

Pengelolaan Aset Berbasis Website Pada Sistem Pendeteksi Aset Berbasis Internet of Things

Handri Toar^{1*}, Ilham Ridha Alamsyah¹

¹Politeknik Negeri Batam, Batam, Indonesia

*Email: toar@polibatam.ac.id

Abstract—Pengelolaan aset merupakan kegiatan yang bertujuan untuk mengidentifikasi, memelihara, mengetahui ketersediaan aset dan informasi aset pada instansi. Setiap aset pada instansi harus selalu dipantau untuk mencegah terjadinya kehilangan aset dan mencegah hal-hal buruk yang tidak diinginkan. Umumnya, pengelolaan data aset masih bersifat konvensional, hal ini dapat memungkinkan terjadinya kesalahan dalam memasukkan data aset. Untuk mengurangi kesalahan dalam memasukkan data aset, dibutuhkan sebuah teknologi yaitu RFID (*Radio Frequency Identification*). RFID digunakan untuk mengidentifikasi aset dan memberikan informasi ketersediaan aset yang melewati suatu ruangan. Data tersebut akan terkirim ke ESP8266 kemudian dilakukan proses pengiriman data ke *server* yang terhubung pada internet, kemudian data tersebut tersimpan pada *database* MySQL (*my structured query language*) secara *realtime* dan data tersebut ditampilkan pada *website*. Untuk menjaga kualitas dari layanan *website*, dilakukan analisa QoS (*Quality of Service*) untuk mengetahui rata-rata *delay* pengiriman data selama 69,5181 ms, rata-rata *jitter* selama 69,5575 ms, dan *packet loss* sebesar 0%. Dari hasil pengujian, RFID *reader* dapat membaca data dari RFID *tag* dengan jarak sejauh 300 cm tanpa penghalang dengan tingkat keakurasian sebesar 100%, menggunakan penghalang kayu dengan tingkat keakurasian 100%, dan menggunakan penghalang besi dengan tingkat keakurasian 92%.

Keyword: ESP8266, MySQL, pengelolaan aset, RFID, *website*

I. PENDAHULUAN

PENGLOLAAN aset merupakan suatu proses untuk mendata dan memberikan nomor inventaris aset, mengetahui informasi aset, dan mengetahui jumlah aset. Untuk mengelola aset, harus dilakukan pemantauan terhadap aset yang ada pada ruangan. Pemantauan aset bertujuan untuk memudahkan dalam mengetahui lokasi aset dan lebih mudah untuk mendata aset [1], [2].

Namun pendataan informasi aset pada umumnya masih bersifat pencatatan manual, hal ini memungkinkan terdapat kesalahan dalam pendataan secara tertulis. Terlebih lagi apabila dalam proses pendataan tersebut terdapat aktifitas berkala yang menyebabkan adanya perbedaan kondisi data [3]. Hal ini dapat mempengaruhi perbedaan data pada pencatatan manual dengan data aktual dari aset [4]. Akibat dari hal ini, suatu instansi dapat mengalami kerugian yang seharusnya dapat diantisipasi. Dari permasalahan tersebut dibutuhkan

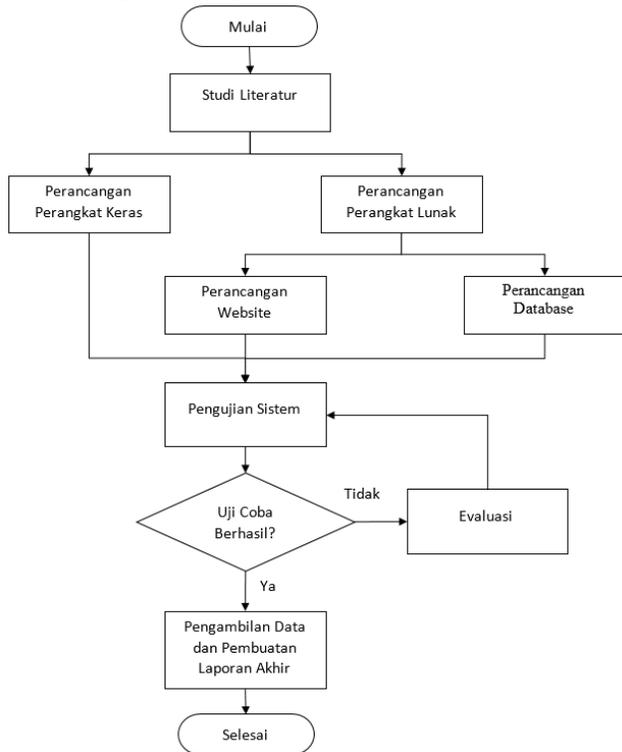
suatu sistem yang dapat digunakan untuk memberikan informasi aset dan melakukan pengelolaan aset dengan mudah.

Upaya untuk memudahkan dalam memantau aset dan mengelola aset menggunakan sebuah teknologi yaitu RFID yang memanfaatkan frekuensi radio untuk memberikan informasi dari aset. Ada beberapa metode yang dapat diimplementasikan pada sistem ini yaitu dengan menggunakan *barcode* dan RFID. Pada penelitian ini menggunakan RFID sebagai metode karena memiliki beberapa kelebihan dibandingkan dengan *barcode* yaitu RFID dapat melakukan *scanning* dari jarak yang jauh, RFID mampu mendeteksi *tag* yang terhalang oleh beberapa objek, dan RFID dapat mendeteksi *multiple tag* dalam satu waktu [5]. Sehingga RFID dapat meningkatkan efisiensi dan memudahkan dalam pembacaan data aset. Kemudian data dari pembacaan RFID *reader* dikirim ke *website* menggunakan jaringan internet dari *router*. RFID *reader* yang digunakan berjumlah dua digunakan pada dua ruangan. Setelah aset tersebut terdeteksi oleh RFID *reader*, maka data tersebut akan dikirim ke *database* dan ditampilkan pada *website*. *Website* berfungsi untuk melakukan pendaftaran aset, untuk mengelola aset pada setiap ruangan, untuk memberikan informasi aset berupa nama aset, nomor inventaris aset, lokasi aset, tanggal dan jam aset masuk. Untuk menjaga kualitas dari layanan *website*, dilakukan analisa QoS menggunakan tiga parameter dasar yaitu *packet loss*, *jitter*, dan *latency* yang digunakan untuk mengetahui kualitas layanan *website* [6].

Penelitian mengenai sistem pendeteksi barang dengan menggunakan RFID telah banyak dilakukan sebelumnya. Salah satunya penelitian, “Rancang Bangun Sistem Peminjaman dan Manajemen Aset Laboratorium Berbasis Implementasi RFID dan Aplikasi Web” [7]. Pada penelitian tersebut sistem pendeteksian dilakukan dengan melakukan *scanning* manual menggunakan RFID RC522 dan menampilkan pada *website*. Penelitian ini mengembangkan dari penelitian sebelumnya dengan membuat sistem yang dapat melakukan *scanning* otomatis, sistem ini juga dapat memantau setiap aset yang ada di beberapa ruangan dengan satu *mikrocontroller* dan melakukan pengukuran kecepatan pengiriman paket data pada protokol HTTP.

