

## Implementasi Konsep *Internet of Things* pada Sistem *Monitoring Banjir* menggunakan Protokol MQTT

Chrisyantar Hasiholan<sup>1</sup>, Rakhmadhany Primananda<sup>2</sup>, Kasyful Amron<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>chrisyantar@gmail.com, <sup>2</sup>rakhmadhany@ub.ac.id, <sup>3</sup>kasyful@ub.ac.id

### Abstrak

Secara umum, *Internet of Things* dapat diartikan sebagai terhubungnya berbagai benda di sekitar dengan sebuah jaringan internet. Untuk menerapkannya, *Internet of Things* membutuhkan suatu jalur komunikasi yang sesuai dengan kebutuhan sistem. Salah satu protokol yang sesuai dengan penerapan konsep *Internet of Things* adalah protokol Message Queue Telemetry Transport (MQTT). Protokol MQTT sering digunakan dalam berbagai sistem yang menggunakan konsep *Internet of Things*, salah satunya adalah sistem *monitoring*. Salah satu aspek dalam kehidupan yang memerlukan suatu sistem *monitoring* adalah bencana alam. Dari berbagai bencana alam yang terjadi di Indonesia, banjir merupakan bencana alam yang cukup sering melanda berbagai kota di Indonesia. Oleh karena itu, penelitian ini mengimplementasikan konsep *Internet of Things* pada sistem *monitoring* banjir dengan menggunakan protokol MQTT. Berdasarkan hasil pengujian, didapatkan kesimpulan bahwa protokol MQTT dapat digunakan pada sistem *monitoring* banjir. Tingkat akurasi yang di dapat dari pengujian sistem adalah 97,801% dengan standar deviasi yang diperoleh sebesar  $\pm 0.0309$  cm. Pada pengujian skalabilitas diperoleh persentase rata-rata tingkat keberhasilan sebesar 100%, 99,87% dan 99,93% pada percobaan publisher 100, 250 dan 500 yang masing-masing dilakukan 3 kali. Dan pada pengujian integritas data, protokol MQTT memperoleh kesamaan data sebesar 100% di setiap percobaan dengan interval 10ms, 100ms dan 1000ms.

**Kata kunci:** *internet of things, sistem monitoring banjir, MQTT, sensor*

### Abstract

*In general, Internet of Things can be interpreted as connecting various objects around with an internet network. To implement it, the Internet of Things requires a communication path to suit the needs of the system. One protocol that suits the application of the concept of Internet of Things is the Message Queue Telemetry Transport (MQTT) protocol. The MQTT protocol is often used in systems that use the concept of the Internet of Things, one of which is the monitoring system. One aspect of life that requires a monitoring system is a natural disaster. From various natural disasters that occurs Indonesia, Flood is a natural disaster that quite often hit many cities in Indonesia. Therefore, this research implements the concept of Internet of Things on flood monitoring system using MQTT protocol. Based on test results, it is concluded that MQTT protocol can be used in flood monitoring system. The accuracy of the system is 97.801% with obtained standard deviation is  $\pm 0.0309$  cm. In scalability testing, obtained the average percentage of success rate of 100%, 99.87% and 99.93% on thread trials 100, 250 and 500, which each performed 3 times. And on data integrity testing, the MQTT protocol obtains the same 100% data in each experiment with intervals of 10ms, 100ms and 1000ms.*

**Keywords:** *internet of things, flood monitoring system, MQTT, sensor*

### 1. PENDAHULUAN

Perkembangan penggunaan internet pada era modern ini berlangsung cepat dan menyentuh berbagai aspek dalam kehidupan masyarakat. Salah satu konsep penggunaan internet yang

tengah berkembang adalah konsep *Internet of Things*. *Internet of Things* (IoT) adalah sebuah konsep dalam pemanfaatan konektivitas internet yang selalu terhubung setiap saat (Rohman et al., 2016). IoT bertujuan untuk menghubungkan perangkat satu dengan yang lainnya melalui internet dengan harapan sistem tersebut dapat

membantu orang-orang dalam melakukan suatu tugas atau pekerjaan. Dalam implementasinya, IoT membutuhkan suatu protokol dalam proses peredaran datanya. Pada penelitian “Sistem Monitoring Suhu Jarak Jauh Berbasis *Internet of Things* menggunakan Protokol MQTT” yang dilakukan oleh Budioko (2016), konsep *Internet of Things* diterapkan pada suatu sistem *monitoring* dengan sensor suhu menggunakan MQTT sebagai protokol.

