Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN: 2548-964X
Vol. 2, No. 10, Oktober 2018, hlm. 4156-4163 http://j-ptiik.ub.ac.id

Analisis Komunikasi Antar IoT Middleware Dengan Node Sensor Kamera Berbasis Raspberry Pi Zero

Yazid[1], Eko Sakti Pramukantoro[2], Ari Kusyanti[3]

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: [1]yzdsamanhudi@student.ub.ac.id, [2]ekosakti@ub.ac.id, [3]ari.kusyanti@ub.ac.id

Abstrak

Pengembangan middleware yang telah dilakukan pada penelitian sebelumnya yang mengacu pada permasalahan interoperabilitas. Permasalahan interoperabilitas yang diangkat yakni mengacu pada perkembangan *middleware* agar dapat menghubungan perangkat yang menggunakan protokol yang berbeda yaitu protokol CoAP, MQTT dan Websocket. Pengujian pada penelitian tersebut menggunakan DHT22 dan nodemcu sebagai sensor. Hasilnya node sensor dan IoT tersebut dapat berkomunikasi dengan baik dengan parameter delay, packet loss dan overhead dari masing masing protokol yang digunakan. Middleware IoT yang dikembangakan sebelumnya selain berangkat dari dari ide syntactical interoperability, juga dikembangan dengan tujuan semantic interoperability, dimana simantik interoperability adalah proses pengiriman data yang mengacu pada konten dan konteks data yang dikirim. Akan tetapi penelitian sebelumnya pengujian middleware terbatas pada syntactical interoperability, dalam penelitian ini dilakukan pengujian semantic, dengan struktur data yang dikirim berbeda dari biasanya dan data yang dikirim berupa gamba, animasi gambar bergerak dan video. Berdasarkan hasil pengujian, middleware mampu menerima data tersebut, dengan hasil Pada saat pengiriman gambar dengan resolusi 1360x768. Delay yang diperoleh paling rendah terdapat pada QoS 2 dan paling tinggi pada QoS 0.002993 detik. Pada saat pengiriman gambar dengan resolusi 1024x768. Delay yang diperoleh paling rendah terdapat pada QoS 2 dan paling tinggi pada QoS 0.003959 detik. Pada saat pengiriman gambar dengan resolusi 800x600. Delay yang diperoleh paling rendah terdapat pada QoS 2 dan paling tinggi pada QoS 0.006523 detik.

Kata kunci: *Internet Of Things, Middleware, MQTT, PiCamera.*

Abstract

Development of middleware that has been done in previous research which refers to interoperability problem. Interoperability issues are raised that refers to the development of middleware in order to connect devices that use different protocols namely CoAP, MQTT and Websocket protocol. Tests on the previous studies used DHT22 and nodemcu as sensors. The result was that node sensor and IoT can communicate well with delay parameters, packet loss and overhead of each protocol used. The previously developed IoT Middleware aside from the idea of syntactical interoperability, is also developed with the aim of semantic interoperability, where semantic interoperability is the process of sending data that refers to the content and context of data sent. However, previous studies of middleware testing were limited to syntactical interoperability, where in this study semantic testing will be done, with data structures sent differently than usual and data sent in the form of images, animated moving images and video. Based on the test results, the middleware is able to receive the data, with the results. At the time of transmitting images with resolution 1360x768. The lowest obtained delay is in QoS 2 and the highest in QoS 0.002993 seconds. When transmitting images with 1024x768 resolution. The lowest obtained delay is in QoS 2 and the highest in QoS 0.003959 seconds. When transmitting images with 800x600 resolution. The lowest received delay is in QoS 2 and highest on QoS 0.006523 seconds.