II. METODE

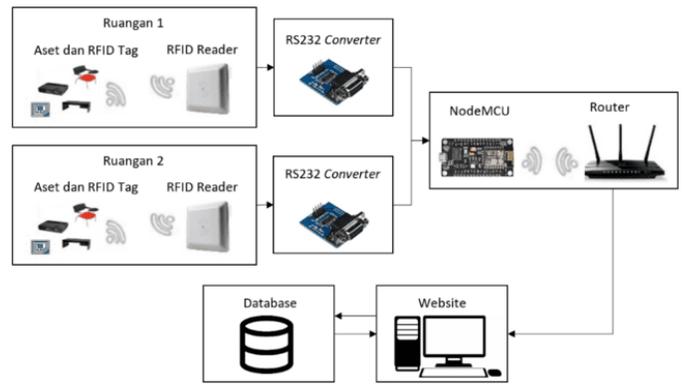
Pelaksanaan pembuatan sistem pendeteksi aset berbasis IoT terdiri dari dari studi literatur yaitu tahapan untuk mencari referensi dalam sebuah proyek. Perancangan perangkat keras yaitu proses menghubungkan beberapa komponen mulai dari mekanik hingga elektrik, Perancangan perangkat lunak terdiri dari merancang desain *website* dan *database*, kemudian dilakukan penggabungan sistem secara keseluruhan. Pengujian sistem dilakukan untuk menguji dari sistem seperti tingkat keakurasian pendeteksian RFID dan proses pengiriman paket data ke *server* sesuai tabel pengujian. Evaluasi yaitu proses untuk mengukur tingkat keberhasilan dari sistem, pengambilan data yaitu proses untuk mengambil beberapa sampel dari pengujian sistem, dan pembuatan laporan akhir. Rancangan Penelitian dapat dilihat pada Gambar 1.



Gambar. 1. Perancangan Sistem Pendeteksi Aset Berbasis IoT

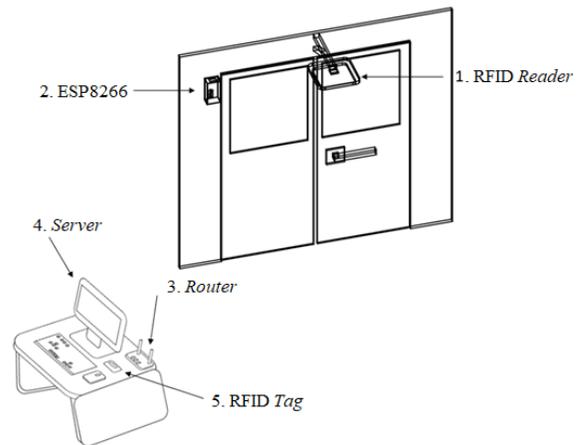
A. Perancangan Perangkat Keras

Penelitian ini menerapkan sistem *web based* dan *internet of things* untuk melakukan pengiriman paket data dengan menganalisa kualitas pengiriman data ke *server* menggunakan metode QoS dan tingkat sensitifitas pendeteksian dari RFID *reader*. Blok diagram sistem ini dapat dilihat pada Gambar 2.



Gambar. 2. Perancangan Perangkat Keras

Sistem ini menggunakan RFID *reader* yang diletakkan di atas pintu setiap ruangan agar lebih mudah dalam melakukan pendeteksian ke setiap aset yang masuk ke ruangan. Data aset yang melewati pintu akan terkirim ke *server* yang berada di ruangan *server*. Ilustrasi sistem ini dapat dilihat pada Gambar 3.



Gambar. 3. Ilustrasi Sistem Pendeteksi Aset Berbasis IoT

1) UHF RFID

RFID merupakan sebuah sensor yang digunakan untuk mengidentifikasi setiap RFID *tag* melalui gelombang radio. RFID *reader* yang digunakan bertipe UHF (*Ultra High Frequency*), frekuensi tipe UHF berkisar antara 902-928 MHz. dari Gambar 4 dan Gambar 5 menunjukkan RFID yang terdiri dari dua bagian yaitu *transceiver/reader* dan *Transponder/tag*. RFID *reader* dapat mengambil data dari RFID *tag* secara otomatis, sehingga pengguna tidak perlu untuk melakukan *scan* secara manual. Setiap RFID *tag* memiliki nilai ID yang unik, sehingga tidak ada yang memiliki ID yang sama. RFID juga dapat meningkatkan keefektifan dalam membaca sebuah ID dari RFID *tag* dari jarak yang jauh dikarenakan dari tipe frekuensi dari RFID dan RF *output* sehingga dapat melakukan *scanning* secara otomatis [8].

Pada penelitian ini, RFID *reader* yang digunakan yaitu Electron HW-VX6330K *Middle Range UHF RFID reader*. RFID ini memiliki frekuensi 902-928 MHz, frekuensi ini dapat menangkap gelombang radio RFID *tag* dari jarak yang jauh walaupun terhalang objek dan dapat menangkap banyak

data RFID tag dalam satu waktu. Jarak deteksi tanpa halangan dari RFID reader adalah 6-8 meter. RFID ini memiliki 7 dBi Circular Antenna [9]. Spesifikasi lengkap dari RFID reader dapat dilihat pada Tabel II. Sedangkan RFID tag yang digunakan yaitu Alien9662 RFID label H3 chip FB00. RFID tag pasif yang berfrekuensi 860-960 MHz. RFID tag ini bekerja pada protokol EPC C1G2 (ISO18000-6C). RFID tag memiliki memori sebesar 512-bits. RFID tag yang digunakan berbentuk antenna dengan ukuran 17x70 mm. Spesifikasi dari RFID tag dapat dilihat pada Tabel III. RFID tag pasif digunakan karena harga yang lebih murah dan ukuran yang lebih kecil dan RFID tag ini dapat ditempel pada setiap aset/barang. RFID memiliki beberapa jenis, setiap jenis RFID memiliki spesifikasi yang berbeda yaitu RFID jenis LF (*Low Frequency*), HF (*High Frequency*), UHF, dan *Active* seperti Tabel I.



