

**IMPLEMENTASI KONSEP *INTERNET OF THINGS* PADA
SISTEM *MONITORING BANJIR* MENGGUNAKAN PROTOKOL
MQTT**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Chrisyantar H.
NIM: 135150200111129



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

IMPLEMENTASI KONSEP *INTERNET OF THINGS* PADA SISTEM *MONITORING*
BANJIR MENGGUNAKAN PROTOKOL MQTT

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:
Chrisyantar H.
NIM: 135150200111129

Skripsi ini telah diuji dan dinyatakan lulus pada
26 Juli 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Rakhmadhany Primananda, S.T.
M.Kom

NIK: 20160986 0406 1 001

Kasyful Amron, S.T, M.Sc
NIP: 19750803 200312 1 003

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001

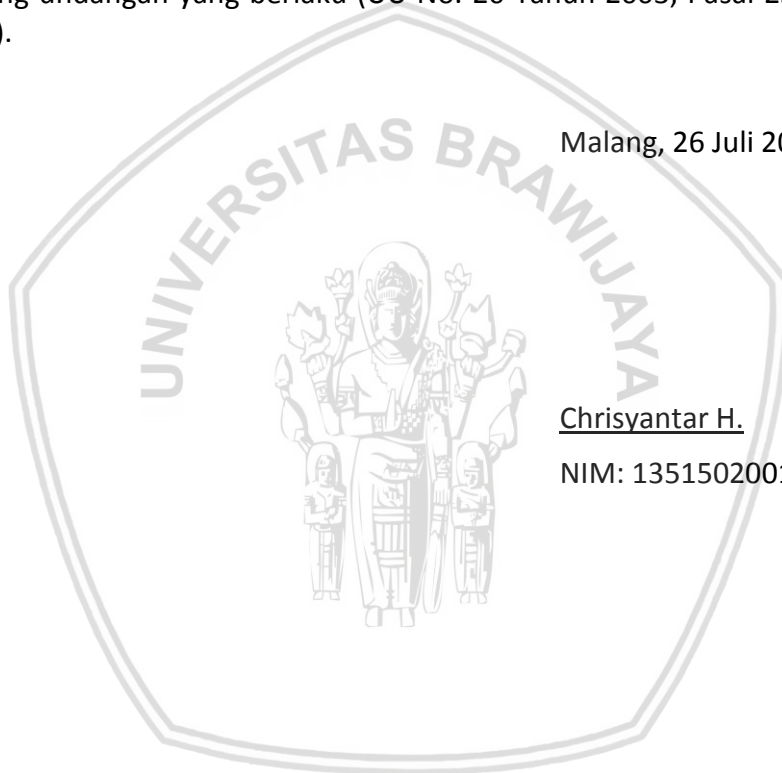


PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 26 Juli 2018



Chrisyantar H.

NIM: 135150200111129

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan YME yang telah melimpahkan berkat-berkatnya sehingga penulis dapat menyelesaikan penelitian dan skripsi dengan judul “Implementasi Konsep *Internet of Things* pada Sistem *Monitoring Banjir* menggunakan Protokol MQTT”. Penulis juga mengucapkan banyak terima kasih kepada pihak-pihak yang terlibat dalam proses penyusunan laporan ini. Ucapan terima kasih penulis ucapkan kepada:

1. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika.
2. Bapak Agus Wahyu Widodo, S.T, M.Cs selaku Ketua Program Studi Teknik Infomatika.
3. Bapak Rakhmadhany Primananda, S.T, M.Kom selaku dosen pembimbing I yang telah membimbing penulis dengan pikiran serta nasihat sehingga penulis dapat menyelesaikan pengerjaan skripsi ini.
4. Bapak Kasyful Amron, S.T, M.Sc selaku dosen pembimbing II yang telah membimbing penulis dengan pikiran serta nasihat sehingga penulis dapat menyelesaikan pengerjaan skripsi ini.
5. Bapak dan Ibu dosen yang telah memberikan ilmu serta didikan selama penulis menempuh pendidikan di Fakultas Ilmu Komputer, Universitas Brawijaya, Malang.
6. Segenap karyawan Fakultas Ilmu Komputer yang telah membantu proses penyelesaian skripsi ini.
7. Kedua orang tua penulis, Bapak Nelson Hutabarat dan Ibu Berlyana Gultom, seluruh anggota keluarga penulis serta teman-teman penulis atas seluruh dukungan doa, nasihat dan kasih sayang yang tiada habisnya kepada penulis.

Dalam proses penyusunan laporan skripsi ini, penulis menyadari bahwa masih terdapat banyak sekali kekurangan dan kesalahan. Penulis menerima kritik dan saran yang dapat membangun dan memperbaiki dalam penyusunan laporan skripsi ini. Semoga laporan ini dapat digunakan dan bermanfaat bagi masyarakat luas. Akhir kata, penulis mengucapkan terima kasih.

Malang, 26 Juli 2018

Penulis

chrisyantar@gmail.com

ABSTRAK

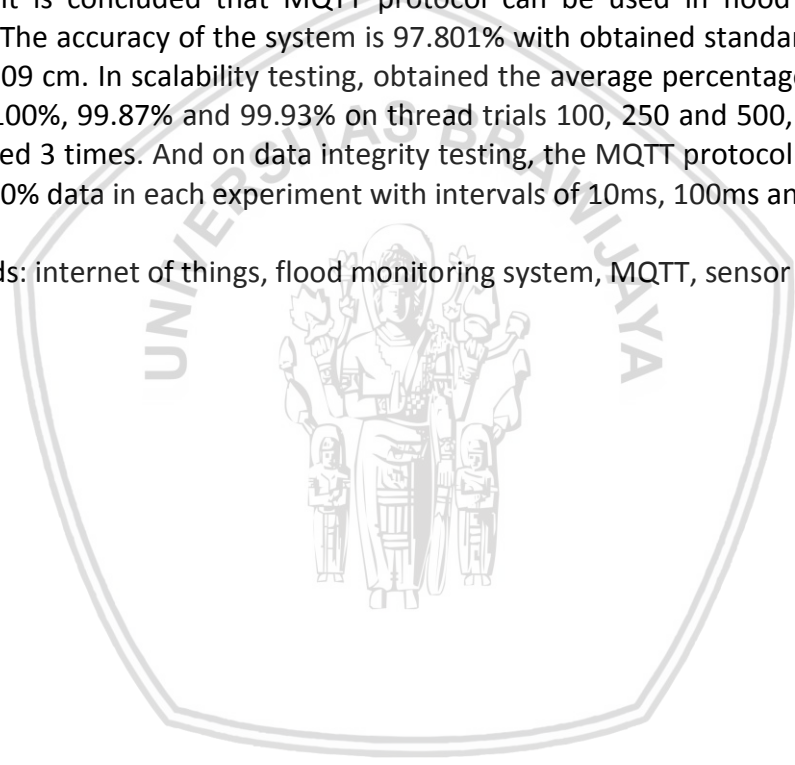
Secara umum, *Internet of Things* dapat diartikan sebagai terhubungnya berbagai benda di sekitar dengan sebuah jaringan internet. Untuk menerapkannya, *Internet of Things* membutuhkan suatu jalur komunikasi yang sesuai dengan kebutuhan sistem. Salah satu protokol yang sesuai dengan penerapan konsep *Internet 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 ± 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 ± 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



DAFTAR ISI

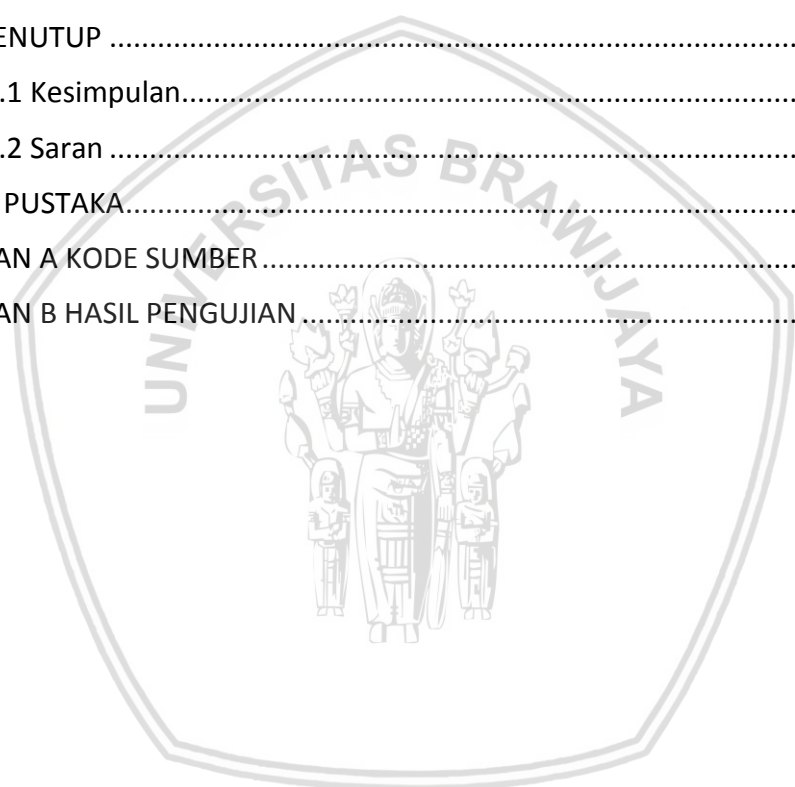
PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
DAFTAR LAMPIRAN	xii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 <i>Internet of Things</i>	5
2.3 Message Queue Telemetry Transport	6
2.4 <i>Broker</i>	8
2.5 Sensor Ultrasonik.....	8
2.6 Akurasi dan Presisi Sensor	9
2.7 Skalabilitas	9
2.8 Integritas Data	10
BAB 3 METODOLOGI	11
3.1 Studi Literatur	11
3.2 Analisis Kebutuhan	12
3.2.1 Kebutuhan Sistem	12
3.3 Perancangan Sistem.....	13
3.4 Implementasi Sistem	15



3.5 Pengujian Sistem.....	15
3.6 Penarikan Kesimpulan	16
BAB 4 REKAYASA KEBUTUHAN.....	17
4.1 Deskripsi Umum.....	17
4.1.1 Perspektif Sistem.....	17
4.1.2 Tujuan.....	17
4.1.3 Ruang Lingkup	17
4.1.4 Batasan Sistem	18
4.1.5 Lingkungan Operasi.....	18
4.1.6 Asumsi dan Ketergantungan	18
4.2 Kebutuhan Sistem.....	18
4.2.1 Kebutuhan Perangkat Keras.....	18
4.2.2 Kebutuhan Perangkat Lunak	18
4.2.3 Kebutuhan Fungsional.....	19
4.2.4 Kebutuhan Non-Fungsional	19
BAB 5 PERANCANGAN DAN IMPLEMENTASI.....	21
5.1 Perancangan Sistem.....	21
5.1.1 Gambaran Umum Sistem	21
5.1.2 <i>Publisher</i>	22
5.1.3 <i>Broker</i>	24
5.1.4 <i>Subscriber</i>	26
5.2 Implementasi Sistem	27
5.2.1 Implementasi Perangkat Keras	27
5.2.2 Implementasi Perangkat Lunak.....	27
BAB 6 PENGUJIAN DAN ANALISIS.....	38
6.1 Pengujian Keseluruhan Sistem	38
6.1.1 Tujuan Pengujian.....	38
6.1.2 Pengujian.....	38
6.1.3 Hasil & Analisis Pengujian	40
6.2 Pengujian Akurasi	42
6.2.1 Tujuan Pengujian.....	42
6.2.2 Pengujian.....	42

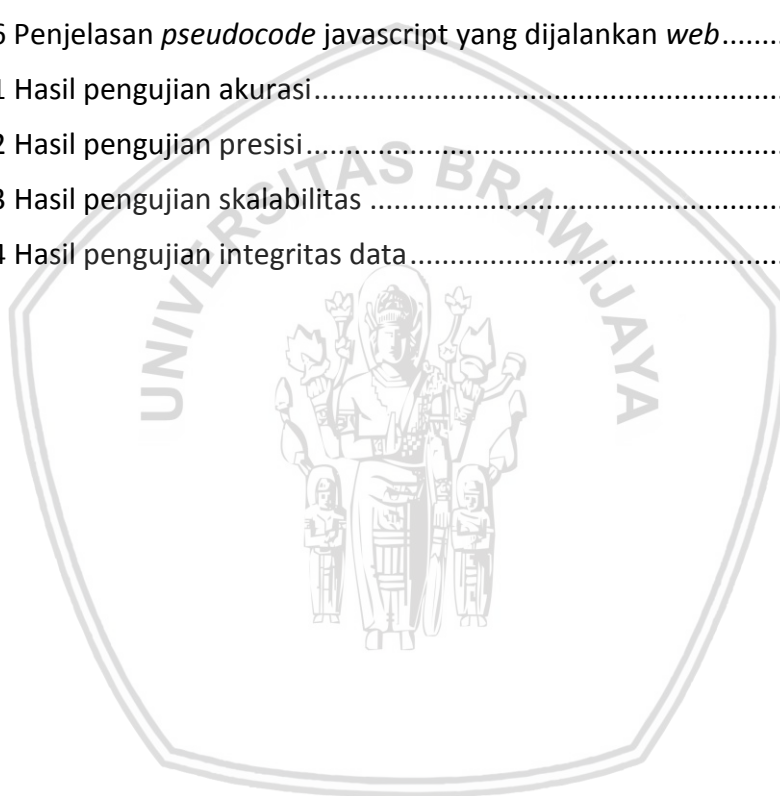


6.2.3 Hasil & Analisis Pengujian	43
6.3 Pengujian Skalabilitas	45
6.3.1 Tujuan Pengujian.....	45
6.3.2 Pengujian.....	45
6.3.3 Hasil & Analisis Pengujian	47
6.4 Pengujian Integritas Data	48
6.4.1 Tujuan Pengujian.....	48
6.4.2 Pengujian.....	48
6.4.3 Hasil & Analisis Pengujian	50
BAB 7 PENUTUP	51
7.1 Kesimpulan.....	51
7.2 Saran	51
DAFTAR PUSTAKA.....	53
LAMPIRAN A KODE SUMBER.....	55
LAMPIRAN B HASIL PENGUJIAN.....	63



DAFTAR TABEL

Tabel 3.1 Spesifikasi perangkat keras	12
Tabel 3.2 Spesifikasi perangkat lunak	13
Tabel 5.1 <i>Pseudocode publisher</i>	28
Tabel 5.2 Penjelasan <i>pseudocode publisher</i>	28
Tabel 5.3 <i>Pseudocode</i> pada javascript konfigurasi	32
Tabel 5.4 Penjelasan <i>pseudocode</i> javascript konfigurasi	33
Tabel 5.5 <i>Pseudocode</i> pada javascript yang dijalankan oleh <i>web</i>	33
Tabel 5.6 Penjelasan <i>pseudocode</i> javascript yang dijalankan <i>web</i>	35
Tabel 6.1 Hasil pengujian akurasi	43
Tabel 6.2 Hasil pengujian presisi	44
Tabel 6.3 Hasil pengujian skalabilitas	47
Tabel 6.4 Hasil pengujian integritas data	50



DAFTAR GAMBAR

Gambar 2.1 Metode komunikasi sistem <i>monitoring</i> banjir dengan IoT.....	6
Gambar 2.2 Metode <i>publish/subscribe</i> dalam MQTT.....	7
Gambar 3.1 Diagram Alir Pelaksanaan Penelitian	11
Gambar 3.2 Ilustrasi Perancangan	14
Gambar 5.1 Gambaran Umum Sistem	21
Gambar 5.2 <i>Design Plan</i> wilayah A	22
Gambar 5.3 Diagram aliran data <i>publisher</i>	23
Gambar 5.4 Diagram aliran data <i>broker</i>	25
Gambar 5.5 Diagram aliran data <i>subscriber</i>	26
Gambar 5.6 <i>Output terminal</i> pada <i>publisher</i>	30
Gambar 5.7 Proses instalasi Mosquitto	30
Gambar 5.8 Proses instalasi Mosquitto lanjutan.....	31
Gambar 5.9 <i>File dependencies</i> dari Mosquitto pada <i>folder directory</i> Mosquitto	31
Gambar 5.10 Mosquitto berhasil dijalankan	32
Gambar 5.11 Tampilan ketika <i>subscriber</i> menerima pesan dari <i>broker</i>	37
Gambar 6.1 <i>Publisher</i> melakukan proses <i>publish</i> topik.....	39
Gambar 6.2 Tampilan awal <i>web subscriber</i>	39
Gambar 6.3 Proses <i>publish</i> dari <i>publisher</i>	40
Gambar 6.4 Hasil pengujian tanpa ada wadah penampung air.....	40
Gambar 6.5 Hasil pengujian dengan wadah penampung berisi air	40
Gambar 6.6 <i>Broker</i> menerima pesan <i>publish</i> dari <i>publisher</i>	41
Gambar 6.7 Proses pengiriman pesan oleh <i>broker</i>	41
Gambar 6.8 <i>Output</i> pada <i>web subscriber</i>	42
Gambar 6.9 Grafik kesalahan pengukuran sensor	44
Gambar 6.10 <i>Thread group</i> pada JMeter.....	46
Gambar 6.11 Konfigurasi <i>sampler</i> MQTT <i>publisher</i>	46
Gambar 6.12 <i>Publisher</i> melakukan proses <i>publish</i>	49
Gambar 6.13 Wireshark menampilkan isi pesan dari pesan <i>publisher</i> yang masuk ke IP <i>broker</i>	49

DAFTAR LAMPIRAN

LAMPIRAN A KODE SUMBER	55
A.1 <i>Publisher</i>	55
A.2 <i>Subscriber</i>	56
LAMPIRAN B HASIL PENGUJIAN	63
B.1 Skalabilitas	63



BAB 1 PENDAHULUAN

1.1 Latar belakang

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 Rachmad (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 Riny 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 (Adi, 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 Choi 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* nantinya akan dipasang di berbagai titik daerah endemik banjir, mengirim data menggunakan Protokol MQTT melalui jaringan ke *broker*, 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.

1.2 Rumusan masalah

Berdasarkan latar belakang, berikut adalah rumusan masalah yang tercakup pada penelitian ini:

1. Bagaimana konsep *Internet of Things* yang menggunakan protokol MQTT dalam jalur komunikasinya dapat diimplementasikan dalam pembangunan sebuah sistem *monitoring* banjir?
2. Bagaimana nilai parameter akurasi, integritas data dan skalabilitas pada implementasi konsep *Internet of Things* menggunakan protokol MQTT pada pembangunan sebuah sistem *monitoring* banjir?

1.3 Tujuan

Tujuan dari penelitian ini adalah:

1. Mengimplementasikan konsep *Internet of Things* pada pembangunan sebuah sistem *monitoring* banjir guna meningkatkan pemahaman terhadap konsep *Internet of Things*, terutama dalam merancang suatu sistem pengamatan ketinggian air dengan mengimplementasikan protokol MQTT dalam jalur komunikasinya.
2. Menguji nilai parameter akurasi, integritas data dan skalabilitas pada implementasi konsep *Internet of Things* dengan menggunakan protokol MQTT pada sistem *monitoring* banjir.

1.4 Manfaat

Manfaat dari penelitian ini adalah:

1. Untuk umum dapat membantu pengembangan pada penelitian terkait topik *Internet of Things* dan protokol MQTT serta menjadi solusi terkait sistem *monitoring* banjir.
2. Untuk peneliti dapat memahami sistem konsep *publish/subscribe* dalam sistem *monitoring* banjir menggunakan protokol MQTT.

1.5 Batasan masalah

Batasan masalah pada penelitian ini antara lain yaitu:

1. Implementasi dilakukan dengan melakukan *publish* dan *subscribe* pada protokol MQTT.
2. *Message Broker* menggunakan Mosquitto.
3. Sensor yang digunakan adalah sensor HC-SR04 dan DHT11
4. Implementasi yang dilakukan untuk mengukur tingkat akurasi sensor, skalabilitas dan integritas data.

1.6 Sistematika pembahasan

BAB I : PENDAHULUAN

Bab ini membahas tentang latar belakang, rumusan masalah penelitian, tujuan dari penelitian, manfaat dari penelitian, batasan masalah dari penelitian dan sistematika dari penelitian.

BAB II : LANDASAN KEPUSTAKAAN

Bab ini membahas tentang dasar teori dan kajian pustaka yang menjadi landasan dalam implementasi metode konsep publish/subscribe pada sistem monitoring banjir menggunakan protokol MQTT.

BAB III : METODOLOGI

Bab ini menjelaskan tentang metode yang digunakan pada penelitian ini dan langkah penerapannya yang diantaranya adalah studi literatur, perancangan simulasi, implementasi, pengujian, hasil dan pembahasan, dan kesimpulan.

BAB IV : REKAYASA KEBUTUHAN

Bab ini menjelaskan tentang deskripsi sistem dan kebutuhan dari sistem pada penelitian ini yang terbagi menjadi kebutuhan perangkat keras, kebutuhan perangkat lunak, kebutuhan fungsional dan kebutuhan non-fungsional.

BAB V : PERANCANGAN DAN IMPLEMENTASI

Bab ini menjelaskan tentang perancangan dan implementasi dari sistem pada penelitian ini. Perancangan dijelaskan mulai dari gambaran umum sampai perancangan komponen sistem. Implementasi menjelaskan implementasi yang dilakukan pada masing-masing komponen sistem.

BAB VI : PENGUJIAN DAN ANALISIS

Bab ini menjelaskan tentang pengujian dan analisa hasil pengujian dari sistem pada penelitian ini. Pengujian dilakukan dengan beberapa skenario untuk mengetahui kebutuhan fungsional dan non-fungsional sistem telah terpenuhi.

BAB VII : PENUTUP

Bab ini menjelaskan tentang kesimpulan dan saran dari sistem berdasarkan pelaksanaan penelitian dari proses perancangan, implementasi, dan pengujian.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Penelitian serupa telah dilakukan oleh Budiarmo (2011) dengan judul penelitian, “Sistem *Monitoring* Tingkat Ketinggian Air Bendungan Berbasis Mikrokontroler”. Penulis menyatakan bahwa pemberian informasi sedini mungkin sangat dibutuhkan. Sistem *monitoring* sebuah bencana memiliki tujuan untuk memberikan informasi mengenai bencana kepada masyarakat/pihak yang terkait yang nantinya informasi tersebut dapat menjadi acuan dalam mempersiapkan diri dan meminimalisir korban jiwa dengan memberikan informasi sedini mungkin akan adanya bencana yang mungkin dapat terjadi.

Penelitian berikutnya dilakukan oleh Budioko (2016) dengan judul, “Sistem *Monitoring* Suhu Jarak Jauh berbasis *Internet of Things* Menggunakan Protokol MQTT”. Penulis memiliki tujuan menggunakan protokol MQTT dalam sistem monitoring suhu jarak jauh skala besar (global dan lokal). Protokol MQTT merupakan protokol pesan ringan dan sederhana, menggunakan arsitektur *publish/subscribe* yang dirancang secara terbuka, mudah dalam pengimplementasiannya, meminimalkan *bandwidth* jaringan dan kebutuhan sumber daya perangkat ketika mencoba untuk menjamin kehandalan dan pengiriman, serta dirasa cocok dalam komunikasi *Machine to Machine* (M2M).

Penelitian selanjutnya dilakukan oleh Supriyadi (2011) dengan judul, “Penggunaan Sensor Ultrasonik sebagai Pendeteksi Ketinggian Air Sungai pada Sistem Peringatan Dini Tanggap Darurat Bencana Banjir”. Penelitian penulis berfokus pada bencana banjir, yang merupakan bencana yang sering melanda Indonesia, sehingga mengakibatkan korban jiwa dan harta benda yang cukup besar secara statistik dan merupakan bencana yang paling sering terjadi di Indonesia. Penulis bertujuan merancang dan merealisasikan sistem *early warning* bencana banjir air sungai yang menggunakan sensor tak sentuh dengan transduser ultrasonik, mampu mengirim data hasil pengukuran jarak jauh dan dapat menggambarkan kondisi *level* air sungai yang terjadi.

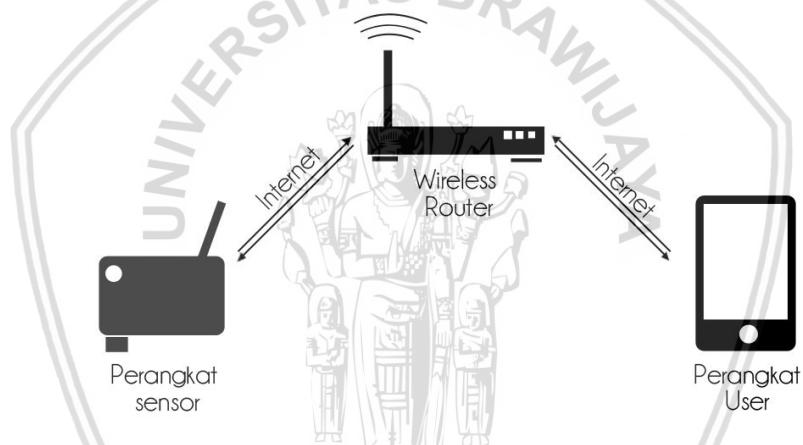
2.2 *Internet of Things*

Internet of Things (IoT) adalah suatu konsep mengenai terhubungnya benda-benda di sekitar kita dengan suatu jaringan. IoT mengacu pada interkoneksi jaringan benda sehari-hari, yang sering dilengkapi dengan kecerdasan (Xia, 2012). IoT merupakan sebuah konsep komputasi yang menggambarkan masa depan dimana setiap obyek fisik dapat terhubung dengan internet dan dapat mengidentifikasi dengan sendirinya antar perangkat yang lain (Sulistyanto et al., 2015).

IoT adalah arsitektur informasi berbasis Internet yang muncul untuk memfasilitasi pertukaran barang dan jasa. IoT bertujuan untuk menyediakan pertukaran “*things*” dengan cara yang aman dan dapat diandalkan, salah satu fungsinya yaitu mengatasi kesenjangan antara obyek dunia fisik dan representasi

nya dalam sebuah sistem informasi. Dalam perkembangannya, IoT kerap digunakan pada suatu sistem *monitoring* berbasis jaringan sebagai jalur komunikasinya. Dengan menggunakan IoT, sebuah sistem *monitoring* dapat digunakan banyak pengguna di berbagai tempat dalam satu waktu, sehingga dapat meningkatkan *availability* dan *accessibility* dari suatu sistem. IoT dapat menghubungkan berbagai objek dan bertukar data. Semakin banyak objek dan perangkat yang dibuat dengan kemampuan sensor dan *wi-fi*, di saat yang sama internet semakin mutakhir dan biaya teknologi semakin kecil, membuat IoT semakin mudah dijangkau (Microsoft.com).

IoT dapat digunakan pada berbagai perangkat yang berbasis sensor dan *wi-fi*, sehingga sistem *monitoring* suatu objek akan semakin mudah di akses dari berbagai tempat melalui internet. Hal tersebut dapat menunjang suatu sistem *monitoring* yang memiliki objek bencana alam seperti banjir. Dengan menggunakan suatu sistem *monitoring* berbasis IoT, penanganan dan penanggulangan banjir dapat lebih tanggap serta proses memperkecil korban materi dan jiwa dapat lebih mudah dan cepat.



Gambar 2.1 Metode komunikasi sistem monitoring banjir dengan IoT

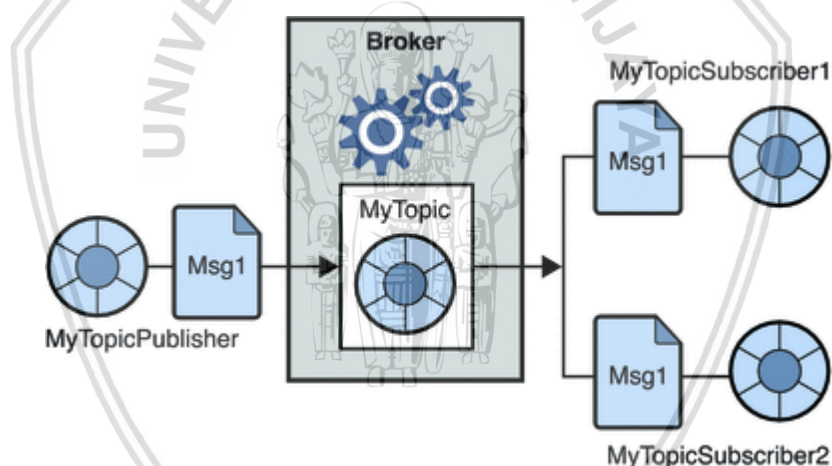
Sistem *monitoring* banjir yang dibuat pada penelitian ini menggunakan IoT sebagai konsep dasar dalam jalur komunikasi pertukaran data yang dilakukan perangkat pengukur dengan perangkat yang digunakan oleh pengguna. Dengan menggunakan konsep IoT, maka sistem memerlukan sebuah protokol komunikasi yang tepat dalam menyalurkan proses pertukaran data dari perangkat pengukur ke perangkat yang digunakan oleh pengguna dalam mengakses sistem *monitoring* ini. Protokol komunikasi yang digunakan adalah Message Queue Telemetry Transport (MQTT).

2.3 Message Queue Telemetry Transport

MQTT atau Message Queue Telemetry Transport adalah protokol yang dibuat oleh Dr. Andy Stanford-Clark dari IBM. MQTT adalah protokol sederhana dan ringan, dirancang untuk perangkat yang terbatas dan *bandwith* rendah, *latency* tinggi dan jaringan yang tidak reliabel. Prinsip desainnya adalah meminimalisir *bandwith* jaringan dan kebutuhan sumber daya perangkat dan memastikan keandalan dengan beberapa tingkat kepastian pengiriman pesan. Prinsip-prinsip

ini membuat MQTT merupakan protokol yang tepat untuk *Machine to Machine* (M2M). (MQTT.org, 2017). Protokol MQTT merupakan protokol pesan sederhana, ringan, mudah diimplementasikan dan mampu menangani ribuan *client* jarak jauh dengan hanya satu server (Budioko, 2016).