Keywords: *Internet Of Things, Middleware, MQTT, PiCamera.*

1. PENDAHULUAN

Perkembangan *Internet of Things* (IoT) dimana merupakan kombinasi antara *ubiquitos computing* dan internet. *Internet of things* (*IoT*)

yang melibatkan berbagai perangkat untuk berinteraksi antar perangkat, seperti akuator, agregator, sensor dan berbagai macam aplikasi ataupun domain. Internet yang merupakan kombinasi antara infratstuktur jaringan yang berkembang dinamis dan skala masif dengan standar dan protokol komunikasi yang mendukung masalah interoperabilitas. Sedangakan things adalah benda atau perangkat baik virtual ataupun konkret yang memiliki atribut, karakteristik identitas yang mampu berkomunikasi antara satu dengan yang lainya melalui sebuah *interface*. Interoperabilitas adalah salah satu yang menjadi permasalahan dalam *internet of things (IoT)*, Berdasarkan penelitian yang telah dilakukan oleh P. Desi (2015) dan IERC (2015), yang telah membagi menjadi tiga bagian dalalm interoperabilitas yaitu *Semantic Interoperability*, *Technical / Network Interoperability* , dan *Syntactical Interoperability*. *Semantic Interoperability* mengacu pada konteks data dan konten data yang dikirim. Technical / Network Interoperability mengacu pada protokol jaringan yang digunakan perangkat untuk saling terhubung antara satu dengan yang lainya dengan meliputi *low power networking protocol* seperti Bluetooth, ZigBee, RFID, dan Zwave, *traditional networking protocols* seperti IEEE 802.11 dan Ethernet. Sedangkan Syntactical Interoperability mengacu pada komunikasi model data yang dikirim dan protokol yang digunakan sebagai pertukaran pesan yang digunakan seperti MQTT, XMPP, CoAP, HTTP dan sebagainya.

Pengembangan *middleware* yang berbasis *event – based* sebagai *gateway* antar antar aplikasi dengan mengguankan protokol *MQTT*, *CoAP* dan *Websocket* dengan judul "Pengembangan *IoT middleware* berbasis event – based dengan protokol komunikasi *CoAP*, *MQTT* dan *Websocket*". Pengembangan middleware diyakini untuk mengatasi masalah interoperabilitas yang ada, seperti menghubungkan antar perangkat yang menggunakan protokol berbeda, seperti *CoAP*, *MQTT* dan *websocket*. *Middleware* tersebut juga sebagai gateway multi-protokol (Husnul, 2017). Pengujian pada penelitian tersebut menggunakan DHT22 dan nodemcu sebagai sensor. Hasilnya node sensor dan IoT tersebut dapat berkomunikasi dengan baik dengan parameter delay, packet loss dan overhead dari masing masing protokol yang digunakan.

Middleware Internet of Things (IoT) yang dikembangakan sebelumnya selain berangkat dari ide *syntactical interoperability*, juga dikembangan dengan tujuan *semantic interoperability*, dimana *semantic interoperability* adalah proses pengiriman data yang mengacu pada konten dan konteks data yang dikirim. Akan tetapi penelitian sebelumnya pengujian *middleware* terbatas pada *syntactical interoperability*, yaitu pertukaran informasi menggunakan protokol yang berbeda, oleh karena itu dalam penelitian ini akan dilakukan pengujian *semantic interoperability* dengan node sensor berbasis kamera. Node sensor Kamera dipilih karena konten yang dihasilkan berbeda dari sensor lainya, dengan data yang dihasilkan dapat beragam dari segi ukuran dan jenis. Dengan demikian Penulis ingin menguji pengiriman hasil gambar berformatkan jpeg, gambar bergerak (animasi) dengan format gif dan video yang didapat dari sensor module kamera raspberry dengan menggunakan protokol MQTT, penulis akan melakukan penerapan publish untuk pengiriman hasil modul kamera menuju *middleware* yang telah dikembangkan.