Gambar. 4. RFID reader



Gambar. 5. RFID tag

TABEL I
JENIS RFID

	LF	HF	UHF	Active
Frequency	125-134.2 KHz	13.56 MHz	850-960 MHz	100 KHz-2.45 GHz
Range	0.2-2m	Up to 1m	Up to 3m	Up to 100m
Memory	64 bits	2048 bits	96 bits	32 bits
Data Rate	Slow	Fast	Fast	Fast
Multiple Tags	Poor	Fast	Fast	Fast
Application	Industrial Application	Item Tracking	Box and Pallet Tracking	Asset Tagging Location System

TABEL II
SPESIFIKASI RFID READER

Parameter	Keterangan
Frekuensi Operasi	902-928 MHz
Dukungan Protokol	ISO18000-6B, ISO18000-6C (EPC C1G2)
Antena	7dBi Circular Antenna
Mode Kerja	Active mode, Trigger mode, and Answer mode
Daya Pancar	1-30 dBm (Adjustable)
Jarak Baca	6-8 meter

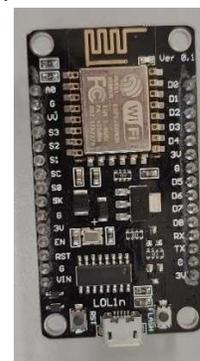
TABEL III
SPESIFIKASI RFID TAG

Parameter	Keterangan
Chip	Alien H3 Chip
Protokol	ISO18000-6C
Frekuensi	860-960 MHz
Memori	512 bits
Daya Pancar	1-30 dBm (Adjustable)
Ukuran	17 × 70mm

2) ESP8266

NodeMCU ESP8266 merupakan salah satu arduino compatible development board yang masih sejenis dengan ESP8266 dan dapat diprogram menggunakan software Arduino IDE. NodeMCU digunakan sebagai salah satu perangkat IoT (*Internet of Things*) dengan menggunakan koneksi dari *Wi-Fi* sehingga NodeMCU dapat mengirim atau bertukar informasi secara *wireless* [10]. Dari Gambar 6, NodeMCU memiliki GPIO (*General Purpose Input-Output*) yang digunakan untuk menentukan pin dari NodeMCU Board yang akan diinisialisasikan pada program Arduino IDE.

Pada penelitian ini, NodeMCU berfungsi sebagai perangkat utama yang digunakan untuk menerima data dari RFID reader dengan menggunakan komunikasi RS232 dan diprogram pada Arduino IDE dengan pemrograman *serial*. Untuk mengkoneksikan RFID reader dengan NodeMCU dibutuhkan satu perangkat keras yaitu RS232 Converter seperti Gambar 7. NodeMCU menerima data dari RFID reader kemudian disimpan pada memori dan diteruskan ke server. NodeMCU memiliki dua buah *Hardware Serial* dan beberapa *Software Serial*. Pada sistem ini menggunakan *Software Serial*, sehingga pin yang digunakan merupakan pin *digital* dari NodeMCU ESP8266.



Gambar. 6. NodeMCU ESP8266



Gambar. 7. RS232 Converter to TTL

3) Router

Router merupakan perangkat keras yang digunakan untuk memberikan jaringan internet ke beberapa perangkat keras lain seperti komputer, laptop, NodeMCU, dan lain-lain. Melalui jaringan internet dari router, perangkat-perangkat keras yang terhubung dengan jaringan internet yang sama dapat melakukan pengiriman data, pertukaran data, dan dapat mengakses perangkat lain dari jarak yang jauh namun masih dalam jangkauan dari router [11]. Pada Gambar 8, frekuensi dari router sebesar 2,4 GHz. Router ini memiliki 4 port yang terdiri dari port internet, LAN 1, LAN 2, dan LAN 3. Port tersebut dapat dihubungkan ke perangkat keras lainnya menggunakan kabel UTP (*Unshielded Twisted Pair*). Router ini juga dapat memberikan internet melalui wireless ke perangkat lainnya menggunakan hotspot. Dengan menggunakan router, sistem dapat menerapkan konsep dari IoT yang dapat melakukan pengiriman/pertukaran data melalui jaringan internet.



Gambar. 8. Router

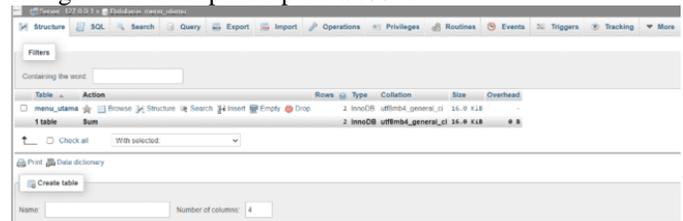
4) PHP

PHP atau *HyperText Preprocessor* merupakan bahasa pemrograman yang digunakan untuk membuat dan mengembangkan halaman website yang dinamis. Kode pada pemrograman PHP dapat bekerja dengan menyelipkan tag HTML (*HyperText Markup Language*), berkas tersebut dapat diberi ekstensi “.php” atau ekstensi yang telah ditetapkan oleh *Apache Web Server*. Dengan ekstensi tersebut, server dapat mengenali bahwa berkas tersebut mempunyai kode PHP. Terdapat beberapa perangkat lunak yang dapat digunakan untuk memprogram PHP yaitu Microsoft Visual Studio, Sublime Text, Notepad++, dan sejenisnya. PHP dapat digunakan pada OS (*Operating System*) Linux, Windows, dan MAC (*Media Access Control*). Untuk mengakses website yang telah diprogram menggunakan PHP dapat dibuka dengan menggunakan perangkat lunak tambahan yang memiliki *Apache* sebagai server [12].

5) MySQL (*My Structure Query Language*)

MySQL merupakan sebuah DBMS (*Database Management System*) yang berfungsi untuk menyimpan, menghapus, dan mengubah data yang ada pada database dengan menggunakan pemrograman SQL [13]. Database MySQL memiliki beberapa tabel yang digunakan untuk menyimpan berbagai data agar data tersebut tersusun dengan rapi dan terstruktur seperti Gambar 9. Untuk mengakses database MySQL dibutuhkan perangkat lunak bebas seperti XAMPP.

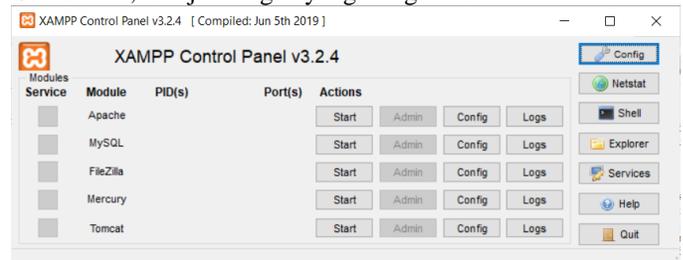
Pada penelitian ini, database MySQL digunakan untuk menyimpan data aset seperti nama aset, nomor inventaris, gambar aset, dan ID tag aset. Data tersebut diperoleh dari website kemudian disimpan pada database. Database juga digunakan untuk menyimpan data-data aset yang masuk ke ruangan dan ditampilkan pada website.