Sistem *monitoring* adalah suatu sistem yang melakukan proses pemantauan secara terus menerus (Mudjahidin et al., 2010). Sistem *monitoring* dibutuhkan dalam proses pemantauan keadaan suatu objek yang diamati guna mendapatkan informasi yang tepat waktu. Berbagai contoh penerapan sistem *monitoring* yaitu Sistem *Monitoring* dan Pengendalian Suhu dan Kelembaban Ruang pada Rumah Walet Berbasis *Android*, *Web* dan *SMA* oleh Atmoko (2013), Implementasi *Wireless Monitoring* Energi Listrik Berbasis *Web Database* oleh Dinata (2015) dan Sistem Pendeteksi Banjir Berbasis sensor ultrasonik dan Mikrokomputer dengan Media Komunikasi *SMS Gateway* oleh Sulistyowati et al. (2015). Sistem *monitoring* juga dapat digunakan dalam memantau ketinggian air dan banjir di berbagai tempat dan menampilkan data yang akurat dengan cepat.

Banjir merupakan masalah yang cukup sering melanda berbagai tempat di Indonesia. Secara umum, banjir adalah suatu kejadian dimana air di dalam saluran meningkat dan melampaui kapasitas daya tampungnya (Seno, 2013). Terdapat berbagai sistem penanggulangan dan peringatan dini banjir, diantaranya Pengembangan Model SIG untuk menentukan Rute Evakuasi Bencana Banjir oleh Mulyanto (2008), Pembangkitan Pola Data Cuaca untuk Sistem Peringatan Dini Banjir oleh Suwarningsih dan Suryawati (2012). Namun, belum ada penelitian yang menggunakan sistem *monitoring* banjir dengan data yang mudah diakses oleh masyarakat. Fenomena banjir kilat (bandang) dan banjir kiriman dapat menjadi masalah dalam sistem tersebut, karena penelitian-penelitian tersebut tidak memiliki sistem *monitoring* dengan data yang mudah diakses oleh pengguna.

Dalam era teknologi informasi dimana informasi dapat disebarkan secara cepat, tentu diperlukan suatu sistem yang dapat menyebarkan informasi mengenai banjir dan ketinggian air di berbagai tempat secara cepat dan mudah diakses. Penulis berpendapat bahwa

konsep *Internet of Things* menggunakan protokol MQTT dapat diterapkan dalam sebuah sistem *monitoring* banjir.

Protokol Message Queue Telemetry Protocol (MQTT) adalah protokol yang sering digunakan dalam penerapan konsep IoT. Protokol MQTT merupakan protokol yang ringan, karena mengirim pesan dengan *header* berukuran kecil yaitu 2 *bytes* (Rochman, 2017). Protokol MQTT bekerja menggunakan konsep *publish/subscribe* (Bandyopadhyay, 2013). Perangkat yang melakukan proses *publish* disebut *publisher*, sedangkan perangkat yang melakukan proses *subscribe* disebut *subscriber*. MQTT berbasis *publish/subscribe* dengan *message-broker* sebagai jembatan antara *publisher* dan *subscriber* (Bandyopadhyay, 2013). Pesan yang melalui proses *publish/subscribe* berupa topik. *Subscriber* dapat memilih topik mana yang ingin dikirim oleh *publisher* melalui *broker*.

Pada penelitian yang membandingkan *round trip*, sumber daya *server* dan *payload size* antara protokol HTTP dan MQTT, protokol MQTT dinyatakan bekerja lebih baik (Yokotani et al., 2016). Protokol MQTT disebutkan memiliki *response time* paling kecil ketika dibandingkan dengan DPWS dan CoAP pada 100 *request* (Fysarakis et al., 2016). Protokol MQTT juga terbukti tepat dalam implementasi konsep IoT, seperti yang dilakukan oleh Kim et al. (2015) dalam penelitian “*IoT Home Gateway for Auto-Configuration and Management of MQTT devices*”. Dalam penelitian “Sistem Monitoring Suhu Jarak Jauh Berbasis *Internet of Things* menggunakan Protokol MQTT”, protokol MQTT disebutkan paling tepat dalam mengalirkan data dari perangkat sensor menuju sebuah jaringan dalam suatu sistem *monitoring* (Budioko, 2016)