2. LANDASAN KEPUSTAKAAN

2.1. Kajian pustaka

Pada penelitian sebelumnya, pengembangan middleware yang dilakukan untuk mengatasi perkembangan *Internet of Things (IoT),* komunikasi antar perangkat hanya pada satu domain. Pada penelitian tersebut maka dikembangkanya middleware dengan multi -protokol untuk mengatasi syntactical interopability dengan tiga protokol yaitu Websocket, CoAP dan MQTT. Hasil dari penelitian ini, dengan menggunakan ketiga protokol tersebut middleware yang dikembangkan dapat memenuhi syarat interoperabilitas. (Anwari, et al., 2017)

Pada penelitian yang kedua, dengan judul "Analisis Performansi dan Scalabilitas pada even-based IoT Middleware. Pada penelitian ini dilakukan pengujian dalam aspek performasnsi dan skalabilitas, dengan hasil bahwa kemampuan *middleware* untuk menangangi *publish* atau *subscibe* dalam satu detik dengan jumlah klien 100 hingga 1000 bergerak naik seiring bertambahnya jumlah klien, ketika jumlah klien bertambah *concurrent middleware* juga bertambah. (Rozi,

et al., 2017)

2.2. Minikomputer

Mikrokomputer merupakan komputer kecil yang memiliki satu atau lebih chip (*British Dictionary*, 2012). Salah satu contoh dari mikrokomputer adalah *Raspberry pi*. *Raspberry pi* merupakan mini komputer dengan ukuran 85.60 mm x 83,98 x 17 mm dan berat 45 g. Papan *Raspbery pi* memiliki sebuah prosesor dengan grafik chip memori RAM dengan berbagai perangkat interface dan konektor yang mendukung perangkat eksternal. Minikomputer yang digunakan pada penelitian ini adalah Raspberry Pi dan Pi Zero.

2.3. PiCamera Raspberry Pi

Ada beberapa modul atau *device* tambahan yang dikeluarkan oleh *raspberry pi foundation,* yang dapat digunakan untuk mengambil video maupun foto dengan resolusi yang tinggi, modul memiliki kamera lima megapiksel yang mendukung 1080p30, 720p60 dan video dengan mode VGA90. Modul ini menempel dengan kabel pita sepanjang 15 cm dan dihubungkan dengan port CSI yang terdapat pada *raspberry pi.*

3. METODOLOGI PENELITIAN

Dalam penelitian ini kita menggunakan alatnya secara langsung (*middlleware*) dan menyusun sebagaimana fungsinya. Hal pertama yang dilakukan adalah konfigurasi kamera modul raspberry pi pada raspberry pi sebagai media publisher dan menghubungkan *publisher* agar bisa terhubung dengan *middleware* yang telah terpasang. Fungsi dari kamera yang terpasang pada media *Publisher* adalah untuk mengahisilkan gambar yang nantinya akan dikirim ke *middleware* sebagai pengujian apakah *middleware* tersebut mampu menerima file berupa gambar. Dalam pengiriman data gamabar ini, sistem pada *publisher* mengguankan protokol MQTT.

3.1. Pengujian Sistem

Sebagai pembuktian Pengujian sistem dilakukan untuk mendapatkan atau menunjukan bahwa *middleware* yang telah dikembangkan dapat digunakan sesuai kebutuhan dan target yang diharapakan. Pengujian ini diharapkan dapat mendapatkan data berupa nilai yang nantinya dapat diukur sebagai tujuan dari penelitian ini. Parameter yang akan diukur adalah tentang performansi pengiriman data gambar menuju middleware dengan protokol mqtt, pegujian ini bertujuan untuk mengetahui kinerja *middleware* dalam menerima berbagai macam file, terutama pada file gamabar (.jpeg), animasi (.gif) dan video yang dihasilkan oleh kamera raspberry pi. Untuk mengetahui performansi dari middleware ini peneliti meihat hasil dari masing-masing QoS *publisher*, dari QoS0, QoS1 dan QoS2 pada sistem yang telah dibuat.

