantaranya yaitu nilai *moisture* yang didapatkan dari sensor Capacitive Soil Moisture, *sunlight* yang didapatkan dari sensor BH1750, dan *temperature and humidity* yang didapatkan dari sensor DHT22.

3.3.4.2 Implementasi Sistem

Pada tahap ini peneliti melakukan implementasi protokol komunikasi MQTT dan HTTP terhadap sistem monitoring tanaman cabai merah berbasis website Node.js. Terdapat beberapa tahapan yang dilakukan, diantaranya yaitu implementasi *hardware* menggunakan mikrokontrol Nodemcu, sensor DHT22, sensor BH1750, dan Capacitive Soil Moisture. Kemudian melakukan implementasi *software* dengan membuat website menggunakan bahasa pemrograman Node.js yang digunakan sebagai *dashboard* monitoring tanaman cabai merah. Selanjutnya melakukan implementasi protokol komunikasi MQTT dan HTTP pada sistem *hardware* dan *software* yang sudah dibuat agar dapat

saling bertukar informasi data sensor. Pada tahap yang terakhir yaitu melakukan *deployment* website Node.js pada Google Cloud Platform agar dapat di akses secara publik melalui internet.

3.3.4.3 Pengujian

Pada tahap ini peneliti menggunakan metode *blackbox*. Penulis melakukan pengujian sistem mulai dari pengiriman data sensor menggunakan protokol komunikasi MQTT dan HTTP oleh nodemcu, sampai menampilkan data sensor tersebut pada halaman *website*. Pada tabel 3.1 merupakan beberapa tahapan kasus pengujian sistem :

Tabel 3. 1 Kasus Pengujian

Kode	Nama Kasus uji
P001	Nodemcu mampu melakukan <i>publish</i> data sensor menggunakan protokol MQTT
P002	Nodemcu mampu melakukan <i>post</i> data sensor menggunakan protokol HTTP
P003	<i>Website</i> mampu menampilkan data sensor dari <i>broker mosquitto</i>
P004	<i>Website</i> mampu menampilkan data sensor dari <i>server</i> melalui <i>API</i>
P005	Wireshark mampu menangkap dan menganalisis paket data MQTT
P006	Wireshark mampu menangkap dan menganalisis paket data HTTP

Pengujian protokol komunikasi MQTT dan HTTP dilakukan dengan cara mengecek informasi pengiriman data sensor pada mikrokontrol, *broker/server* web, dan *website* Node.js. Pengujian dilakukan dengan menghubungkan terlebih dahulu mikrokontrol Nodemcu ke jaringan *internet*, kemudian melakukan pengambilan dan pengiriman data sensor menggunakan protokol komunikasi MQTT dan HTTP. Pengujian juga dilakukan menggunakan *software* Wireshark

untuk memastikan data sensor yang dikirimkan oleh Nodemcu dapat terbaca atau terekam oleh *software* Wireshark.