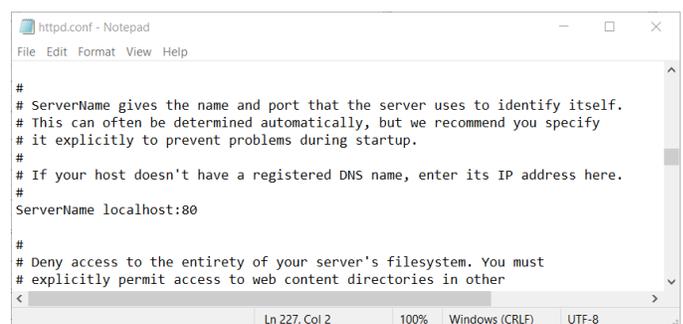
Gambar. 9. Tampilan MySQL

6) XAMPP

XAMPP merupakan perangkat lunak bebas yang digunakan sebagai server untuk menjalankan Apache, MySQL, PHPMyAdmin, Mercury, Tomcat, dan FileZilla sebagai localhost [13]. XAMPP secara default akan menjalankan HTTP pada port 80, HTTPS pada port 443, dan MySQL pada port 3306 seperti pada Gambar 10. User dapat mengubah HTTP port dengan cara mengubah pada *Config Apache* seperti Gambar 11, menjadi angka yang diinginkan.



Gambar. 10. Tampilan XAMPP

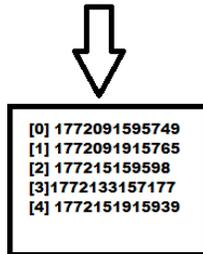


Gambar. 11. Config XAMPP

7) Parsing

Parsing merupakan suatu proses untuk memisah suatu masukan yang menghasilkan beberapa uraian yang digunakan untuk proses selanjutnya [14]. Pada penelitian ini, parsing digunakan untuk memisah beberapa data dari RFID tag pada saat pendeteksian banyak tag. Ketika terdapat 5 RFID tag yang terdeteksi, maka data semua RFID tag tersebut akan tersambung. Maka dilakukan parsing untuk memisah kelima data RFID tag. Parsing dilakukan pada pemrograman PHP dengan menggunakan fungsi explode. Fungsi ini akan memisah string dengan sebuah tanda pemisah yang ditelah ditentukan, kemudian data tersebut akan terpecah menjadi array seperti Gambar 12. Data yang telah menjadi array tersebut akan terkirim ke database MySQL.

!1772091595749!1772091915765!177215159598!1772133157177!1772151915939



Gambar. 12. Contoh Parsing Data

8) QoS (Quality of Service)

QoS merupakan metode pengukuran untuk mengetahui seberapa baik proses pengiriman data ke suatu tujuan dan untuk mendefinisikan karakteristik dan sifat suatu servis. QoS digunakan untuk mengukur sekumpulan atribut kinerja yang telah dispesifikasikan dan diasosiasikan dengan suatu servis. QoS menggunakan beberapa parameter yaitu packet loss, jitter, dan latency [15].

Packet loss merupakan parameter yang digunakan untuk menggambarkan kondisi yang menunjukkan jumlah paket atau data yang hilang [15]. Kondisi ini terjadi karena error pada website, adanya tabrakan data, dan lain sebagainya. Indeks dan kategori packet loss ditunjukkan pada Tabel IV. Persamaan perhitungan packet loss:

$$Packet\ loss = \frac{(\text{Paket dikirim} - \text{Paket diterima}) \times 100\%}{\text{Paket yang dikirim}} \quad (1)$$

TABEL IV
INDEKS KATEGORI PACKET LOSS

Kategori Degradasi	Packet Loss (%)	Indeks
Sangat Bagus	0-2	4
Bagus	3-14	3
Sedang	15-24	2
Jelek	>25	1

Latency merupakan waktu yang dibutuhkan data untuk menempuh jarak dari NodeMCU ke website. Latency dapat dipengaruhi oleh jarak, website, dan waktu proses yang lama [15]. Indeks dan kategori latency ditunjukkan pada Tabel V. Persamaan perhitungan rata-rata delay:

$$Average\ delay = \frac{Total\ delay}{Total\ packets\ receive} \quad (2)$$

TABEL V
INDEKS KATEGORI LATENCY

Kategori Degradasi	Delay (ms)	Indeks
Sangat Bagus	<150	4
Bagus	150-300	3
Sedang	300-400	2
Jelek	>450	1

Jitter merupakan variasi delay dan sangat berhubungan dengan latency, yang menunjukkan banyaknya variasi delay pada transmisi data. Jitter diakibatkan oleh variasi dalam panjang antrian data, waktu pengolahan data, dan waktu penghimpunan ulang data-data diakhir perjalanan [15]. Indeks dan kategori jitter ditunjukkan pada Tabel VI. Persamaan perhitungan jitter:

$$Jitter = \frac{Total\ delay}{Total\ packets\ receive} \quad (3)$$

$$Total\ Delay = Delay - Average\ delay \quad (4)$$

TABEL VI
INDEKS KATEGORI JITTER

Kategori Degradasi	Jitter (ms)	Indeks
Sangat Bagus	0	4
Bagus	1-75	3
Sedang	76-125	2
Jelek	126-225	1

B. Perancangan Perangkat Lunak

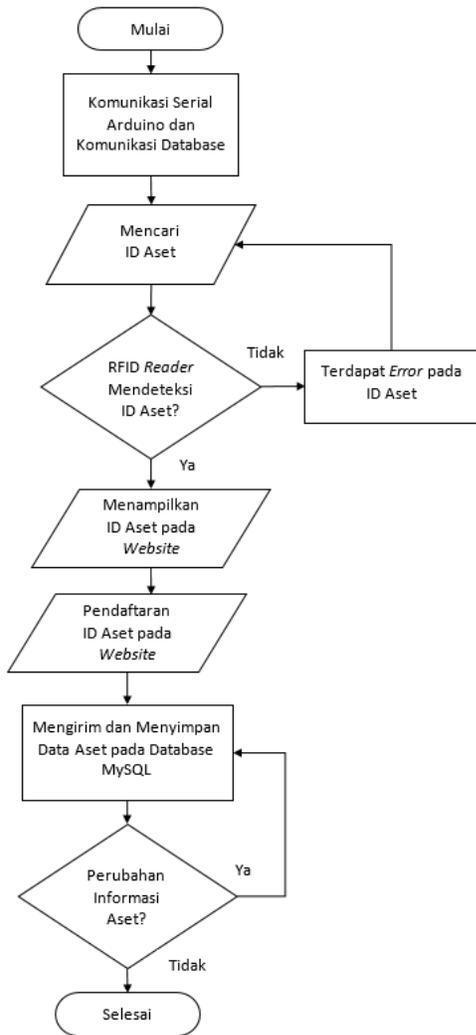
Perancangan perangkat lunak merupakan merancang perangkat yang berada di dalam sistem. Pada sistem ini, perancangan perangkat lunak terbagi menjadi dua yaitu perancangan website dan perancangan database. Perancangan website akan menjelaskan bagaimana membuat website hingga bagaimana cara menggunakan website. Perancangan database akan menjelaskan bagaimana pengaplikasian database MySQL pada sistem ini dan bagaimana menyimpan dan mengolah data-data aset pada database MySQL.