Protokol MQTT menggunakan konsep *publish/subscribe* dalam komunikasinya. *Publish/subscribe* adalah sebuah pola pertukaran pesan dimana pengirim pesan disebut *publisher* dan penerima pesan disebut *subscriber* (Rochman et al., 2017). Protokol MQTT merupakan protokol berbasis topik, yang menggunakan karakter *string* untuk menyediakan layanan topik secara hirarkis. Protokol MQTT juga memiliki fasilitas *subscribe* dengan banyak topik. MQTT menyediakan layanan berbasis *end-to-end Quality of service*, tergantung seberapa prioritas keandalan suatu pesan untuk sampai ke *client*, MQTT menyediakan 3 level QOS. QOS level 0 berarti pesan dikirim minimal 1 kali, QOS level 1 berarti pesan dikirim 1 kali dan menunggu *acknowledge* dari sisi *client*, lalu kembali mengirim pesan selanjutnya, dan QOS level 2 berarti pesan dikirim 1 kali, menunggu *acknowledge* dari sisi *client*, jika pesan tidak sampai maka *server* akan mengirim pesan kembali sampai *acknowledge* dari sisi *client* menyatakan telah diterima (Hunkeler, 2008)



Gambar 2.2 Metode *publish/subscribe* dalam MQTT

Sumber: docs.oracle.com

Oleh karena karakteristik MQTT tersebut, MQTT sering dipakai pada perangkat *mobile* dan sensor berbasis IoT dimana penggunaan sumber daya dan jaringan sangat diminimalisir, seperti pada sistem *monitoring* banjir yang dibuat pada penelitian ini. Dalam sistem *monitoring* yang dibuat pada penelitian ini, MQTT berperan sebagai pengatur/protokol dalam metode aliran data dalam sistem komunikasi yang digunakan. MQTT berperan sebagai gerbang yang mengatur aliran data yang masuk dan keluar. Sistem *monitoring* yang dibuat dalam penelitian ini memiliki perangkat sensor yang tersebar dan ditempatkan di titik-titik endemik banjir. MQTT digunakan sebagai protokol komunikasi yang mengirimkan data dari perangkat sensor menuju perangkat yang digunakan oleh

pengguna. Setiap sensor memiliki identitas *publisher* yang unik dan mengirimkan topik yang berbeda antara *publisher* satu dan lainnya. Dalam proses pertukaran data antara *publisher* dan *subscriber*, protokol MQTT membutuhkan sebuah perantara yang disebut *broker*. *Broker* MQTT yang digunakan sebagai perantara di sistem dalam penelitian ini adalah *Broker* Mosquitto.

2.4 Broker

Dalam konsep *publish/subscribe*, *publisher* dan *subscriber* tidak terhubung secara langsung. *Publisher* mengirim pesan kepada *subscriber* melalui *broker*. *Broker* berfungsi seperti *server*, dengan tugas sebagai perantara *publisher* dan *subscriber* untuk melanjutkan pesan dari *publisher* ke *subscriber* dan mengelola *subscription* dari *subscriber* (Tarigan, 2014). *Broker* memiliki peranan sebagai suatu jembatan dan portal yang bertugas mengatur aliran data yang masuk dan keluar antara *publisher* dan *subscriber*. *Broker* mengatur aliran data antara *publisher* dan *subscriber* berdasarkan topik yang diperlukan oleh *subscriber*. Ketika *publisher* mengirimkan suatu data dengan topik yang sesuai dengan topik yang dibutuhkan oleh *subscriber*, *broker* akan langsung meneruskan data tersebut ke *subscriber*. Sebaliknya, ketika *publisher* mengirimkan data dengan topik yang tidak dibutuhkan oleh *subscriber*, *broker* tidak akan meneruskan data tersebut ke *subscriber*.

Dalam penelitian ini, *broker* yang digunakan adalah Mosquitto. Mosquitto adalah *broker* pesan *open source* (berlisensi EPL / EDL) yang menerapkan protokol MQTT versi 3.1 dan 3.1.1. MQTT menyediakan metode ringan untuk melakukan olah pesan menggunakan model *publish/subscribe*. Hal ini membuatnya cocok untuk pesan *Internet of Things* seperti sensor daya rendah atau perangkat *mobile* seperti telepon, mikrokontroler atau mikrokomputer seperti Arduino dan Raspberry. (mosquitto.org).

Broker Mosquitto digunakan di sistem dalam penelitian ini berdasarkan penelitian "Implementasi Metode *Failover* pada *Broker* Protokol MQTT Dengan ActiveMQ" (Rakhman et al., 2018) yang membandingkan *broker* Mosquitto dengan ActiveMQ. Penelitian tersebut menghasilkan bahwa *broker* Mosquitto memiliki tingkat penggunaan sumber daya yang lebih rendah dibandingkan dengan *broker* ActiveMQ.

2.5 Sensor Ultrasonik

Sensor Ultrasonik adalah sensor yang menggunakan gelombang ultrasonik untuk mengukur suatu jarak. Metode umum dalam pengukuran jarak oleh sensor ultrasonik adalah sebagai berikut: gelombang ultrasonik dipancarkan dari sensor ultrasonik. Gelombang tersebut merambat melalui medium transmisi (udara), dan terpantulkan oleh permukaan di depannya. Gelombang tersebut kembali dan diterima transduser. Waktu tempuh dari gelombang tersebut proposional dalam pengukuran jarak, dengan syarat kecepatan suaranya diketahui. (Webster, D., 1994)

Pada penelitian yang dilakukan oleh Velasco, S., et al. (2003), kecepatan gelombang suara dalam medium transmisi udara bergantung kepada temperatur pada udara tersebut. Penelitian tersebut menguji kecepatan suara pada beberapa variasi temperatur udara dan dihasilkan persamaan sebagai berikut:

$$c_s(m/s) = (331 \pm 3 m/s) + (0.60 \pm 0.06 \frac{m/s}{^\circ C})t \quad (2.1)$$

Dimana c_s adalah konstanta kecepatan suara dalam m/s dan t adalah temperature dalam $^\circ C$.

2.6 Akurasi dan Presisi Sensor

Akurasi adalah seberapa dekat hasil yang diperoleh dari suatu pengukuran dengan nilai sebenarnya atau nilai yang dianggap benar (*accepted value*) (Hanifah, 2016). Pengukuran tingkat akurasi tidak menghasilkan kuantitas atau nilai kuantitas numerik, dan pengukuran yang dilakukan dapat dinyatakan lebih akurat ketika nilai tingkat akurasi yang dihasilkan memiliki nilai pengukuran kesalahan yang lebih kecil (*International Vocabulary of Metrology*, 2008).

Dalam suatu sistem yang melakukan pengukuran terhadap suatu objek, tingkat akurasi menjadi salah satu aspek yang harus diperhitungkan. Akurasi sensor adalah selisih maksimum antara nilai aktual dan nilai yang ditunjukkan oleh sensor. Tingkat akurasi dapat dinyatakan dengan menggunakan persentase dari skala sepenuhnya ataupun secara absolut. (Carr, 1998).

Presisi adalah tingkat kedekatan antara nilai hasil pengukuran yang dilakukan dengan pengukuran yang berulang pada kondisi yang ditentukan sebelumnya. Pengukuran presisi biasanya menghasilkan angka yang di dapat dari mengukur impresi, seperti standar deviasi, variasi atau koefisien dari variasi pada kondisi yang ditentukan sebelumnya. (*International Vocabulary of Metrology*, 2008).

Dalam sistem yang dibuat dalam penelitian ini, akurasi dan presisi dijadikan salah satu parameter dalam pengujian sistem karena dalam suatu sistem yang melakukan pengukuran terhadap suatu objek, tingkat akurasi menjadi salah satu aspek yang harus diperhitungkan (Carr, 1998).

2.7 Skalabilitas

Skalabilitas adalah salah satu atribut yang diperlukan dalam suatu sistem yang memiliki banyak pengguna. Skalabilitas adalah kemampuan sistem dalam mengakomodasi peningkatan jumlah elemen atau objek, untuk memproses volume permintaan yang meningkat atau rentan terhadap peningkatan permintaan yang pesat. Sebuah sistem memiliki skalabilitas yang baik jika dapat berfungsi dengan baik, yaitu tanpa penundaan yang tidak semestinya. (Bondi, 2000).

Dalam suatu sistem yang digunakan umum, komunikasi yang taktis harus berjalan lancar dimana setiap pesan yang dikirimkan harus tersampaikan ke penerima. Oleh karena itu, dibutuhkan skalabilitas yang tinggi. Skalabilitas juga di definisikan sebagai kemampuan suatu sistem dalam menangani perubahan suatu

jaringan seperti jumlah node dan topologi area yang semakin meluas. (Agustini, 2017).

Dalam suatu sistem berbasis *Internet of Things*, skalabilitas adalah sebuah parameter yang dibutuhkan untuk mengetahui seberapa besar kapasitas sistem untuk menangani proses-proses ketika terjadi perubahan menjadi lebih besar dari sebelumnya. (Rozi et al., 2017)

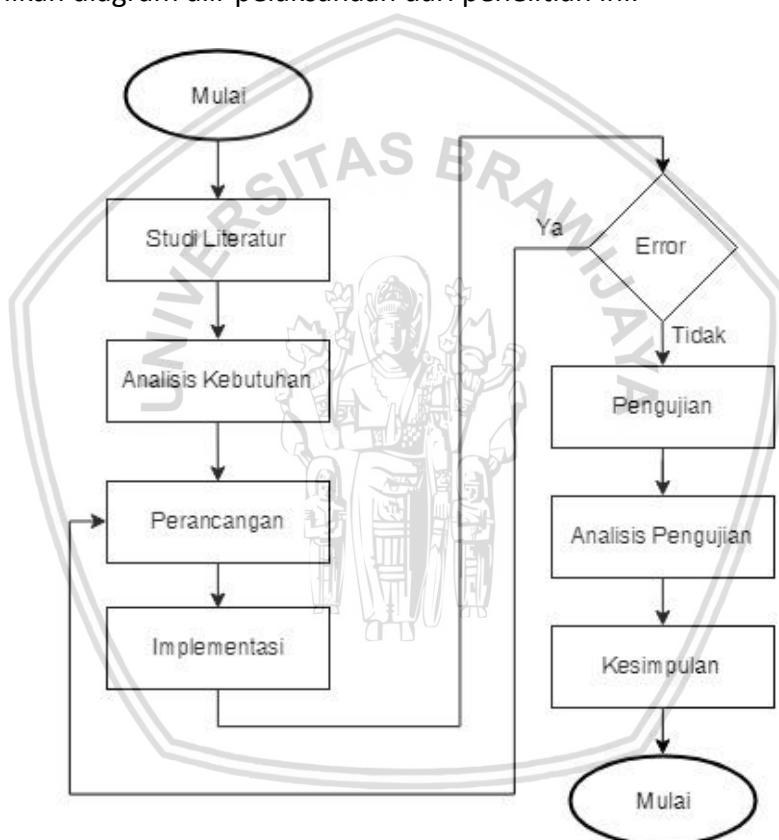
2.8 Integritas Data

Integritas data adalah keakuratan dan konsistensi data yang tersimpan, ditunjukkan dengan tidak adanya perubahan data antara dua *update* data yang tersimpan. Integritas data diperlukan suatu sistem pada tahap perancangannya melalui penggunaan peraturan dan prosedur standar, dan dikelola melalui penggunaan pengecekan kesalahan dan rutinitas validasi. (*Data Integrity Seminar*, 2014).

Dalam suatu proses pengiriman atau penyimpanan data, seringkali terdapat resiko perubahan yang tidak diinginkan terhadap data. Hal tersebut seringkali terjadi pada level fisik ataupun media saluran yang digunakan, yang disebabkan karena gangguan (*noise*) pada proses penyimpanan atau pengiriman data itu sendiri. Agar dapat mendeteksi kerusakan data tersebut, diperlukan suatu cara untuk menghitung suatu nilai terhadap data yang diberikan dengan nilai-nilai tersebut dikirim bersama-sama untuk dicek penerima apakah data yang diterima sama dengan aslinya. (Wijayanto, 2007).

BAB 3 METODOLOGI

Bab ini akan membahas mengenai tahapan-tahapan yang digunakan dalam pembuatan sistem “Implementasi Konsep *Internet of Things* pada Sistem *Monitoring Banjir* menggunakan protokol MQTT”. Langkah pertama yang akan dilakukan adalah mengumpulkan teori-teori pendukung sebagai studi literatur. Selanjutnya akan dilakukan analisis dan pembuatan rancangan simulasi yang tepat dan sesuai dengan hasil analisis. Setelah analisis dan rancangan simulasi selesai, akan dilakukan implementasi simulasi. Setelah itu akan dilakukan pengujian sistem dan analisis dari hasil pengujian tersebut. Sebagai langkah terakhir, akan dibuat kesimpulan dan saran sebagai dasar untuk penelitian selanjutnya. Gambar 3.1 menampilkan diagram alir pelaksanaan dari penelitian ini.



Gambar 3.1 Diagram Alir Pelaksanaan Penelitian

3.1 Studi Literatur

Dalam langkah ini dilakukan pembelajaran teori-teori yang mendukung penelitian ini dalam rangka memperluas pengetahuan dalam pembuatan sistem di penelitian ini. Beberapa teori yang dibahas antara lain:

1. *Publish/subscribe*
2. MQTT (Message Queue Telemetry Transport)
3. *Broker Mosquitto*

4. Akurasi sensor
5. Skalabilitas
6. Integritas Data

Studi literatur diperlukan karena dalam sebuah penelitian diperlukan pengkajian dan penggunaan metode dari penelitian terkait yang telah dilakukan sebelumnya sehingga proses penelitian dapat lebih tepat dan minim kesalahan. Dalam studi literatur yang dilakukan, referensi diambil dari beberapa penelitian yang terkait dengan penelitian ini, yakni paper, jurnal, buku maupun *website* yang terpercaya.

3.2 Analisis Kebutuhan

Analisis kebutuhan adalah elemen-elemen atau bagian-bagian dasar dari suatu penelitian yang harus dipenuhi sebelum penelitian dilakukan lebih mendalam, guna sebagai pedoman dan memperkuat dasar teori yang telah disampaikan. Dalam penelitian ini, elemen yang harus dipenuhi adalah informasi mengenai data yang dihasilkan dari sensor pengukur ketinggian air.

3.2.1 Kebutuhan Sistem

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

- a. Perangkat sensor pada *publisher* mampu mengukur ketinggian air.
- b. *Publisher* mampu mengirimkan pesan ke *broker* melalui *wireless network* dengan jaringan *wi-fi*.
- c. *Broker* mampu menerima pesan yang dikirimkan oleh *publisher* melalui *wireless network* dengan jaringan *wi-fi*.
- d. *Subscriber* mampu melakukan proses *subscribe* pada topik yang diinginkan ke *broker*.
- e. *Broker* mampu mengirimkan pesan ke *subscriber* sesuai dengan topik yang diinginkan.

Tabel 3.1 Spesifikasi perangkat keras

Nama Komponen	Spesifikasi
Prosesor	Intel® Celeron® CPU 1007 @ 1.50 GHz
Memory (RAM)	2.0 GB DDR 3
Harddisk	465GB Hitachi HGST HTS545050A7E680 (SATA)
Sensor	HC-SR04 DHT11
Mikrokomputer	Raspberry Pi 3

Spesifikasi perangkat keras yang digunakan dalam penelitian ini dapat dilihat pada Tabel 3.1. Dalam penelitian ini, penulis menggunakan sensor pengukur jarak HC-SR04. Sensor ini dipilih karena pada sebuah sistem yang melakukan pengukuran jarak dalam skala besar, sensor HC-SR04 memiliki tingkat akurasi yang cukup baik. Jika dibandingkan dengan sensor jarak HRLV-MaxSonar-EZ yang memiliki tingkat akurasi diatas HC-SR04 yaitu mencapai 1 mm, HC-SR04 dianggap lebih tepat digunakan pada sistem ini dikarenakan pengukuran ketinggian banjir tidak membutuhkan pengukuran yang akurasinya mencapai 1 mm.

Tabel 3.2 Spesifikasi perangkat lunak

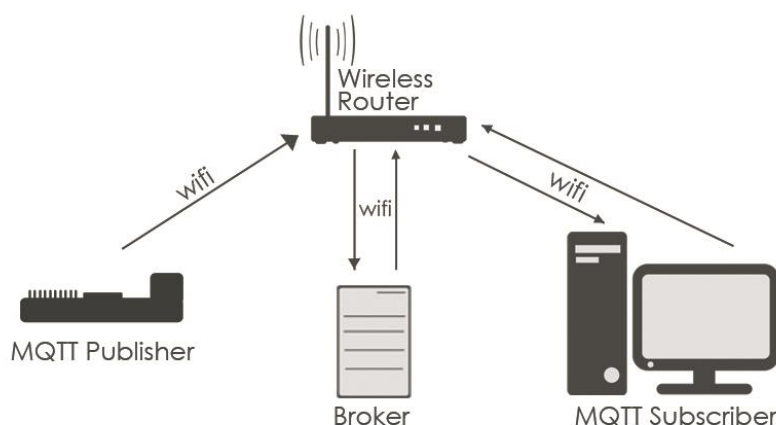
Nama Komponen	Spesifikasi
Sistem Operasi	Windows Embedded 8.1 Industry Pro Raspbian Jessie
<i>Broker</i>	Mosquitto 1.4.7
Editor Pemrograman	Notepad++ 7.5.1 Python 2.7
<i>Testing Tool</i>	Apache JMeter 3.3 Wireshark 2.4.2

Spesifikasi perangkat lunak pada penelitian ini ditampilkan pada Tabel 3.2. Raspberry Pi 3 digunakan sebagai mikrokomputer dalam penelitian ini karena Raspberry Pi 3 merupakan mikrokomputer yang memiliki *built-in wireless connection* sehingga dalam proses pengukuran ketinggian air akan lebih mudah karena tidak perlukan perangkat jaringan tambahan dalam melakukan koneksi ke broker. Berbeda dengan mikrokontroler yang juga sering digunakan dalam implementasi *Internet of Things* lainnya, yaitu Arduino yang memerlukan perangkat dan konfigurasi tambahan ketika mikrokomputer akan melakukan *wireless connection*.

Penggunaan *Broker* Mosquitto pada penelitian ini berdasarkan hasil penelitian yang membandingkan *Broker* Mosquitto dengan *broker* MQTT lainnya yaitu ActiveMQ terkait performa. Dihilangkan bahwa *broker* Mosquitto memiliki performa yang lebih tinggi.

3.3 Perancangan Sistem

Implementasi Konsep *Internet of Things* pada Sistem *Monitoring* Banjir menggunakan Protokol MQTT memerlukan perancangan yang tepat dan efisien dalam penerapannya. Oleh karena itu peneliti membuat perancangan ini dengan tepat dan efisien serta menggunakan ilustrasi perancangan seperti yang ditampilkan pada Gambar 3.2 sebagai pedoman dalam perancangan sistem yang akan dibuat.



Gambar 3.2 Ilustrasi Perancangan

A. MQTT Publisher

MQTT *Publisher* adalah sensor dan mikrokomputer yang berfungsi untuk mengirimkan pesan ke *broker*. Sensor berfungsi mengambil data yang diperlukan (ketinggian air) dan mikrokomputer berfungsi untuk mengolah data tersebut agar dapat dikirimkan melalui *wireless router* dengan jaringan *wi-fi* ke MQTT *broker*. MQTT *Publisher* pada sistem ini dijalankan pada mikrokomputer karena pada implementasinya MQTT *Publisher* akan menerima nilai yang dihasilkan sensor yang diletakkan pada daerah-daerah endemik banjir, sehingga dengan menggunakan mikrokomputer yang merupakan perangkat dengan ukuran dan penggunaan sumber daya yang kecil merupakan pilihan yang cocok dan fleksibel dalam instalasinya.

B. Wireless Router

Wireless Router berfungsi sebagai jalur penghubung dari ketiga komponen utama dalam sistem ini, yaitu MQTT *Publisher*, MQTT *Broker* dan MQTT *Subscriber*. *Wireless Router* digunakan karena dalam proses menjalankan sistem ini akan memiliki banyak *publisher* pada implementasinya. Jika dibandingkan dengan *router* yang tidak *wireless* yaitu menggunakan kabel, *wireless router* dapat memudahkan penerapan dan menekan biaya produksi.

C. Broker

Broker berfungsi sebagai penerima pesan yang dikirimkan oleh MQTT *Publisher* dan sekaligus pihak yang akan mengirimkan pesan berupa topik yang diminta oleh MQTT *Subscriber*. *Broker* digunakan karena sebuah sistem MQTT memerlukan *broker* yang berfungsi sebagai *server* yang akan meneruskan data yang dikirimkan oleh MQTT *publisher* ke MQTT *subscriber*.

D. MQTT Subscriber

MQTT *Subscriber* adalah pihak yang akan melakukan *request* ke MQTT *Broker* dan akan menerima pesan yang dikirimkan oleh MQTT *Broker*. MQTT *Subscriber* pada sistem ini akan dijalankan pada sebuah laman *web*. Dalam proses komunikasi antara *broker* dengan *subscriber* dalam bentuk laman *web*, digunakan *websockets* yang mendukung komunikasi dengan laman *web*. Laman *web* dipilih karena pengguna dapat mengakses laman *web* dimana saja dan kapan saja.

3.4 Implementasi Sistem

Setelah semua data dan kebutuhan sistem telah terkumpul secara lengkap, maka Implementasi Konsep *Internet of Things* pada Sistem *Monitoring Banjir* menggunakan Protokol MQTT dapat dilakukan dengan data yang telah diperoleh sebagai objek. Implementasi sistem yang dilakukan akan meliputi beberapa rincian yaitu:

1. Implementasi proses pengukuran oleh sensor. Proses pengukuran dilakukan dengan menempatkan sensor di atas tabung pengukur, sehingga sensor akan mengukur jarak dari atas tabung pengukur ke permukaan air dibawahnya.
2. Implementasi proses pengiriman data dari mikrokomputer menuju *broker* menggunakan konsep *publish/subscribe* MQTT, dilakukan dengan menghubungkan mikrokomputer yang berperan sebagai *publisher* ke *broker* melalui jaringan *wireless* dan mengirimkan data berbasis topik melalui protokol MQTT menuju *broker* yang digunakan.
3. Konfigurasi pengiriman data dari *broker* menuju *subscriber* yang melakukan *subscribe* menggunakan konsep *publish/subscribe* MQTT.
4. *Subscriber* menerima topik yang di *subscribe* dari *broker* dan ditampilkan dalam bentuk *web* menggunakan konsep *publish/subscribe* MQTT. Untuk menciptakan komunikasi antara *broker* dan *subscriber* melalui *web*, digunakan protokol yang mendukung komunikasi dengan *web* yaitu *websockets*.

Implementasi sistem dilakukan guna menerapkan perancangan pada sistem yang telah dilakukan sebelumnya, sehingga penelitian tidak melenceng dari perancangan yang telah dilakukan sebelumnya.

3.5 Pengujian Sistem

Pengujian sistem dilakukan guna mengetahui sejauh mana implementasi sistem telah dijalankan dan untuk mengetahui tingkat performa dari sistem yang sedang diteliti. Pengujian akan menghasilkan data yang selanjutnya dapat digunakan sebagai dasar analisis.

Pengujian yang akan dilakukan pada sistem ini adalah sebagai berikut:

1. Akurasi
2. Skalabilitas
3. Integritas data

Dari poin-poin tersebut, akan dilakukan pengujian sistem dengan beberapa skenario sebagai berikut:

1. Pengujian dari sistem dengan MQTT *Publisher* dapat mengirimkan pesan ke MQTT *broker*. Pengujian ini dilakukan untuk mengetahui apakah MQTT *Publisher* dapat mengirimkan pesan ke MQTT *Broker*. Jika MQTT *Publisher* tidak dapat mengirimkan pesan ke MQTT *Broker*, maka sistem akan gagal.
2. Pengujian dari sistem dengan MQTT *Broker* dapat menerima pesan yang dikirimkan MQTT *Publisher*. Pengujian ini dilakukan untuk mengetahui apakah MQTT *Broker* dapat menerima pesan dari MQTT *Publisher*. Jika MQTT *Broker* tidak dapat menerima pesan dari MQTT *Publisher*, maka sistem akan gagal.
3. Pengujian dari sistem dengan MQTT *Subscriber* dapat melakukan *subscribe* dan menerima pesan dari *broker*. Pengujian ini dilakukan untuk mengetahui apakah MQTT *Subscriber* dapat melakukan proses *subscribe* dan menerima pesan dari MQTT *Broker*. Jika MQTT *Subscriber* tidak dapat melakukan proses *subscribe* atau menerima pesan dari MQTT *Broker*, maka sistem akan gagal.
4. Pengujian untuk menentukan akurasi sensor dengan mengukur nilai aktual dari ketinggian air dan dibandingkan dengan nilai pengukuran dari sensor. Pengujian dilakukan dengan melakukan pengukuran menggunakan sensor sebanyak 10 kali pada titik acuan yang berbeda-beda. Lalu pada pengukuran tingkat presisi, pengujian dilakukan dengan melakukan pengukuran menggunakan sensor sebanyak 10 kali pada kondisi dan titik acuan yang tetap. Pengujian ini bertujuan menghasilkan tingkat akurasi pada sistem.
5. Pengujian untuk menentukan skalabilitas dengan MQTT *Broker* dapat melayani banyak MQTT *Publisher*. Pengujian dilakukan dengan menggunakan 100 *publisher*, 250 *publisher* dan 500 *publisher* dan sebanyak 3 kali pada masing-masing pengujian. Pengujian dilakukan dengan menggunakan *thread* yaitu pengiriman proses *publish* ke *broker* dalam jumlah yang telah ditentukan pada satu waktu. Pengujian ini bertujuan menghasilkan tingkat skalabilitas pada sistem.
6. Pengujian untuk menentukan Integritas data dengan membandingkan data yang dikirim oleh *publisher* dengan data yang diterima oleh *broker*. Pengujian dilakukan sebanyak 3 kali dengan perbedaan *interval* pengiriman pada setiap pengiriman data masing-masing 10 ms, 100 ms dan 100 ms. Pengujian ini bertujuan menghasilkan tingkat integritas data pada sistem.

3.6 Penarikan Kesimpulan

Setelah tahapan-tahapan dalam metodologi telah diselesaikan secara menyeluruh, maka dapat ditarik kesimpulan dari data-data yang diperoleh dari tahapan sebelumnya sebagai acuan dan referensi. Mengapa protokol MQTT dipilih sebagai metode pengiriman data pada sistem monitoring banjir. Kesimpulan akan di validasi sesuai dengan data-data yang telah diperoleh dan diuji pada penelitian.

BAB 4 REKAYASA KEBUTUHAN

4.1 Deskripsi Umum

Pada bab ini akan dijelaskan mengenai hal – hal yang harus dipenuhi untuk melakukan perancangan hingga implementasi. Dengan harapan perancangan dan implementasi pada penelitian ini dapat berjalan dengan baik.

4.1.1 Perspektif Sistem

Sistem ini dapat dinyatakan bekerja apabila semua komponen dari sistem berjalan dengan baik. Mulai dari komponen *publisher*, yaitu sensor dan mikrokomputer, *broker* pada PC/Laptop sampai *subscriber*, yaitu halaman *web*. Sensor bekerja dengan mengambil data dari ketinggian air di suatu lokasi, lalu informasi tersebut diolah pada mikrokomputer dan dikirim melalui sebuah jaringan menuju *broker*. *Broker* menerima dan menyimpan informasi tersebut lalu mengirim informasi tersebut ke *subscriber* yang melakukan *subscribe*. *Subscriber* akan menerima informasi tersebut dalam sebuah halaman *web* yang diakses. Mekanisme komunikasi pada sistem ini menggunakan protokol MQTT, sehingga *publisher* dan *subscriber* tidak terhubung secara langsung.

4.1.2 Tujuan

Bab ini bertujuan untuk menjelaskan sistematis penelitian ini secara rinci. Mulai dari deskripsi umum, alat dan komponen yang diperlukan, proses berjalannya sistem, antarmuka serta kebutuhan fungsional dan non-fungsional dari sistem. Dokumentasi ini bertujuan agar pengembangan sistem dapat lebih fokus dan terarah.

4.1.3 Ruang Lingkup

Sistem ini bertujuan untuk mengetahui alur dan proses dari implementasi konsep *publish subscribe* menggunakan protokol MQTT dalam sistem *monitoring* banjir pada lingkungan kerja yang nyata. Sistem akan mulai bekerja dengan sensor melakukan pengukuran lalu mikrokomputer akan mengolah data tersebut dengan kemudian mengirimkan data tersebut ke *broker* dan sampai kepada pengguna yang mengakses suatu halaman *web* tertentu. Metode komunikasi yang digunakan adalah *publish/subscribe* dimana protokol yang digunakan adalah MQTT. Pembuatan dan pengembangan sistem ini ditujukan kepada penguji, pengembang dan pengguna sehingga dalam beberapa waktu ke depan penelitian ini dapat dikembangkan dalam aspek yang lebih luas seperti *Internet of Things*, penanggulangan dan pencegahan banjir maupun penelitian lainnya yang lebih kompleks.

4.1.4 Batasan Sistem

Batasan sistem untuk perancangan dan implementasi pada penelitian ini antara lain:

1. Sistem akan mulai bekerja ketika *broker* telah menerima data sensor dari mikrokomputer.
2. Sensor dan mikrokomputer dapat bekerja bila memiliki daya yang cukup.
3. *Publisher*, *Broker* dan *Subscriber* dapat berkomunikasi apabila terhubung dalam satu jaringan yang sama.

4.1.5 Lingkungan Operasi

Syarat kebutuhan lingkungan untuk mendukung kebutuhan sistem ini adalah sebagai berikut:

1. Jarak air dari sensor tidak boleh kurang dari 2cm dan tidak lebih dari 400cm agar sensor dapat mengukur ketinggian air.
2. Kondisi sensor tidak boleh basah agar tidak mengganggu kinerja dari sensor.

4.1.6 Asumsi dan Ketergantungan

1. Sensor dapat melakukan proses pengukuran ketinggian air.
2. *Broker*, *Publisher* dan *Subscriber* harus terhubung dalam satu jaringan yang sama.

4.2 Kebutuhan Sistem

Bagian ini menjelaskan kebutuhan yang diperlukan oleh sistem agar dapat bekerja. Kebutuhan terkait yang akan dijabarkan adalah kebutuhan perangkat keras, kebutuhan perangkat lunak serta kebutuhan fungsional sistem.

4.2.1 Kebutuhan Perangkat Keras

1. Mikrokomputer Raspberry Pi 3 yang berfungsi sebagai kontroler untuk mengatur kerja sensor dan pengiriman data melalui modul *wireless network*.
2. Sensor HC-SR04 berfungsi untuk mengukur data ketinggian air.
3. Sensor DHT-11 berfungsi untuk mengukur temperatur yang digunakan sebagai penunjang sensor HC-SR04
4. Satu buah Komputer/Laptop yang berfungsi sebagai *broker*.
5. Satu buah Komputer/Laptop/Smartphone dengan *web browser* yang berfungsi sebagai *subscriber*.

4.2.2 Kebutuhan Perangkat Lunak

1. Python berfungsi untuk menjalankan program pada mikrokomputer.
2. *Broker* Mosquitto berfungsi untuk membangun sebuah MQTT *Broker*.
3. Apache JMeter berfungsi untuk alat uji dari skalabilitas sistem.
4. Notepad++ berfungsi untuk editor code html dan javascript.