3.2. Skenario 1 Pengiriman Gambar Dengan Resolusi 1360x768

Pengambilan gambar dengan resolusi 1360x768 yang didapatkan dari hasil sensor kamera dari sisi *Publisher,* akan diteruskan menuju *middleware*, pengiriman dilakukan secara terus menerus sebanyak 10 kali dengan jeda waktu 10 detik. Pengujian yang diinginkan adalah melihat performa dari masing-masing QoS dari *publisher*. Data yang diingikan tersebut nantinya berupa nilai dari *delay, jitter* dan *throughput.*

3.3. Skenario 2 Pengiriman Gambar Dengan Resolusi 1024x768

Pengambilan gambar dengan resolusi 1024x768 yang didapatkan dari hasil sensor kamera dari sisi *Publisher,* akan diteruskan menuju *middleware*, pengiriman dilakukan secara terus menerus sebanyak 10 kali dengan jeda waktu 10 detik. Pengujian yang diinginkan adalah melihat performa dari masing-masing QoS dari *publisher*. Data yang diingikan tersebut nantinya berupa nilai dari *delay, jitter* dan *throughput*.

3.4 Skenario 3 Pengiriman Gambar Dengan Resolusi 700x600

Pengambilan gambar dengan resolusi 1360x768 yang didapatkan dari hasil sensor kamera dari sisi *Publisher,* akan diteruskan menuju *middleware*, pengiriman dilakukan secara terus menerus sebanyak 10 kali dengan jeda waktu 10 detik. Pengujian yang diinginkan adalah melihat performa dari masing-masing QoS dari *publisher*. Data yang diingikan tersebut nantinya berupa nilai dari *delay, jitter* dan *throughput*.

3.5 Skenario Pengiriman Animasi

Publisher mengambil gambar dan sebanyak 10 kali kemudian diconvert menjadi animasi, kemudian dikirim ke *middleware*. Besar file animasi ini rata-rata 9,8 MiB atau setara dengan (10,171,187.2 byte). Kemudian pengambilan data performa QoS dengan parameter *delay, throughput* dan *jitter*.

3.6 Skenario Pengiriman Video

Publisher mengambil video berdurasi 60 detik, dan kemudian dikirim ke *middleware*, rata-rata ukuran file gambar berikut sebesar 8,6 MiB setara dengan (9,017,753.6 byte). Pengambilan data yang diharapkan adalah Mengetahui perfoma QoS yang dihasilakan dengan parameter *delay, jitter* dan *throughput*.

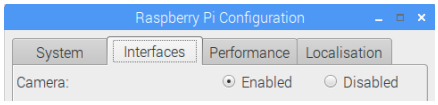
4. PERANCANGAN LINGKUNGAN UJI

4.1 Penyusunan Alat

Penyusunan node sensor adalah menghubungkan tiap komponen yang telah didefinisikan sebelumnya. Sepertipada gambar 4.1, Secara urut peneliti menghubungkan kabel fleksi pada sensor kamera dan dihubungkan pada soket kamera pi zero. Penelitian ini menggunakan kamera raspberry versi 1.3 dengan kabel fleksi HBV-Raspberry -10FPC. Agar modul sensor kamera pada pi zero bisa digunakan,pastikan pada konfigurasi raspberry modul sensor kamera telah diaktifkan.

Gambar 4.1 Pemasangan sensor kamera pada pi zero

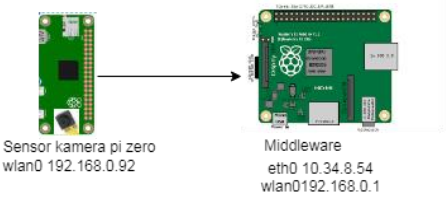
Untuk mengatifkan fungsi kamera pada pi-zero dapat dilakukan pada konfigurasi *interface* raspberry. *Enable* kamera raspberry, agar sensor bisa terbaca. Seperti pada gambar 4.2.