Perancangan Website

Sistem Pendeteksi Aset Berbasis IoT menggunakan website sebagai UI untuk menampilkan informasi lokasi aset, jam masuk aset, dan jumlah setiap aset dalam satu ruangan. Pembuatan website menggunakan pemrograman PHP, menggunakan Bootstrap sebagai framework, dan menggunakan HTTP sebagai protokol pengiriman data. Website memiliki dua level yaitu level admin dan level member, setiap level tersebut memiliki hak akses yang berbeda. Hak akses pada level admin yaitu dapat melakukan pendaftaran pada setiap aset, dapat melakukan perubahan informasi aset, dapat melakukan penghapusan aset, dapat melihat informasi lokasi dari aset, dan dapat melihat jumlah setiap aset di dalam ruangan.

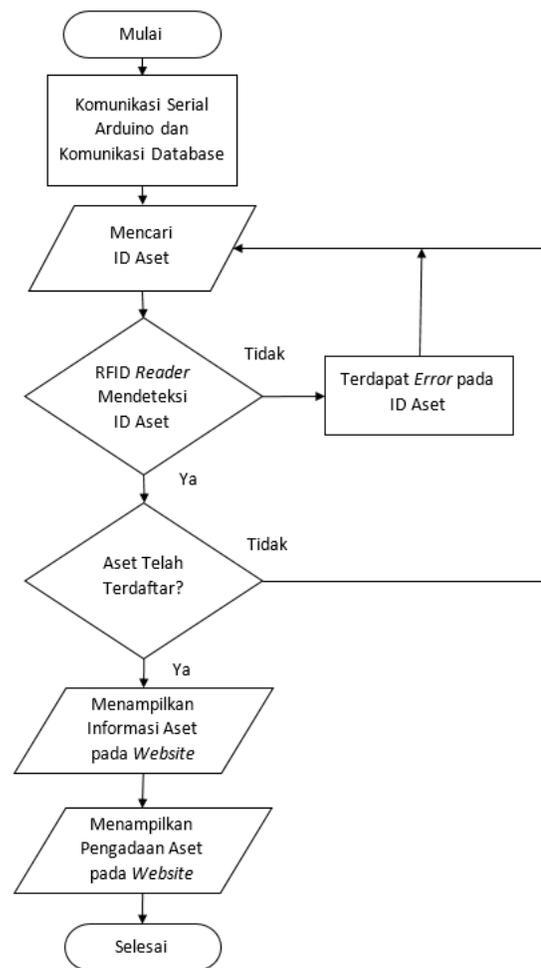
Proses saat melakukan pendaftaran aset yaitu melakukan koneksi NodeMCU dan koneksi server, kemudian RFID reader dalam kondisi stand by untuk mencari ID aset, jika RFID reader mendeteksi aset maka data tersebut akan terkirim

ke *server* dan ditampilkan pada *website*, jika *RFID reader* terdapat *error* pada saat melakukan pendeteksian maka dilakukan pendeteksian ulang, kemudian ID aset yang telah tampil pada *website* dapat didaftarkan pada *website*, kemudian data yang tersebut akan terkirim ke *database* MySQL dengan metode *POST* dan menyimpan pada *database* MySQL. Jika ingin melakukan perubahan informasi pada aset tersebut dapat dilakukan pada *website* dan akan diperbarui di *database* MySQL. *Flowchart* proses pendaftaran aset dapat dilihat pada Gambar 13.



Gambar 13. Perancangan Hak Akses Admin pada Website

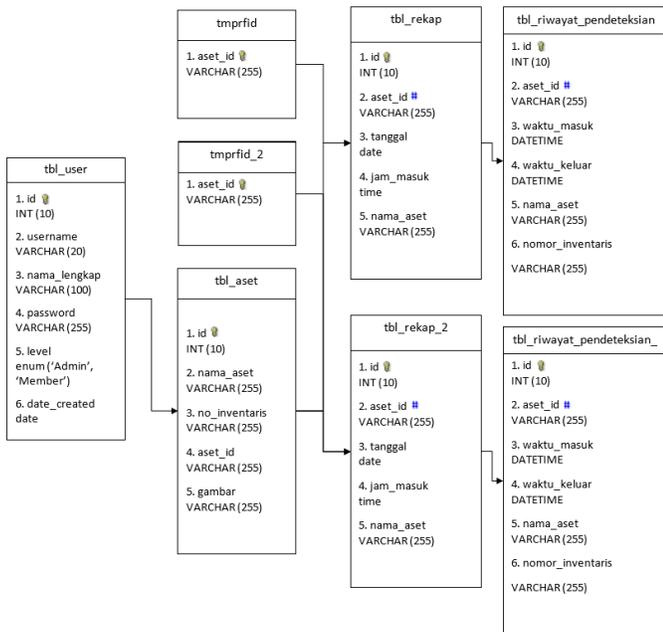
Proses untuk melihat informasi aset yaitu melakukan koneksi *NodeMCU* dan koneksi *server*, kemudian *RFID reader* dalam kondisi *stand by* untuk mencari ID aset, jika *RFID reader* mendeteksi aset maka data tersebut akan terkirim ke *server*, kemudian dilakukan *filter* aset yang telah terdaftar di *database* MySQL, jika data aset tersebut terdaftar maka *website* akan menampilkan informasi lokasi aset dan menampilkan jumlah setiap aset di dalam ruangan, jika data aset tersebut belum terdaftar maka *website* tidak akan menampilkan informasi aset tersebut dan jumlah setiap aset di dalam ruangan tidak akan bertambah. *Flowchart* proses untuk melihat informasi aset dapat dilihat pada Gambar 14.



Gambar 14. Perancangan Hak Akses User pada Website

C. Perancangan Database MySQL

Sistem Pendeteksi Aset Berbasis IoT menggunakan *database* untuk mempermudah dalam manajemen data dari aset dan mempermudah dalam mengelompokkan informasi lokasi aset. *Database* yang digunakan yaitu *MySQL* yang bersifat *localhost*, untuk mengakses *database* ini membutuhkan *server* *XAMPP*. Sistem ini memiliki delapan tabel *database* *tbl_user* berfungsi untuk menyimpan data dari setiap *user admin* dan *user member*. *Tbl_aset* berfungsi untuk menyimpan seluruh data aset. *Tbl_rekap* dan *tbl_rekap_2* berfungsi untuk merekap hasil pendeteksian di ruangan 1 dan ruangan 2. *Tbl_riwayat_pendeteksian* dan *tbl_riwayat_pendeteksian_2* sama seperti *tbl_rekap* dan *tbl_rekap_2* perbedaannya hanya pada waktu_masuk dan waktu_keluar. *Tmprfid* dan *tmprfid_2* berfungsi untuk menyimpan pendeteksian dari sensor. Kemudian dilakukan relasi antara *tbl_aset* dengan *tbl_rekap* dan *tbl_riwayat_pendeteksian* dengan tujuan agar saat terjadi pendeteksian kemudian direkap, lalu *tbl_rekap* akan mengambil data nama_aset di *tbl_aset* untuk menampilkan nama dari aset dan untuk mengelompokkan jumlah aset berdasarkan nama aset. Tabel-tabel dari *database* dapat dilihat pada Gambar 15.