Pada penelitian ini, penulis menerapkan konsep *publish/subscribe* menggunakan protokol MQTT pada sistem *monitoring* banjir. Perangkat yang dijadikan *publisher* adalah mikrokomputer Raspberry Pi yang tersambung ke sensor ultrasonik pengukur jarak, *broker* yang digunakan adalah Mosquitto dan *subscriber* dapat menampilkan data tersebut melalui sebuah *web*. Protokol MQTT dipilih karena dalam penelusuran penulis terkait topik penggunaan protokol pada sebuah sistem yang menggunakan sensor, protokol MQTT dinyatakan tepat dan sesuai. Dengan data pengukuran sensor yang berupa beberapa *digit* angka, protokol MQTT

disebutkan cocok dalam penerapan sistem *monitoring* dengan *payload size* yang kecil.

Berdasarkan deskripsi yang telah dijabarkan sebelumnya, penulis menggunakan judul penelitian “Implementasi Konsep *Internet of Things* pada Sistem *Monitoring Banjir* menggunakan Protokol MQTT”. Sistem *monitoring* banjir yang dimaksud menggunakan protokol MQTT berbasis *web*, dengan menggunakan sensor ultrasonik sebagai pengukur ketinggian air dan Raspberry Pi sebagai mikrokomputer. Sensor dan mikrokomputer sebagai *publisher* dipasang di berbagai titik daerah endemik banjir, mengirim data menggunakan protokol MQTT melalui jaringan ke *broker* yang aktif pada perangkat PC/laptop dan *broker* akan mengirimkan data tersebut ke *web* yang dapat diakses oleh *subscriber*. Diharapkan penelitian ini dapat menjadi solusi dalam masalah banjir di berbagai daerah.

## 2. ANALISIS KEBUTUHAN

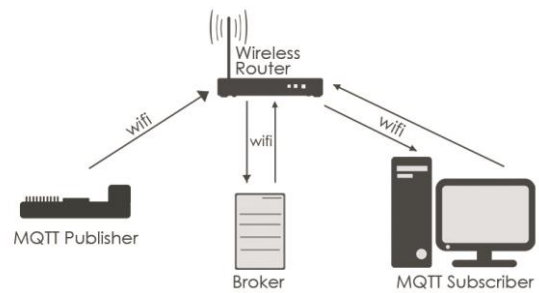
Dalam penelitian Implementasi Konsep *Internet of Things* pada Sistem *Monitoring Banjir* menggunakan Protokol MQTT dibutuhkan sistem yang memadai, yaitu:

- MQTT *Publisher* mampu mengirimkan pesan ke MQTT *Broker* melalui *wireless network* dengan jaringan *wi-fi*.
- MQTT *Broker* mampu menerima pesan yang dikirimkan oleh MQTT *Publisher* melalui *wireless network* dengan jaringan *wi-fi*.
- MQTT *Subscriber* mampu melakukan proses *subscribe* pada topik yang diinginkan ke MQTT *Broker*.
- MQTT *Broker* mampu mengirimkan pesan ke MQTT *Subscriber* sesuai dengan topik yang diinginkan.

## 3. PERANCANGAN DAN IMPLEMENTASI

### 3.1. Gambaran Umum Sistem

Gambar 1 menjelaskan gambaran penggunaan protokol MQTT pada sistem secara umum. Komponen-komponen yang terdapat pada protokol MQTT meliputi *publisher*, *broker* dan *subscriber* yang melakukan komunikasi menggunakan jaringan *wifi* yang disediakan *wireless router*. *Publisher* melakukan *publish message* yang berbasis topik ke *broker*, lalu *broker* akan menerima *publish message* tersebut.