Gambar 4.2 konfigurasi kamera raspberry

4.2 Topologi Jaringan

Topologi jaringan ini digunakan sebagai gambaran umum penelitian yang dilakukan, dengan menghubungkan kedua perangkat yang digunakan, dari sisi kamera sebagai *publisher* dan *middleware* sebagai broker atau penyimpanan sementara, sebelum data dikirim menuju data senter. Dapat dilihat Pada gambar 4.3 yang menjelaskan bagaimana sensor dapat terhubung dan mengirim data ke *middleware.*



Gambar 4.3 Topologi Jaringan Sensor

Pada penelitian ini *middleware* dipasang pada Raspberry Pi 3 model B, selain menjadi broker, middleware juga sebagai *access point* untuk memberikan koneksi pada sisi *publisher*. Raspberry pi 3 dengan menggunakan microSD 8GB dan menggunakan sistem operasi raspibian jessie, memori 8GB tersebut juga berfungsi sebagai media penyimpanan data sekaligus menjadi penyimpanan perangkat lunak yang dibutuhkan peneliti. Komunikasi pada penelitian ini menggunakan *network interface* wlan0 yang menghubungkan antara *middleware* dengan sensor, dengan IP static yang telah ditentukan oleh peneliti 192.168.0.1. Ada tiga port yang terbuka ketika middleware tersebut kita jalankan, port 3000 untuk Websocket, port 5683 untuk port CoAP, dan port yang digunakan untuk MQTT adalah 1883. Namun pada penelitian ini hanya menggunakan protokol MQTT, dimana port yang dibutuhkan hanya 1883. Raspi zero yang menjadi publisher hasil sensor kamera mendapatkan IP 192.168.0.92 yang nantinya digunakan sebagai penhubung antara *publisher* dengan *middleware*.

5. HASIL DAN PEMBAHASAN

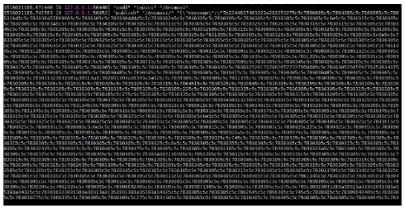
Hasil dan pembahasa berfungsi untuk melaporkan teknik pengujian dan pengolahan data sesuai dengan skenario yang telah ditentukan. Penarikan kesimpulan ini menggunakan representasi uji F atau yang sering

disebut sebagai uji anova. Yang bertujuan untuk mengetahui perbedaan varian dari dua kelompok sampel atau lebih. Kaidah pengambilan keputusan uji F dapat dilakukan dengan membandingakan nilai signifikansi maupun dengan membandingkan nilai kritis dengan nilai F hitung.

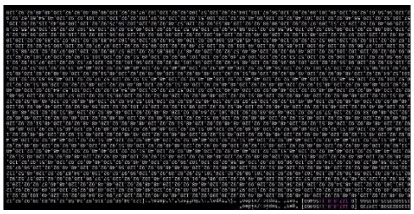
5.1 Analisis Hasil Pengiriman

Dari hasil pengambilan data baik berupa gambar, animasi bergerak, dan video dari sensor kamera, berhasil dikirim menggunakan format json, terbukti dengan pengecekan pada *monitoring* redis-cli pada gambar dibawa. Redis digunakan sebagai *buffer* atau penyimpanan sementara data yang dikirim oleh *publisher*.

Gambar 5-1 Hasil Pengiriman Gambar



Gambar 5-2 Hasil Pengiriman Animasi Gambar Bergerak



Gambar 5.3 Hasil Pengiriman Video

5.2 Skenario 1 Pengiriman Gambar Dengan Resolusi 1360x768

Untuk menarik kesimpulan Pada penelitian ini, pengujian menggunakan perhitungan statistika dengan metode anova, pengiriman dengan resolusi 1360x728, berawal dari perhitungan rata-rata dan standar deviasi dari setiap data baik QoS1, QoS2 maupun QoS3, pada table 1. Dilanjutkan dengan uji kesamaan varian data, pada table 2, terakhir dengan uji data mengunakan anova.