Gambar. 15. Perancangan Database MySQL

III. HASIL DAN PEMBAHASAN

A. Pengujian RFID Reader

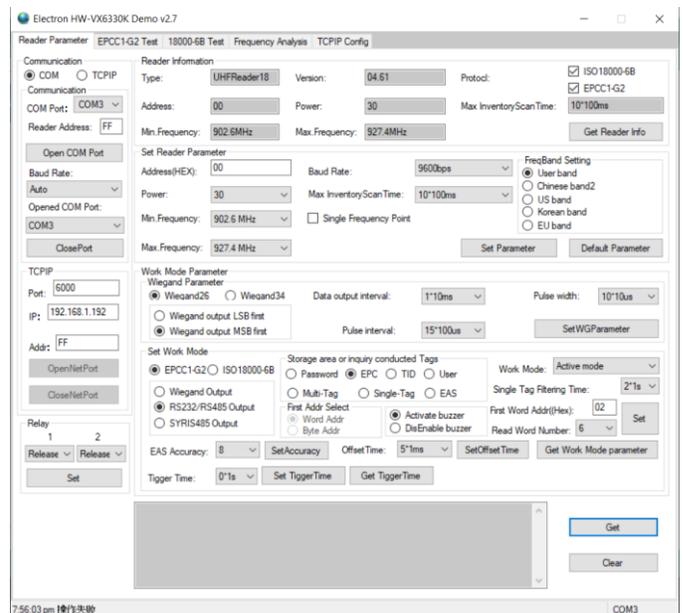
Pengujian ini dilakukan menggunakan dua RFID reader dengan jarak antar reader sejauh 300 cm, dapat dilihat pada Gambar 16. Pengujian ini dilakukan dengan mendeteksi setiap barang yang telah ditempel RFID tag seperti Gambar 17 sejauh 60, 120, 180, 240, dan 300 cm. Pengujian ini hanya dilakukan sampai 300 cm karena pengaplikasian RFID reader yang diletakkan di plafon depan pintu dan jarak antara plafon sampai ke lantai yaitu 280 cm. Pengujian dilakukan dengan melakukan setting pada setiap parameter reader seperti RF Output Power dapat disesuaikan menjadi 30 dBm, Baud Rate yang digunakan sebesar 9600bps, Set Work Mode RS232/RS485 Output, Work Mode Active, dan Single Tag Filtering Time selama 2,1 s. Setting RFID reader dapat dilihat pada Gambar 18.



Gambar. 16. Jarak dua RFID reader



Gambar. 17. RFID Tag yang digunakan untuk pengujian



Gambar 18. Setting RFID reader

Pengujian ini dilakukan dengan beberapa tahapan yaitu pendeteksian RFID reader tanpa penghalang, pendeteksian RFID reader dengan penghalang, dan hasil pembacaan RFID reader pada website. Pada pengujian dengan penghalang menggunakan dua objek yaitu kayu dengan tebal 18 mm seperti Gambar 20 dan besi dengan tebal 15 mm seperti Gambar 19. Pada pengujian pembacaan pada website dilakukan dengan memantau website ketika melakukan pendeteksian. Tampilan website yang digunakan seperti Gambar 21 dan tampilan dari database seperti Gambar 22.



Gambar. 19. Penghalang Besi



Gambar. 20. Penghalang Kayu

No	Nama Aset	Nomor Inventaris	Gambar Aset	Tanggal	Jam Aset Masuk
1	Komputer	RTF22.DELL5040.001		2022-07-09	12:37:38
2	Komputer	RTF22.DELL5040.001		2022-07-09	12:38:30

Gambar. 21. Tampilan Form Pendeteksian

id	aset_id	nama_aset	no_inventaris	waktu_masuk	waktu_keluar
1	1772091595749	Komputer	RTF22.DELL5040.001	2022-07-02 19:20:55	2022-07-02 19:20:55
2	1772091915785	Komputer	RTF22.DELL5040.002	2022-07-02 19:21:06	2022-07-02 19:21:09
3	177215159598	Komputer	RTF22.DELL5040.003	2022-07-02 19:21:40	2022-07-02 19:21:38
4	1772151915939	Komputer	RTF22.DELL5040.005	2022-07-02 19:32:37	2022-07-02 19:32:37
5	1772133157177	Komputer	RTF22.DELL5040.004	2022-07-02 19:22:10	2022-07-02 19:22:10
6	177247636173243151	Kursi	RTF22.KURSI.001	2022-07-02 19:28:28	2022-07-02 19:28:31
7	1772199559	Meja	RTF22.MEJA.001	2022-07-02 19:25:36	2022-07-02 19:25:28
8	177223	Meja	RTF22.MEJA.003	2022-07-02 19:16:09	2022-07-02 19:16:03
9	1772136357193	Komputer	RTF22.DELL5040.006	2022-07-02 19:36:05	2022-07-02 19:35:04
10	1772193159124	Komputer	RTF22.DELL5040.007	2022-07-02 19:32:30	2022-07-02 19:32:30
11	17721522359176	Komputer	RTF22.DELL5040.008	2022-07-02 19:22:36	2022-07-02 19:22:36
12	177219635983	Komputer	RTF22.DELL5040.009	2022-07-02 19:23:44	2022-07-02 19:23:44
13	17721525559148	Komputer	RTF22.DELL5040.010	2022-07-02 19:23:44	2022-07-02 19:23:44
14	17723715959177	Kursi	RTF22.KURSI.010	2022-07-02 19:30:48	2022-07-02 19:29:50
15	17723719159193	Kursi	RTF22.KURSI.009	2022-07-02 19:28:42	2022-07-02 19:28:40
16	17724319161	Kursi	RTF22.KURSI.007	2022-07-02 19:30:46	2022-07-02 19:30:44

Gambar. 22. Tampilan Database Hasil Pendeteksian

Hasil pengujian pendeteksian RFID reader tanpa penghalang dengan melakukan pendeteksian sebanyak 30 RFID tag pada jarak 60, 120, 180, 240, dan 300 cm berhasil dengan persentase keberhasilan sebesar 100% dapat dilihat pada Tabel VII. Dari pengujian ini dapat diketahui bahwa RFID yang digunakan mampu membaca data dari RFID tag melalui gelombang frekuensi sejauh 300 cm.

TABEL VII
HASIL PENGUJIAN RFID TANPA PENGHALANG

Jumlah RFID tag yang akan diuji	Jarak baca RFID reader dengan RFID tag (cm)	Hasil Pendeteksian	
		Berhasil	Gagal
30	60	30	0
	120	30	0
	180	30	0
	240	30	0
	300	30	0

Hasil pengujian pendeteksian RFID reader dengan penghalang dilakukan sebanyak 30 data disetiap penghalang dan setiap jarak. Dari hasil tersebut didapatkan persentase pendeteksian RFID reader dengan penghalang kayu sebesar 100% dan penghalang besi sebesar 92%. RFID reader dapat mendeteksi RFID tag dengan penghalang non logam dengan baik. Namun dengan penghalang besi RFID reader tidak dapat mendeteksi dengan sempurna karena frekuensi radio dipantulkan oleh benda logam. Hasil pendeteksian dapat dilihat pada Tabel VIII. Dari pengujian ini dapat dilihat bahwa RFID reader mampu membaca data RFID tag yang terhalang benda logam dengan baik, namun kurang baik ketika terhalang oleh objek logam. RFID reader kurang baik dalam mendeteksi benda logam dikarenakan frekuensi radio akan terganggu untuk membaca RFID tag yang terhalang oleh logam.