4.2.3 Kebutuhan Fungsional

1. Fungsi mengoleksi data ketinggian air oleh sensor.
 - a. Penjelasan
Dalam fungsi ini, sistem harus dapat mengukur ketinggian air di suatu lokasi dan melakukan proses input data tersebut ke mikrokomputer.
 - b. Respon sistem
Data dari sensor yang berisi informasi mengenai ketinggian air yang telah diukur akan diolah dan ditampilkan oleh mikrokomputer.
2. Fungsi mengirimkan data *publisher* ke *broker*.
 - a. Penjelasan
Dalam fungsi ini *publisher* melakukan *publish* data sensor yang telah diberikan topik ke *broker* melalui protokol MQTT.
 - b. Respon sistem
Data sensor yang dikirimkan oleh *publisher* akan diterima oleh *broker* sebagai *payload* dari topik yang dikirim.
3. Fungsi *web subscriber* mengirimkan *request* untuk *subscribe* topik ke *broker*.
 - a. Penjelasan
Dalam fungsi ini *web* yang berperan sebagai *subscriber* harus dapat melakukan *subscribe* topik ke *broker*.
 - b. Respon sistem
Broker menampilkan *output* ketika *web* yang berperan sebagai *subscriber* tersebut melakukan *subscribe*.
4. Fungsi *broker* merespon *subscribe* dari *web subscriber* dengan mengirimkan *payload* dari topik yang di *request* oleh *web*, berisi data sensor menggunakan protokol MQTT.
 - a. Penjelasan
Fungsi ini mengharuskan *broker* dapat merespon *subscribe* topik dari *web subscriber* dengan mengirimkan *payload* yang berisi data sensor ke *web subscriber* menggunakan protokol MQTT.
 - b. Respon sistem
Broker menampilkan *output* ketika berhasil mengirimkan respon ke *web subscriber*. Data sensor yang diterima *subscriber* ditampilkan pada *web*.

4.2.4 Kebutuhan Non-Fungsional

1. Kebutuhan Akurasi
Kebutuhan non-fungsional akurasi adalah sistem dapat melakukan pengukuran yang akurat ketika hasil pengukuran sesuai dengan keadaan sebenarnya.
2. Kebutuhan Skalabilitas
Kebutuhan non-fungsional skalabilitas adalah sistem dapat bekerja dengan skalabilitas baik ketika proses keseluruhan sistem dapat menangani banyak *publisher* dengan tingkat keberhasilan yang tinggi

3. Kebutuhan Integritas Data

Kebutuhan non-fungsional integritas data adalah sistem dapat mengirimkan data yang berintegritas ke *subscriber*, yaitu ketika data yang dikirimkan oleh *publisher* sama dengan data yang diterima *subscriber*.

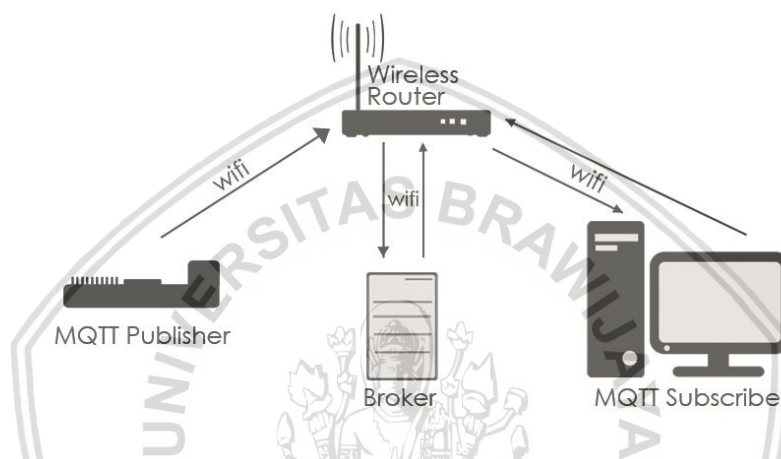


BAB 5 PERANCANGAN DAN IMPLEMENTASI

5.1 Perancangan Sistem

Bab ini menjelaskan mengenai perancangan dari sistem secara keseluruhan. Perancangan yang akan dijelaskan meliputi gambaran umum dari sistem, perancangan perangkat keras, perancangan perangkat lunak dan juga komponen-komponen yang terdapat pada sistem.

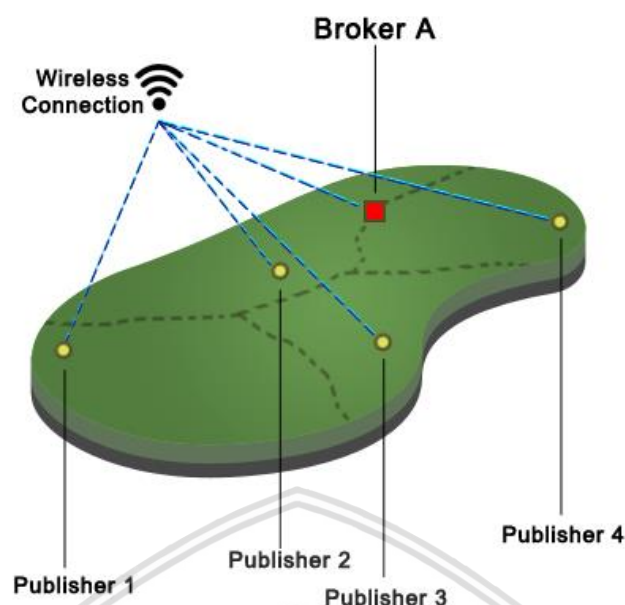
5.1.1 Gambaran Umum Sistem



Gambar 5.1 Gambaran Umum Sistem

Gambar 5.1 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 *wi-fi* yang disediakan *wireless router*. *Publisher* akan melakukan *publish message* yang berbasis topik dan memiliki tipe data *string* ke *broker*, lalu *broker* akan menerima *publish message* tersebut. *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 5.2 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, garis putus-putus adalah batas dari tiap-tiap daerah yang dapat ditangani oleh setiap *publisher* dan setiap *publisher* dan *broker* terhubung dengan *wireless connection*.



Gambar 5.2 Design Plan wilayah A

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

5.1.2 Publisher

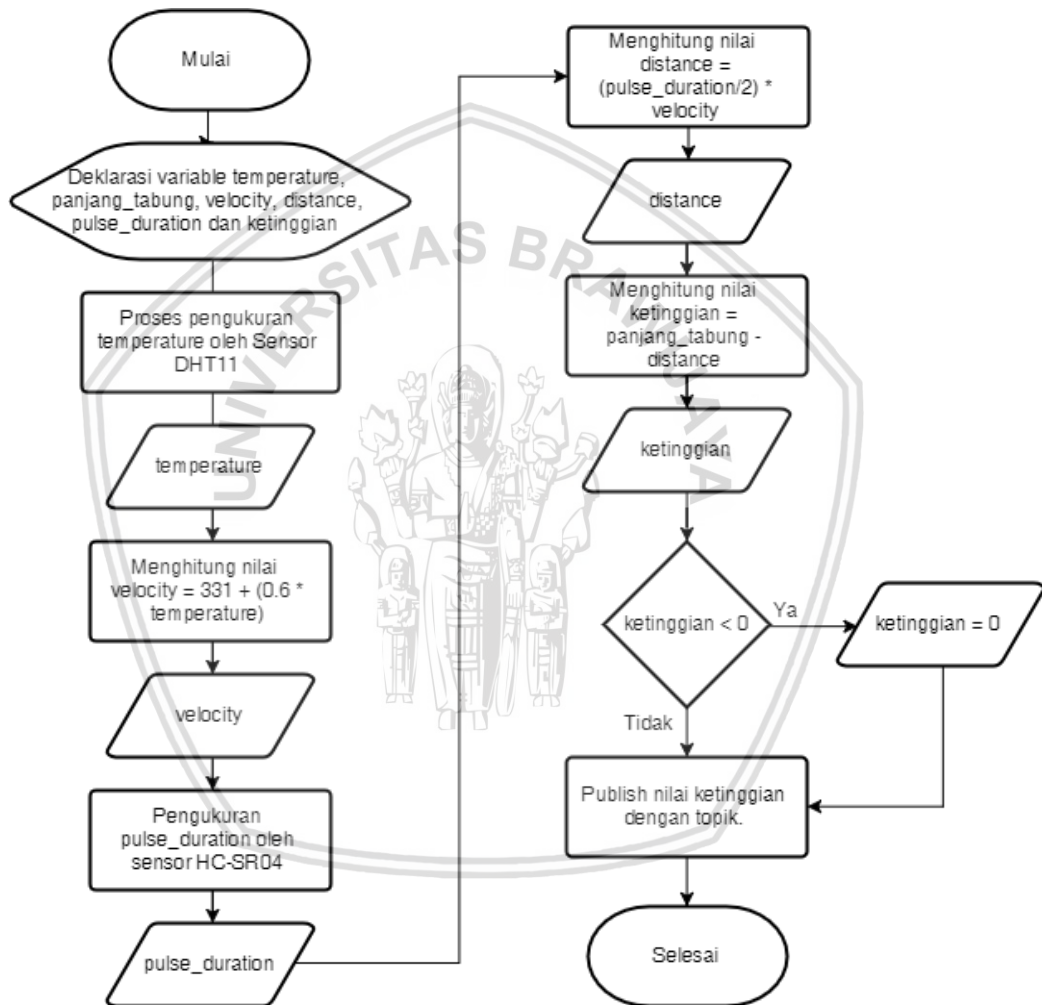
Publisher pada sistem ini merupakan perangkat sensor dan mikrokomputer yang berfungsi mengambil data pengukuran ketinggian air dan mengirimkannya ke *broker*. Sensor yang digunakan pada sistem yaitu sensor HC-SR04 sebagai pengukur ketinggian air dan sensor DHT11 sebagai pengukur temperatur udara untuk menunjang pengukuran ketinggian air. Sedangkan mikrokomputer yang digunakan adalah Raspberry Pi 3. Sensor akan mengukur ketinggian air dan *publisher* akan melakukan proses *publish* setiap 5 detik, sehingga QoS yang digunakan *publisher* adalah QoS 0, karena walaupun dalam suatu waktu *subscriber* tidak menerima pesan yang dikirimkan *publisher*, dalam 5 detik selanjutnya *publisher* akan kembali mengirimkan pesan dengan *payload* yang telah terbaharui.

Proses pengukuran yang dilakukan oleh sensor HC-SR04, memerlukan variabel temperature yang diukur oleh sensor DHT-11. Sensor HC-SR04 menggunakan gelombang suara dalam menentukan jarak, dan penentuan tersebut dihitung melalui waktu tempuh gelombang suara dari sensor menuju permukaan yang diukur. Dalam menentukan waktu tempuh, diperlukan variabel kecepatan, yang pada penelitian ini adalah kecepatan suara. Pada penelitian "A Computer-Assisted Experiment for the Measurement of the Temperature dependence of the Speed of Sound in Air" oleh Velasco et al. (2003) disebutkan bahwa kecepatan suara pada medium udara, bergantung pada temperatur pada medium udara tersebut.

Dari penelitian tersebut didapatkan persamaan kecepatan suara sebagai berikut:

$$c_s(m/s) = (331 \pm 3 m/s) + (0.60 \pm 0.06 \frac{m/s}{^\circ C})t \quad (5.1)$$

Dimana c_s adalah konstanta kecepatan suara dalam m/s dan t adalah temperature dalam $^\circ C$. Setelah didapatkan persamaan dalam menentukan kecepatan suara, diagram aliran data pada *publisher* ditampilkan pada Gambar 5.2



Gambar 5.3 Diagram aliran data *publisher*

Penjelasan secara runtut dari diagram aliran data pada *publisher* yaitu sebagai berikut:

1. Deklarasi variable temperature, panjang_tabung, velocity, distance, pulse_duration dan ketinggian. Variable temperature adalah suhu udara dalam tabung pengukur yang nantinya akan dijadikan salah satu variable dalam menentukan variable velocity. Variable panjang_tabung adalah panjang dari tabung pengukur yang telah ditentukan sebelumnya dan akan menjadi salah satu variable dalam proses yang menghasilkan variable



ketinggian. Variable distance adalah variable yang dihasilkan dari pengukuran sensor ke permukaan air. Proses penentuan variable distance membutuhkan variable velocity dan pulse_duration. Variable pulse_duration adalah variable yang dihasilkan dari pengukuran sensor ke permukaan air. Sensor menghitung waktu tempuh yang dibutuhkan oleh sinar ultrasonik ke permukaan air dan kembali lagi ke sensor. Variable ketinggian adalah variable akhir dari keseluruhan proses pengukuran yang akan dijadikan value dari topik yang akan di *publish*.

2. Proses pengukuran suhu udara dalam tabung yang menghasilkan value dari variable temperature yang akan digunakan dalam proses penentuan value dari variable velocity.
3. Proses penentuan value velocity yang berdasarkan value dari variable temperature dan rumus kecepatan suara sebagai berikut:

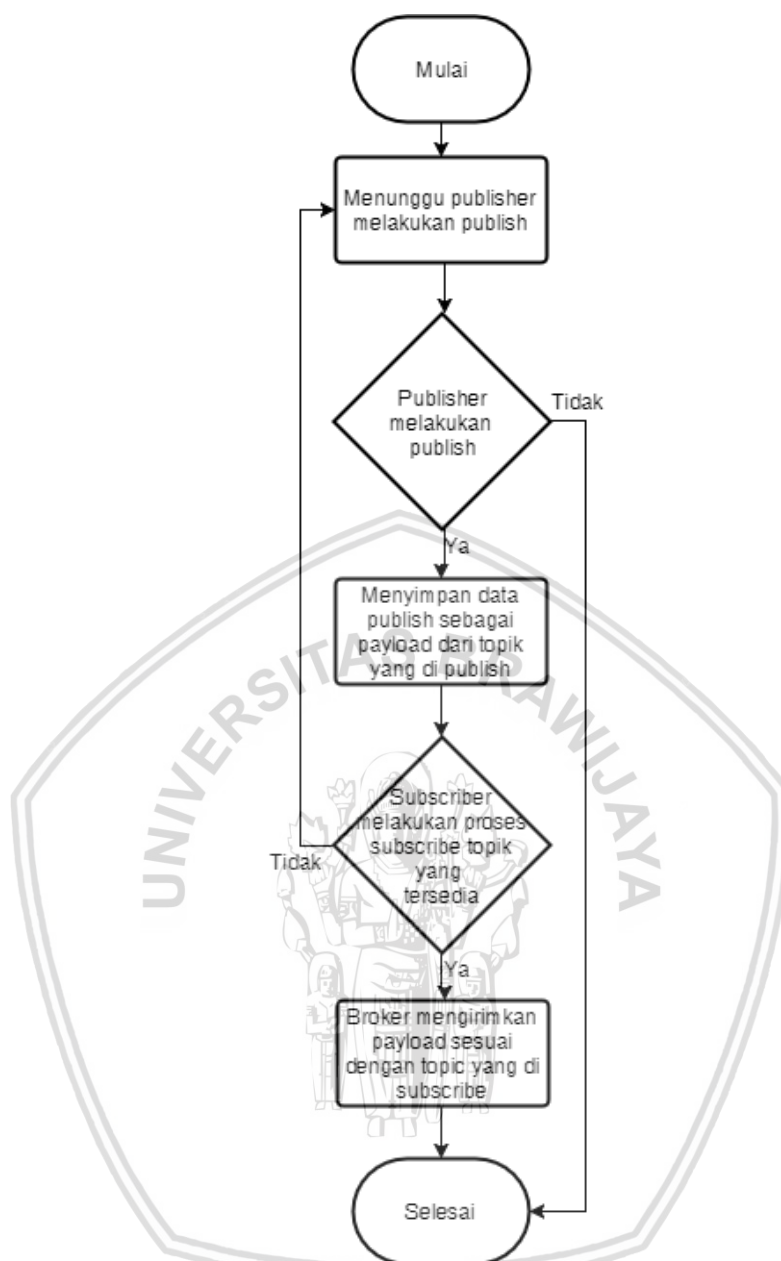
$$v = 331 + (0.6t) \quad (5.2)$$

v adalah velocity atau kecepatan, 331 adalah kecepatan suara pada medium udara dan 0.6 adalah kenaikan kecepatan per t (temperature dalam celsius).

4. Pengukuran pulse_duration dilakukan dengan sensor memancarkan sinar ultrasonik dan menghitung selang waktu dari terpancarnya sinar sampai kembalinya sinar tersebut ke sensor.
5. Penentuan *value* dari variabel distance dengan mengalikan *value* dari variabel velocity dengan *value* dari pulse_duration dibagi 2. Pembagian oleh 2 dilakukan karena value dari pulse_duration dihasilkan dari perhitungan waktu tempuh dari terpancarnya sinar ultrasonik sampai kembali ke sensor, sedangkan dalam penentuan jarak hanya diperlukan waktu tempuh dari terpancarnya sinar ultrasonik sampai mengenai permukaan air yang sedang diukur.
6. Pengukuran *value* dari variable ketinggian, yang dihasilkan dari pengurangan panjang_tabung dengan jarak sensor dengan permukaan air (distance).
7. Proses *publish* dengan *value* ketinggian.

5.1.3 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 5.3



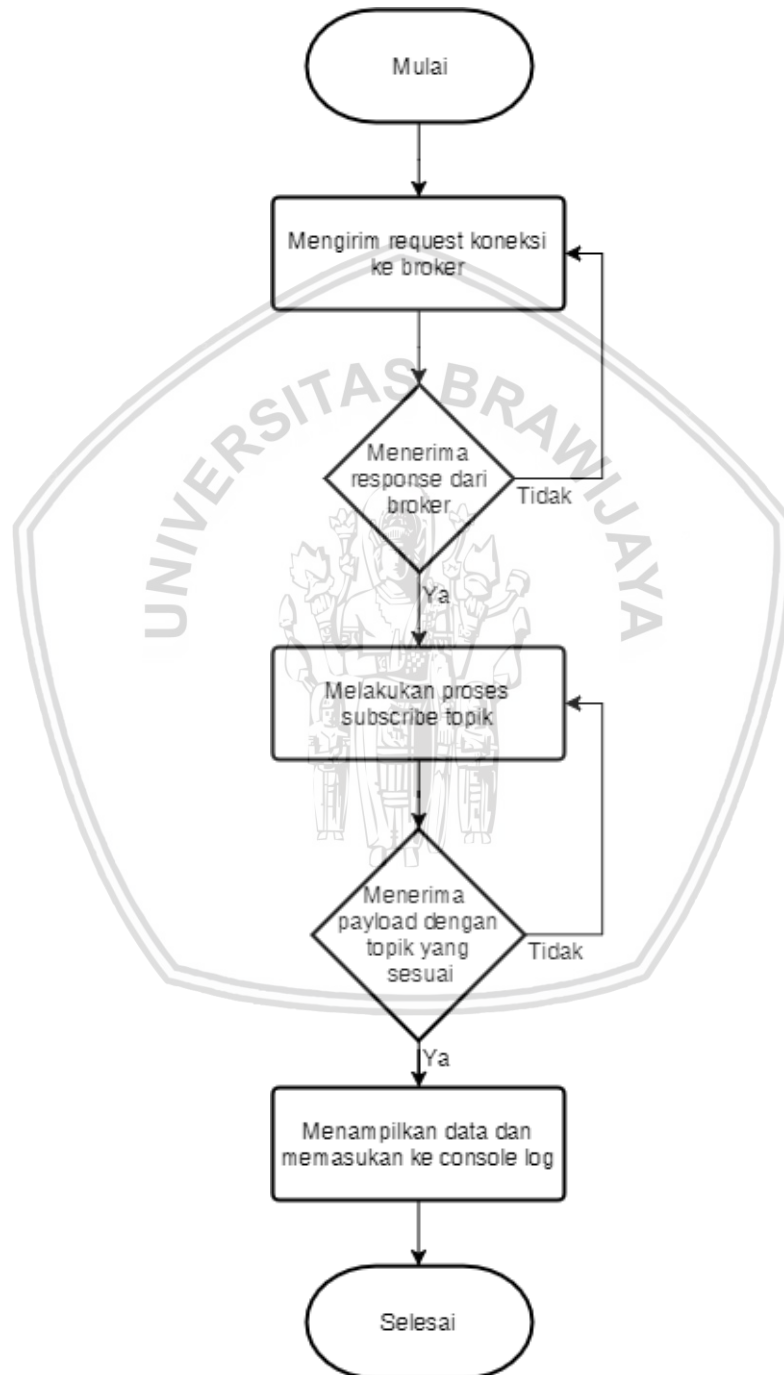
Gambar 5.4 Diagram aliran data *broker*

Penjelasan secara runtut dari diagram aliran data pada *broker* adalah sebagai berikut:

1. *Broker* dalam keadaan *idle* dan menunggu adanya *publisher* yang melakukan *publish* ke *broker*.
2. Ketika *publisher* melakukan proses *publish* ke *broker*, *broker* menerima topik dan *payload* dari topik tersebut.
3. Ketika *subscriber* melakukan proses *subscribe* ke *broker*, *broker* mengirimkan *payload* dan topik yang dikirimkan oleh *publisher* kepada *subscriber*.

5.1.4 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, data akan ditampilkan pada *interface browser*. Aliran data pada *subscriber* dapat dilihat pada gambar 5.4.



Gambar 5.5 Diagram aliran data *subscriber*

Penjelasan secara runtut dari diagram aliran data pada *subscriber* adalah sebagai berikut:



1. *Subscriber* mengirimkan *request* koneksi ke *broker*. Ketika *broker* melakukan respons, maka proses akan dilanjutkan ke tahap berikutnya. Jika *broker* tidak melakukan respons maka *broker* sedang tidak aktif dan *subscriber* akan mengirimkan *request* terus menerus.
2. *Subscriber* melakukan proses *subscribe* topik ke *broker*. Jika *broker* mengirimkan topik yang di *subscribe* oleh *subscriber*, proses akan dilanjutkan ke tahap selanjutnya. Jika *broker* tidak mengirimkan topik yang di *subscribe* oleh *subscriber*, maka *subscriber* akan melakukan proses *subscribe* lagi ke *broker*.
3. Setelah topik yang di *subscribe* ke *broker* diterima oleh *subscriber*, *web* akan menampilkan *payload* dari topik yang di *subscribe* di *interface*. Proses komunikasi yang dilakukan antara *broker* dan *web* menggunakan *websockets*.

5.2 Implementasi Sistem

Setelah semua tahapan pada perancangan sistem telah dilakukan, maka implementasi sistem dapat dilakukan. Terdapat dua bagian dari sistem yang akan di implementasi, yaitu implementasi perangkat keras dan implementasi perangkat lunak.

5.2.1 Implementasi Perangkat Keras

Proses implementasi pada bagian ini dilakukan sesuai dengan proses perancangan yang telah dilakukan pada tahap sebelumnya. *Publisher* yang berupa sensor HC-SR04, sensor DHT11 dan mikrokomputer Raspberry Pi 3 harus terhubung agar dapat melakukan proses *publish* data ke *broker* menggunakan jaringan *wireless*.

Mikrokomputer Raspberry Pi 3 dihubungkan ke sensor HC-SR04 dan juga sensor DHT11 menggunakan kabel *jumper* dan *breadboard*. Setelah semua pin dari sensor HC-SR04 dan DHT11 telah terhubung ke mikrokomputer Raspberry Pi 3, maka agar perangkat dapat menyala dibutuhkan adaptor *power supply* dengan tegangan minimal sebesar 5 Volt dan arus 2 Ampere.

5.2.2 Implementasi Perangkat Lunak

Pada implementasi perangkat lunak, terdapat 3 bagian besar yang harus dilakukan. Bagian-bagian tersebut meliputi implementasi perangkat lunak pada *publisher*, implementasi perangkat lunak pada *broker* dan implementasi perangkat lunak pada *subscriber*.

5.2.2.1 Implementasi Perangkat Lunak pada *Publisher*

Implementasi perangkat lunak pada *publisher* dilakukan pada aplikasi Python 2.7 yang telah ditambahkan *library* Adafruit_DHT agar Python dapat melakukan komunikasi dengan sensor DHT11 dan juga *library* paho.mqtt.client agar Python dapat melakukan proses *publish* ke *broker*. Implementasi perangkat lunak pada

publisher dilakukan di mikrokomputer Raspberry Pi 3 dengan sistem operasi Raspbian Jessie.

Tabel 5.1 Pseudocode publisher

1	MULAI
2	IMPORT RPi.GPIO, paho.mqtt.client, time, Adafruit_DHT
3	SET ip broker
4	SET nama client
5	CONNECT ip broker
6	CETAK pesan proses dimulai
7	WHILE True
8	AKSES Adafruit_DHT dengan argument 11 dan pin 17
9	SET temperature= Adafruit_DHT(11,17)
10	SET GPIO.setmode: GPIO.BCM
11	SET TRIG pada pin 23
12	SET ECHO pada pin 24
13	DELAY 5 detik
14	SET TRIG= True
15	DELAY 0.00001 detik
16	SET TRIG= False
17	WHILE ECHO == 0
18	SET pulse_start = time.time
19	WHILE ECHO == 1
20	SET pulse_end = time.time
21	SET pulse_duration = pulse_end - pulse_start
22	SET velocity = 331 + (0.6 * temperature)
23	SET distance = ((pulse_duration/2) * velocity) * 100
24	SET panjang = 91.4
25	SET pelampung = 1.3
26	SET ketinggian = panjang - (distance + pelampung)
27	IF ketinggian < 0 THEN
28	SET ketinggian = 0
29	CETAK temperature
30	CETAK ketinggian
31	SET topik = banjir/ub
32	PUBLISH topik, ketinggian
33	SELESAI WHILE
34	SELESAI

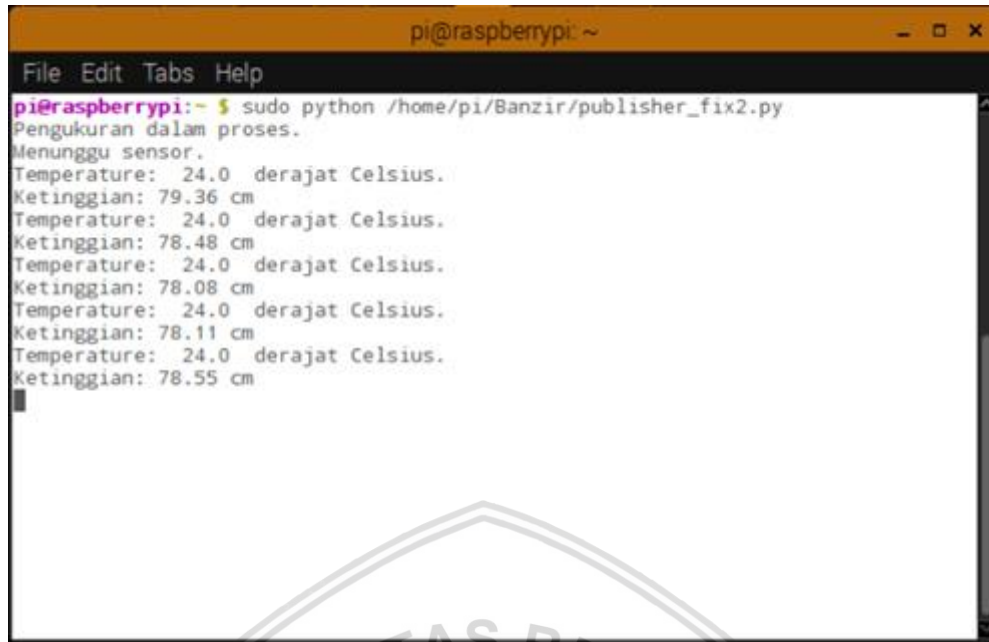
Penjelasan secara runtut dari *pseudocode* pada Tabel 5.1 ditampilkan pada tabel 5.2.

Tabel 5.2 Penjelasan pseudocode publisher

Baris 2	Melakukan import <i>library-library</i> yang dibutuhkan sistem. <i>Library</i> RPi.GPIO berfungsi agar python dapat melakukan komunikasi dengan GPIO Raspberry. <i>Library</i> paho.mqtt.client berfungsi agar python dapat menjadi <i>client</i> MQTT. <i>Library</i> time berfungsi agar python dapat menjalankan suatu perintah dengan berdasarkan waktu yang telah di <i>set</i> . <i>Library</i> Adafruit_DHT berfungsi agar python dapat berkomunikasi dengan sensor DHT melalui pin pada Raspberry Pi 3.
---------	---

Baris 3-4	Deklarasi ip dari <i>broker</i> . Deklarasi nama <i>client</i> untuk perangkat <i>publisher</i> .
Baris 5	Melakukan <i>request</i> koneksi ke ip <i>broker</i> yang telah di atur sebelumnya.
Baris 9-10	Deklarasi variabel temperature dan humidity dan pemanggilan perintah DHT_11 yang berarti sensor yang digunakan adalah DHT11 dan deklarasi nomor pin yang digunakan sensor DHT11 yaitu pin 17.
Baris 11-12	Deklarasi variable TRIG dan ECHO pada pin 23 dan 24.
Baris 13	Pemanggilan fungsi <i>time.sleep</i> yang berfungsi agar melakukan <i>delay</i> ke perintah selanjutnya selama 5 detik. Berfungsi agar komponen sensor dapat bekerja optimal pada setiap pengukuran.
Baris 14-16	Perintah untuk menyalakan pin TRIG untuk 0.00001 ($1\mu s$) lalu mematikannya lagi.
Baris 17-21	Melakukan perhitungan waktu yang dibutuhkan gelombang ultrasonik yang dikeluarkan sensor pada waktu dinyalakan dengan rumus $pulse_duration = pulse_end - pulse_start$
Baris 22-23	Melakukan perhitungan <i>value</i> velocity yaitu $331 + (0.6 * temperature)$. Setelah <i>value</i> dari velocity ditemukan, maka perintah pengukuran jarak di deklarasi dengan rumus $(pulse_duration/2) * velocity * 100$
Baris 26	Melakukan perhitungan pengukuran ketinggian dengan rumus panjang tabung – jarak
Baris 31-32	Melakukan proses <i>publish</i> dengan topik dan <i>payload</i> hasil pengukuran ketinggian

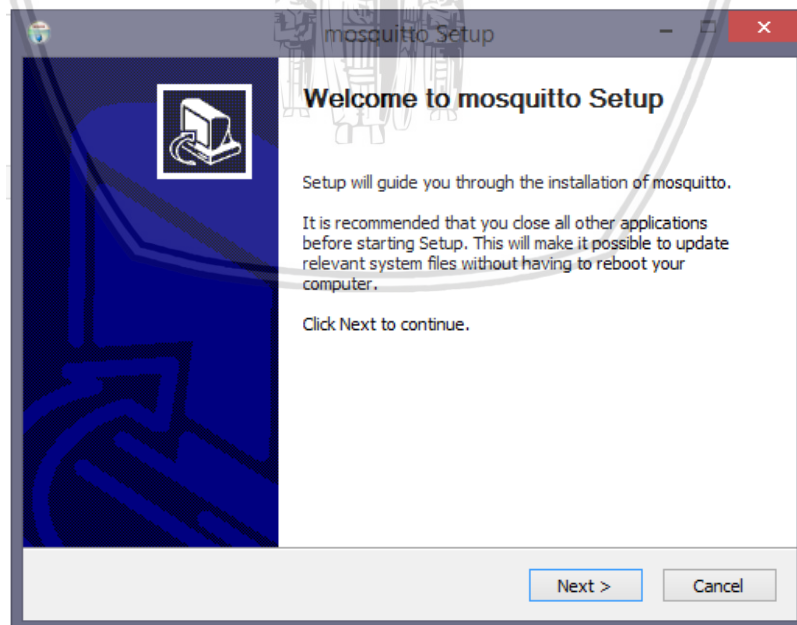
Setelah program berhasil dijalankan, maka akan dihasilkan *output* pada *terminal* seperti pada gambar 5.6.