Tabel 1. Perhitungan rata-rata dan standar deviasi

	N	Mean	Std. Deviation
QoS0	30	2381.10	728.632
QoS1	30	2463.87	711.861
QoS2	30	2140.43	613.654
Total	90	2328.47	692.709

Tabel 2. Uji kesaam data varian

Levene Statistic	df1	df2	Sig.
.477	2	87	.622

Uji kesaman data sebagai syarat membandingakan taraf signifikansi, dengan nilai tetap 0.05, kriteria yang digunakan yaitu apabila nilai signifikansi lebih besar dari 0.05 maka data mempnyai varian yang sama. Table 2, adalah hasil uji kesaman data varian (uji homogenis), dari hasil uji data ini dinyatakan dengan hasil signifikansi (0.622) lebih besar dari taraf signifikansinya (0,05) maka data ini memiliki nilai variansi yang sama.

Pada table 3 merupakan hasil uji anova dengan data jitter yang diperoleh dari sekenario 1360, dengan hasil yang diperoleh dibawah juga menyatakan hasil yang sama dengan hasil hitung throughput dan delay, karena nilai F hitung lebih rendah dari nilai F table, dan nilai singnifikansi lebih rendah dari nilai taraf signifikansinya, maka tidak ada perbedaan jika mengguankan QoS0, QoS1 maupun QoS2.

Tabel 3. Pengujian Througput

	Mean Square	F	Sig.
Between Groups	846899.433	1.797	.172
Within Groups	471407.236		
Total			

Tabel 4. Pengujian Delay

	Mean Square	F	Sig.
Between Groups	279.922	.827	.441
Within Groups	338.649		
Total			

Tabel 5. Pengujian Jitter

	Mean Square	F	Sig.
Between Groups	.106	1.000	.372
Within Groups	.106		
Total			

Dengan Demikian pada skenario ini dapat disimpulkan tidak ada perbedaan pengiriman gambar dengan menggunakan QoS0, QoS1, maupun QoS2.

5.3 Skenario 2 Pengiriman Gambar Dengan Resolusi 1024x768

Dengan skenario ini didapatkan hasil sebagai berikut.

Tabel 6. Perhitungan rata-rata dan standar deviasi

	N	Mean	Std. Deviation	Std. Error
QoS0	30	1454.23	435.475	79.506
QoS1	30	1485.97	428.920	78.310
QoS2	30	1514.77	437.987	79.965
Total	90	1484.99	429.958	45.322

Tabel 7. Uji kesaam data varian

Levene Statistic	df1	df2	Sig.
.097	2	87	.908

Tabel 8. Pengujian Througput

	Mean Square	F	Sig.
Between Groups	27503.644	.146	.864
Within Groups	188481.123		
Total			

Tabel 9. Pengujian Delay

	Mean Square	F	Sig.
Between Groups	166.114	.345	.709
Within Groups	482.042		
Total			

Tabel 10. Pengujian Jitter

	Mean Square	F	Sig.
Between Groups	1.650	.841	.435
Within Groups	1.962		
Total			

Pada table 6. Terlihat hasil yang sama dengan skenario pertama, tidak ada perbedaan meskipun dengan menggunakan resolusi gambar yang berbeda. Dilanjutkan dengan hasil pada table 7, dan seterusnya. Yang memastikan nilai dari performa masing masing parameter tidak ada perbedaan.

5.4 Performa Video, Animasi, Gambar

Skenario ini menghasilkan hal yang berbeda, dilihat dari hasil rata rata dan standart deviasi yang berbeda, beda halnya dengan analisis 1 dan 2.