TABEL VIII
HASIL PENGUJIAN RFID DENGAN PENGHALANG

Jumlah RFID tag yang akan diuji	Penghalang	Jarak baca RFID reader dengan RFID tag (cm)	Hasil Pendeteksian	
			Berhasil	Gagal
30	Kayu 18mm	60	30	0
		120	30	0
		180	30	0
		240	30	0
		300	30	0
	Besi 15mm	60	28	2
		120	28	2
		180	29	1
		240	27	3
		300	26	4

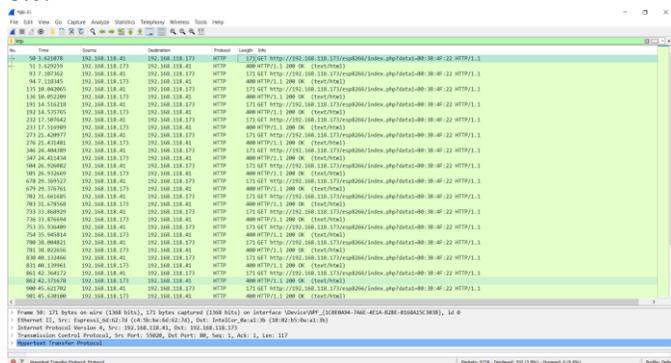
Pengujian pembacaan pada website dilakukan dengan membandingkan tampilan kode unik pada aplikasi Gambar 18 dengan tampilan website dan database seperti pada Gambar 21 dan Gambar 22. Dari hasil pengujian pada Tabel IX, bahwa kode unik dari RFID tag sesuai dengan kode unik yang tampil pada website. Sehingga dari pengujian ini memastikan bahwa kode unik dari RFID tag tidak berubah-ubah.

TABEL IX
HASIL PENDETEKSIAN KODE RFID TAG PADA WEBSITE

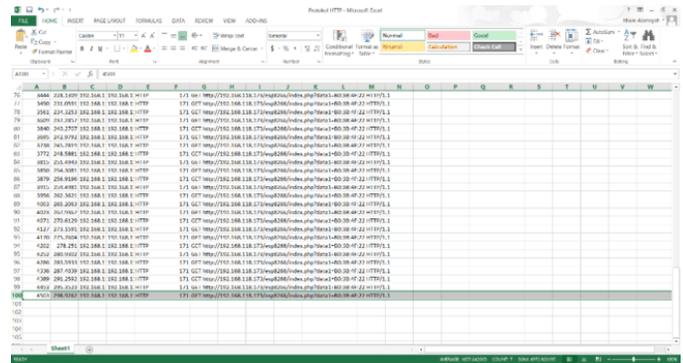
No	Kode unik yang akan dipindai	Jarak baca RFID reader dengan RFID tag	Kode unik yang tampil pada website	Keterangan
1	1772091595749	60	1772091595749	Sesuai
2	1772091915765	120	1772091915765	Sesuai
3	177215159598	180	177215159598	Sesuai
4	1772133157177	240	1772133157177	Sesuai
5	1772151915939	300	1772151915939	Sesuai
6	1772136357193	60	1772136357193	Sesuai
7	1772193159124	120	1772193159124	Sesuai
8	17721522359176	180	17721522359176	Sesuai
9	177219635983	240	177219635983	Sesuai
10	17721525559148	300	17721525559148	Sesuai
11	1772199559	60	1772199559	Sesuai
12	1772379559185	120	1772379559185	Sesuai
13	177223	180	177223	Sesuai
14	1772417914216	240	1772417914216	Sesuai
15	17723712759201	300	17723712759201	Sesuai
16	6119119120	60	6119119120	Sesuai
17	17721378114123	120	17721378114123	Sesuai
18	1772212555973	180	1772212555973	Sesuai
19	1772212235957	240	1772212235957	Sesuai
20	17721912759	300	17721912759	Sesuai
21	177247636173243	60	177247636173243	Sesuai
22	151	120	151	Sesuai
23	177247956149113	180	177247956149113	Sesuai
24	109	240	109	Sesuai
25	1772432236190	300	1772432236190	Sesuai
26	177241636112	60	177241636112	Sesuai
27	1772417918167	120	1772417918167	Sesuai
28	177241316135	180	177241316135	Sesuai
29	17724319161	240	17724319161	Sesuai
30	1772431596185	300	1772431596185	Sesuai
31	17723719159193	60	17723719159193	Sesuai
32	17723715959177	120	17723715959177	Sesuai

D. Pengujian QoS

Pengujian ini dilakukan dengan menggunakan topologi star pada protokol HTTP. Pengujian ini dilakukan dengan mendeteksi tag sebanyak 100 data selama 5 menit. Data tag tersebut terdeteksi secara *realtime* dengan rentang waktu sekitar 2 detik. Data yang terbaca oleh wireshark sebanyak 100 data dapat dilihat pada Gambar 23 dan Gambar 24. Data tersebut di *export* ke excel untuk melihat data yang lebih jelas seperti, data yang dikirim dan diterima sesuai artinya tidak ada paket data yang hilang sehingga persentase *packet loss* sebesar 0%.



Gambar 23. Tampilan data pada wireshark



Gambar 24. Tampilan data pada Excel

Rata-rata *delay delay* didapat dari rumus penjumlahan *total delay* dibagi banyaknya paket data yang masuk seperti (2). Kemudian masukkan data tersebut ke rumus dan didapatkan hasil *total delay* sebesar 363,44076 dibagi dengan banyaknya jumlah paket data yang masuk yaitu sebanyak 5228. Didapat hasil rata-rata *delay* sebesar 0,0695181 s. Data tersebut diubah ke millisecond menjadi 69,5181 ms. Nilai tersebut memastikan bahwa *delay* pengiriman data sensor ke *server* sangat cepat.

$$Average\ delay = \frac{363,44076}{5228} \tag{5}$$

$$Average\ delay = 0,0695181\ s$$

$$Average\ delay = 69,5181\ ms$$

Rata-rata *Jitter* didapat dari rumus (3) dan (4). Kemudian data hasil analisa QoS dimasukkan maka didapat *total jitter* sebesar 363,646633 dibagi dengan banyaknya jumlah paket data yang masuk yaitu 5228. Didapat hasil rata-rata *jitter* sebesar 0,0695575 s. Data tersebut diubah ke millisecond menjadi 69,5575 ms. Dari nilai tersebut, artinya variasi ping dari waktu ke waktu cukup rendah sehingga setiap data yang masuk ke *server* didapat dengan cepat.