Gambar 5.6 Output terminal pada publisher

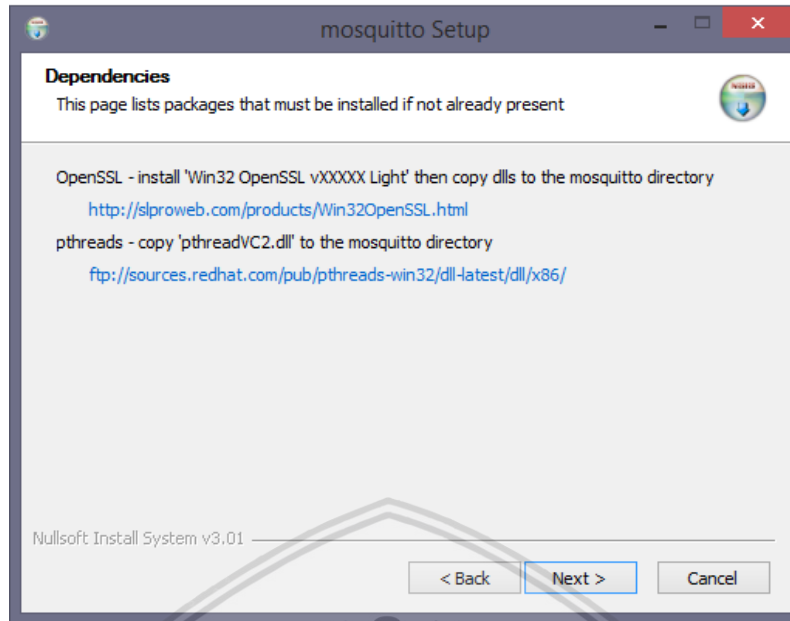
5.2.2.2 Implementasi Perangkat Lunak pada Broker

Implementasi perangkat lunak pada *broker* dilakukan di sebuah laptop yang menjalankan sistem operasi Windows. *Broker* yang digunakan pada sistem adalah broker Mosquitto 1.4.7 dengan tahapan instalasi sebagai berikut. Gambar 5.7 menampilkan proses awal instalasi *Broker* Mosquitto 1.4.7.



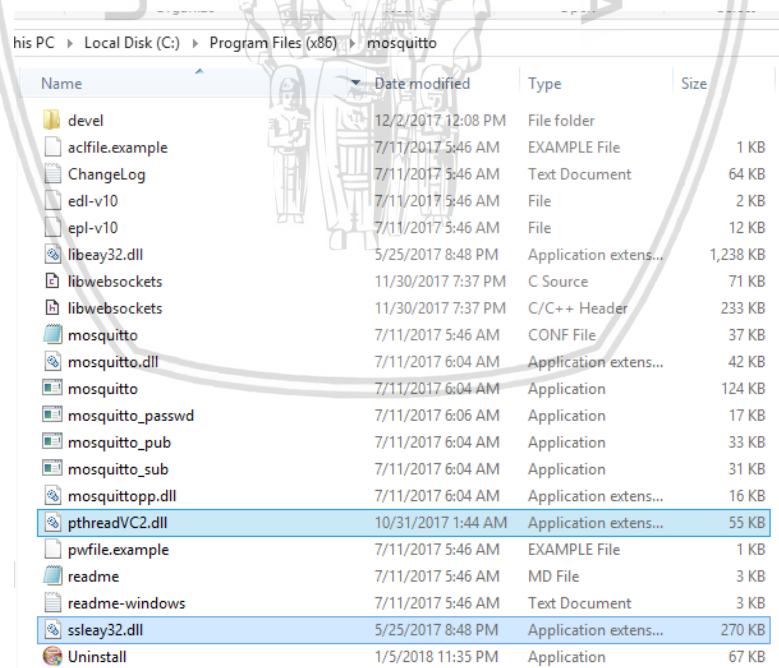
Gambar 5.7 Proses instalasi Mosquitto





Gambar 5.8 Proses instalasi Mosquitto lanjutan

Pada gambar 5.8 dapat dilihat bahwa Mosquitto memiliki *dependencies file* yaitu pthreadVC2.dll dan sslsleay32.dll yang harus disediakan pada *folder directory* Mosquitto sebelum dapat dijalankan. *Dependencies file* yang telah diletakkan pada *folder directory* dapat dilihat pada Gambar 5.9.



Gambar 5.9 File dependencies dari Mosquitto pada folder directory Mosquitto

Setelah proses instalasi selesai dilakukan, diperlukan konfigurasi agar *broker* Mosquitto dapat melakukan komunikasi dengan *web* nantinya. Perlu ditambahkan *listener* pada *broker* Mosquitto yang mengarah pada *port websockets* yaitu 8000, sehingga *broker* akan mempunyai 2 *listener* yang mendengarkan *publisher* dan



subscriber pada *port* MQTT yaitu 1883 dan juga *port websockets* yaitu 8000. Pada file *mosquitto.conf* ditambahkan kode program sebagai berikut

```
listener 8000
Protocol websockets
```

Setelah konfigurasi penambahan *listener* Mosquitto pada *file* *mosquitto.conf* berhasil dilakukan, maka Mosquitto dapat dijalankan dengan tambahan perintah yang dapat dilihat pada Gambar 5.10.



Gambar 5.10 Mosquitto berhasil dijalankan

5.2.2.3 Implementasi Perangkat Lunak pada *Subscriber*

Implementasi perangkat lunak pada *subscriber* dilakukan pada javascript yang nantinya akan menampilkan output dalam bentuk *interface web*. Agar *web* dapat melakukan komunikasi dengan *broker*, diperlukan javascript tambahan dari Paho.MQTT. *Pseudocode* pada *file* javascript konfigurasi dapat dilihat pada tabel 5.3.

Tabel 5.3 *Pseudocode* pada javascript konfigurasi

1	MULAI
2	SET host = '192.168.43.189'
3	SET port = 8000
4	SET topik = '#'
5	SET path = /mqtt
6	SET useTLS = false
7	SET username = null
8	SET password = null
9	SET cleansession = true
	SELESAI

Penjelasan dari *pseudocode* pada javascript konfigurasi ditampilkan pada tabel 5.4.

Tabel 5.4 Penjelasan *pseudocode* javascript konfigurasi

Baris 2-3	Perintah menentukan ip dari <i>broker</i> dan juga <i>port</i> yang digunakan
Baris 4	Menentukan topik yang akan di <i>subscribe</i> oleh <i>subscriber</i>
Baris 5-8	Melakukan konfigurasi mengenai <i>authentication subscriber</i> .

Tabel 5.5 menampilkan kode *pseudocode* dari file `mqtt_js` yang berisi dari alur perintah yang akan dijalankan oleh *web*.

Tabel 5.5 *Pseudocode* pada javascript yang dijalankan oleh *web*

1	MULAI
2	DEKLARASI mqtt
3	SET reconnecttimeout = 2 detik
4	SET UB = {lat,lng}
5	SET icongrey = directory icon grey
6	SET icongreen = directory icon green
7	SET iconyellow = directory icon yellow
8	SET iconred = directory icon red
9	SET topik1 = banjir/ub
10	DEKLARASI array payload
11	SET payload = array length(2)
12	DEKLARASI marker1
13	DEKLARASI map
14	
15	FUNGSI initmap
16	SET MAP = new google.maps.Map dengan attribute
17	map zoom, maxzoom, minzoom dan center
18	SET marker1 = new google.maps.Marker dengan
19	attribute marker position=UB, icon=icongrey,
20	scaledSize=80,80, anchor=40,40, label=null, map=map
21	SELESAI FUNGSI
22	
23	FUNGSI MQTTconnect
24	SET mqtt = new Paho.MQTT.Client dengan
25	parameter host, port, path, client id
26	SET options = timeout=3, useSSL=useTLS,
27	cleanSession=cleansession, onSuccess= CALL fungsi
28	onConnect, onFailure= FUNGSI message CETAK pesan
29	error, setTimeout dengan parameter fungsi MQTTconnect
30	dan reconnectTimeout
31	SET mqtt.onConnectionLost = CALL fungsi
32	onConnectionLost
33	SET mqtt.onMessageArrived = CALL fungsi
34	onMessageArrived
35	IF username != null THEN
36	SET options.userName = username
37	SET options.password = password
38	SELESAI IF
39	CETAK pada console.log (host, port, path, TLS,
40	username, password)
41	CALL fungsi mqtt.connect

```

34     SELESAI fungsi
35
36     FUNGSI onConnect
37         CALL fungsi mqtt.subscribe dengan parameter
38 topik dan SET QOS=0
39     SELESAI FUNGSI
40
41     FUNGSI onConnectionLost
42         SET setTimeout dengan parameter MQTTconnect dan
43 reconnectTimeout
44         CETAK pada console.log pesan error
45     SELESAI fungsi
46
47     FUNGSI onMessageArrived
48         CALL fungsi topiksorting
49     SELESAI FUNGSI
50
51     FUNGSI topiksorting
52         SWITCH topik
53             CASE topik1
54                 UNSHIFT array payload1 dengan value
55 payload
56                 IF payload1 pada indeks 0 <= 10 THEN
57                     SET marker1 icon = NULL
58                     SET marker1 label = NULL
59                     SET marker1 icon = icongreen
60                     SET marker1 label = payload1 indeks
61 0, " cm"
62                     SET marker1 map = map
63                     SET setInterval dengan parameter
64 nomessage1, 20000
65                     SELESAI IF
66                     IF payload1 pada indeks 0 > 10 dan <= 20
67 THEN
68                         SET marker1 icon = NULL
69                         SET marker1 label = NULL
70                         SET marker1 icon = iconyellow
71                         SET marker1 label = payload1 indeks
72 0, " cm"
73                         SET marker1 map = map
74                         SET setInterval dengan parameter
75 nomessage1, 20000
76                     SELESAI IF
77                     IF payload1 pada indeks 0 > 20 dan <= 100
78 THEN
79                         SET marker1 icon = NULL
80                         SET marker1 label = NULL
81                         SET marker1 icon = iconRED
82                         SET marker1 label = payload1 indeks
83 0, " cm"
84                         SET marker1 map = map
85                         SET setInterval dengan parameter
86 nomessage1, 20000

```

```

87         SELESAI IF
88         IF payload1 pada indeks 0 > 100 THEN
89             SET marker1 icon = NULL
90             SET marker1 label = NULL
91             SET marker1 icon = iconred
92             SET marker1 label = payload1 indeks
93 0, " cm"
94             SET marker1 map = map
95             SET setInterval dengan parameter
96 nomessage1, 20000
97         SELESAI IF
98         CETAK pada console.log topik, payload1
99 pada indeks 0, " cm"
100        BREAK
101        SELESAI CASE
102        SELESAI SWITCH
103        SELESAI FUNGSI
104
105        FUNGSI nomessagecol
106        SET marker1 icon = NULL
107        SET marker1 label = NULL
108        SET marker1 icon = NULL
109        SET clearInterval dengan parameter nomessagecol
110        SELESAI FUNGSI
111        SELESAI
112

```

Penjelasan dari *pseudocode* pada javascript yang dijalankan oleh web ditampilkan pada Tabel 5.6.

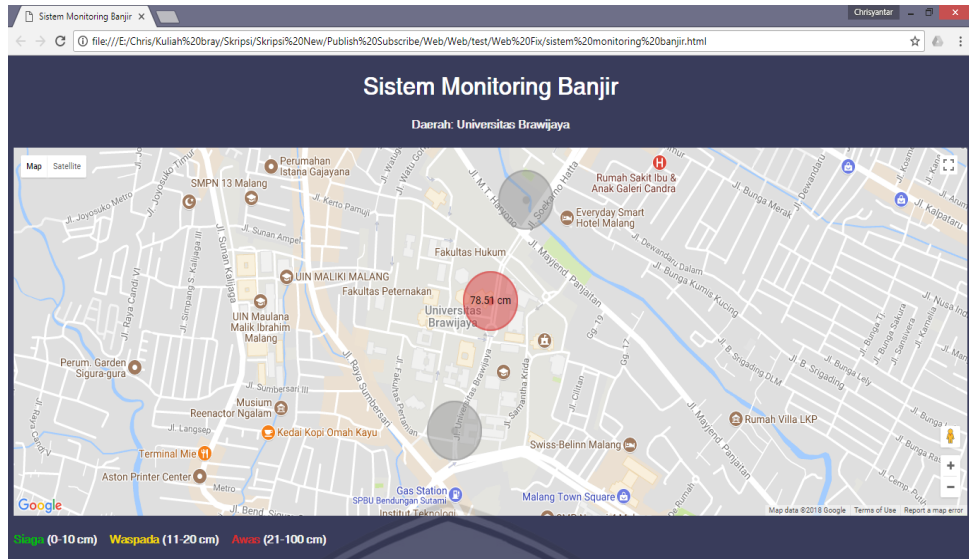
Tabel 5.6 Penjelasan *pseudocode* javascript yang dijalankan web

Baris 2-13	Melakukan deklarasi variabel yang dibutuhkan. Terdapat variabel yang berisi kordinat dari peletakan sensor nantinya, berupa <i>latitude</i> dan <i>longitude</i> yang dibutuhkan Google Maps dalam menentukan kordinat peletakan <i>marker</i> . Pada variabel topik, di deklarasikan topik-topik yang nantinya <i>payload</i> dari topik tersebut akan ditampilkan pada <i>interface</i> .
Baris 15-18	Fungsi <i>initMap</i> yang berisi perintah untuk menampilkan google maps dengan pengaturan <i>size</i> dari <i>maps</i> , <i>min zoom</i> , <i>max zoom</i> dan juga <i>marker</i> dengan posisi sesuai dengan <i>value</i> dari variable yang berisi <i>latitude</i> dan <i>longitude</i> yang telah di deklarasikan sebelumnya. Ketika <i>subscriber</i> belum mendapat <i>payload</i> topik dari <i>broker</i> , maka <i>warn marker</i> akan berwarna abu-abu serta label pada <i>marker</i> tersebut akan kosong.
Baris 20-34	Fungsi yang berisi perintah untuk menjalankan koneksi ke <i>broker</i> dengan konfigurasi <i>host</i> dan <i>port</i> yang telah di konfigurasi sebelumnya pada file <i>config.js</i> . Juga terdapat perintah untuk melakukan pemberian nama <i>client subscriber</i>



	dengan <i>random integer</i> . Jika proses perjalanan koneksi gagal, fungsi akan memanggil fungsi <i>on connection lost</i> dan menulis pesan gagal pada <i>console log</i> dan melakukan koneksi ulang dengan <i>timeout</i> yang telah ditentukan. Jika koneksi ke <i>broker</i> berhasil, sistem akan memanggil fungsi <i>on connect</i> dan menuliskan pesan berhasil ke <i>console log</i> .
Baris 36-39	Fungsi <i>on connect</i> yang akan dipanggil ketika koneksi ke <i>broker</i> telah berjalan. Berisi perintah untuk melakukan proses <i>subscribe</i> dari topik yang telah ditentukan pada file <i>config.js</i> sebelumnya.
Baris 47-103	Fungsi <i>on message arrived</i> yang berisi perintah yang akan dijalankan ketika <i>broker</i> mengirimkan pesan dari topik yang telah di <i>subscribe</i> . Fungsi ini berisi perintah-perintah untuk melakukan sortir topik dimana ketika topik yang dikirim oleh <i>broker</i> adalah topik A, maka perintah yang dijalankan adalah perintah terkait <i>marker 1</i> , dan seterusnya. Pada fungsi ini juga terdapat perintah untuk melakukan perubahan elemen dari <i>marker</i> pada <i>interface</i> Google Maps sesuai dengan <i>payload</i> dari topik yang diterima dan menuliskan topik dan <i>payload</i> -nya pada <i>console log</i> . Pada fungsi ini juga terdapat pemanggilan fungsi untuk melakukan <i>reset</i> di setiap elemen dari <i>marker</i> dengan <i>interval</i> 20000 ms dihitung dari diterimanya pesan dari <i>broker</i> dengan topik yang telah di sortir. Perintah tersebut berfungsi agar ketika <i>broker</i> tidak mengirimkan pesan dalam 20000ms, pengguna dapat mengetahuinya.
Baris 105-110	Berisi fungsi <i>nomessage</i> yang dipanggil ketika <i>broker</i> tidak mengirimkan pesan dengan topik yang telah di sortir selama 20000ms. Berisi perintah untuk <i>reset</i> elemen-elemen pada <i>marker</i> .

Setelah implementasi perangkat lunak pada *subscriber* berhasil dijalankan, *browser* akan menampilkan *output* seperti pada Gambar 5.11.



Gambar 5.11 Tampilan ketika *subscriber* menerima pesan dari *broker*



BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini akan dijelaskan mengenai pengujian sistem dan analisis dari data yang dihasilkan oleh pengujian. Pengujian diperlukan agar dapat diketahui apakah sistem dapat bekerja sesuai dengan kebutuhan dan perancangan sebelumnya. Pengujian sistem terbagi menjadi dua skenario yaitu skenario pengujian kebutuhan fungsional dan skenario pengujian kebutuhan non-fungsional. Setelah skenario pengujian dilakukan, akan didapatkan data yang akan di analisis.

Pengujian sistem yang akan dilakukan memiliki dua skenario, yaitu skenario pengujian fungsionalitas sistem dan pengujian non-fungsionalitas sistem. Pengujian fungsionalitas sistem adalah pengujian keseluruhan sistem yang meliputi pengukuran ketinggian air oleh sensor, *publisher* melakukan proses *publish* dan *subscriber* melakukan proses *subscribe*. Pengujian non-fungsional sistem meliputi pengujian akurasi, skalabilitas dan integritas data.

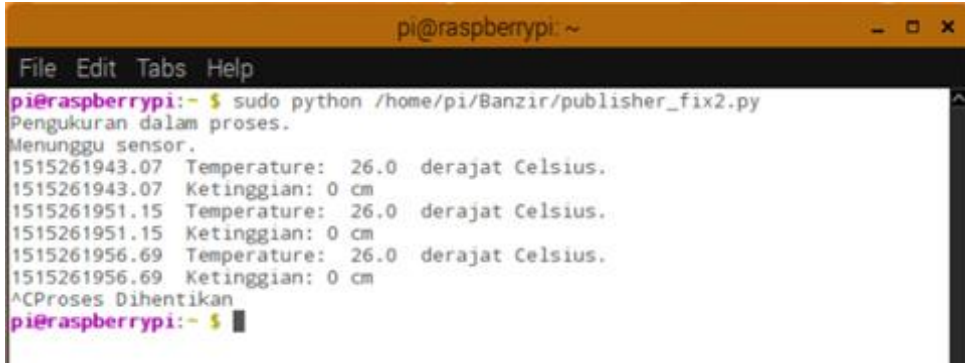
6.1 Pengujian Keseluruhan Sistem

6.1.1 Tujuan Pengujian

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.

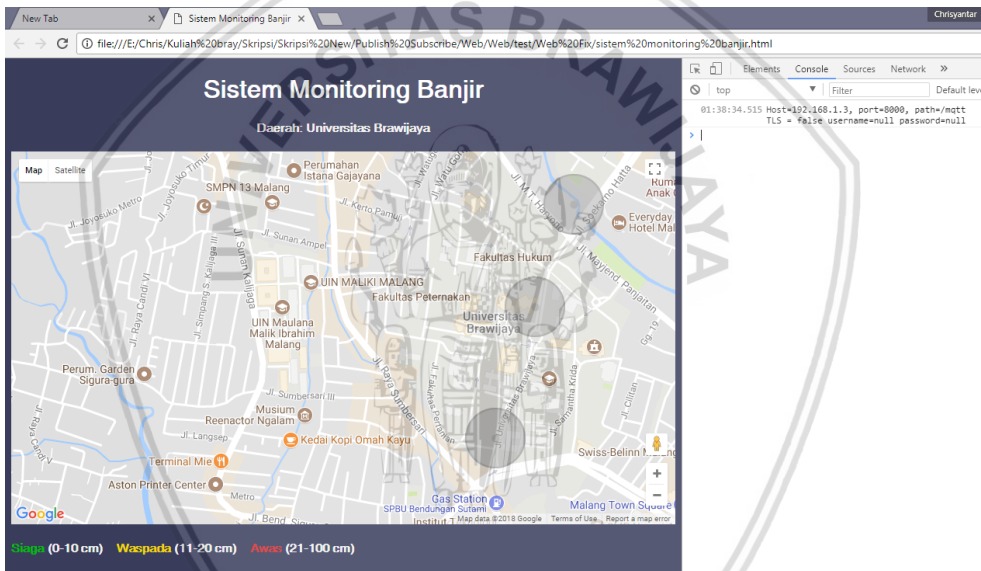
6.1.2 Pengujian

Skenario pengujian pengukuran ketinggian air dilakukan dengan menjalankan *file python* di *terminal* pada mikrokomputer. Pada pengujian pertama, dilakukan tanpa ada wadah penampung air dibawah sensor. Dilanjutkan dengan adanya wadah penampung berisi air di bawah sensor. Setelah proses pengukuran air oleh sensor telah berhasil, akan dilanjutkan dengan melakukan proses *publish* oleh *publisher*. Pengujian dilakukan dengan *publisher* menjalankan *file python* pada *terminal* yang berguna mengirimkan pesan berbasis topik ke *broker* sebanyak 3 kali seperti pada Gambar 6.1.



Gambar 6.1 *Publisher* melakukan proses *publish* topik

Setelah proses *publish* oleh *publisher* berhasil dilakukan, pengujian dilanjutkan dengan melakukan proses *subscribe* oleh *subscriber* dengan cara menjalankan *web subscriber* yang telah dibuat dengan tampilan awal seperti pada Gambar 6.2.



Gambar 6.2 Tampilan awal *web subscriber*

Setelah *web* selesai dimuat, *publisher* akan kembali dijalankan untuk mengirimkan data ke *broker* seperti pada Gambar 6.3, dan *broker* akan meneruskan pesan tersebut ke *subscriber*.



```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~$ sudo python /home/pi/Banzir/publisher_fix2.py
Pengukuran dalam proses.
Menunggu sensor.
1515262774.14 Temperature: 26.0 derajat Celsius.
1515262774.14 Ketinggian: 80.53 cm
1515262779.67 Temperature: 25.0 derajat Celsius.
1515262779.67 Ketinggian: 80.93 cm
1515262785.21 Temperature: 25.0 derajat Celsius.
1515262785.21 Ketinggian: 81.05 cm
^CProses Dihentikan
pi@raspberrypi:~$
    
```

Gambar 6.3 Proses publish dari *publisher*

6.1.3 Hasil & Analisis Pengujian

Hasil dari pengujian pengukuran ketinggian air adalah *terminal* menampilkan *output* sesuai dengan hasil pengukuran ketinggian air yang dilakukan oleh sensor. Hasil pengujian dapat dilihat pada Gambar 6.4 dan Gambar 6.5.

```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~$ sudo python /home/pi/Banzir/publisher_fix2.py
Pengukuran dalam proses.
Menunggu sensor.
Temperature: 24.0 derajat Celsius.
Ketinggian: 0.28 cm
Temperature: 24.0 derajat Celsius.
Ketinggian: 0.37 cm
Temperature: 24.0 derajat Celsius.
Ketinggian: 0.3 cm
^CProses Dihentikan
pi@raspberrypi:~$
    
```

Gambar 6.4 Hasil pengujian tanpa ada wadah penampung air

Pada gambar 6.4 dapat dilihat *terminal* menunjukkan hasil pengukuran sensor tanpa ada wadah penampung air di bawah sensor. Pengujian dilakukan 3 kali pada setiap skenario yang telah dibuat sebelumnya.

```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~$ sudo python /home/pi/Banzir/publisher_fix2.py
Pengukuran dalam proses.
Menunggu sensor.
Temperature: 25.0 derajat Celsius.
Ketinggian: 14.48 cm
Temperature: 25.0 derajat Celsius.
Ketinggian: 14.85 cm
Temperature: 25.0 derajat Celsius.
Ketinggian: 14.86 cm
^CProses Dihentikan
pi@raspberrypi:~$
    
```

Gambar 6.5 Hasil pengujian dengan wadah penampung berisi air



Gambar 6.5 menampilkan hasil dari pengujian pengukuran ketinggian air dengan skenario kedua yaitu dengan adanya wadah penampung berisi air dibawah sensor. Terlihat perbedaan dari *output* hasil pengujian pengukuran ketinggian air pada skenario 1 dan 2, sehingga dapat disimpulkan proses pengujian pengukuran ketinggian air telah berhasil.

Selanjutnya dalam proses *publish*, *command prompt* akan menampilkan *output* pada *broker* yang telah menerima pesan *publish* dari *publisher*. Tampilan dari *command prompt* pada *broker* dapat dilihat pada Gambar 6.6.

```

C:\mosquitto\build\src\Release>mosquitto.exe -v -c mosquitto.conf
1515261925: mosquitto version 1.4.7 (build date Tue 12/19/2017 11:16:54.07) starting
1515261925: Config loaded from mosquitto.conf.
1515261925: Opening ipv6 listen socket on port 1883.
1515261925: Opening ipv4 listen socket on port 1883.
1515261925: Opening websockets listen socket on port 8080.
1515261933: New connection from 192.168.1.18 on port 1883.
1515261933: New client connected from 192.168.1.18 as raspi (c1, k60).
1515261933: Sending CONNACK to raspi (0, 0)
1515261944: Received PUBLISH from raspi (d0, q0, r0, n0, 'banjir/ub', ... (1 bytes))
1515261952: Received PUBLISH from raspi (d0, q0, r0, n0, 'banjir/ub', ... (1 bytes))
1515261957: Received PUBLISH from raspi (d0, q0, r0, n0, 'banjir/ub', ... (1 bytes))
    
```

Gambar 6.6 Broker menerima pesan *publish* dari *publisher*

Pada Gambar 6.6 dapat dilihat bahwa *broker* menerima *request* koneksi dari *publisher*. Setelah mengirimkan respon *CONNACK*, *broker* menerima pesan *PUBLISH* dari *publisher* sebanyak 3 kali. Hal ini menandakan bahwa proses pengujian *publish* topik berhasil.

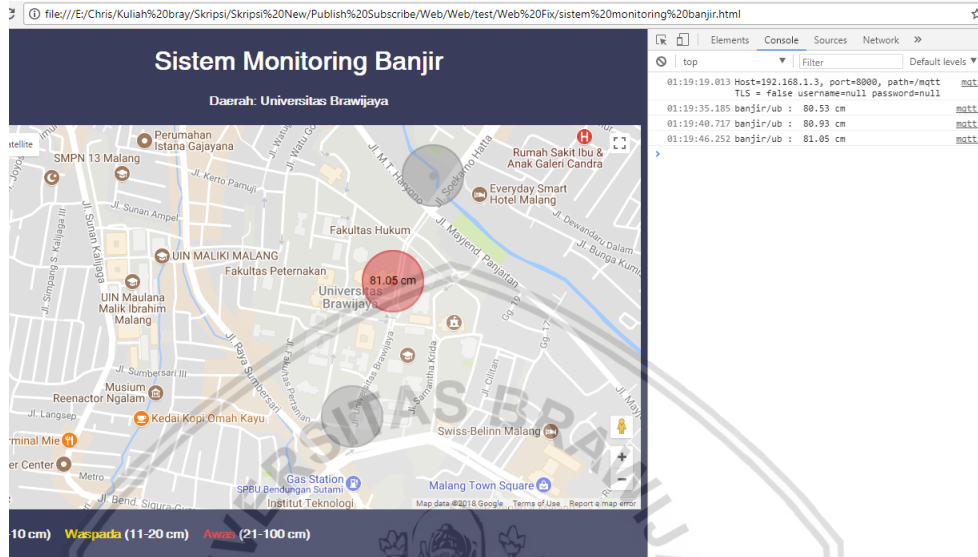
Selanjutnya adalah hasil dari pengujian proses *subscribe* topik oleh *subscriber* yang ditampilkan pada *web subscriber*. Setelah *publisher* berhasil melakukan *publish*, *broker* akan meneruskan pesan tersebut ke *subscriber* yang telah melakukan *request* topik ke *broker*. Tampilan *command prompt* pada *broker* ditampilkan pada Gambar 6.7.

```

C:\Windows\system32\cmd.exe - mosquitto.exe -v -c mosquitto.conf
1515262757: Opening websockets listen socket on port 8080.
1515262759: New client connected from 192.168.1.3 as web_62 (c1, k60).
1515262759: Sending CONNACK to web_62 (0, 0)
1515262759: Received SUBSCRIBE from web_62
1515262759: # (QoS 0)
1515262759: web_62 0 #
1515262759: Sending SUBACK to web_62
1515262769: New connection from 192.168.1.18 on port 1883.
1515262769: New client connected from 192.168.1.18 as raspi (c1, k60).
1515262769: Sending CONNACK to raspi (0, 0)
1515262775: Received PUBLISH from raspi (d0, q0, r0, n0, 'banjir/ub', ... (5 bytes))
1515262775: Sending PUBLISH to web_62 (d0, q0, r0, n0, 'banjir/ub', ... (5 bytes))
1515262780: Received PUBLISH from raspi (d0, q0, r0, n0, 'banjir/ub', ... (5 bytes))
1515262780: Sending PUBLISH to web_62 (d0, q0, r0, n0, 'banjir/ub', ... (5 bytes))
1515262786: Received PUBLISH from raspi (d0, q0, r0, n0, 'banjir/ub', ... (5 bytes))
1515262786: Sending PUBLISH to web_62 (d0, q0, r0, n0, 'banjir/ub', ... (5 bytes))
1515262819: Received PINGREQ from web_62
1515262819: Sending PINGRESP to web_62
    
```

Gambar 6.7 Proses pengiriman pesan oleh *broker*

Pada gambar 6.7 dapat dilihat bahwa setelah *broker* menerima pesan *PUBLISH* dari *client publisher* yang bernama *raspi*, *broker* langsung meneruskan pesan tersebut dengan *PUBLISH* menuju ke *client subscriber* yang bernama *web_62* yang sebelumnya sudah melakukan *request SUBSCRIBE*. Setelah *broker* berhasil meneruskan pesan dari *publisher* ke *subscriber*, *web subscriber* akan menampilkan pesan tersebut seperti pada gambar berikut.



Gambar 6.8 Output pada web subscriber

Pada Gambar 6.8 dapat dilihat bahwa *output* pada *marker* berubah dan menampilkan label dengan *payload* yang sama dengan *payload* yang dikirimkan oleh *publisher* sehingga pengujian proses *subscribe* oleh *subscriber* telah berhasil. Dari pengujian keseluruhan sistem didapatkan bahwa keseluruhan fungsi dari kebutuhan sistem dapat berjalan dengan baik, sehingga dapat disimpulkan bahwa sistem telah berhasil dibangun.

6.2 Pengujian Akurasi

6.2.1 Tujuan Pengujian

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 standar deviasi dari sensor.

6.2.2 Pengujian

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. Pengujian dilakukan pada wadah penampung air dengan ketinggian air maksimal 59,4 cm. Pengujian presisi dilakukan dengan

melakukan pengukuran ketinggian air menggunakan sensor sebanyak 10 kali dengan kondisi ketinggian air yang tidak berubah.