Tabel 11. Perhitungan rata-rata dan standar deviasi

	N	Mean	Std. Deviation	Std. Error
Animasi	30	3932.10	999.471	182.478
Video	30	5206.80	202.561	36.982
Gambar	30	2140.43	613.654	112.037
Total	90	3759.78	1435.721	151.338

Tabel 12. Uji kesaam data varian

Levene Statistic	df1	df2	Sig.
14.560	2	87	.000

Uji kesaman data sebagai syarat membandingakan taraf signifikansi, dengan nilai tetap 0.05, kriteria yang digunakan yaitu apabila nilai signifikansi lebih besar dari 0.05 maka data mempunyai varian yang sama.

Tabel 13. Pengujian Througput

	Mean Square	F	Sig.
Between Groups	142375340.689	150.763	.000
Within Groups	41079812.867		
Total	183455153.556		

Tabel 14. Pengujian Delay

	Mean Square	F	Sig.
Between Groups	136792.063	69.002	. 000
Within Groups	86236.249		
Total	223028.312		

Tabel 15. Pengujian Jitter

	Mean Square	F	Sig.
Between Groups	.000	. 233	. 793
Within Groups	.000		
Total	.000		

Pada table 12, 13 dan 14 untuk menarik kesimpulan dengan ketentuan jika f hitung lebih besar dari f table, maka pernyataan tidak ada perbedaan dalam performa dalam pengiriman tersebut, namun disini kita mendapatkan hasil yang berbeda, nilai F hitung pada delay dan throgput yang didapatkan lebih besar dari f table, terdapat perbedaan antara pengiriman gambar, video dan anmasi tersebut.

6. KESIMPULAN

Berdasarkan hasil penerapan pengiriman hasil gambar yang didapat dari modul kamera raspberry dan pengujian yang telah dilakukan, maka diambil kesimpulan sebagai berikut:

1. Kamera sukses mengambil gambar dengan menghubungkan kabel fleksi HBV-Raspberry -10FPC mengaktifkan sensor kamera pada konfigurasi raspberry pi.
2. Middleware mampu menerima berbagai type file hasil sensor kamera raspberry pi, dengan semantik data yang berbeda dari pengiriman data sensor suhu. Hasil data yang dihasilkan sensor kamera diantaranya gambar (.jpeg), animasi (.gif) dn video. Kinerja dari pengiriman tersebut bervariasi, dari berbagai pengujian yang telah dilalkukan dengan parameter throughput, delay, dan jitter dengan resolusi gambar yang berbeda-beda. Hasil pengujian ini dianalisis mengguankan wireshark. Parameter throughput dianalisa menggunakan fitur yang telah tersedia pada wireshark yaitu capture properties, sedangaan untuk parameter delay dan jitter, menggunakan table time delta yang tersedia saat melakukan *capturing*. File hasil *capturing* wireshark harus di exsport terlebih dahulu menjadi MS Office Excel untuk mempermudah pengolahan data.
3. Berdasarkan analisa yang telah dilakukan dengan 3 skenario, maka didapatkan hasil seperti berikut
   a. Pada saat pengiriman gambar dengan resolusi 1360x768. Rata-rata throughput yang diperoleh pada setia percobaan yang telah dilakukan berkisar 2328 kbps, pada QoS 1 hasil throughput yang didapatkan lebih besar dari QoS lainya yaitu 2463 kbps. Sedangakan rata-rata delay yang diperoleh pada skenario ini adalah 16.613 ms, dengan nilai delay paling rendah terdapat pada QoS 1 dan paling tinggi pada QoS 0 dengan waktu 20.1 ms. Terakhir analisis jitter, pada skenario ini diperoleh 0.0343 ms. Dengan nilai jitter terendah terdapat pada QoS 1, dengan nilai delay yang tinggi namun pada QoS 2 kita dapatkan hasil jitter yang sangan rendah, hal ini dikarenaakn total variasi delay yang didapatkan saat melakukan pengujian berulang ulang.
   b. Pada saat pengiriman gambar dengan resolusi 1024x768. Rata-rata throughput yang diperoleh pada setia percobaan yang telah dilakukan berkisa 1484 kbps, pada QoS 1 hasil throughput yang didapatkan lebih besar dari QoS lainya yaitu 1485 kbps. Sedangakan rata-rata delay yang diperoleh pada skenario ini adalah 25.34 ms, dengan nilai delay paling rendah terdapat pada QoS 2 dan paling tinggi pada QoS 0 dengan waktu 27.43 ms. Terakhir analisis jitter, pada skenario ini diperoleh 0.172 ms. Dengan nilai jitter terendah terdapat pada QoS 0, dengan nilai delay yang tinggi namun padda QoS 0 juga dapatkan hasil jitter yang sangat rendah dibandingakan dengan jitter pada QoS 1 dan 2, hal ini dikarenaakn total variasi delay yang