$$Jitter = \frac{363,646633}{5228} \tag{6}$$

$$Jitter = 0,0695575\ s$$

$$Jitter = 69,5575\ ms$$

IV. KESIMPULAN

Berdasarkan hasil dari pengujian yang telah dilakukan, didapatkan beberapa poin kesimpulan yaitu sistem mampu menerapkan *database* MySQL secara *realtime* dengan cara NodeMCU ESP8266 mengirimkan data ke *server* dengan menggunakan metode GET kemudian *server* akan mengolah data tersebut dan terkirim ke *database* MySQL secara *realtime*. Data yang diuji sebanyak 30 RFID tag sejauh 300cm tanpa menggunakan penghalang dengan tingkat keakurasian sebesar 100%. Pengujian dengan penghalang besi dengan jarak sejauh 300cm dengan tingkat keakurasian sebesar 92% dan dengan penghalang kayu dengan tingkat keakurasian

sebesar 100%. Pengujian tersebut mampu menampilkan hasil pendeteksian pada *website* dengan tingkat keberhasilan sebesar 100%.

Website mampu menampilkan informasi aset yang terdeteksi RFID *reader* dengan cara mengambil data dari hasil pendeteksian RFID *reader* secara *realtime* kemudian melakukan relasi tabel dengan tabel daftar aset yang bertujuan ketika RFID *tag* tersebut terdeteksi dan telah terdaftar pada tabel daftar aset, maka *website* akan mengambil beberapa data dari tabel daftar aset seperti nama aset, nomor inventaris aset sebagai informasi aset. *Website* juga menampilkan waktu masuk aset dan waktu keluar aset. Sistem akan mengetahui aset masuk ruangan apabila terdeteksi sekali, dan mengetahui aset keluar ruangan apabila terdeteksi sebanyak dua kali.

Analisa QoS dilakukan dengan menggunakan tiga parameter yaitu *packet loss*, *latency*, dan *jitter*. Analisa pada parameter *packet loss* dilakukan dengan hasil 0% dan kategori sangat bagus, artinya saat pengiriman data dan penerimaan data tidak ada data yang hilang. Analisa pada parameter *latency* dilakukan dengan menggunakan rumus seperti (2) didapat hasil rata-rata delay sebesar 69,5181 ms, artinya waktu yang dibutuhkan paket data pada saat pengiriman dan penerimaan data masuk dalam kategori sangat bagus. Analisa pada parameter *jitter* dilakukan dengan menggunakan rumus (3) didapat hasil rata-rata jitter sebesar 69,5575 ms, artinya banyaknya variasi *delay* pada saat transmisi data masuk dalam kategori bagus.

REFERENSI

- [1] A. Q. Gbadamosi *et al.*, "IoT for predictive assets monitoring and maintenance: An implementation strategy for the UK rail industry," *Autom Constr*, vol. 122, p. 103486, Feb. 2021, doi: 10.1016/J.AUTCON.2020.103486.
- [2] E. Calixto, "Asset Management," *Gas and Oil Reliability Engineering*, pp. 703–767, 2016, doi: 10.1016/B978-0-12-805427-7.00008-7.
- [3] K. P. Aji, U. Darusalam, and N. D. Nathasia, "Perancangan Sistem Presensi Untuk Pegawai Dengan RFID Berbasis IoT Menggunakan NodeMCU ESP8266," *JOINTECS (Journal of Information Technology and Computer Science)*, vol. 5, no. 1, p. 25, 2020, doi: 10.31328/jointecs.v5i1.1222.
- [4] D. A. Detasari, A. Y. Ridwan, and B. Santosa, "Perancangan Sistem Real Time Monitoring Stock Berbasis Teknologi Rfid (Studi Kasus Di Pt Tmmin Plant Sunter 1) Design," vol. 7, no. 2, pp. 6165–6172, 2020.
- [5] H. Setiadi, Y. Priyandari, and S. I. Cahyono, "Implementation of Parking System Based on Radio Frequency Identification (RFID) at the Faculty of Engineering Sebelas," *ITSMART: Jurnal Ilmiah Teknologi dan Informasi*, vol. 6, no. 1, pp. 39–44, 2017.
- [6] I. Nurhaida and I. Ichsan, "Congestion Control Pada Jaringan Komputer Berbasis Multi-Protocol Label Switching (Mpls)," *Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, vol. 11, no. 1, pp. 77–88, 2020, doi: 10.24176/simet.v11i1.3671.
- [7] J. E. Elektro *et al.*, "Darwin: Rancang Bangun Sistem Peminjaman dan Manajemen ... 80," vol. 05, no. 2, pp. 80–90, 2021.
- [8] W. T. Kusuma, Y. Yulia, and R. Lim, "Penggunaan Ultra High Frequency RFID pada Sistem Inventarisasi Creatrix Organizer," *Jurnal Infra*, vol. 8, no. 2, pp. 149–154, 2020.
- [9] N. B. Webber, "General Description," *Fluid Mechanics for Civil Engineers*, pp. 20–20, 2020, doi: 10.1201/9781315273426-11.
- [10] P. H. Rantellinggi, F. F. Paiki, and Y. Gadi, "Telematika Pemantau Suhu Menggunakan NodeMcu , IoT Dan Cayenne Pada Rack Server," vol. 13, no. 2, pp. 80–90, 2020.
- [11] F. Ramadhan, I. Ardiansah, and R. Kastaman, "Perancangan Purwarupa Alat Penyiraman Otomatis pada Tanaman Pisang dengan Internet of Things (IoT)," *Jurnal Rekayasa Sistem Industri*, vol. 8, no. 2, pp. 75–80, 2019, doi: 10.26593/jrsi.v8i2.3224.75-80.
- [12] R. Y. Endra, Y. Aprilinda, Y. Y. Dharmawan, and W. Ramadhan, "Analisis Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website," *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, vol. 11, no. 1, p. 48, 2021, doi: 10.36448/expert.v11i1.2012.
- [13] H. Riyadli, A. Arliyana, and F. E. Saputra, "Rancang Bangun Sistem Informasi Keuangan Berbasis WEB," *Jurnal Sains Komputer dan Teknologi Informasi*, vol. 3, no. 1, pp. 98–103, 2020, doi: 10.33084/jsakti.v3i1.1770.
- [14] R. S. Poliyama, F. E. P. Surusa, and ..., "Rancang Bangun Alat Sistem Monitor Lampu Jalan Umum Tenaga Surya Berbasis Teknologi Lo-Ra," *Jambura Journal of ...*, vol. 3, pp. 34–40, 2021.
- [15] A. R. Maulana, H. Walidainy, M. Irhamsyah, F. Fathurrahman, and A. Bintang, "Analisis Quality of Service (Qos) Jaringan Internet Pada Website E-Learning Universitas Syiah Kuala Berbasis Wireshark," *Jurnal Komputer, Informasi Teknologi, dan Elektro*, vol. 6, no. 2, pp. 27–30, 2021, doi: 10.24815/kitektro.v6i2.22284.