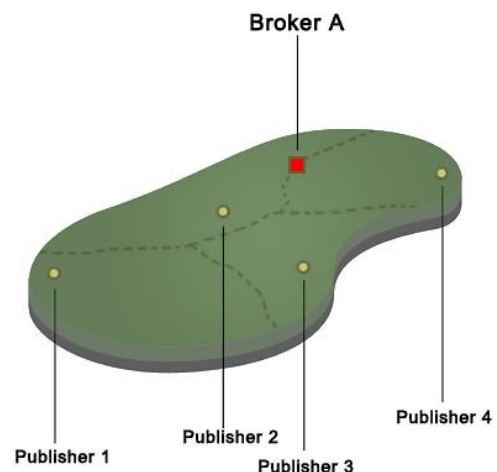


Gambar 1. Gambaran Umum Sistem

*Subscriber* akan melakukan proses *subscribe* topik dengan mengirimkan *request* topik tertentu ke *broker*, lalu *broker* akan menerima *request subscribe* topik tersebut dan mengirimkan *message* dari *publisher* dengan topik yang sesuai kepada *subscriber*.

Tiap-tiap *publisher* ditempatkan pada daerah-daerah endemik banjir pada suatu wilayah. Dalam suatu wilayah ditempatkan juga satu *broker*. Tiap-tiap *publisher* akan mengirimkan data pengukuran air pada daerah *publisher* tersebut ke *broker* dalam wilayah tersebut. Setiap *publisher* tersebut akan melakukan proses *publish* dengan topik yang berbeda. Gambar 2 menampilkan *design plan* dari sistem *monitoring* yang dibangun. Area berwarna hijau adalah suatu wilayah A, kotak merah adalah *broker* pada wilayah A, lingkaran kuning adalah *publisher-publisher* pada wilayah A dan garis putus-putus adalah batas dari tiap-tiap daerah yang dapat ditangani oleh setiap *publisher*.

Setiap *publisher* dalam wilayah A terhubung dengan jaringan *wireless* ke *broker* pada wilayah A. Jaringan *wireless* dipilih karena pada penerapannya lebih menghemat biaya dalam membangun jaringannya dan jaringan *wireless* lebih fleksibel karena tidak memerlukan



Gambar 2. Design Plan wilayah A

suatu kabel penghubung jaringan dari *publisher* ke *broker*.

### 3.2. Perancangan *Publisher*

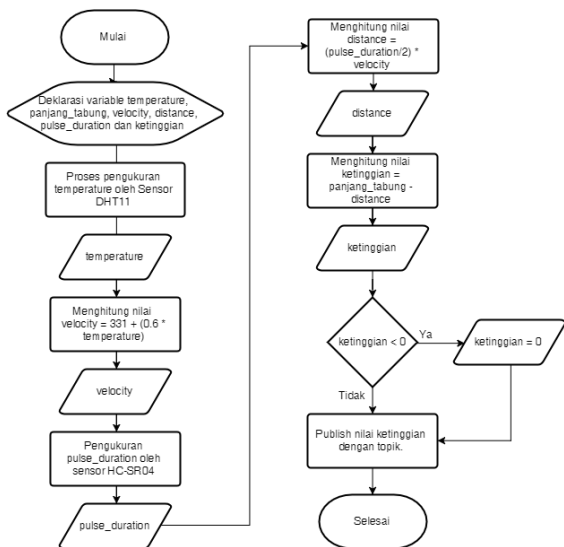
*Publisher* pada sistem ini merupakan perangkat sensor dan mikrokomputer yang berfungsi mengambil data pengukuran ketinggian air dan mengirimkannya ke *broker*. Sensor yang digunakan pada sistem yaitu sensor HC-SR04 sebagai pengukur ketinggian air dan sensor DHT11 sebagai pengukur temperatur udara untuk menunjang pengukuran ketinggian air. Sedangkan mikrokomputer yang digunakan adalah Raspberry Pi 3. perancangan aliran data dari mikrokomputer sebagai *publisher* ke *broker* yang dapat dilihat pada Gambar 3.