didapatkan saat melakukan pengujian berulang ulang.

c. Pada saat pengiriman gambar dengan resolusi 800x600. Rata-rata throughput yang diperoleh pada setia percobaan yang telah dilakukan berkisar 860.4 kbps, pada QoS 1 hasil throughput yang didapatkan lebih besar dari QoS lainya yaitu 867.5 kbps. Sedangakan rata-rata delay yang diperoleh pada skenario ini adalah 35.64 ms, dengan nilai delay paling rendah terdapat pada QoS 1 dan paling tinggi pada QoS 2 dengan waktu 37.82 ms. Terakhir analisis jitter, pada skenario ini diperoleh 3.21 ms. Dengan nilai jitter terendah terdapat pada QoS 0, dengan nilai delay yang tinggi namun padda QoS 0 juga dapatkan hasil jitter yang sangat rendah dibandingakan dengan jitter pada QoS 1 dan 2, hal ini dikarenaakn total variasi delay yang didapatkan saat melakukan pengujian berulang ulang.

d. Namun dengan bervariasinya hasil yang didapatkan dari ketiga scenario tersebut, tidak membedakan adanya performa dari masing masing QoS, dari analisis menggunakan probabilitas statistika, dengan menggunakan metode anova one-way kita dapatkan hasil hipotesis yang telah ditentukan, diketahuinya data yang homogen (sama) dan dilihat dari perbandingan hasil F hitung yang lebih kecil dari F table, dan nilai probabilitas (*p-value*) yang lebih besar dari taraf signifikanya menyatakan tidak ada perbandingan performa pada masing - masing skenario, sehingga untuk mengirimkan gambar dengan resolusi yang bervariasi bisa menggunakan QoS0, QoS1, maupun QoS2.

DAFTAR PUSTAKA

A., B. K., A, S. R. & Yahya, W., 2016. *Implementasi aplikasi pengiriman file gambar menggunakan protokol MQTT,* s.l.: j-ptiik.ub.ac.id.

Amron, K., Sakti, E. & Data, M., 2016. Pemodelan Dan Analisis Wireless Mesh Network Dengan Arsitektur Publish-Subscribe Dengan Protokol MQTT. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK),* 3(-), p. 2.

Anwari, H., P., E. S. & Hanafi, M. H., 2017. *Pengembangan Iot Middleware Berbasis Event-Based dengan Protokol komunikasi CoAP, MQTT dan websocket,* s.l.: j-ptiik.ub.ac.id.

Aziz, B., 2015. A Formal Model and Analysis an IoT Protocol,. *e-journal university of Portsmouth* , -(-), pp. 3-4.

Rozi, M. F., P., E. S. & Amron, K., 2017. *Analisis Performansi dan skalabilitas pada event-based IoT middleware,* s.l.: j-ptiik.ub.ac.id.

Singh, M., A, R. M., L, S. V. & p, B., 2015. Secure MQTT For Internet Of Things (IoT). *fifth International Conference on Communication System and Network Technologies,* 747(-), p. 1.