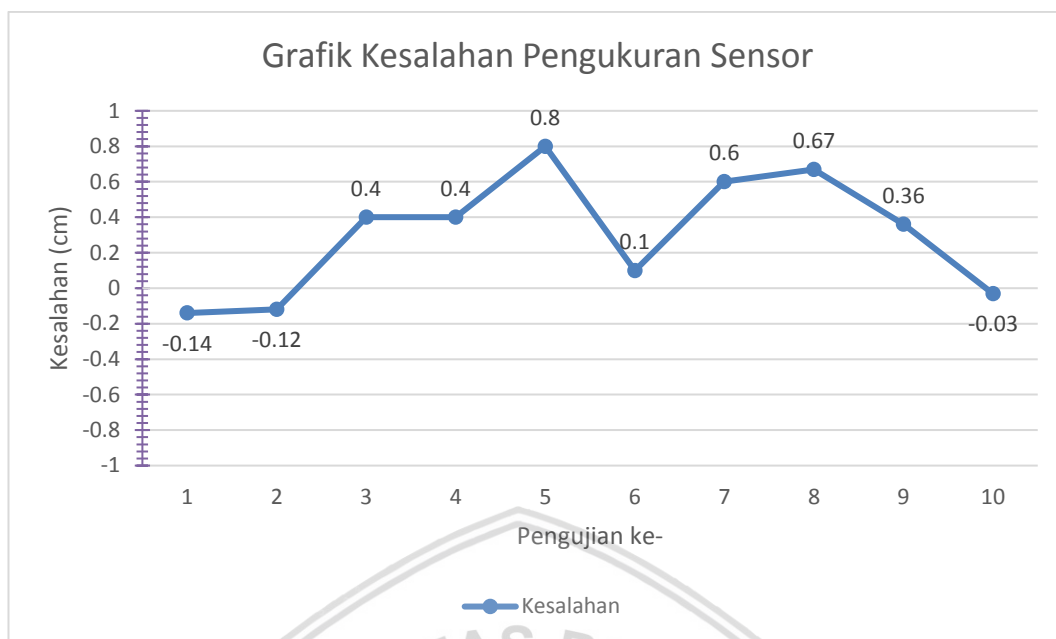
6.2.3 Hasil & Analisis Pengujian

Hasil dari pengujian akurasi yang dilakukan sebanyak 10 kali dapat dilihat pada Tabel 6.1.

Tabel 6.1 Hasil pengujian akurasi

No.	Nilai Aktual (cm)	Nilai Sensor (cm)	Kesalahan (cm)	Persentase Kesalahan (%)	Persentase Akurasi (%)
1.	2	1.86	-0.14	7	93
2.	5	4.88	-0.12	2,4	97,6
3.	10	10.4	0,4	4	96
4.	17	17.4	0,4	2,35	97,65
5.	30	30.8	0,8	2,67	97,33
6.	39	39.1	0,1	0,27	99,73
7.	45	45.6	0,6	1,3	98,7
8.	51	51.67	0,67	1,31	98,69
9.	56	56.36	0,36	0,64	99,36
10.	59.4	59.37	-0,03	0,05	99,95
Rata-rata			0,362	2,19	97,801

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. Grafik kesalahan dapat dilihat pada Gambar 6.9.



Gambar 6.9 Grafik kesalahan pengukuran sensor

Hasil pengujian presisi dapat dilihat dari tabel 6.2. Dapat dilihat bahwa pengukuran ketinggian air masih memiliki selisih dengan nilai acuan yang telah diukur sebelumnya menggunakan instrumen ukur penggaris.

Tabel 6.2 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
Rata-rata		59,377	

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 diatas, akan dilakukan perhitungan standar deviasi dengan persamaan sebagai berikut:



$$SD = \sqrt{\frac{\sum(x_i - \bar{x})^2}{n-1}} \quad (6.1)$$

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. Setelah dilakukan perhitungan, maka didapatkan hasil sebagai berikut

$$SD = \sqrt{\frac{0.00861}{9}} \quad (6.2)$$

$$SD = \pm 0.0309$$

Dari hasil pengujian akurasi didapatkan rata-rata akurasi sistem sebesar 97.801% dan memiliki standar deviasi sebagaimana dinyatakan dalam hasil pengujian presisi sebesar ± 0.0309 cm, sehingga didapatkan analisis sebagai berikut:

- Dari 10 kali pengujian akurasi, kesalahan terbesar didapatkan pada pengujian ke-1 pada titik acuan 2 cm, dan kesalahan terkecil didapatkan pada pengujian ke-10 pada titik acuan 59,4 cm. Dapat di analisis bahwa semakin jauh ketinggian air dari muka sensor, pengukuran semakin tidak akurat. Sedangkan semakin dekat ketinggian air dengan muka sensor, pengukuran semakin akurat.

- Ketidakstabilan pengukuran yang dilakukan oleh sensor dapat diakibatkan oleh deviasi dari sensor itu sendiri ataupun dari ketidakstabilan permukaan air yang diukur.

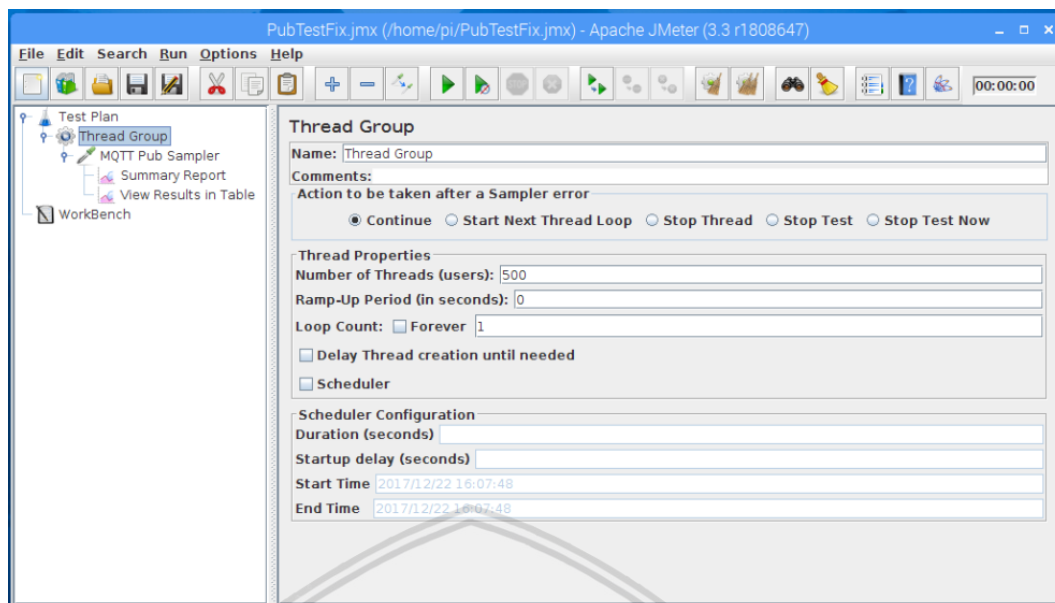
6.3 Pengujian Skalabilitas

6.3.1 Tujuan Pengujian

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.

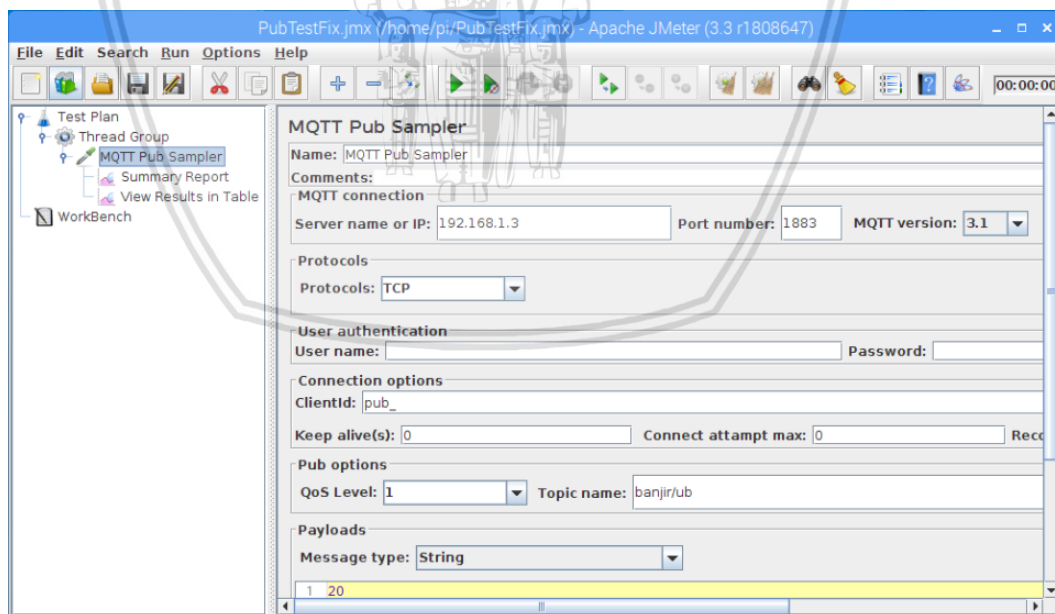
6.3.2 Pengujian

Pengujian dilakukan dengan menggunakan *plugin* MQTT pada aplikasi Apache JMeter 3.3 yang dijalankan pada perangkat *publisher* yaitu mikrokomputer Raspberry Pi 3 dan perangkat *broker* yaitu laptop Asus X450CA.



Gambar 6.10 Thread group pada JMeter

Pada gambar 6.10 dapat dilihat konfigurasi *thread group* pada JMeter dengan jumlah *thread* 500 dan *ramp-up period* sebesar 0 detik yang berarti setiap *thread* diproses tanpa *interval delay*. Selanjutnya pada gambar 6.11 terdapat MQTT Pub Sampler dengan konfigurasi IP broker 192.168.1.3, QoS level 1 dan topik yang dikirim yaitu "banjir/ub".



Gambar 6.11 Konfigurasi sampler MQTT Publisher



6.3.3 Hasil & Analisis Pengujian

Setelah pengujian skalabilitas dilakukan, aplikasi JMeter dapat menulis hasil dari pengerjaan *thread group* pada satu *file*. Hasil dari pengujian skalabilitas dapat dilihat pada Tabel 6.3.

Tabel 6.3 Hasil pengujian skalabilitas

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

Pada Tabel 6.3 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 pengeujian ke- 3 pada *thread group* 250. Sedangkan rata-rata persentase tingkat keberhasilan sebesar 99.93%. Pengujian skalabilitas pada sistem terbilang sangat baik karena persentase tingkat keberhasilan dan rata-rata tingkat keberhasilan secara keseluruhan lebih dari 99%.

Pada pengujian skalabilitas yang dilakukan, tidak terlihat tren yang signifikan pada persentase tingkat keberhasilan sistem menangani banyak *publisher*. Dapat diasumsikan bahwa fluktuasi yang terjadi pada persentase tingkat keberhasilan sistem menangani banyak *publisher* dikarenakan pada jaringan yang digunakan terjadi gangguan pada sinyal jaringan *wi-fi* yang digunakan dikarenakan pada lokasi pengujian terdapat bidang yang dapat menghalangi jaringan seperti dinding dan benda elektronik lainnya.

6.4 Pengujian Integritas Data

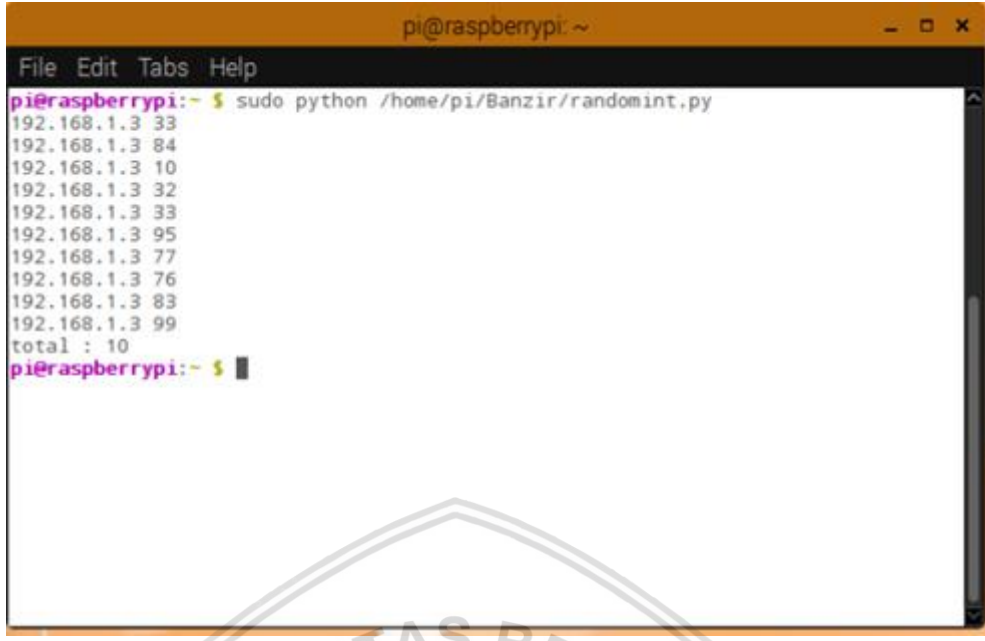
6.4.1 Tujuan Pengujian

Pengujian integritas data memiliki tujuan untuk mengetahui apakah protokol MQTT merupakan protokol yang reliabel dalam pengiriman suatu data. Pengujian dilakukan dengan *publisher* melakukan proses *publish* ke *broker* dengan topik "banjir/ub" dan isi dari pesan adalah *integer* dengan *range* 0 sampai 100 secara acak. *Interval* antara pengiriman data nya masing-masing 1000 ms, 100 ms dan 10 ms dan menampilkan isi dari pesan pada *output terminal*.

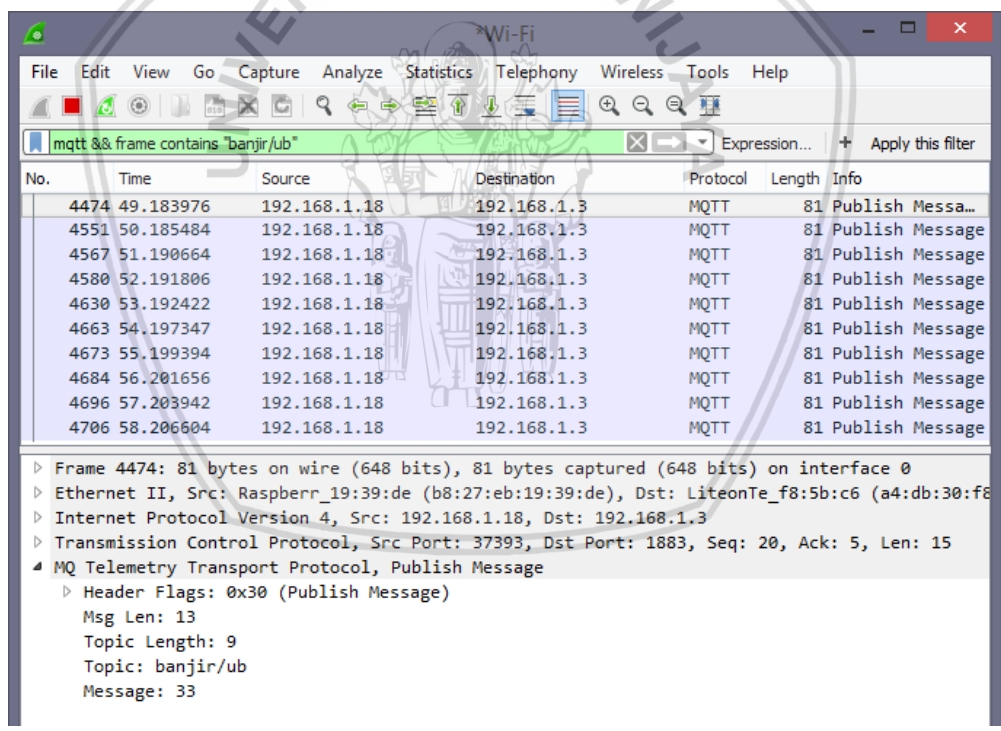
6.4.2 Pengujian

Untuk dapat membandingkan data yang dikirim oleh *publisher* dan data yang diterima oleh *broker*, pengujian dilakukan menggunakan aplikasi Wireshark 2.4.2 untuk menangkap paket yang diterima oleh ip *broker*. *Filter* yang digunakan berisikan nama protokol dan isi paket terkait topik yang di *publish* oleh *publisher* yaitu "banjir/ub".

```
mqtt && frame contains "banjir/ub"
```



Gambar 6.12 *Publisher* melakukan proses *publish*



Gambar 6.13 Wireshark menampilkan isi pesan dari pesan *publisher* yang masuk ke *IP broker*

Gambar 6.12 menunjukkan *terminal* menampilkan *ip broker* dan isi dari pesan *publish* yang dikirim. Lalu pada gambar 6.13 dapat dilihat bahwa wireshark yang telah menerapkan *filter* yang telah dijelaskan sebelumnya dan dapat membuka isi pesan yang masuk ke *ip broker*, sehingga proses perbandingan data dapat dilakukan.

6.4.3 Hasil & Analisis Pengujian

Setelah pengujian dilakukan, hasil dari pengujian integritas data ditampilkan pada Tabel 6.4.

Tabel 6.4 Hasil pengujian integritas data

No.	Interval 1000 ms		Interval 100 ms		Interval 10 ms	
	Data Publisher	Data Broker	Data Publisher	Data Broker	Data Publisher	Data Broker
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

Dari hasil perbandingan data yang dikirim *publisher* dan data yang diterima oleh *broker* pada setiap perbedaan *interval* menghasilkan kesamaan data 100%, sehingga dapat disimpulkan bahwa protokol MQTT merupakan protokol yang reliabel dalam proses pengiriman data.

BAB 7 PENUTUP

Bab ini akan menjelaskan kesimpulan berdasarkan dengan tahap-tahap yang telah dilakukan sebelumnya yaitu, kebutuhan, perancangan, implementasi dan pengujian. Terdapat juga saran pengembangan yang berkaitan dengan penelitian yang berhubungan.

7.1 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 yang telah dibuat. Dalam penerapannya, dibutuhkan elemen-elemen yang saling berkaitan yaitu *publisher*, *broker* dan *subscriber*. Pada penelitian ini, perangkat *publisher* yang digunakan adalah Raspberry Pi 3, sensor ultrasonik HC-SR04. *Broker* yang digunakan adalah Mosquitto 1.4.7 dan ditampilkan pada laman *web* yang berperan sebagai *subscriber*. Semua elemen tersebut dihubungungkan dengan jaringan *wi-fi*. Elemen-elemen tersebut dapat digunakan dalam implementasi konsep *Internet of Things* pada sistem *monitoring* banjir menggunakan protokol MQTT.
2. Rata-rata persentase akurasi sensor HC-SR04 dalam mengukur ketinggian air dari 10 kali percobaan adalah 97,801% dengan tingkat presisi dalam standar deviasi sebesar ± 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 dalam menangani 100 *publisher* dalam satu waktu memiliki persentase tingkat keberhasilan sebesar 100%, pada penanganan 250 *publisher* sebesar 99,87% dan pada penanganan 500 *publisher* diperoleh persentase 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 percobaan pengiriman data dengan *interval* 10ms, 100ms, 1000ms. Keberhasilan sistem dalam menjaga integritas data disebabkan oleh protokol MQTT yang reliabel dalam proses pengiriman data.

7.2 Saran

Beberapa saran yang dapat dilakukan dalam pengembangan dengan penelitian yang terkait dijelaskan sebagai berikut:

1. Dilakukan integrasi data dari *broker* ke dalam suatu *database*, sehingga data dapat diolah menjadi statistik dengan rentang waktu yang lebih besar.

2. Digunakan jaringan *wi-fi* dengan sinyal yang lebih stabil agar skalabilitas sistem dapat ditingkatkan.
3. Digunakan sensor lain dengan tingkat akurasi yang lebih besar dalam proses pengukuran, sehingga proses pengukuran dapat lebih akurat.



DAFTAR PUSTAKA

- Atmoko, R. A., 2013. *Sistem Monitoring dan Pengendalian Suhu dan Kelembaban Ruang pada Rumah Walet Berbasis Android, Web, dan SMS*. Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2013, 283-290.
- Arasada, B., 2017. *Aplikasi Sensor Ultrasonik untuk Deteksi Posisi Jarak pada Ruang menggunakan Arduino Uno*. Jurnal Teknik Elektro Vol. 6, No. 2, 137-145. Teknik Elektro, Fakultas Teknik. Universitas Negeri Surabaya.
- Bandyopadhyay, S. dan 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, pp. 334-340.
- Bondi, A. B., 2000. *Characteristics of scalability and their impact on performance*. Proceedings of the 2nd International Workshop on Software and Performance, 195-203.
- Budiarso, Z. dan Nurraharjo, E., 2011. *Sistem Monitoring Tingkat Ketinggian Air Bendungan Bebas Mikrokontroler*. Jurnal Dinamika Informatika Vol. 3, No. 1.
- Budioko, T., 2016. *Sistem Monitoring Suhu Jarak Jauh Berbasis Internet of Things Menggunakan Protokol MQTT*. Teknik Komputer. STMIK AKAKOM.
- Carr, J. J. dan John M. B., 1998. *Sensor Terminology*. Introduction to Biomedical Equipment Technology, Third Edition.
- Dinata, I. dan Sunanda, W., 2015. *Implementasi Wireless Monitoring Energi Listrik Berbasis Web Database*. Teknik Elektro, Fakultas Teknik Elektro. Universitas Bangka Belitung.
- Fysarakis, K., Askoxylakis, I., Soutatos, O., Papaefstathiou, I., Manifavas, C. & Katos, V., 2016. *Comparing Standardized Approaches over a Common M2M Application*. Global Communications Conference (GLOBECOM), 2016 IEEE.
- Hunkeler, U., Truong, H. L. & Stanford-Clark, A., 2008. *Communication Systems Software and Middleware and Workshops*. MQTT-S - A Publish/Subscribe Protocol For Wireless Sensor Networks, 791-798.
- 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, Vol 11, No. 1, 75-83.
- Mulyanto, A., 2008. *Pengembangan Model SIG untuk Menentukan Rute Evakuasi Bencana Banjir*. Perencanaan Wilayah dan Kota, Fakultas Teknik. Universitas Diponegoro Semarang.

- Rahmana, A., 2009. *Peranan Teknologi Informasi dalam Peningkatan Daya Saing Usaha Kecil Menengah*. Teknik Industri, Fakultas Teknik. Universitas Widyatama.
- Rakhman, M.H., Yahya, W. & Amron, K., 2018. *Implementasi Metode Failover pada Broker Protokol MQTT dengan ActiveMQ*. Teknik Informatika, Fakultas Ilmu Komputer. Universitas Brawijaya
- Rochman, H. A., Primananda, R. & Nurwasito, H., 2017. *Sistem Kendali Berbasis Mikrokontroler Menggunakan Protokol MQTT pada Smarthome*. Teknik Informatika, Fakultas Ilmu Komputer. Universitas Brawijaya.
- Rohman, F. dan Iqbal, M., 2016. *Implementasi IoT Dalam Rancang Bangun Sistem Monitoring Panel Surya Berbasis Arduino*. Teknik Elektro, Fakultas Teknik. Universitas Muria Kudus.
- Seno, A., 2013. *Karakterisasi Bencana Banjir Bandang di Indonesia*. Jurnal Sains dan Teknologi Indonesia Vol. 15, No. 1, 42-51.
- Sulistiyanto, M. P. T., Nugraha, D. A., Sari, N., Karima, N. & Asrori, W., 2015. *Implementasi IoT (Internet of Things) dalam pembelajaran di Universitas Kanjuruhan Malang*. SMARTICS Journal Vol. 1, No. 1, 20-23.
- Sulistiyowati, R., Sujono, H. A. & Musthofa, A. K., 2015. *Sistem Pendeteksi Banjir Berbasis Sensor Ultrasonik dan Mikrokontroler dengan Media Komunikasi SMS Gateway*. Seminar Nasional Sains dan Teknologi Terapan III 2015, 49-58.
- Supriyadi, T., 2011. *Penggunaan Sensor Ultrasonik sebagai Pendeteksi Ketinggian Air Sungai pada Sistem Peringatan Dini Tanggap Darurat Bencana Banjir*. Teknik Elektro. Politeknik Negeri Bandung.
- Suwarningsih, W. dan Suryawati, E., 2012. *Pembangkitan Pola Data Cuaca untuk Sistem Peringatan Dini Banjir*. Lembaga Ilmu Pengetahuan Indonesia.
- Tarigan, S. O., Sitepu, H. I. & Hutagalung, M., 2014. *Pengukuran Kinerja Sistem Publish/Subscribe Menggunakan Protokol MQTT (Message Queue Telemetry Transport)*. Jurnal Telematika, Vol. 9 no. 1. Institut Teknologi Harapan Bangsa. Bandung.
- Velasco, S., Roman, F. L., Gonzalez, A. & White, J. A. 2003. *A Computer-Assisted for the Measurement of the Temperature Dependence of the Speed of Sound in Air*. Am. J. Phys., Vol. 72, No. 2, 276-279.
- Wijayanto, I. S., 2007. *Penggunaan CRC32 dalam Integritas Data*. Teknik Informatika. Institut Teknologi Bandung.
- Xia, F., Yang, L. T., Wang, L. & Vinel, A., 2012. *Internet of Things*. International Journal of Communication Systems Vol. 25, No. 9, 1101-1102.
- 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).

LAMPIRAN A KODE SUMBER

A.1 Publisher

```
1 import RPi.GPIO as GPIO
2 import paho.mqtt.client as mqtt
3 import time
4 import Adafruit_DHT
5
6 broker_address="192.168.1.11"
7 client = mqtt.Client("raspi")
8 client.connect(broker_address)
9
10 print "Pengukuran dalam proses."
11 print "Menunggu sensor."
12
13 try:
14     while True :
15
16         humidity, temperature =
17 Adafruit_DHT.read_retry(Adafruit_DHT.DHT11,17)
18         humidity = round(humidity, 2)
19         temperature = round(temperature, 1)
20
21         GPIO.setmode(GPIO.BCM)
22
23         TRIG = 23
24         ECHO = 24
25         GPIO.setwarnings(False)
26         GPIO.setup(TRIG,GPIO.OUT)
27         GPIO.setup(ECHO,GPIO.IN)
28
29         GPIO.output(TRIG, False)
30
31         time.sleep(5)
32
33         GPIO.output(TRIG, True)
34         time.sleep(0.00001)
35         GPIO.output(TRIG, False)
36
37         while GPIO.input(ECHO)==0:
38             pulse_start = time.time()
39
40         while GPIO.input(ECHO)==1:
41             pulse_end = time.time()
42
43         pulse_duration = pulse_end - pulse_start
44
45         velocity = 331 + (0.6 * temperature)
46
47         distance = ((pulse_duration/2) * velocity)
48
49 * 100
50
51
52
```

```

53
54
55     panjang = 91.4
56     pelampung = 1.3
57
58     ketinggian = panjang - (distance +
59 pelampung)
60     ketinggian = round(ketinggian, 2)
61
62     if (ketinggian <= 0):
63         ketinggian = 0
64     print "Temperature: ",temperature, "
65 derajat Celsius."
66     print "Ketinggian:",ketinggian ,"cm"
67     client.publish("banjir/ub", ketinggian)
68
69     GPIO.cleanup()
70
71 except KeyboardInterrupt:
72     print "Proses Dihentikan"

```

A.2 Subscriber

```

1   var mqtt;
2   var reconnectTimeout = 2000;
3   var UB = {lat: -7.952628, lng: 112.614441};
4   var suhat = {lat: -7.949811, lng: 112.615515};
5   var veteran = {lat: -7.956209, lng: 112.613347};
6   var icongrey = 'web fix/grey.png';
7   var icongreen = 'web fix/green.png';
8   var iconyellow = 'web fix/yellow.png';
9   var iconred = 'web fix/red.png';
10
11   var topic1 = 'banjir/ub';
12   var topic2 = 'banjir/suhat';
13   var topic3 = 'banjir/veteran';
14
15   payload1 = new Array(2);
16   payload2 = new Array(2);
17   payload3 = new Array(2);
18
19   var marker1;
20   var marker2;
21   var marker3;
22
23   var map;
24
25
26
27   function initMap() {
28       map = new
29 google.maps.Map(document.getElementById('map'), {
30         zoom: 16,
31         maxZoom: 17,
32         minZoom: 15,