### 3.3. Perancangan *Broker*

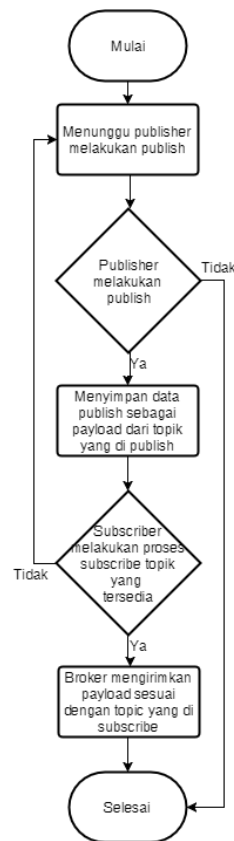
*Broker* yang digunakan pada sistem ini adalah *broker* Mosquitto yang diaktifkan pada perangkat laptop, berperan sebagai MQTT *broker*. *Broker* ini berfungsi sebagai penerima data *publish* dari *publisher* dan juga sebagai pengirim data ke *subscriber* yang melakukan *subscribe* dengan topik yang tersedia. Perancangan aliran data *broker* dapat dilihat pada Gambar 4.

### 3.4. Perancangan *Subscriber*

*Subscriber* pada sistem ini adalah *web* yang diakses menggunakan laptop/smartphone. *Web* tersebut akan melakukan proses *subscribe* topik ke *broker*. Setelah *payload* dari topik diterima,



Gambar 3. Diagram Aliran Data *Publisher*



Gambar 4. Diagram Aliran Data *Broker*

data akan ditampilkan pada *interface browser*. Aliran data pada *subscriber* dapat dilihat pada Gambar 5.

## 4. HASIL PENGUJIAN DAN PEMBAHASAN

### 4.1 Pengujian Keseluruhan Sistem

Pengujian ini bertujuan untuk menguji bagaimana sistem dapat melakukan fungsinya dalam mengukur ketinggian air sampai mengirimkan data ke *subscriber*. Pengujian ini meliputi pengujian sensor mengukur ketinggian air, *publisher* melakukan proses *publish*, dan *subscriber* melakukan proses *subscribe*. Dalam pengujian ini *publisher*, *broker* dan *subscriber* terhubung ke dalam satu jaringan yang sama. Pengujian dilakukan untuk mengetahui apakah sistem dapat menjalankan fungsi keseluruhan secara utuh.

Pengujian dilakukan dengan menjalankan sensor dengan 2 kondisi yaitu dengan tidak ada air dan ada air di wadah penampung dibawah sensor. Lalu setelah pengambilan data sensor, *publisher* melakukan proses *publish* dan *subscriber* melakukan proses *subscribe*.

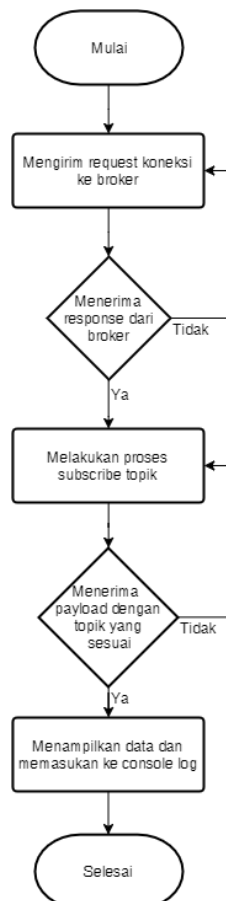
Dari pengujian keseluruhan sistem yang

telah dilakukan, diperoleh hasil bahwa sistem dapat melakukan tugasnya. Sensor dapat mengukur ketinggian air di bawahnya, *publisher* Raspberry Pi 3 dapat menjalankan perintah pada python yang berisi perintah melakukan proses *publish* ke *broker*, lalu *subscriber* dapat melakukan proses *subscribe* dan menampilkan data yang di dapat pada *interface web*.

#### 4.2 Pengujian Akurasi

Tujuan dari pengujian akurasi adalah untuk mengetahui apakah sensor dapat memberikan nilai ketinggian air yang sesuai ketika dibandingkan dengan nilai ketinggian air sebenarnya. Pengujian akurasi memiliki skenario tambahan yaitu pengujian presisi yang bertujuan untuk mengetahui tingkat presisi dari sensor.

Pengujian akurasi dilakukan dengan membandingkan hasil pengukuran menggunakan instrumen ukur penggaris dengan pengukuran yang dilakukan oleh sensor pada kondisi ketinggian air yang sama. Pengujian dilakukan sebanyak 10 kali dengan nilai ketinggian air acuan hasil dari pengukuran manual menggunakan instrumen ukur penggaris.