```



```
33         center: UB})
34
35
36         marker1 = new google.maps.Marker({
37     position: UB,
38     icon: 'web fix/grey.png',
39     scaledSize: new google.maps.Size(80,
40     80),
41     anchor: new google.maps.Point(40, 40),
42     label: null,
43     map: map
44     });
45
46     marker2 = new google.maps.Marker({
47     position: suhat,
48     icon: 'web fix/grey.png',
49     scaledSize: new google.maps.Size(80,
50     80),
51     anchor: new google.maps.Point(40, 40),
52     label: null,
53     map: map
54     });
55
56     marker3 = new google.maps.Marker({
57     position: veteran,
58     icon: 'web fix/grey.png',
59     scaledSize: new google.maps.Size(80,
60     80),
61     anchor: new google.maps.Point(40, 40),
62     label: null,
63     map: map
64     });
65
66     };
67
68     function MQTTconnect() {
69     if (typeof path == "undefined") {
70     path = '/mqtt';
71     }
72     mqtt = new Paho.MQTT.Client(
73     host,
74     port,
75     path,
76     "web_" + parseInt(Math.random() *
77     100, 10)
78     );
79     var options = {
80     timeout: 3,
81     useSSL: useTLS,
82     cleanSession: cleansession,
83     onSuccess: onConnect,
84     onFailure: function (message) {
85     $('#status').val("Connection failed:
86     " + message.errorMessage + "Retrying");
87     setTimeout(MQTTconnect,
88     reconnectTimeout);
89     }
```

```
90     };
91
92     mqtt.onConnectionLost = onConnectionLost;
93     mqtt.onMessageArrived = onMessageArrived;
94
95     if (username != null) {
96         options.userName = username;
97         options.password = password;
98     }
99     console.log("Host="+ host + ", port=" + port
100 + ", path=" + path + " TLS = " + useTLS + "
101 username=" + username + " password=" + password);
102     mqtt.connect(options);
103 }
104
105 function onConnect() {
106     $('#status').val('Connected to ' + host + ':' +
107 + port + path);
108     // Connection succeeded; subscribe to our
109 topic
110     mqtt.subscribe(topic, {qos: 0});
111
112     $('#topic').val(topic);
113 }
114
115 function onConnectionLost(response) {
116     setTimeout(MQTTconnect, reconnectTimeout);
117     $('#status').val("connection lost: " +
118 responseObject.errorMessage + ". Reconnecting");
119
120 };
121
122
123
124 function onMessageArrived(message) {
125
126     var topic = message.destinationName;
127     var payload = message.payloadString;
128
129
130     topicsorting();
131
132     function topicsorting () {
133         switch (topic) {
134             case topic1 :
135                 payload1.unshift(payload);
136                 if (payload1[0] <= 10) {
137
138                     marker1.setIcon(null);
139                     marker1.setLabel(null);
140
141                     marker1.setIcon(icongreen);
142                     marker1.setLabel(payload1[0]+ '
143 cm');
144                     marker1.setMap(map);
145
146                 setInterval(nomessagecol,20000);
```

```
147
148         };
149
150         if (payload1[0] > 10 && payload1[0]
151 <= 20) {
152
153             marker1.setIcon(null);
154             marker1.setLabel(null);
155
156             marker1.setIcon(iconyellow);
157             marker1.setLabel(payload1[0]+ '
158 cm');
159             marker1.setMap(map);
160
161             setInterval(nomessagecol,20000);
162             };
163
164         if (payload1[0] > 20 && payload1[0]
165 <= 100) {
166
167             marker1.setIcon(null);
168             marker1.setLabel(null);
169
170             marker1.setIcon(iconred);
171             marker1.setLabel(payload1[0]+ '
172 cm');
173             marker1.setMap(map);
174             setInterval(nomessagecol,20000);
175             };
176
177         if (payload1[0] > 100) {
178
179             marker1.setIcon(null);
180             marker1.setLabel(null);
181
182             marker1.setIcon(icongrey);
183             marker1.setLabel(payload1[0]+ '
184 cm');
185             marker1.setMap(map);
186
187             setInterval(nomessagecol,20000);
188             };
189         console.log(topic, ": ", payload1[0]+
190 ' cm');
191         break;
192
193         case topic2 :
194             payload2.unshift(payload);
195             if (payload2[0] <= 10) {
196
197                 marker2.setIcon(null);
198                 marker2.setLabel(null);
199
200                 marker2.setIcon(icongreen);
201                 marker2.setLabel(payload2[0]+ '
202 cm');
203
```

```
204         marker2.setMap(map);
205
206         setInterval(nomessageco2,20000);
207         };
208
209         if (payload2[0] > 10 && payload2[0]
210 <= 20) {
211
212             marker2.setIcon(null);
213             marker2.setLabel(null);
214
215             marker2.setIcon(iconyellow);
216             marker2.setLabel(payload2[0]+ '
217 cm');
218             marker2.setMap(map);
219
220             setInterval(nomessageco2,20000);
221             };
222
223             if (payload2[0] > 20 && payload2[0]
224 <= 100) {
225
226                 marker2.setIcon(null);
227                 marker2.setLabel(null);
228
229                 marker2.setIcon(iconred);
230                 marker2.setLabel(payload2[0]+ '
231 cm');
232                 marker2.setMap(map);
233
234                 setInterval(nomessageco2,20000);
235                 };
236
237                 if (payload2[0] > 100) {
238
239                     marker2.setIcon(null);
240                     marker2.setLabel(null);
241
242                     marker2.setIcon(icongrey);
243                     marker2.setLabel(payload2[0]+ '
244 cm');
245                     marker2.setMap(map);
246
247                     setInterval(nomessageco2,20000);
248                     };
249                     console.log(topic, ": ", payload2[0]+ '
250 cm');
251                     break;
252
253                     case topic3 :
254                         payload3.unshift(payload);
255                         if (payload3[0] <= 10) {
256
257                             marker3.setIcon(null);
258                             marker3.setLabel(null);
259
260                             marker3.setIcon(icongreen);
```

```
261         marker3.setLabel(payload3[0]+ '
262 cm');
263         marker3.setMap(map);
264
265         setInterval(nomessageco3,20000);
266         };
267
268         if (payload3[0] > 10 && payload3[0]
269 <= 20) {
270
271             marker3.setIcon(null);
272             marker3.setLabel(null);
273
274             marker3.setIcon(iconyellow);
275             marker3.setLabel(payload3[0]+ '
276 cm');
277             marker3.setMap(map);
278
279             setInterval(nomessageco3,20000);
280             };
281
282             if (payload3[0] > 20 && payload3[0]
283 <= 100) {
284
285                 marker3.setIcon(null);
286                 marker3.setLabel(null);
287
288                 marker3.setIcon(iconred);
289                 marker3.setLabel(payload3[0]+ '
290 cm');
291                 marker3.setMap(map);
292
293                 setInterval(nomessageco3,20000);
294                 };
295
296                 if (payload3[0] > 100) {
297
298                     marker3.setIcon(null);
299                     marker3.setLabel(null);
300
301                     marker3.setIcon(icongrey);
302                     marker3.setLabel(payload3[0]+ '
303 cm');
304                     marker3.setMap(map);
305
306                     setInterval(nomessageco3,20000);
307                     };
308                     console.log(topic, ": ", payload3[0]+
309 ' cm');
310                     break;
311                 };
312
313             };
314
315
316
317
```

```
318     };
319
320     function nomessagecol() {
321         marker1.setIcon(null);
322         marker1.setLabel(null);
323         marker1.setIcon(icongrey);
324         clearInterval(nomessagecol);
325
326     };
327
328     function nomessageco2() {
329         marker2.setIcon(null);
330         marker2.setLabel(null);
331         marker2.setIcon(icongrey);
332         clearInterval(nomessageco1);
333
334
335     };
336
337     function nomessageco3() {
338         marker3.setIcon(null);
339         marker3.setLabel(null);
340         marker3.setIcon(icongrey);
341         clearInterval(nomessageco1);
342     };
343
344
345
346
347     $(document).ready(function() {
348         MQTTconnect();
349
350
351     });
```


LAMPIRAN B HASIL PENGUJIAN

B.1 Skalabilitas

B.1.1 Thread 100 Pengujian ke – 1

threadName	success	Latency
Thread Group 1-2	true	5
Thread Group 1-7	true	7
Thread Group 1-6	true	7
Thread Group 1-8	true	7
Thread Group 1-4	true	7
Thread Group 1-5	true	7
Thread Group 1-9	true	6
Thread Group 1-10	true	10
Thread Group 1-1	true	13
Thread Group 1-3	true	10
Thread Group 1-11	true	10
Thread Group 1-12	true	12
Thread Group 1-14	true	18
Thread Group 1-17	true	13
Thread Group 1-13	true	14
Thread Group 1-15	true	13
Thread Group 1-16	true	7
Thread Group 1-18	true	8
Thread Group 1-19	true	8
Thread Group 1-20	true	58
Thread Group 1-21	true	55
Thread Group 1-27	true	93
Thread Group 1-24	true	93
Thread Group 1-25	true	99
Thread Group 1-23	true	122
Thread Group 1-22	true	126
Thread Group 1-30	true	78
Thread Group 1-28	true	80
Thread Group 1-26	true	83
Thread Group 1-32	true	79

Thread Group 1-33	true	118
Thread Group 1-31	true	113
Thread Group 1-29	true	120
Thread Group 1-35	true	89
Thread Group 1-34	true	116
Thread Group 1-39	true	72
Thread Group 1-41	true	71
Thread Group 1-37	true	71
Thread Group 1-42	true	54
Thread Group 1-40	true	49
Thread Group 1-38	true	83
Thread Group 1-36	true	89
Thread Group 1-45	true	58
Thread Group 1-43	true	68
Thread Group 1-47	true	31
Thread Group 1-46	true	61
Thread Group 1-49	true	42
Thread Group 1-44	true	112
Thread Group 1-48	true	53
Thread Group 1-52	true	48
Thread Group 1-50	true	51
Thread Group 1-51	true	47
Thread Group 1-53	true	48
Thread Group 1-54	true	50
Thread Group 1-57	true	51
Thread Group 1-59	true	51
Thread Group 1-56	true	51
Thread Group 1-55	true	51
Thread Group 1-58	true	48
Thread Group 1-61	true	36
Thread Group 1-64	true	10

Thread Group 1-66	true	7
Thread Group 1-62	true	35
Thread Group 1-67	true	22
Thread Group 1-60	true	94
Thread Group 1-65	true	11
Thread Group 1-63	true	35
Thread Group 1-68	true	24
Thread Group 1-70	true	10
Thread Group 1-69	true	18
Thread Group 1-72	true	23
Thread Group 1-71	true	16
Thread Group 1-73	true	11
Thread Group 1-74	true	17
Thread Group 1-75	true	16
Thread Group 1-76	true	19
Thread Group 1-78	true	7
Thread Group 1-77	true	16
Thread Group 1-79	true	5
Thread Group 1-80	true	67
Thread Group 1-81	true	103
Thread Group 1-83	true	117
Thread Group 1-82	true	23
Thread Group 1-86	true	22
Thread Group 1-85	true	27
Thread Group 1-87	true	25
Thread Group 1-89	true	24
Thread Group 1-92	true	9
Thread Group 1-90	true	19
Thread Group 1-91	true	28
Thread Group 1-84	true	37
Thread Group 1-94	true	28
Thread Group 1-88	true	61
Thread Group 1-96	true	25
Thread Group 1-93	true	32
Thread Group 1-95	true	33

Thread Group 1-98	true	31
Thread Group 1-97	true	21
Thread Group 1-99	true	24
Thread Group 1-100	true	16

Success rate = 100%
Min. Latency = 5 ms
Max. Latency = 126 ms
Mean Latency = 43.18 ms

B.1.2 Thread 100 Pengujian ke - 2

threadName	success	Latency
Thread Group 1-1	true	18
Thread Group 1-3	true	12
Thread Group 1-5	true	20
Thread Group 1-8	true	9
Thread Group 1-4	true	68
Thread Group 1-2	true	19
Thread Group 1-7	true	23
Thread Group 1-6	true	158
Thread Group 1-9	true	89
Thread Group 1-11	true	88
Thread Group 1-14	true	83
Thread Group 1-13	true	100
Thread Group 1-10	true	86
Thread Group 1-17	true	75
Thread Group 1-15	true	87
Thread Group 1-36	true	129
Thread Group 1-42	true	128
Thread Group 1-12	true	130
Thread Group 1-38	true	134
Thread Group 1-18	true	136
Thread Group 1-16	true	139
Thread Group 1-31	true	130
Thread Group 1-37	true	121
Thread Group 1-28	true	122



Thread Group 1-21	true	120
Thread Group 1-49	true	123
Thread Group 1-23	true	124
Thread Group 1-40	true	136
Thread Group 1-29	true	130
Thread Group 1-39	true	147
Thread Group 1-32	true	147
Thread Group 1-33	true	149
Thread Group 1-34	true	157
Thread Group 1-47	true	141
Thread Group 1-19	true	150
Thread Group 1-48	true	163
Thread Group 1-22	true	162
Thread Group 1-35	true	153
Thread Group 1-45	true	159
Thread Group 1-27	true	170
Thread Group 1-30	true	169
Thread Group 1-25	true	180
Thread Group 1-41	true	136
Thread Group 1-20	true	160
Thread Group 1-26	true	175
Thread Group 1-24	true	172
Thread Group 1-43	true	145
Thread Group 1-44	true	182
Thread Group 1-46	true	192
Thread Group 1-51	true	153
Thread Group 1-50	true	150
Thread Group 1-53	true	150
Thread Group 1-52	true	156
Thread Group 1-55	true	154
Thread Group 1-54	true	168
Thread Group 1-58	true	154
Thread Group 1-56	true	168
Thread Group 1-60	true	165
Thread Group 1-57	true	169

Thread Group 1-61	true	168
Thread Group 1-59	true	174
Thread Group 1-63	true	173
Thread Group 1-62	true	175
Thread Group 1-65	true	177
Thread Group 1-64	true	164
Thread Group 1-69	true	159
Thread Group 1-66	true	156
Thread Group 1-67	true	160
Thread Group 1-68	true	142
Thread Group 1-73	true	132
Thread Group 1-72	true	137
Thread Group 1-70	true	138
Thread Group 1-71	true	133
Thread Group 1-74	true	128
Thread Group 1-77	true	120
Thread Group 1-76	true	112
Thread Group 1-75	true	129
Thread Group 1-79	true	109
Thread Group 1-80	true	110
Thread Group 1-78	true	134
Thread Group 1-82	true	103
Thread Group 1-85	true	87
Thread Group 1-81	true	96
Thread Group 1-83	true	81
Thread Group 1-87	true	80
Thread Group 1-86	true	88
Thread Group 1-84	true	95
Thread Group 1-88	true	78
Thread Group 1-90	true	80
Thread Group 1-91	true	78
Thread Group 1-93	true	77
Thread Group 1-94	true	77
Thread Group 1-92	true	75
Thread Group 1-89	true	79

Thread Group 1-98	true	61
Thread Group 1-100	true	68
Thread Group 1-99	true	67
Thread Group 1-95	true	92
Thread Group 1-96	true	102
Thread Group 1-97	true	88

Success rate = 100%
Min. Latency = 9 ms
Max. Latency = 192 ms
Mean Latency = 122.15 ms

B.1.3 Thread 100 Pengujian ke - 3

threadName	success	Latency
Thread Group 1-3	true	10
Thread Group 1-4	true	14
Thread Group 1-7	true	10
Thread Group 1-5	true	10
Thread Group 1-2	true	18
Thread Group 1-8	true	12
Thread Group 1-10	true	9
Thread Group 1-13	true	9
Thread Group 1-6	true	25
Thread Group 1-1	true	9
Thread Group 1-22	true	9
Thread Group 1-9	true	10
Thread Group 1-11	true	22
Thread Group 1-19	true	12
Thread Group 1-21	true	31
Thread Group 1-15	true	8
Thread Group 1-20	true	26
Thread Group 1-18	true	8
Thread Group 1-12	true	34
Thread Group 1-16	true	7
Thread Group 1-17	true	18

Thread Group 1-14	true	26
Thread Group 1-25	true	8
Thread Group 1-24	true	6
Thread Group 1-23	true	22
Thread Group 1-28	true	7
Thread Group 1-29	true	10
Thread Group 1-27	true	32
Thread Group 1-26	true	5
Thread Group 1-30	true	10
Thread Group 1-31	true	10
Thread Group 1-32	true	6
Thread Group 1-33	true	6
Thread Group 1-34	true	11
Thread Group 1-37	true	5
Thread Group 1-35	true	9
Thread Group 1-38	true	6
Thread Group 1-36	true	16
Thread Group 1-40	true	7
Thread Group 1-39	true	9
Thread Group 1-41	true	12
Thread Group 1-42	true	10
Thread Group 1-43	true	12
Thread Group 1-44	true	8
Thread Group 1-47	true	6
Thread Group 1-45	true	10
Thread Group 1-48	true	5
Thread Group 1-46	true	9
Thread Group 1-52	true	9
Thread Group 1-49	true	5
Thread Group 1-50	true	12
Thread Group 1-51	true	7
Thread Group 1-53	true	6
Thread Group 1-54	true	6
Thread Group 1-58	true	10
Thread Group 1-57	true	12



Thread Group 1-55	true	33
Thread Group 1-60	true	25
Thread Group 1-61	true	29
Thread Group 1-65	true	27
Thread Group 1-64	true	29
Thread Group 1-63	true	19
Thread Group 1-62	true	27
Thread Group 1-66	true	23
Thread Group 1-67	true	45
Thread Group 1-59	true	54
Thread Group 1-68	true	61
Thread Group 1-56	true	73
Thread Group 1-70	true	27
Thread Group 1-71	true	13
Thread Group 1-72	true	11
Thread Group 1-69	true	12
Thread Group 1-74	true	27
Thread Group 1-73	true	10
Thread Group 1-75	true	14
Thread Group 1-76	true	15
Thread Group 1-77	true	16
Thread Group 1-78	true	17
Thread Group 1-79	true	6
Thread Group 1-81	true	5
Thread Group 1-80	true	17
Thread Group 1-82	true	5
Thread Group 1-83	true	15
Thread Group 1-87	true	9
Thread Group 1-86	true	8
Thread Group 1-84	true	7
Thread Group 1-85	true	6
Thread Group 1-88	true	8
Thread Group 1-89	true	13
Thread Group 1-91	true	10
Thread Group 1-92	true	5

Thread Group 1-90	true	17
Thread Group 1-95	true	10
Thread Group 1-93	true	6
Thread Group 1-94	true	7
Thread Group 1-96	true	12
Thread Group 1-99	true	13
Thread Group 1-97	true	19
Thread Group 1-100	true	6
Thread Group 1-98	true	6

Success rate = 100%
 Min. Latency = 5 ms
 Max. Latency = 73 ms
 Mean Latency = 14.78 ms

B.1.4 Thread 250 Pengujian ke - 1

threadName	success	Latency
Thread Group 1-1	true	7
Thread Group 1-3	true	9
Thread Group 1-2	true	7
Thread Group 1-5	true	12
Thread Group 1-4	true	9
Thread Group 1-6	true	5
Thread Group 1-7	true	13
Thread Group 1-9	true	8
Thread Group 1-8	true	5
Thread Group 1-10	true	7
Thread Group 1-11	true	6
Thread Group 1-12	true	10
Thread Group 1-14	true	11
Thread Group 1-13	true	25
Thread Group 1-15	true	10
Thread Group 1-16	true	12
Thread Group 1-17	true	22
Thread Group 1-18	true	11
Thread Group 1-19	true	18



Thread Group 1-20	true	12
Thread Group 1-21	true	16
Thread Group 1-23	true	15
Thread Group 1-24	true	22
Thread Group 1-22	true	35
Thread Group 1-26	true	39
Thread Group 1-25	true	39
Thread Group 1-27	true	35
Thread Group 1-28	true	33
Thread Group 1-30	true	44
Thread Group 1-33	true	48
Thread Group 1-29	true	50
Thread Group 1-34	true	50
Thread Group 1-35	true	51
Thread Group 1-31	true	63
Thread Group 1-37	true	65
Thread Group 1-38	true	57
Thread Group 1-32	true	58
Thread Group 1-39	true	67
Thread Group 1-43	true	43
Thread Group 1-42	true	49
Thread Group 1-36	true	98
Thread Group 1-40	true	46
Thread Group 1-45	true	44
Thread Group 1-44	true	42
Thread Group 1-41	true	51
Thread Group 1-47	true	39
Thread Group 1-46	true	71
Thread Group 1-48	true	17
Thread Group 1-51	true	25
Thread Group 1-53	true	77
Thread Group 1-50	true	73
Thread Group 1-49	true	46
Thread Group 1-60	true	74
Thread Group 1-61	true	35

Thread Group 1-57	true	29
Thread Group 1-59	true	31
Thread Group 1-55	true	34
Thread Group 1-56	true	44
Thread Group 1-62	true	26
Thread Group 1-58	true	15
Thread Group 1-66	true	18
Thread Group 1-64	true	15
Thread Group 1-52	true	30
Thread Group 1-54	true	34
Thread Group 1-68	true	16
Thread Group 1-63	true	41
Thread Group 1-65	true	25
Thread Group 1-67	true	10
Thread Group 1-71	true	15
Thread Group 1-73	true	19
Thread Group 1-72	true	10
Thread Group 1-70	true	37
Thread Group 1-69	true	6
Thread Group 1-74	true	16
Thread Group 1-77	true	13
Thread Group 1-75	true	14
Thread Group 1-76	true	6
Thread Group 1-78	true	5
Thread Group 1-79	true	7
Thread Group 1-80	true	9
Thread Group 1-82	true	6
Thread Group 1-81	true	8
Thread Group 1-84	true	12
Thread Group 1-83	true	10
Thread Group 1-85	true	11
Thread Group 1-86	true	12
Thread Group 1-87	true	17
Thread Group 1-90	true	16
Thread Group 1-88	true	18



Thread Group 1-89	true	19
Thread Group 1-91	true	25
Thread Group 1-92	true	26
Thread Group 1-93	true	34
Thread Group 1-95	true	27
Thread Group 1-96	true	24
Thread Group 1-98	true	29
Thread Group 1-94	true	33
Thread Group 1-97	true	34
Thread Group 1-101	true	46
Thread Group 1-99	true	38
Thread Group 1-103	true	49
Thread Group 1-100	true	41
Thread Group 1-107	true	34
Thread Group 1-102	true	40
Thread Group 1-106	true	43
Thread Group 1-104	true	42
Thread Group 1-105	true	47
Thread Group 1-112	true	51
Thread Group 1-109	true	51
Thread Group 1-111	true	66
Thread Group 1-115	true	58
Thread Group 1-113	true	35
Thread Group 1-110	true	43
Thread Group 1-108	true	81
Thread Group 1-114	true	46
Thread Group 1-117	true	34
Thread Group 1-119	true	9
Thread Group 1-118	true	13
Thread Group 1-120	true	26
Thread Group 1-116	true	70
Thread Group 1-122	true	15
Thread Group 1-125	true	17
Thread Group 1-123	true	17
Thread Group 1-128	true	31

Thread Group 1-121	true	38
Thread Group 1-127	true	19
Thread Group 1-126	true	15
Thread Group 1-129	true	15
Thread Group 1-131	true	15
Thread Group 1-124	true	5
Thread Group 1-130	true	23
Thread Group 1-133	true	6
Thread Group 1-132	true	27
Thread Group 1-134	true	8
Thread Group 1-137	true	9
Thread Group 1-139	true	8
Thread Group 1-138	true	21
Thread Group 1-140	true	8
Thread Group 1-141	true	8
Thread Group 1-142	true	17
Thread Group 1-143	true	15
Thread Group 1-135	true	11
Thread Group 1-136	true	12
Thread Group 1-145	true	25
Thread Group 1-146	true	9
Thread Group 1-147	true	13
Thread Group 1-144	true	34
Thread Group 1-148	true	10
Thread Group 1-149	true	9
Thread Group 1-151	true	9
Thread Group 1-150	true	9
Thread Group 1-152	true	19
Thread Group 1-154	true	17
Thread Group 1-153	true	17
Thread Group 1-155	true	5
Thread Group 1-156	true	9
Thread Group 1-157	true	10
Thread Group 1-159	true	23
Thread Group 1-158	true	17



Thread Group 1-160	true	7
Thread Group 1-161	true	15
Thread Group 1-162	true	5
Thread Group 1-163	true	9
Thread Group 1-164	true	7
Thread Group 1-165	true	11
Thread Group 1-166	true	13
Thread Group 1-167	true	6
Thread Group 1-170	true	8
Thread Group 1-168	true	7
Thread Group 1-169	true	14
Thread Group 1-172	true	5
Thread Group 1-171	true	8
Thread Group 1-173	true	16
Thread Group 1-174	true	14
Thread Group 1-176	true	9
Thread Group 1-178	true	12
Thread Group 1-175	true	8
Thread Group 1-177	true	8
Thread Group 1-181	true	6
Thread Group 1-179	true	6
Thread Group 1-180	true	9
Thread Group 1-182	true	13
Thread Group 1-183	true	5
Thread Group 1-184	true	10
Thread Group 1-185	true	5
Thread Group 1-188	true	5
Thread Group 1-186	true	17
Thread Group 1-187	true	8
Thread Group 1-189	true	6
Thread Group 1-190	true	7
Thread Group 1-191	true	7
Thread Group 1-192	true	5
Thread Group 1-195	true	9
Thread Group 1-196	true	8

Thread Group 1-200	true	10
Thread Group 1-199	true	6
Thread Group 1-197	true	16
Thread Group 1-205	true	11
Thread Group 1-202	true	13
Thread Group 1-203	true	9
Thread Group 1-193	true	146
Thread Group 1-209	true	13
Thread Group 1-204	true	9
Thread Group 1-201	true	12
Thread Group 1-198	true	20
Thread Group 1-206	true	17
Thread Group 1-208	true	14
Thread Group 1-194	true	29
Thread Group 1-207	true	22
Thread Group 1-210	true	14
Thread Group 1-211	true	33
Thread Group 1-213	true	13
Thread Group 1-212	true	6
Thread Group 1-215	true	16
Thread Group 1-214	true	6
Thread Group 1-216	true	8
Thread Group 1-217	true	6
Thread Group 1-218	true	6
Thread Group 1-219	true	7
Thread Group 1-222	true	6
Thread Group 1-220	true	36
Thread Group 1-221	true	6
Thread Group 1-225	true	7
Thread Group 1-223	true	18
Thread Group 1-224	true	16
Thread Group 1-226	true	5
Thread Group 1-228	true	19
Thread Group 1-227	true	14
Thread Group 1-229	true	6



Thread Group 1-230	true	16
Thread Group 1-231	true	14
Thread Group 1-234	true	7
Thread Group 1-235	true	7
Thread Group 1-236	true	7
Thread Group 1-232	true	10
Thread Group 1-233	true	9
Thread Group 1-237	true	10
Thread Group 1-238	true	8
Thread Group 1-239	true	7
Thread Group 1-240	true	12
Thread Group 1-242	true	6
Thread Group 1-241	true	7
Thread Group 1-243	true	9
Thread Group 1-245	true	6
Thread Group 1-244	true	9
Thread Group 1-246	true	8
Thread Group 1-248	true	9
Thread Group 1-249	true	8
Thread Group 1-247	true	7
Thread Group 1-250	true	8

Success rate = 100%
Min. Latency = 5 ms
Max. Latency = 146 ms
Mean Latency = 21.468 ms

B.1.5 Thread 250 Pengujian ke - 2

threadName	success	Latency
Thread Group 1-4	true	8
Thread Group 1-3	true	11
Thread Group 1-1	true	12
Thread Group 1-2	true	34
Thread Group 1-7	true	10
Thread Group 1-10	true	11
Thread Group 1-8	true	18

Thread Group 1-5	true	25
Thread Group 1-9	true	11
Thread Group 1-11	true	20
Thread Group 1-6	true	21
Thread Group 1-13	true	19
Thread Group 1-12	true	11
Thread Group 1-16	true	14
Thread Group 1-14	true	19
Thread Group 1-19	true	14
Thread Group 1-15	true	16
Thread Group 1-18	true	49
Thread Group 1-17	true	24
Thread Group 1-23	true	8
Thread Group 1-20	true	34
Thread Group 1-22	true	13
Thread Group 1-25	true	10
Thread Group 1-21	true	9
Thread Group 1-26	true	29
Thread Group 1-27	true	6
Thread Group 1-24	true	8
Thread Group 1-31	true	6
Thread Group 1-30	true	9
Thread Group 1-33	true	14
Thread Group 1-34	true	14
Thread Group 1-32	true	13
Thread Group 1-29	true	18
Thread Group 1-28	true	15
Thread Group 1-37	true	16
Thread Group 1-36	true	20
Thread Group 1-35	true	15
Thread Group 1-39	true	18
Thread Group 1-44	true	6
Thread Group 1-43	true	13
Thread Group 1-40	true	18
Thread Group 1-38	true	5



Thread Group 1-41	true	10
Thread Group 1-45	true	6
Thread Group 1-42	true	11
Thread Group 1-51	true	7
Thread Group 1-46	true	7
Thread Group 1-52	true	9
Thread Group 1-48	true	10
Thread Group 1-47	true	9
Thread Group 1-49	true	13
Thread Group 1-50	true	8
Thread Group 1-55	true	7
Thread Group 1-56	true	17
Thread Group 1-54	true	27
Thread Group 1-57	true	8
Thread Group 1-59	true	8
Thread Group 1-60	true	22
Thread Group 1-53	true	4
Thread Group 1-58	true	28
Thread Group 1-64	true	19
Thread Group 1-67	true	17
Thread Group 1-62	true	20
Thread Group 1-69	true	5
Thread Group 1-68	true	16
Thread Group 1-63	true	29
Thread Group 1-66	true	50
Thread Group 1-73	true	13
Thread Group 1-71	true	18
Thread Group 1-72	true	13
Thread Group 1-61	true	13
Thread Group 1-75	true	13
Thread Group 1-74	true	29
Thread Group 1-70	true	26
Thread Group 1-76	true	31
Thread Group 1-65	true	18
Thread Group 1-78	true	8

Thread Group 1-77	true	19
Thread Group 1-79	true	16
Thread Group 1-80	true	10
Thread Group 1-81	true	17
Thread Group 1-83	true	21
Thread Group 1-82	true	10
Thread Group 1-85	true	10
Thread Group 1-84	true	18
Thread Group 1-86	true	6
Thread Group 1-87	true	7
Thread Group 1-88	true	11
Thread Group 1-91	true	11
Thread Group 1-89	true	5
Thread Group 1-90	true	7
Thread Group 1-94	true	7
Thread Group 1-92	true	7
Thread Group 1-93	true	5
Thread Group 1-95	true	5
Thread Group 1-96	true	10
Thread Group 1-119	true	5
Thread Group 1-118	true	4
Thread Group 1-117	true	4
Thread Group 1-116	true	7
Thread Group 1-115	true	9
Thread Group 1-114	true	7
Thread Group 1-113	true	6
Thread Group 1-110	true	8
Thread Group 1-112	true	6
Thread Group 1-111	true	11
Thread Group 1-109	true	10
Thread Group 1-105	true	12
Thread Group 1-108	true	8
Thread Group 1-106	true	7
Thread Group 1-107	true	7
Thread Group 1-103	true	9



Thread Group 1-99	true	10
Thread Group 1-101	true	12
Thread Group 1-104	true	6
Thread Group 1-102	true	9
Thread Group 1-100	true	8
Thread Group 1-98	true	9
Thread Group 1-97	true	10
Thread Group 1-121	true	16
Thread Group 1-124	true	12
Thread Group 1-120	true	23
Thread Group 1-122	true	36
Thread Group 1-125	true	23
Thread Group 1-127	true	5
Thread Group 1-123	true	15
Thread Group 1-126	true	10
Thread Group 1-128	true	21
Thread Group 1-129	true	10
Thread Group 1-132	true	12
Thread Group 1-130	true	10
Thread Group 1-131	true	13
Thread Group 1-133	true	14
Thread Group 1-134	true	6
Thread Group 1-136	true	14
Thread Group 1-135	true	6
Thread Group 1-137	true	7
Thread Group 1-139	true	5
Thread Group 1-138	true	14
Thread Group 1-142	true	5
Thread Group 1-144	true	9
Thread Group 1-141	true	5
Thread Group 1-140	true	5
Thread Group 1-143	true	15
Thread Group 1-147	true	15
Thread Group 1-149	true	17
Thread Group 1-145	true	20

Thread Group 1-146	true	23
Thread Group 1-148	true	10
Thread Group 1-151	true	8
Thread Group 1-152	true	9
Thread Group 1-150	true	11
Thread Group 1-154	true	10
Thread Group 1-153	true	9
Thread Group 1-155	true	5
Thread Group 1-156	true	11
Thread Group 1-158	true	12
Thread Group 1-161	true	13
Thread Group 1-159	true	7
Thread Group 1-160	true	8
Thread Group 1-163	true	10
Thread Group 1-157	true	11
Thread Group 1-166	true	4
Thread Group 1-164	true	9
Thread Group 1-165	true	8
Thread Group 1-162	true	31
Thread Group 1-169	true	11
Thread Group 1-167	true	16
Thread Group 1-168	true	6
Thread Group 1-171	true	5
Thread Group 1-170	true	7
Thread Group 1-172	true	5
Thread Group 1-173	true	7
Thread Group 1-174	true	13
Thread Group 1-177	true	5
Thread Group 1-178	true	5
Thread Group 1-176	true	6
Thread Group 1-175	true	8
Thread Group 1-179	true	15
Thread Group 1-183	true	11
Thread Group 1-181	true	8
Thread Group 1-180	true	14