Gambar 5. Diagram Aliran Data *Subscriber*

Tabel 1. Hasil Pengujian Akurasi

No.	Nilai Aktual (cm)	Nilai Sensor (cm)	Persentase Akurasi (%)
1.	2	1.86	93
2.	5	4.88	97,6
3.	10	10.4	96
4.	17	17.4	97,65
5.	30	30.8	97,33
6.	39	39.1	99.73
7.	45	45.6	98,7
8.	51	51.67	98,69
9.	56	56.36	99,36
10.	59.4	59.37	99,95
		<b>Rata-rata</b>	<b>97,801</b>

Pengujian dilakukan pada wadah penampung air dengan ketinggian air maksimal 59,4 cm. Hasil pengujian akurasi dapat dilihat pada Tabel 1.

Dari pengujian akurasi didapatkan hasil persentase akurasi terkecil sebesar 93% yaitu pada pengujian ke- 1 pada titik acuan 2 cm dan persentase akurasi terbesar pada pengujian ke-10 pada titik acuan 59.4 cm yaitu sebesar 99.95%. Dari pengujian akurasi yang dilakukan 10 kali, di dapatkan rata-rata persentase akurasi sebesar 97,801% dengan rata-rata kesalahan sebesar 0.362 cm.

Pengujian presisi dilakukan dengan melakukan pengukuran ketinggian air menggunakan sensor sebanyak 10 kali dengan kondisi ketinggian air yang tidak berubah. Pada pengujian presisi, sensor mengukur ketinggian air dengan selisih sebesar 0,09 cm antara nilai pengukuran minimum dan nilai pengukuran maksimal. Dari hasil pengujian presisi, akan dilakukan perhitungan standar deviasi dengan rumus pada Persamaan (1)

$$SD = \sqrt{\frac{\sum(x_i - \bar{x})^2}{n-1}} \tag{1}$$

Dimana SD adalah standar deviasi,  $\bar{x}$  adalah nilai rata-rata dari hasil pengukuran sensor,  $x_i$  adalah nilai dari setiap pengukurannya dan  $n$  adalah jumlah pengukuran yang dilakukan. Hasil pengujian presisi dapat dilihat pada Tabel 2.

Setelah dilakukan perhitungan, maka didapatkan hasil sebagaimana dinyatakan pada

Tabel 2. Hasil Pengujian Presisi

No.	Nilai Aktual (cm)	Nilai Sensor (cm)	$(X_i - \bar{X})^2$
1.	59,4	59,34	0.001369
2.	59,4	59,39	0.000169
3.	59,4	59,37	0.000049
4.	59,4	59,41	0.001089
5.	59,4	59,39	0.000169
6.	59,4	59,35	0.000729
7.	59,4	59,41	0.001089
8.	59,4	59,32	0.003249
9.	59,4	59,39	0.000169
10.	59,4	59,40	0.000529
<b>Rata-rata</b>	<b>59,377</b>		

Persamaan (2). Pada pengujian presisi, didapatkan standar deviasi sebesar ± 0.0309 cm.

$$SD = \sqrt{\frac{0.00861}{9}}$$

$$SD = \pm 0.0309 \tag{2}$$

#### 4.2 Pengujian Skalabilitas

Pengujian skalabilitas bertujuan untuk mengetahui seberapa banyak *publisher* yang dapat ditangani *broker* dalam satu waktu. Pengujian dilakukan dengan menggunakan rekayasa jumlah *publisher* karena pada implementasinya *broker* akan menangani banyak *publisher* yang terus mengirimkan data pada waktu yang berdekatan. Pengujian dilakukan dengan variasi jumlah *publisher* sebanyak 100, 200 dan 500 dengan masing-masing pengujian dilakukan sebanyak 3 kali.

Pada Tabel 3 dapat dilihat hasil dari pengujian skalabilitas pada jumlah *publisher* 100, 250 dan 500. Pada 100 *publisher*, rata-rata *thread* yang dapat di tangani oleh *broker* sebesar 100 *publisher*, dengan persentase keberhasilan sebesar 100%. Pada *publisher* 250, rata-rata *thread* yang dapat di tangani oleh *broker* sebesar 249 dengan persentase keberhasilan sebesar 99,87%. Pada *publisher* 500, rata-rata *thread* yang dapat ditangani oleh *broker* sebesar 499 dan persentase keberhasilan 99,93%.

Dari hasil pengujian skalabilitas di atas, di dapatkan persentase tingkat keberhasilan paling kecil sebesar 99.6% yaitu pada pengujian ke- 3 pada *thread group* 250. Sedangkan rata-rata persentase tingkat keberhasilan sebesar 99.93%.

Tabel 3. Hasil Pengujian Skalabilitas

<b>Thread group 100</b>		
<i>Publisher</i>	Responded	Tingkat keberhasilan (%)
100	100	100
100	100	100
100	100	100
<b>Rata-rata</b>	<b>100</b>	<b>100</b>
<b>Thread Group 250</b>		
<i>Publisher</i>	Responded	Tingkat keberhasilan (%)
250	250	100
250	250	100
250	249	99,6
<b>Rata-rata</b>	<b>249</b>	<b>99,87</b>
<b>Thread Group 500</b>		
<i>Publisher</i>	Responded	Tingkat keberhasilan (%)
500	500	100
500	499	99,8
500	500	100
<b>Rata-rata</b>	<b>499</b>	<b>99,93</b>

Pengujian skalabilitas pada sistem terbilang sangat baik karena persentase tingkat keberhasilan dan rata-rata tingkat keberhasilan secara keseluruhan lebih dari 99%.

#### 4.2 Pengujian Integritas Data

Pengujian integritas data memiliki tujuan untuk mengetahui apakah protokol MQTT merupakan protokol yang reliabel dalam pengiriman suatu data. Hasil pengujian integritas data dapat dilihat pada Tabel 4.

Tabel 4. Hasil Pengujian Integritas Data

No.	Interval 1000 ms		Interval 100 ms		Interval 10 ms	
	Data <i>publisher</i>	Data <i>Broker</i>	Data <i>Publis her</i>	Data <i>Brok er</i>	Data <i>Publis her</i>	Data <i>Brok er</i>
1.	77	77	21	21	82	82
2.	14	14	44	44	51	51
3.	94	94	58	58	22	22
4.	76	76	24	24	85	85
5.	19	19	85	85	55	55
6.	53	53	30	30	8	8
7.	13	13	4	4	42	42
8.	94	94	2	2	53	53
9.	96	96	8	8	58	58
10	73	73	87	87	27	27

Pengujian dilakukan dengan *publisher* melakukan proses *publish* ke *broker* dengan topik “banjir/ub” dan isi pesan dalam *integer* pada range 0 sampai 100 secara acak. *Interval* antar pengiriman data masing-masing 1000 ms, 100 ms dan 10 ms.

## 5. KESIMPULAN

Setelah tahapan-tahapan pada penelitian ini selesai dilakukan, maka kesimpulan yang dapat ditarik adalah sebagai berikut:

1. Konsep *Internet of Things* dengan menggunakan protokol MQTT dapat diterapkan pada sistem *monitoring* banjir yang dibuat sesuai dengan perancangan. Dalam penerapannya, dibutuhkan elemen-elemen yang berkaitan yaitu *publisher*, *broker* dan *subscriber*. Pada penelitian ini, *publisher* yang digunakan adalah Raspberry Pi 3, sensor ultrasonik HC-SR04. *Broker* yang digunakan adalah Mosquitto 1.4.7 dan laman *web* yang berperan sebagai *subscriber*. Semua elemen tersebut dihubungkan dengan jaringan *wi-fi*. Elemen-elemen tersebut dapat digunakan dalam implementasi konsep *Internet of Things* menggunakan protokol MQTT pada sistem *monitoring* banjir.
2. Rata-rata persentase akurasi sensor HC-SR04 dalam mengukur ketinggian air dari 10 kali percobaan adalah 97,801% dengan standar deviasi  $\pm 0.0309$  cm. Kesalahan yang terjadi pada pengukuran oleh sensor dapat disebabkan oleh jarak permukaan air dengan muka sensor dan ketidakstabilan permukaan air yang diukur oleh sensor.
3. Kemampuan sistem menangani 100 *publisher* dalam satu waktu memiliki tingkat keberhasilan sebesar 100%, pada penanganan 250 *publisher* sebesar 99,87% dan pada penanganan 500 *publisher* diperoleh tingkat keberhasilan sebesar 99,93%. Fluktuasi pada persentase tingkat keberhasilan sistem menangani banyak *publisher* disebabkan oleh gangguan sinyal jaringan *wi-fi* yang digunakan saat pengujian skalabilitas dilakukan.
4. Dalam pengujian integritas data, diperoleh kesamaan data 100% dari hasil pengiriman data dengan *interval* 10ms, 100ms, 1000ms. Keberhasilan sistem dalam menjaga integritas data disebabkan oleh protokol MQTT yang reliabel dalam proses pengiriman data.