Thread Group 1-182	true	18
Thread Group 1-185	true	11
Thread Group 1-187	true	5
Thread Group 1-186	true	29
Thread Group 1-188	true	6
Thread Group 1-184	true	17
Thread Group 1-189	true	5
Thread Group 1-190	true	10
Thread Group 1-191	true	6
Thread Group 1-193	true	7
Thread Group 1-192	true	46
Thread Group 1-196	true	6
Thread Group 1-198	true	5
Thread Group 1-194	true	69
Thread Group 1-197	true	7
Thread Group 1-199	true	7
Thread Group 1-195	true	6
Thread Group 1-201	true	7
Thread Group 1-204	true	9
Thread Group 1-205	true	8
Thread Group 1-203	true	11
Thread Group 1-206	true	10
Thread Group 1-200	true	11
Thread Group 1-207	true	25
Thread Group 1-202	true	8
Thread Group 1-210	true	6
Thread Group 1-209	true	12
Thread Group 1-208	true	21
Thread Group 1-211	true	14
Thread Group 1-213	true	24
Thread Group 1-215	true	14
Thread Group 1-214	true	9
Thread Group 1-212	true	23
Thread Group 1-217	true	9
Thread Group 1-216	true	19

Thread Group 1-221	true	10
Thread Group 1-218	true	9
Thread Group 1-222	true	11
Thread Group 1-220	true	8
Thread Group 1-219	true	9
Thread Group 1-223	true	11
Thread Group 1-224	true	9
Thread Group 1-225	true	9
Thread Group 1-226	true	13
Thread Group 1-227	true	7
Thread Group 1-228	true	7
Thread Group 1-230	true	10
Thread Group 1-229	true	6
Thread Group 1-232	true	6
Thread Group 1-233	true	5
Thread Group 1-234	true	7
Thread Group 1-231	true	13
Thread Group 1-235	true	9
Thread Group 1-236	true	12
Thread Group 1-237	true	6
Thread Group 1-238	true	6
Thread Group 1-239	true	9
Thread Group 1-241	true	8
Thread Group 1-240	true	13
Thread Group 1-242	true	6
Thread Group 1-243	true	8
Thread Group 1-246	true	6
Thread Group 1-245	true	7
Thread Group 1-244	true	22
Thread Group 1-247	true	9
Thread Group 1-248	true	5
Thread Group 1-249	true	4
Thread Group 1-250	true	5

Success rate = 100%

Min. Latency = 4 ms



Max. Latency = 69 ms
 Mean Latency = 12.392 ms

B.1.6 Thread 250 Pengujian ke - 3

threadName	success	Latency
Thread Group 1-6	true	12
Thread Group 1-1	true	10
Thread Group 1-5	true	8
Thread Group 1-2	true	20
Thread Group 1-4	true	8
Thread Group 1-3	true	20
Thread Group 1-8	true	41
Thread Group 1-7	true	41
Thread Group 1-10	true	5
Thread Group 1-9	true	7
Thread Group 1-11	true	7
Thread Group 1-15	true	8
Thread Group 1-12	true	10
Thread Group 1-14	true	33
Thread Group 1-19	true	19
Thread Group 1-13	true	13
Thread Group 1-18	true	20
Thread Group 1-16	true	15
Thread Group 1-17	true	10
Thread Group 1-20	true	16
Thread Group 1-21	true	17
Thread Group 1-22	true	16
Thread Group 1-23	true	7
Thread Group 1-24	true	8
Thread Group 1-25	true	10
Thread Group 1-27	true	8
Thread Group 1-26	true	11
Thread Group 1-29	true	10
Thread Group 1-31	true	8
Thread Group 1-28	true	6

Thread Group 1-32	true	7
Thread Group 1-30	true	7
Thread Group 1-33	true	10
Thread Group 1-36	true	7
Thread Group 1-35	true	8
Thread Group 1-37	true	7
Thread Group 1-34	true	9
Thread Group 1-38	true	12
Thread Group 1-40	true	7
Thread Group 1-39	true	5
Thread Group 1-41	true	6
Thread Group 1-43	true	6
Thread Group 1-44	true	8
Thread Group 1-42	true	8
Thread Group 1-45	true	9
Thread Group 1-46	true	6
Thread Group 1-48	true	19
Thread Group 1-49	true	8
Thread Group 1-47	true	10
Thread Group 1-51	true	7
Thread Group 1-50	true	5
Thread Group 1-52	true	8
Thread Group 1-54	true	12
Thread Group 1-53	true	14
Thread Group 1-55	true	5
Thread Group 1-56	true	6
Thread Group 1-58	true	11
Thread Group 1-57	true	6
Thread Group 1-60	true	5
Thread Group 1-59	true	11
Thread Group 1-62	true	8
Thread Group 1-61	true	10
Thread Group 1-63	true	7
Thread Group 1-64	true	19
Thread Group 1-69	true	10



Thread Group 1-67	true	8
Thread Group 1-70	true	16
Thread Group 1-65	true	10
Thread Group 1-71	true	7
Thread Group 1-68	true	13
Thread Group 1-73	true	8
Thread Group 1-66	true	9
Thread Group 1-72	true	15
Thread Group 1-74	true	21
Thread Group 1-75	true	10
Thread Group 1-76	true	8
Thread Group 1-77	true	13
Thread Group 1-78	true	34
Thread Group 1-79	true	25
Thread Group 1-80	true	25
Thread Group 1-82	true	32
Thread Group 1-84	true	29
Thread Group 1-81	true	31
Thread Group 1-85	true	31
Thread Group 1-87	true	37
Thread Group 1-83	true	31
Thread Group 1-88	true	37
Thread Group 1-89	true	55
Thread Group 1-86	true	35
Thread Group 1-90	true	36
Thread Group 1-91	true	34
Thread Group 1-93	true	34
Thread Group 1-92	true	27
Thread Group 1-95	true	29
Thread Group 1-94	true	34
Thread Group 1-96	true	40
Thread Group 1-97	true	43
Thread Group 1-98	true	41
Thread Group 1-99	true	49
Thread Group 1-101	true	46

Thread Group 1-100	true	39
Thread Group 1-103	true	36
Thread Group 1-102	true	43
Thread Group 1-104	true	89
Thread Group 1-107	true	56
Thread Group 1-105	true	56
Thread Group 1-108	true	98
Thread Group 1-109	true	70
Thread Group 1-106	true	71
Thread Group 1-117	true	71
Thread Group 1-115	true	61
Thread Group 1-116	true	69
Thread Group 1-110	true	70
Thread Group 1-111	true	68
Thread Group 1-112	true	75
Thread Group 1-122	true	86
Thread Group 1-120	true	91
Thread Group 1-114	true	87
Thread Group 1-113	true	101
Thread Group 1-118	true	97
Thread Group 1-125	true	90
Thread Group 1-121	true	79
Thread Group 1-129	true	83
Thread Group 1-123	true	84
Thread Group 1-134	true	93
Thread Group 1-119	true	85
Thread Group 1-131	true	86
Thread Group 1-130	true	83
Thread Group 1-132	true	87
Thread Group 1-127	true	87
Thread Group 1-133	true	86
Thread Group 1-126	true	87
Thread Group 1-135	true	91
Thread Group 1-128	true	92
Thread Group 1-137	true	99



Thread Group 1-136	true	109
Thread Group 1-124	true	112
Thread Group 1-138	true	133
Thread Group 1-140	true	131
Thread Group 1-139	true	131
Thread Group 1-141	true	126
Thread Group 1-142	true	143
Thread Group 1-143	true	141
Thread Group 1-144	true	153
Thread Group 1-145	true	104
Thread Group 1-149	true	111
Thread Group 1-147	true	100
Thread Group 1-146	true	112
Thread Group 1-148	true	98
Thread Group 1-150	true	109
Thread Group 1-151	true	60
Thread Group 1-153	true	53
Thread Group 1-152	true	53
Thread Group 1-156	true	51
Thread Group 1-158	true	62
Thread Group 1-155	true	65
Thread Group 1-160	true	91
Thread Group 1-154	true	82
Thread Group 1-159	true	66
Thread Group 1-162	true	65
Thread Group 1-166	true	68
Thread Group 1-161	true	70
Thread Group 1-165	true	68
Thread Group 1-164	true	64
Thread Group 1-169	true	63
Thread Group 1-174	true	58
Thread Group 1-172	true	58
Thread Group 1-163	true	70
Thread Group 1-173	true	62
Thread Group 1-170	true	53

Thread Group 1-168	true	64
Thread Group 1-175	true	64
Thread Group 1-171	true	45
Thread Group 1-178	true	41
Thread Group 1-177	true	63
Thread Group 1-167	true	41
Thread Group 1-179	true	38
Thread Group 1-183	true	34
Thread Group 1-176	true	26
Thread Group 1-180	true	33
Thread Group 1-182	true	37
Thread Group 1-185	true	24
Thread Group 1-184	true	24
Thread Group 1-186	true	24
Thread Group 1-181	true	19
Thread Group 1-187	true	24
Thread Group 1-189	true	22
Thread Group 1-190	true	10
Thread Group 1-188	true	10
Thread Group 1-191	true	15
Thread Group 1-192	true	12
Thread Group 1-194	true	8
Thread Group 1-193	true	13
Thread Group 1-195	true	37
Thread Group 1-197	true	4
Thread Group 1-201	true	9
Thread Group 1-200	true	7
Thread Group 1-202	true	7
Thread Group 1-207	true	5
Thread Group 1-208	true	9
Thread Group 1-205	true	8
Thread Group 1-204	true	7
Thread Group 1-209	true	9
Thread Group 1-210	true	12
Thread Group 1-196	true	7



Thread Group 1-199	true	12
Thread Group 1-206	true	14
Thread Group 1-198	true	8
Thread Group 1-203	true	14
Thread Group 1-211	true	21
Thread Group 1-212	true	6
Thread Group 1-213	true	7
Thread Group 1-214	true	8
Thread Group 1-215	true	15
Thread Group 1-217	true	7
Thread Group 1-216	true	7
Thread Group 1-219	true	5
Thread Group 1-218	true	9
Thread Group 1-220	true	8
Thread Group 1-221	true	5
Thread Group 1-224	true	10
Thread Group 1-225	true	5
Thread Group 1-223	true	17
Thread Group 1-222	true	6
Thread Group 1-227	true	7
Thread Group 1-226	true	9
Thread Group 1-228	true	18
Thread Group 1-229	true	9
Thread Group 1-230	true	11
Thread Group 1-231	true	5
Thread Group 1-232	true	10
Thread Group 1-233	true	6
Thread Group 1-236	true	5
Thread Group 1-234	true	6
Thread Group 1-238	true	6
Thread Group 1-235	true	6
Thread Group 1-237	true	5
Thread Group 1-239	true	15
Thread Group 1-241	true	4
Thread Group 1-240	true	6

Thread Group 1-243	true	9
Thread Group 1-245	true	9
Thread Group 1-244	true	7
Thread Group 1-246	true	7
Thread Group 1-247	true	6
Thread Group 1-248	true	11
Thread Group 1-249	true	9
Thread Group 1-242	true	9
Thread Group 1-157	false	0
Thread Group 1-250	true	4

Success rate = 99.6%
 Min. Latency = 4 ms
 Max. Latency = 153 ms
 Mean Latency = 35.02 ms

B.1.7 Thread 500 Pengujian ke - 1

threadName	success	Latency
Thread Group 1-1	true	32
Thread Group 1-2	true	30
Thread Group 1-3	true	56
Thread Group 1-10	true	12
Thread Group 1-4	true	65
Thread Group 1-5	true	52
Thread Group 1-6	true	52
Thread Group 1-9	true	19
Thread Group 1-8	true	22
Thread Group 1-12	true	26
Thread Group 1-11	true	31
Thread Group 1-17	true	41
Thread Group 1-7	true	42
Thread Group 1-20	true	45
Thread Group 1-15	true	49
Thread Group 1-16	true	49
Thread Group 1-14	true	53
Thread Group 1-19	true	52



Thread Group 1-13	true	54
Thread Group 1-18	true	62
Thread Group 1-21	true	41
Thread Group 1-22	true	58
Thread Group 1-26	true	59
Thread Group 1-30	true	60
Thread Group 1-31	true	64
Thread Group 1-23	true	58
Thread Group 1-28	true	55
Thread Group 1-25	true	55
Thread Group 1-27	true	46
Thread Group 1-33	true	48
Thread Group 1-24	true	49
Thread Group 1-29	true	52
Thread Group 1-32	true	54
Thread Group 1-35	true	52
Thread Group 1-34	true	52
Thread Group 1-36	true	55
Thread Group 1-37	true	54
Thread Group 1-38	true	45
Thread Group 1-39	true	52
Thread Group 1-43	true	56
Thread Group 1-42	true	59
Thread Group 1-45	true	62
Thread Group 1-46	true	63
Thread Group 1-40	true	68
Thread Group 1-41	true	70
Thread Group 1-44	true	74
Thread Group 1-48	true	67
Thread Group 1-51	true	52
Thread Group 1-49	true	44
Thread Group 1-53	true	40
Thread Group 1-47	true	93
Thread Group 1-50	true	99
Thread Group 1-52	true	86

Thread Group 1-56	true	41
Thread Group 1-54	true	54
Thread Group 1-59	true	25
Thread Group 1-55	true	33
Thread Group 1-58	true	36
Thread Group 1-65	true	17
Thread Group 1-60	true	19
Thread Group 1-63	true	33
Thread Group 1-61	true	17
Thread Group 1-57	true	66
Thread Group 1-70	true	9
Thread Group 1-64	true	9
Thread Group 1-62	true	23
Thread Group 1-68	true	18
Thread Group 1-69	true	12
Thread Group 1-67	true	9
Thread Group 1-66	true	5
Thread Group 1-71	true	9
Thread Group 1-75	true	7
Thread Group 1-74	true	9
Thread Group 1-76	true	10
Thread Group 1-77	true	7
Thread Group 1-78	true	5
Thread Group 1-79	true	8
Thread Group 1-80	true	6
Thread Group 1-84	true	6
Thread Group 1-81	true	7
Thread Group 1-82	true	8
Thread Group 1-85	true	7
Thread Group 1-86	true	13
Thread Group 1-83	true	13
Thread Group 1-88	true	5
Thread Group 1-87	true	9
Thread Group 1-89	true	5
Thread Group 1-90	true	8



Thread Group 1-91	true	7
Thread Group 1-92	true	9
Thread Group 1-94	true	4
Thread Group 1-93	true	11
Thread Group 1-95	true	7
Thread Group 1-96	true	6
Thread Group 1-98	true	7
Thread Group 1-119	true	5
Thread Group 1-118	true	7
Thread Group 1-117	true	7
Thread Group 1-116	true	6
Thread Group 1-115	true	9
Thread Group 1-114	true	9
Thread Group 1-113	true	6
Thread Group 1-111	true	5
Thread Group 1-112	true	4
Thread Group 1-109	true	5
Thread Group 1-110	true	6
Thread Group 1-120	true	6
Thread Group 1-108	true	6
Thread Group 1-107	true	6
Thread Group 1-106	true	5
Thread Group 1-105	true	7
Thread Group 1-104	true	5
Thread Group 1-103	true	7
Thread Group 1-102	true	5
Thread Group 1-101	true	6
Thread Group 1-99	true	6
Thread Group 1-100	true	5
Thread Group 1-97	true	15
Thread Group 1-73	true	6
Thread Group 1-121	true	14
Thread Group 1-72	true	14
Thread Group 1-125	true	13
Thread Group 1-124	true	8

Thread Group 1-126	true	6
Thread Group 1-123	true	11
Thread Group 1-122	true	16
Thread Group 1-127	true	8
Thread Group 1-128	true	8
Thread Group 1-129	true	9
Thread Group 1-130	true	5
Thread Group 1-131	true	8
Thread Group 1-132	true	8
Thread Group 1-133	true	5
Thread Group 1-135	true	6
Thread Group 1-134	true	13
Thread Group 1-136	true	7
Thread Group 1-138	true	8
Thread Group 1-137	true	7
Thread Group 1-139	true	6
Thread Group 1-141	true	5
Thread Group 1-142	true	14
Thread Group 1-140	true	21
Thread Group 1-144	true	11
Thread Group 1-143	true	15
Thread Group 1-145	true	12
Thread Group 1-146	true	10
Thread Group 1-147	true	22
Thread Group 1-151	true	34
Thread Group 1-150	true	11
Thread Group 1-152	true	15
Thread Group 1-153	true	16
Thread Group 1-154	true	18
Thread Group 1-148	true	17
Thread Group 1-149	true	20
Thread Group 1-157	true	5
Thread Group 1-155	true	7
Thread Group 1-158	true	8
Thread Group 1-156	true	10



Thread Group 1-160	true	6
Thread Group 1-159	true	7
Thread Group 1-165	true	11
Thread Group 1-162	true	10
Thread Group 1-163	true	6
Thread Group 1-164	true	6
Thread Group 1-161	true	10
Thread Group 1-166	true	12
Thread Group 1-169	true	5
Thread Group 1-167	true	17
Thread Group 1-170	true	8
Thread Group 1-168	true	11
Thread Group 1-171	true	8
Thread Group 1-173	true	4
Thread Group 1-172	true	11
Thread Group 1-175	true	6
Thread Group 1-176	true	6
Thread Group 1-174	true	9
Thread Group 1-177	true	16
Thread Group 1-178	true	11
Thread Group 1-179	true	8
Thread Group 1-180	true	6
Thread Group 1-181	true	15
Thread Group 1-182	true	7
Thread Group 1-183	true	18
Thread Group 1-184	true	20
Thread Group 1-186	true	6
Thread Group 1-185	true	6
Thread Group 1-187	true	8
Thread Group 1-188	true	9
Thread Group 1-191	true	5
Thread Group 1-190	true	6
Thread Group 1-192	true	13
Thread Group 1-189	true	8
Thread Group 1-193	true	7

Thread Group 1-194	true	5
Thread Group 1-196	true	18
Thread Group 1-198	true	11
Thread Group 1-197	true	23
Thread Group 1-199	true	7
Thread Group 1-204	true	15
Thread Group 1-200	true	30
Thread Group 1-203	true	30
Thread Group 1-202	true	40
Thread Group 1-206	true	39
Thread Group 1-205	true	29
Thread Group 1-201	true	29
Thread Group 1-195	true	36
Thread Group 1-207	true	31
Thread Group 1-209	true	40
Thread Group 1-208	true	43
Thread Group 1-211	true	38
Thread Group 1-212	true	38
Thread Group 1-210	true	40
Thread Group 1-214	true	41
Thread Group 1-215	true	50
Thread Group 1-213	true	41
Thread Group 1-216	true	42
Thread Group 1-219	true	42
Thread Group 1-218	true	47
Thread Group 1-217	true	53
Thread Group 1-220	true	48
Thread Group 1-222	true	46
Thread Group 1-221	true	45
Thread Group 1-223	true	46
Thread Group 1-224	true	47
Thread Group 1-226	true	40
Thread Group 1-227	true	51
Thread Group 1-228	true	45
Thread Group 1-230	true	57



Thread Group 1-229	true	59
Thread Group 1-225	true	61
Thread Group 1-232	true	55
Thread Group 1-231	true	60
Thread Group 1-233	true	71
Thread Group 1-234	true	69
Thread Group 1-236	true	74
Thread Group 1-235	true	74
Thread Group 1-237	true	78
Thread Group 1-238	true	78
Thread Group 1-239	true	85
Thread Group 1-241	true	71
Thread Group 1-240	true	70
Thread Group 1-242	true	80
Thread Group 1-243	true	74
Thread Group 1-244	true	77
Thread Group 1-246	true	75
Thread Group 1-245	true	76
Thread Group 1-247	true	77
Thread Group 1-250	true	72
Thread Group 1-248	true	71
Thread Group 1-249	true	74
Thread Group 1-252	true	71
Thread Group 1-251	true	80
Thread Group 1-253	true	84
Thread Group 1-254	true	80
Thread Group 1-255	true	85
Thread Group 1-257	true	97
Thread Group 1-256	true	103
Thread Group 1-258	true	105
Thread Group 1-261	true	110
Thread Group 1-259	true	109
Thread Group 1-260	true	112
Thread Group 1-262	true	131
Thread Group 1-264	true	132

Thread Group 1-263	true	133
Thread Group 1-265	true	140
Thread Group 1-267	true	143
Thread Group 1-266	true	142
Thread Group 1-268	true	138
Thread Group 1-269	true	144
Thread Group 1-271	true	138
Thread Group 1-272	true	139
Thread Group 1-273	true	141
Thread Group 1-274	true	145
Thread Group 1-270	true	145
Thread Group 1-277	true	150
Thread Group 1-275	true	161
Thread Group 1-278	true	167
Thread Group 1-276	true	171
Thread Group 1-279	true	178
Thread Group 1-280	true	179
Thread Group 1-281	true	175
Thread Group 1-283	true	191
Thread Group 1-282	true	195
Thread Group 1-284	true	195
Thread Group 1-285	true	193
Thread Group 1-286	true	197
Thread Group 1-287	true	193
Thread Group 1-291	true	198
Thread Group 1-289	true	202
Thread Group 1-290	true	193
Thread Group 1-288	true	220
Thread Group 1-292	true	207
Thread Group 1-294	true	190
Thread Group 1-293	true	187
Thread Group 1-295	true	196
Thread Group 1-296	true	200
Thread Group 1-297	true	199
Thread Group 1-298	true	194



Thread Group 1-299	true	200
Thread Group 1-300	true	203
Thread Group 1-302	true	198
Thread Group 1-304	true	199
Thread Group 1-305	true	199
Thread Group 1-303	true	200
Thread Group 1-307	true	202
Thread Group 1-301	true	202
Thread Group 1-309	true	210
Thread Group 1-306	true	215
Thread Group 1-310	true	216
Thread Group 1-311	true	214
Thread Group 1-308	true	235
Thread Group 1-314	true	216
Thread Group 1-313	true	214
Thread Group 1-312	true	227
Thread Group 1-315	true	212
Thread Group 1-316	true	218
Thread Group 1-317	true	227
Thread Group 1-320	true	226
Thread Group 1-318	true	226
Thread Group 1-319	true	223
Thread Group 1-322	true	223
Thread Group 1-321	true	228
Thread Group 1-324	true	219
Thread Group 1-325	true	222
Thread Group 1-323	true	226
Thread Group 1-326	true	226
Thread Group 1-327	true	228
Thread Group 1-328	true	217
Thread Group 1-329	true	201
Thread Group 1-330	true	196
Thread Group 1-331	true	180
Thread Group 1-335	true	164
Thread Group 1-336	true	164

Thread Group 1-334	true	153
Thread Group 1-333	true	157
Thread Group 1-332	true	148
Thread Group 1-340	true	152
Thread Group 1-339	true	158
Thread Group 1-338	true	174
Thread Group 1-337	true	169
Thread Group 1-343	true	142
Thread Group 1-344	true	149
Thread Group 1-341	true	154
Thread Group 1-342	true	160
Thread Group 1-345	true	144
Thread Group 1-346	true	138
Thread Group 1-349	true	120
Thread Group 1-352	true	125
Thread Group 1-353	true	129
Thread Group 1-350	true	135
Thread Group 1-355	true	114
Thread Group 1-356	true	117
Thread Group 1-354	true	113
Thread Group 1-359	true	110
Thread Group 1-351	true	127
Thread Group 1-360	true	117
Thread Group 1-357	true	113
Thread Group 1-362	true	94
Thread Group 1-363	true	95
Thread Group 1-358	true	109
Thread Group 1-361	true	112
Thread Group 1-366	true	97
Thread Group 1-348	true	124
Thread Group 1-347	true	135
Thread Group 1-364	true	96
Thread Group 1-367	true	188
Thread Group 1-383	true	111
Thread Group 1-376	true	147



Thread Group 1-379	true	134
Thread Group 1-377	true	152
Thread Group 1-380	true	135
Thread Group 1-378	true	120
Thread Group 1-370	true	189
Thread Group 1-372	true	133
Thread Group 1-369	true	190
Thread Group 1-375	true	175
Thread Group 1-392	true	18
Thread Group 1-389	true	18
Thread Group 1-393	true	16
Thread Group 1-387	true	28
Thread Group 1-373	true	170
Thread Group 1-371	true	192
Thread Group 1-374	true	195
Thread Group 1-391	true	35
Thread Group 1-394	true	49
Thread Group 1-390	true	46
Thread Group 1-384	true	187
Thread Group 1-386	true	45
Thread Group 1-388	true	49
Thread Group 1-396	true	21
Thread Group 1-381	true	205
Thread Group 1-385	true	202
Thread Group 1-368	true	270
Thread Group 1-365	true	279
Thread Group 1-399	true	27
Thread Group 1-400	true	17
Thread Group 1-395	true	38
Thread Group 1-382	true	19
Thread Group 1-397	true	10
Thread Group 1-402	true	10
Thread Group 1-403	true	11
Thread Group 1-401	true	11
Thread Group 1-405	true	13

Thread Group 1-404	true	11
Thread Group 1-406	true	7
Thread Group 1-408	true	8
Thread Group 1-407	true	13
Thread Group 1-409	true	10
Thread Group 1-410	true	5
Thread Group 1-412	true	4
Thread Group 1-411	true	8
Thread Group 1-413	true	7
Thread Group 1-414	true	12
Thread Group 1-418	true	5
Thread Group 1-420	true	5
Thread Group 1-416	true	9
Thread Group 1-398	true	5
Thread Group 1-421	true	6
Thread Group 1-422	true	10
Thread Group 1-424	true	9
Thread Group 1-426	true	11
Thread Group 1-425	true	11
Thread Group 1-428	true	11
Thread Group 1-427	true	11
Thread Group 1-432	true	12
Thread Group 1-430	true	11
Thread Group 1-431	true	11
Thread Group 1-415	true	9
Thread Group 1-434	true	16
Thread Group 1-429	true	19
Thread Group 1-417	true	13
Thread Group 1-433	true	25
Thread Group 1-435	true	16
Thread Group 1-436	true	6
Thread Group 1-437	true	6
Thread Group 1-439	true	7
Thread Group 1-438	true	9
Thread Group 1-440	true	6



Thread Group 1-441	true	10
Thread Group 1-423	true	7
Thread Group 1-442	true	11
Thread Group 1-419	true	7
Thread Group 1-445	true	7
Thread Group 1-444	true	6
Thread Group 1-443	true	10
Thread Group 1-446	true	6
Thread Group 1-448	true	6
Thread Group 1-447	true	5
Thread Group 1-450	true	6
Thread Group 1-449	true	7
Thread Group 1-451	true	8
Thread Group 1-453	true	8
Thread Group 1-452	true	11
Thread Group 1-455	true	9
Thread Group 1-454	true	12
Thread Group 1-456	true	8
Thread Group 1-457	true	9
Thread Group 1-458	true	8
Thread Group 1-459	true	12
Thread Group 1-460	true	11
Thread Group 1-461	true	18
Thread Group 1-462	true	6
Thread Group 1-468	true	14
Thread Group 1-464	true	11
Thread Group 1-466	true	15
Thread Group 1-467	true	16
Thread Group 1-469	true	16
Thread Group 1-463	true	15
Thread Group 1-471	true	14
Thread Group 1-465	true	27
Thread Group 1-470	true	12
Thread Group 1-473	true	11
Thread Group 1-472	true	13

Thread Group 1-475	true	11
Thread Group 1-474	true	16
Thread Group 1-476	true	12
Thread Group 1-478	true	8
Thread Group 1-479	true	6
Thread Group 1-477	true	13
Thread Group 1-480	true	5
Thread Group 1-481	true	7
Thread Group 1-482	true	7
Thread Group 1-484	true	7
Thread Group 1-483	true	8
Thread Group 1-485	true	8
Thread Group 1-486	true	10
Thread Group 1-487	true	5
Thread Group 1-488	true	4
Thread Group 1-489	true	8
Thread Group 1-493	true	18
Thread Group 1-490	true	7
Thread Group 1-491	true	18
Thread Group 1-495	true	9
Thread Group 1-497	true	14
Thread Group 1-496	true	19
Thread Group 1-492	true	6
Thread Group 1-498	true	6
Thread Group 1-494	true	10
Thread Group 1-499	true	8
Thread Group 1-500	true	4

Success rate = 100%
Min. Latency = 4 ms
Max. Latency = 279 ms
Mean Latency = 61.91 ms



B.1.8 Thread 500 Pengujian ke - 2

threadName	success	Latency
Thread Group 1-2	true	8
Thread Group 1-1	true	7
Thread Group 1-3	true	7
Thread Group 1-5	true	6
Thread Group 1-7	true	6
Thread Group 1-4	true	29
Thread Group 1-11	true	11
Thread Group 1-6	true	12
Thread Group 1-15	true	9
Thread Group 1-9	true	5
Thread Group 1-16	true	27
Thread Group 1-19	true	30
Thread Group 1-12	true	35
Thread Group 1-20	true	19
Thread Group 1-14	true	16
Thread Group 1-8	true	14
Thread Group 1-13	true	39
Thread Group 1-18	true	29
Thread Group 1-17	true	22
Thread Group 1-21	true	31
Thread Group 1-22	true	36
Thread Group 1-24	true	27
Thread Group 1-10	true	14
Thread Group 1-28	true	18
Thread Group 1-26	true	18
Thread Group 1-29	true	20
Thread Group 1-25	true	22
Thread Group 1-32	true	11
Thread Group 1-23	true	12
Thread Group 1-30	true	36
Thread Group 1-27	true	29
Thread Group 1-33	true	9
Thread Group 1-35	true	15

Thread Group 1-37	true	7
Thread Group 1-31	true	9
Thread Group 1-38	true	7
Thread Group 1-34	true	5
Thread Group 1-36	true	9
Thread Group 1-40	true	8
Thread Group 1-39	true	12
Thread Group 1-41	true	9
Thread Group 1-42	true	10
Thread Group 1-43	true	7
Thread Group 1-45	true	11
Thread Group 1-44	true	12
Thread Group 1-47	true	6
Thread Group 1-46	true	6
Thread Group 1-48	true	5
Thread Group 1-49	true	8
Thread Group 1-50	true	11
Thread Group 1-51	true	9
Thread Group 1-52	true	7
Thread Group 1-54	true	10
Thread Group 1-53	true	6
Thread Group 1-55	true	16
Thread Group 1-56	true	8
Thread Group 1-58	true	11
Thread Group 1-59	true	10
Thread Group 1-60	true	9
Thread Group 1-57	true	20
Thread Group 1-61	true	6
Thread Group 1-62	true	6
Thread Group 1-64	true	9
Thread Group 1-63	true	5
Thread Group 1-65	true	16
Thread Group 1-66	true	7
Thread Group 1-67	true	4
Thread Group 1-68	true	6



Thread Group 1-69	true	5
Thread Group 1-71	true	8
Thread Group 1-73	true	10
Thread Group 1-70	true	12
Thread Group 1-74	true	8
Thread Group 1-75	true	14
Thread Group 1-72	true	14
Thread Group 1-77	true	5
Thread Group 1-76	true	7
Thread Group 1-79	true	10
Thread Group 1-78	true	5
Thread Group 1-80	true	5
Thread Group 1-81	true	4
Thread Group 1-82	true	5
Thread Group 1-83	true	7
Thread Group 1-85	true	5
Thread Group 1-84	true	6
Thread Group 1-86	true	7
Thread Group 1-88	true	6
Thread Group 1-89	true	7
Thread Group 1-87	true	7
Thread Group 1-90	true	6
Thread Group 1-91	true	5
Thread Group 1-92	true	11
Thread Group 1-96	true	10
Thread Group 1-95	true	7
Thread Group 1-93	true	7
Thread Group 1-97	true	7
Thread Group 1-103	true	6
Thread Group 1-101	true	6
Thread Group 1-100	true	7
Thread Group 1-98	true	8
Thread Group 1-99	true	7
Thread Group 1-94	true	9
Thread Group 1-105	true	10