## DAFTAR PUSTAKA

- Atmoko, R. A. 2013. Sistem Monitoring dan Pengendalian Suhu dan Kelembaban Ruang pada Rumah Walet Berbasis Android, Web dan SMA. *Seminar Nasional Teknologi Informasi & Komunikasi Terapan*, 16 November, Semarang, Indonesia. 283-290.
- Bandyopadhyay, S. & Bhattacharyya, A. 2013. Lightweight Internet Protocols for Web Enablement of Sensors using Constrained Gateway Devices. *International Conference on Computing, Networking and Communications, Workshops Cyber Physical System*, 28-31 January, San Diego, US. 334-340.
- Budioko, T. 2016. Sistem Monitoring Suhu Jarak Jauh Berbasis Internet of Things Menggunakan Protokol MQTT. *Seminar Nasional Riset Teknologi Informasi*, 30 Juli, Yogyakarta, Indonesia. 535-538.
- Dinata, I. & Sunanda, W. 2015. Implementasi Wireless Monitoring Energi Listrik Berbasis Web Database. *Skripsi*. Fakultas Teknik Elektro Universitas Bangka Belitung.
- Fysarakis, K., Askoxylakis, I., Soutatos, O., Papaefstathiou, I., Manifavas, C. & Katos, V. 2016. Which IoT Protocol? Comparing Standardized Approaches over a Common M2M Application. *Global Communications Conference (GLOBECOM)*, 4-8 December, Washington, US. 1-7.
- Kim, S., Choi, H. & Rhee, W. 2015. IoT Gateway for Auto-Configuration and Management of MQTT Devices. *IEEE Conference on Wireless Sensors (ICWiSe)*, 24-26 August, Melaka, Malaysia. 12-17.
- Mudjahidin, M. & Putra, N. D. P. 2010. Rancang Bangun Sistem Informasi Monitoring Perkembangan Proyek Berbasis Web Studi Kasus di Dinas Bina Marga dan Pemantusan. *Jurnal Teknik Industri*, 11(1), 75-83.
- Mulyanto, A. 2008. Pengembangan Model SIG untuk Menentukan Rute Evakuasi Bencana Banjir (Studi Kasus: Kec. Semarang Barat, Kota Semarang). *Skripsi*. Fakultas Teknik Universitas Diponegoro, Semarang.

- Rochman, H. A., Primananda, R. & Nurwasito, H. 2017. Sistem Kendali Berbasis Mikrokontroler menggunakan Protokol MQTT pada Smarthome. *Skripsi*. Fakultas Ilmu Komputer Universitas Brawijaya, Malang.
- Rohman, F. & Iqbal, M. 2016. Implementasi IoT dalam Rancang Bangun Sistem Monitoring Panel Surya berbasis Arduino. *Skripsi*. Fakultas Teknik. Universitas Muria Kudus.
- Seno, A. 2013. Karakterisasi Bencana Banjir Bandang di Indonesia. *Jurnal Sains dan Teknologi Indonesia*, 15(1), 42-51.
- Sulistiyowati, R., Sujono, H. A. & Musthofa, A. K. 2015. Sistem Pendeteksi Banjir berbasis Sensor Ultrasonik dan Mikrokontroler dengan Media Komunikasi SMS Gateway. *Skripsi*. Fakultas Teknologi Industri Institut Teknologi Adhi Tama Surabaya.
- Suwarningsih, W. & Suryawati, E. 2012. Pembangkitan Pola Data Cuaca untuk Sistem Peringatan Dini Banjir. *Jurnal Informatika, Sistem Kendali dan Komputer*, 6(1), 9-14.
- Yokotani, T. & Sasaki, Y. 2016. Comparison with HTTP and MQTT on Required Network Resources for IoT. *The 2016 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, 13-15 September, Bandung, Indonesia. 1-6.