Thread Group 1-102	true	6
Thread Group 1-114	true	9
Thread Group 1-116	true	13
Thread Group 1-110	true	12
Thread Group 1-106	true	22
Thread Group 1-117	true	12
Thread Group 1-104	true	16
Thread Group 1-115	true	6
Thread Group 1-107	true	33
Thread Group 1-109	true	40
Thread Group 1-113	true	10
Thread Group 1-112	true	52
Thread Group 1-108	true	9
Thread Group 1-111	true	22
Thread Group 1-119	true	9
Thread Group 1-122	true	11
Thread Group 1-118	true	10
Thread Group 1-123	true	9
Thread Group 1-121	true	7
Thread Group 1-120	true	9
Thread Group 1-124	true	7
Thread Group 1-126	true	4
Thread Group 1-129	true	7
Thread Group 1-127	true	7
Thread Group 1-128	true	6
Thread Group 1-125	true	8
Thread Group 1-130	true	6
Thread Group 1-131	true	8
Thread Group 1-132	true	7
Thread Group 1-134	true	5
Thread Group 1-133	true	6
Thread Group 1-135	true	7
Thread Group 1-136	true	5
Thread Group 1-139	true	5
Thread Group 1-137	true	13



Thread Group 1-138	true	5
Thread Group 1-142	true	4
Thread Group 1-140	true	21
Thread Group 1-141	true	7
Thread Group 1-146	true	15
Thread Group 1-144	true	15
Thread Group 1-151	true	22
Thread Group 1-150	true	20
Thread Group 1-149	true	17
Thread Group 1-143	true	29
Thread Group 1-145	true	31
Thread Group 1-147	true	39
Thread Group 1-148	true	50
Thread Group 1-153	true	36
Thread Group 1-154	true	21
Thread Group 1-152	true	18
Thread Group 1-159	true	24
Thread Group 1-160	true	62
Thread Group 1-156	true	65
Thread Group 1-158	true	109
Thread Group 1-162	true	110
Thread Group 1-155	true	119
Thread Group 1-165	true	232
Thread Group 1-157	true	229
Thread Group 1-163	true	201
Thread Group 1-170	true	207
Thread Group 1-167	true	225
Thread Group 1-168	true	242
Thread Group 1-164	true	222
Thread Group 1-166	true	257
Thread Group 1-161	true	256
Thread Group 1-172	true	243
Thread Group 1-178	true	238
Thread Group 1-176	true	239
Thread Group 1-175	true	240

Thread Group 1-174	true	241
Thread Group 1-173	true	251
Thread Group 1-181	true	244
Thread Group 1-179	true	254
Thread Group 1-180	true	240
Thread Group 1-193	true	210
Thread Group 1-197	true	217
Thread Group 1-171	true	216
Thread Group 1-189	true	218
Thread Group 1-185	true	223
Thread Group 1-188	true	222
Thread Group 1-192	true	222
Thread Group 1-186	true	225
Thread Group 1-184	true	230
Thread Group 1-182	true	231
Thread Group 1-191	true	231
Thread Group 1-177	true	229
Thread Group 1-183	true	232
Thread Group 1-190	true	232
Thread Group 1-187	true	237
Thread Group 1-196	true	234
Thread Group 1-194	true	235
Thread Group 1-201	true	239
Thread Group 1-206	true	239
Thread Group 1-209	true	242
Thread Group 1-198	true	285
Thread Group 1-199	true	282
Thread Group 1-195	true	281
Thread Group 1-211	true	246
Thread Group 1-216	true	246
Thread Group 1-212	true	243
Thread Group 1-217	true	247
Thread Group 1-219	true	249
Thread Group 1-202	true	277
Thread Group 1-222	true	255



Thread Group 1-207	true	284
Thread Group 1-215	true	272
Thread Group 1-208	true	271
Thread Group 1-204	true	276
Thread Group 1-205	true	288
Thread Group 1-210	true	277
Thread Group 1-214	true	301
Thread Group 1-213	true	261
Thread Group 1-218	true	263
Thread Group 1-200	true	310
Thread Group 1-224	true	258
Thread Group 1-220	true	265
Thread Group 1-203	true	316
Thread Group 1-221	true	258
Thread Group 1-225	true	257
Thread Group 1-223	true	256
Thread Group 1-227	true	256
Thread Group 1-226	true	261
Thread Group 1-228	true	270
Thread Group 1-230	true	262
Thread Group 1-229	true	265
Thread Group 1-234	true	268
Thread Group 1-231	true	287
Thread Group 1-235	true	220
Thread Group 1-233	true	263
Thread Group 1-232	true	229
Thread Group 1-236	true	275
Thread Group 1-169	false	0
Thread Group 1-238	true	185
Thread Group 1-237	true	191
Thread Group 1-239	true	203
Thread Group 1-241	true	197
Thread Group 1-240	true	208
Thread Group 1-242	true	195
Thread Group 1-251	true	158

Thread Group 1-246	true	157
Thread Group 1-249	true	154
Thread Group 1-252	true	151
Thread Group 1-248	true	198
Thread Group 1-244	true	223
Thread Group 1-245	true	226
Thread Group 1-250	true	161
Thread Group 1-254	true	155
Thread Group 1-243	true	255
Thread Group 1-247	true	216
Thread Group 1-253	true	153
Thread Group 1-255	true	152
Thread Group 1-259	true	155
Thread Group 1-256	true	157
Thread Group 1-257	true	155
Thread Group 1-258	true	169
Thread Group 1-261	true	151
Thread Group 1-260	true	152
Thread Group 1-263	true	155
Thread Group 1-262	true	142
Thread Group 1-264	true	144
Thread Group 1-271	true	128
Thread Group 1-280	true	129
Thread Group 1-266	true	148
Thread Group 1-267	true	154
Thread Group 1-268	true	141
Thread Group 1-279	true	115
Thread Group 1-277	true	124
Thread Group 1-281	true	117
Thread Group 1-265	true	134
Thread Group 1-282	true	119
Thread Group 1-274	true	119
Thread Group 1-286	true	128
Thread Group 1-285	true	112
Thread Group 1-283	true	131



Thread Group 1-290	true	113
Thread Group 1-292	true	114
Thread Group 1-297	true	112
Thread Group 1-288	true	127
Thread Group 1-287	true	139
Thread Group 1-295	true	88
Thread Group 1-289	true	94
Thread Group 1-294	true	97
Thread Group 1-293	true	88
Thread Group 1-272	true	204
Thread Group 1-276	true	241
Thread Group 1-273	true	247
Thread Group 1-278	true	247
Thread Group 1-269	true	257
Thread Group 1-270	true	257
Thread Group 1-275	true	118
Thread Group 1-284	true	118
Thread Group 1-304	true	98
Thread Group 1-296	true	93
Thread Group 1-300	true	99
Thread Group 1-303	true	88
Thread Group 1-301	true	96
Thread Group 1-307	true	91
Thread Group 1-306	true	106
Thread Group 1-309	true	88
Thread Group 1-308	true	88
Thread Group 1-305	true	93
Thread Group 1-313	true	79
Thread Group 1-298	true	79
Thread Group 1-299	true	79
Thread Group 1-291	true	80
Thread Group 1-312	true	63
Thread Group 1-302	true	80
Thread Group 1-318	true	53
Thread Group 1-316	true	79

Thread Group 1-315	true	82
Thread Group 1-323	true	53
Thread Group 1-314	true	68
Thread Group 1-320	true	57
Thread Group 1-322	true	61
Thread Group 1-310	true	53
Thread Group 1-319	true	56
Thread Group 1-311	true	59
Thread Group 1-317	true	48
Thread Group 1-321	true	62
Thread Group 1-327	true	37
Thread Group 1-325	true	33
Thread Group 1-328	true	43
Thread Group 1-326	true	47
Thread Group 1-324	true	47
Thread Group 1-333	true	28
Thread Group 1-330	true	26
Thread Group 1-332	true	25
Thread Group 1-329	true	28
Thread Group 1-331	true	26
Thread Group 1-334	true	25
Thread Group 1-336	true	21
Thread Group 1-335	true	26
Thread Group 1-339	true	14
Thread Group 1-340	true	17
Thread Group 1-338	true	16
Thread Group 1-337	true	18
Thread Group 1-341	true	21
Thread Group 1-342	true	13
Thread Group 1-343	true	8
Thread Group 1-344	true	6
Thread Group 1-345	true	13
Thread Group 1-347	true	10
Thread Group 1-346	true	8
Thread Group 1-348	true	22



Thread Group 1-350	true	5
Thread Group 1-349	true	6
Thread Group 1-352	true	5
Thread Group 1-351	true	5
Thread Group 1-353	true	8
Thread Group 1-355	true	8
Thread Group 1-354	true	10
Thread Group 1-357	true	5
Thread Group 1-356	true	6
Thread Group 1-358	true	7
Thread Group 1-360	true	7
Thread Group 1-361	true	7
Thread Group 1-359	true	13
Thread Group 1-363	true	4
Thread Group 1-362	true	5
Thread Group 1-364	true	5
Thread Group 1-366	true	8
Thread Group 1-365	true	10
Thread Group 1-367	true	7
Thread Group 1-368	true	6
Thread Group 1-370	true	6
Thread Group 1-371	true	4
Thread Group 1-369	true	6
Thread Group 1-372	true	5
Thread Group 1-373	true	7
Thread Group 1-375	true	4
Thread Group 1-374	true	9
Thread Group 1-376	true	9
Thread Group 1-378	true	5
Thread Group 1-377	true	5
Thread Group 1-379	true	10
Thread Group 1-380	true	4
Thread Group 1-381	true	8
Thread Group 1-382	true	11
Thread Group 1-384	true	5

Thread Group 1-383	true	9
Thread Group 1-385	true	10
Thread Group 1-387	true	44
Thread Group 1-388	true	51
Thread Group 1-386	true	17
Thread Group 1-389	true	5
Thread Group 1-391	true	10
Thread Group 1-392	true	10
Thread Group 1-390	true	5
Thread Group 1-393	true	5
Thread Group 1-394	true	7
Thread Group 1-395	true	6
Thread Group 1-396	true	13
Thread Group 1-397	true	4
Thread Group 1-399	true	14
Thread Group 1-401	true	6
Thread Group 1-400	true	5
Thread Group 1-402	true	10
Thread Group 1-398	true	9
Thread Group 1-403	true	6
Thread Group 1-404	true	6
Thread Group 1-405	true	5
Thread Group 1-406	true	4
Thread Group 1-407	true	9
Thread Group 1-410	true	4
Thread Group 1-408	true	9
Thread Group 1-411	true	5
Thread Group 1-409	true	5
Thread Group 1-412	true	7
Thread Group 1-413	true	5
Thread Group 1-414	true	15
Thread Group 1-416	true	6
Thread Group 1-415	true	21
Thread Group 1-417	true	7
Thread Group 1-418	true	6



Thread Group 1-419	true	7
Thread Group 1-420	true	6
Thread Group 1-421	true	4
Thread Group 1-422	true	6
Thread Group 1-423	true	4
Thread Group 1-424	true	5
Thread Group 1-425	true	11
Thread Group 1-426	true	4
Thread Group 1-427	true	6
Thread Group 1-428	true	4
Thread Group 1-429	true	6
Thread Group 1-430	true	4
Thread Group 1-431	true	5
Thread Group 1-432	true	6
Thread Group 1-433	true	5
Thread Group 1-435	true	6
Thread Group 1-436	true	5
Thread Group 1-434	true	5
Thread Group 1-438	true	4
Thread Group 1-439	true	8
Thread Group 1-437	true	8
Thread Group 1-441	true	9
Thread Group 1-443	true	10
Thread Group 1-442	true	8
Thread Group 1-440	true	7
Thread Group 1-444	true	7
Thread Group 1-446	true	8
Thread Group 1-447	true	8
Thread Group 1-445	true	6
Thread Group 1-448	true	4
Thread Group 1-449	true	6
Thread Group 1-451	true	7
Thread Group 1-450	true	4
Thread Group 1-452	true	9
Thread Group 1-454	true	8

Thread Group 1-453	true	11
Thread Group 1-456	true	9
Thread Group 1-455	true	10
Thread Group 1-457	true	18
Thread Group 1-458	true	20
Thread Group 1-459	true	18
Thread Group 1-460	true	14
Thread Group 1-462	true	27
Thread Group 1-461	true	29
Thread Group 1-463	true	29
Thread Group 1-465	true	33
Thread Group 1-466	true	33
Thread Group 1-467	true	33
Thread Group 1-464	true	34
Thread Group 1-468	true	36
Thread Group 1-469	true	37
Thread Group 1-470	true	34
Thread Group 1-471	true	40
Thread Group 1-473	true	43
Thread Group 1-472	true	48
Thread Group 1-474	true	50
Thread Group 1-475	true	61
Thread Group 1-476	true	53
Thread Group 1-477	true	63
Thread Group 1-478	true	63
Thread Group 1-480	true	68
Thread Group 1-479	true	63
Thread Group 1-481	true	70
Thread Group 1-482	true	78
Thread Group 1-485	true	81
Thread Group 1-484	true	84
Thread Group 1-483	true	87
Thread Group 1-486	true	86
Thread Group 1-487	true	89
Thread Group 1-489	true	88



Thread Group 1-488	true	80
Thread Group 1-491	true	76
Thread Group 1-490	true	76
Thread Group 1-492	true	66
Thread Group 1-493	true	67
Thread Group 1-494	true	62
Thread Group 1-495	true	51
Thread Group 1-496	true	46
Thread Group 1-497	true	50
Thread Group 1-499	true	40
Thread Group 1-500	true	39
Thread Group 1-498	true	40

Success rate = 99,8%

Min. Latency = 4 ms

Max. Latency = 316 ms

Mean Latency = 73.06 ms

B.1.9 Thread 500 Pengujian ke - 3

threadName	success	Latency
Thread Group 1-3	true	20
Thread Group 1-1	true	21
Thread Group 1-2	true	38
Thread Group 1-5	true	38
Thread Group 1-4	true	63
Thread Group 1-8	true	37
Thread Group 1-7	true	31
Thread Group 1-6	true	32
Thread Group 1-10	true	39
Thread Group 1-18	true	41
Thread Group 1-16	true	42
Thread Group 1-11	true	42
Thread Group 1-15	true	51
Thread Group 1-14	true	53
Thread Group 1-21	true	80
Thread Group 1-17	true	80

Thread Group 1-9	true	87
Thread Group 1-22	true	94
Thread Group 1-20	true	95
Thread Group 1-12	true	98
Thread Group 1-23	true	116
Thread Group 1-25	true	115
Thread Group 1-26	true	103
Thread Group 1-24	true	109
Thread Group 1-19	true	109
Thread Group 1-27	true	101
Thread Group 1-13	true	102
Thread Group 1-28	true	99
Thread Group 1-30	true	96
Thread Group 1-29	true	99
Thread Group 1-31	true	102
Thread Group 1-34	true	109
Thread Group 1-32	true	114
Thread Group 1-33	true	112
Thread Group 1-36	true	109
Thread Group 1-35	true	111
Thread Group 1-37	true	109
Thread Group 1-38	true	105
Thread Group 1-40	true	108
Thread Group 1-41	true	102
Thread Group 1-39	true	103
Thread Group 1-42	true	101
Thread Group 1-43	true	100
Thread Group 1-44	true	102
Thread Group 1-45	true	90
Thread Group 1-47	true	93
Thread Group 1-46	true	83
Thread Group 1-48	true	93
Thread Group 1-50	true	88
Thread Group 1-51	true	94
Thread Group 1-49	true	114



Thread Group 1-56	true	88
Thread Group 1-57	true	91
Thread Group 1-62	true	89
Thread Group 1-58	true	106
Thread Group 1-60	true	96
Thread Group 1-59	true	98
Thread Group 1-65	true	90
Thread Group 1-55	true	154
Thread Group 1-52	true	160
Thread Group 1-54	true	147
Thread Group 1-66	true	95
Thread Group 1-53	true	157
Thread Group 1-63	true	99
Thread Group 1-61	true	99
Thread Group 1-72	true	79
Thread Group 1-68	true	98
Thread Group 1-73	true	99
Thread Group 1-67	true	105
Thread Group 1-69	true	101
Thread Group 1-70	true	106
Thread Group 1-75	true	91
Thread Group 1-76	true	81
Thread Group 1-77	true	85
Thread Group 1-64	true	146
Thread Group 1-88	true	76
Thread Group 1-85	true	78
Thread Group 1-81	true	104
Thread Group 1-82	true	83
Thread Group 1-83	true	83
Thread Group 1-78	true	123
Thread Group 1-86	true	77
Thread Group 1-74	true	104
Thread Group 1-93	true	56
Thread Group 1-95	true	52
Thread Group 1-79	true	135

Thread Group 1-84	true	83
Thread Group 1-97	true	62
Thread Group 1-87	true	79
Thread Group 1-91	true	138
Thread Group 1-71	true	141
Thread Group 1-89	true	124
Thread Group 1-80	true	147
Thread Group 1-99	true	88
Thread Group 1-153	true	7
Thread Group 1-154	true	11
Thread Group 1-152	true	6
Thread Group 1-151	true	6
Thread Group 1-148	true	5
Thread Group 1-149	true	7
Thread Group 1-150	true	6
Thread Group 1-144	true	8
Thread Group 1-147	true	8
Thread Group 1-145	true	7
Thread Group 1-143	true	6
Thread Group 1-140	true	7
Thread Group 1-141	true	8
Thread Group 1-142	true	6
Thread Group 1-139	true	8
Thread Group 1-138	true	7
Thread Group 1-136	true	7
Thread Group 1-125	true	5
Thread Group 1-135	true	9
Thread Group 1-137	true	5
Thread Group 1-131	true	8
Thread Group 1-134	true	7
Thread Group 1-133	true	7
Thread Group 1-106	true	16
Thread Group 1-128	true	16
Thread Group 1-124	true	11
Thread Group 1-127	true	19

Thread Group 1-123	true	8
Thread Group 1-126	true	9
Thread Group 1-132	true	16
Thread Group 1-121	true	52
Thread Group 1-130	true	18
Thread Group 1-129	true	21
Thread Group 1-119	true	50
Thread Group 1-120	true	57
Thread Group 1-122	true	22
Thread Group 1-115	true	54
Thread Group 1-118	true	25
Thread Group 1-109	true	29
Thread Group 1-90	true	29
Thread Group 1-114	true	60
Thread Group 1-117	true	56
Thread Group 1-116	true	53
Thread Group 1-107	true	20
Thread Group 1-108	true	105
Thread Group 1-102	true	105
Thread Group 1-98	true	50
Thread Group 1-92	true	59
Thread Group 1-94	true	57
Thread Group 1-111	true	71
Thread Group 1-110	true	70
Thread Group 1-100	true	94
Thread Group 1-101	true	49
Thread Group 1-96	true	52
Thread Group 1-105	true	85
Thread Group 1-103	true	77
Thread Group 1-104	true	84
Thread Group 1-155	true	36
Thread Group 1-156	true	31
Thread Group 1-113	true	24
Thread Group 1-146	true	19
Thread Group 1-161	true	23

Thread Group 1-164	true	30
Thread Group 1-160	true	25
Thread Group 1-157	true	27
Thread Group 1-159	true	28
Thread Group 1-173	true	38
Thread Group 1-180	true	40
Thread Group 1-178	true	39
Thread Group 1-175	true	35
Thread Group 1-172	true	40
Thread Group 1-176	true	37
Thread Group 1-174	true	47
Thread Group 1-162	true	35
Thread Group 1-165	true	41
Thread Group 1-158	true	40
Thread Group 1-112	true	43
Thread Group 1-166	true	39
Thread Group 1-177	true	47
Thread Group 1-163	true	37
Thread Group 1-181	true	40
Thread Group 1-171	true	38
Thread Group 1-179	true	33
Thread Group 1-167	true	39
Thread Group 1-184	true	35
Thread Group 1-183	true	45
Thread Group 1-182	true	42
Thread Group 1-170	true	62
Thread Group 1-168	true	53
Thread Group 1-169	true	62
Thread Group 1-186	true	26
Thread Group 1-187	true	27
Thread Group 1-185	true	22
Thread Group 1-189	true	18
Thread Group 1-188	true	16
Thread Group 1-197	true	16
Thread Group 1-195	true	21



Thread Group 1-192	true	23
Thread Group 1-193	true	22
Thread Group 1-190	true	15
Thread Group 1-191	true	23
Thread Group 1-200	true	14
Thread Group 1-202	true	23
Thread Group 1-196	true	18
Thread Group 1-198	true	16
Thread Group 1-194	true	20
Thread Group 1-204	true	18
Thread Group 1-203	true	16
Thread Group 1-199	true	17
Thread Group 1-201	true	23
Thread Group 1-208	true	14
Thread Group 1-206	true	14
Thread Group 1-207	true	13
Thread Group 1-205	true	15
Thread Group 1-211	true	13
Thread Group 1-210	true	12
Thread Group 1-212	true	15
Thread Group 1-217	true	8
Thread Group 1-209	true	86
Thread Group 1-218	true	13
Thread Group 1-216	true	20
Thread Group 1-214	true	7
Thread Group 1-213	true	9
Thread Group 1-220	true	27
Thread Group 1-215	true	6
Thread Group 1-221	true	19
Thread Group 1-222	true	15
Thread Group 1-223	true	12
Thread Group 1-219	true	13
Thread Group 1-225	true	9
Thread Group 1-224	true	8
Thread Group 1-226	true	9

Thread Group 1-227	true	7
Thread Group 1-228	true	5
Thread Group 1-229	true	6
Thread Group 1-230	true	8
Thread Group 1-232	true	14
Thread Group 1-231	true	10
Thread Group 1-236	true	7
Thread Group 1-233	true	8
Thread Group 1-235	true	6
Thread Group 1-237	true	5
Thread Group 1-234	true	14
Thread Group 1-240	true	5
Thread Group 1-239	true	7
Thread Group 1-238	true	9
Thread Group 1-241	true	13
Thread Group 1-242	true	21
Thread Group 1-245	true	9
Thread Group 1-244	true	8
Thread Group 1-243	true	8
Thread Group 1-247	true	8
Thread Group 1-246	true	10
Thread Group 1-250	true	8
Thread Group 1-248	true	13
Thread Group 1-251	true	7
Thread Group 1-249	true	19
Thread Group 1-252	true	6
Thread Group 1-253	true	9
Thread Group 1-254	true	13
Thread Group 1-255	true	12
Thread Group 1-256	true	16
Thread Group 1-260	true	14
Thread Group 1-259	true	15
Thread Group 1-257	true	20
Thread Group 1-262	true	18
Thread Group 1-264	true	18



Thread Group 1-263	true	16
Thread Group 1-258	true	22
Thread Group 1-261	true	20
Thread Group 1-265	true	19
Thread Group 1-269	true	27
Thread Group 1-266	true	27
Thread Group 1-267	true	35
Thread Group 1-268	true	20
Thread Group 1-271	true	28
Thread Group 1-270	true	19
Thread Group 1-276	true	19
Thread Group 1-277	true	14
Thread Group 1-274	true	16
Thread Group 1-275	true	16
Thread Group 1-278	true	19
Thread Group 1-279	true	18
Thread Group 1-273	true	15
Thread Group 1-272	true	24
Thread Group 1-280	true	17
Thread Group 1-281	true	17
Thread Group 1-283	true	19
Thread Group 1-286	true	10
Thread Group 1-282	true	8
Thread Group 1-285	true	6
Thread Group 1-288	true	8
Thread Group 1-287	true	7
Thread Group 1-289	true	13
Thread Group 1-291	true	18
Thread Group 1-290	true	15
Thread Group 1-292	true	9
Thread Group 1-293	true	8
Thread Group 1-294	true	11
Thread Group 1-295	true	9
Thread Group 1-297	true	6
Thread Group 1-299	true	16

Thread Group 1-300	true	18
Thread Group 1-298	true	21
Thread Group 1-301	true	23
Thread Group 1-302	true	29
Thread Group 1-303	true	27
Thread Group 1-305	true	28
Thread Group 1-306	true	35
Thread Group 1-307	true	31
Thread Group 1-308	true	33
Thread Group 1-309	true	43
Thread Group 1-311	true	39
Thread Group 1-312	true	38
Thread Group 1-314	true	48
Thread Group 1-313	true	55
Thread Group 1-315	true	52
Thread Group 1-316	true	51
Thread Group 1-317	true	47
Thread Group 1-318	true	53
Thread Group 1-319	true	56
Thread Group 1-321	true	58
Thread Group 1-320	true	56
Thread Group 1-322	true	53
Thread Group 1-324	true	60
Thread Group 1-323	true	61
Thread Group 1-326	true	62
Thread Group 1-325	true	63
Thread Group 1-327	true	58
Thread Group 1-328	true	64
Thread Group 1-296	true	69
Thread Group 1-304	true	69
Thread Group 1-310	true	70
Thread Group 1-284	true	78
Thread Group 1-329	true	91
Thread Group 1-332	true	97
Thread Group 1-331	true	97

Thread Group 1-330	true	107
Thread Group 1-334	true	107
Thread Group 1-335	true	130
Thread Group 1-338	true	92
Thread Group 1-337	true	120
Thread Group 1-336	true	132
Thread Group 1-333	true	173
Thread Group 1-339	true	82
Thread Group 1-340	true	85
Thread Group 1-344	true	95
Thread Group 1-346	true	96
Thread Group 1-342	true	108
Thread Group 1-352	true	106
Thread Group 1-341	true	104
Thread Group 1-348	true	107
Thread Group 1-347	true	108
Thread Group 1-343	true	107
Thread Group 1-345	true	109
Thread Group 1-353	true	108
Thread Group 1-350	true	117
Thread Group 1-351	true	121
Thread Group 1-355	true	126
Thread Group 1-357	true	147
Thread Group 1-349	true	151
Thread Group 1-358	true	154
Thread Group 1-364	true	163
Thread Group 1-362	true	164
Thread Group 1-369	true	164
Thread Group 1-356	true	161
Thread Group 1-360	true	160
Thread Group 1-370	true	162
Thread Group 1-361	true	164
Thread Group 1-363	true	161
Thread Group 1-367	true	152
Thread Group 1-359	true	153

Thread Group 1-354	true	153
Thread Group 1-368	true	155
Thread Group 1-365	true	158
Thread Group 1-371	true	140
Thread Group 1-372	true	149
Thread Group 1-373	true	146
Thread Group 1-366	true	143
Thread Group 1-380	true	132
Thread Group 1-386	true	140
Thread Group 1-377	true	131
Thread Group 1-381	true	138
Thread Group 1-378	true	139
Thread Group 1-379	true	157
Thread Group 1-374	true	172
Thread Group 1-382	true	136
Thread Group 1-385	true	147
Thread Group 1-384	true	138
Thread Group 1-376	true	122
Thread Group 1-383	true	136
Thread Group 1-387	true	93
Thread Group 1-375	true	98
Thread Group 1-389	true	82
Thread Group 1-388	true	86
Thread Group 1-392	true	82
Thread Group 1-390	true	71
Thread Group 1-391	true	102
Thread Group 1-394	true	86
Thread Group 1-393	true	73
Thread Group 1-399	true	61
Thread Group 1-395	true	103
Thread Group 1-398	true	77
Thread Group 1-401	true	69
Thread Group 1-397	true	82
Thread Group 1-396	true	88
Thread Group 1-402	true	62

Thread Group 1-400	true	88
Thread Group 1-404	true	66
Thread Group 1-406	true	66
Thread Group 1-405	true	63
Thread Group 1-403	true	95
Thread Group 1-411	true	109
Thread Group 1-407	true	107
Thread Group 1-409	true	110
Thread Group 1-412	true	108
Thread Group 1-413	true	127
Thread Group 1-408	true	125
Thread Group 1-410	true	121
Thread Group 1-414	true	122
Thread Group 1-415	true	130
Thread Group 1-417	true	85
Thread Group 1-416	true	82
Thread Group 1-419	true	132
Thread Group 1-421	true	161
Thread Group 1-424	true	165
Thread Group 1-420	true	175
Thread Group 1-429	true	176
Thread Group 1-422	true	188
Thread Group 1-425	true	206
Thread Group 1-423	true	209
Thread Group 1-430	true	212
Thread Group 1-428	true	219
Thread Group 1-418	true	223
Thread Group 1-432	true	210
Thread Group 1-426	true	199
Thread Group 1-427	true	199
Thread Group 1-433	true	203
Thread Group 1-431	true	204
Thread Group 1-435	true	207
Thread Group 1-436	true	178
Thread Group 1-434	true	183

Thread Group 1-437	true	178
Thread Group 1-439	true	181
Thread Group 1-438	true	181
Thread Group 1-440	true	181
Thread Group 1-441	true	190
Thread Group 1-443	true	200
Thread Group 1-442	true	193
Thread Group 1-446	true	200
Thread Group 1-447	true	201
Thread Group 1-444	true	201
Thread Group 1-445	true	198
Thread Group 1-449	true	197
Thread Group 1-452	true	197
Thread Group 1-451	true	198
Thread Group 1-450	true	199
Thread Group 1-453	true	199
Thread Group 1-448	true	201
Thread Group 1-454	true	202
Thread Group 1-455	true	196
Thread Group 1-456	true	194
Thread Group 1-458	true	191
Thread Group 1-457	true	191
Thread Group 1-459	true	193
Thread Group 1-460	true	188
Thread Group 1-461	true	193
Thread Group 1-463	true	191
Thread Group 1-464	true	191
Thread Group 1-465	true	182
Thread Group 1-466	true	184
Thread Group 1-468	true	183
Thread Group 1-469	true	174
Thread Group 1-472	true	172
Thread Group 1-471	true	173
Thread Group 1-473	true	172
Thread Group 1-475	true	172



Thread Group 1-476	true	173
Thread Group 1-467	true	198
Thread Group 1-462	true	218
Thread Group 1-474	true	178
Thread Group 1-478	true	172
Thread Group 1-470	true	174
Thread Group 1-477	true	164
Thread Group 1-482	true	166
Thread Group 1-481	true	170
Thread Group 1-480	true	168
Thread Group 1-479	true	168
Thread Group 1-484	true	164
Thread Group 1-483	true	166
Thread Group 1-485	true	155
Thread Group 1-489	true	154
Thread Group 1-492	true	151
Thread Group 1-487	true	156
Thread Group 1-493	true	152
Thread Group 1-491	true	148
Thread Group 1-486	true	165
Thread Group 1-495	true	147
Thread Group 1-488	true	142
Thread Group 1-494	true	146
Thread Group 1-490	true	155
Thread Group 1-496	true	138
Thread Group 1-497	true	150
Thread Group 1-498	true	142
Thread Group 1-500	true	141
Thread Group 1-499	true	154

Total:
Mean Success rate = 99.93%
Mean Min. Latency = 5 ms
Mean Max Latency = 175.22 ms
Mean Latency = 51.62 ms

Success rate = 100%
Min. Latency = 5 ms
Max. Latency = 223 ms
Mean Latency = 80.63 ms

