

Implementasi *Constrained Application Protocol* (CoAP) pada Sistem Pengamatan Kelembaban Tanah

Yosef Febri Wiryawan¹, Dany Primanita Kartikasari², Mahendra Data³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹abi.wiryawan@gmail.com, ²dany.jalin@ub.ac.id, ³mahendra.data@ub.ac.id

Abstrak

Saat ini teknologi telah dikembangkan untuk membantu pekerjaan manusia di segala macam bidang. Salah satunya adalah di bidang pertanian. Masih kurangnya implementasi teknologi pada bidang agrikultur misalnya dalam hal observasi kelembaban tanah yang masih dilakukan secara manual. *Wireless Sensor Network* (WSN) adalah suatu arsitektur jaringan yang cocok untuk dibangun dalam menyelesaikan masalah tersebut. Di dalam WSN kita dapat membuat suatu sistem monitoring yang terdiri dari sensor, mikrokontroler dan komunikasi nirkabel antar node – node. Namun hal yang terpenting adalah bagaimana membuat suatu komunikasi yang handal di dalam sistem tersebut. Berdasarkan tujuan tersebut kita akan mengimplementasikan *Constrained Application Protocol* (CoAP), sebuah *lighweight protocol* sebagai protokol *web transfer* untuk sistem ini. Mikrokontroler Wemos DI dan sensor YL-69 digunakan untuk mengumpulkan data kelembaban tanah dan mengirimkan data tersebut ke *server* menggunakan modul *bluetooth* HC-05. Kemudian *server* dan *client* akan melakukan pertukaran data secara wireless menggunakan protokol CoAP. Untuk melihat performa protokol CoAP kita menguji pertukaran data yang terjadi antara *server* dan *client* pada beberapa skenario. Berdasarkan hasil pengujian dapat dilihat terdapat kenaikan *delay* yang signifikan pada ukuran *payload* lebih dari 64 byte karena adanya mekanisme *blockwise* pada protokol CoAP. Selain itu, jarak juga meningkatkan *delay* dalam proses transmisi data.

Kata kunci: kelembaban tanah, *Node Sensor*, *Wireless Sensor Network* (WSN), *Constrained Application Protocol* (CoAP)

Abstract

Recently, technology has been developed to facilitate human life in every aspects including agriculture. Implementation of technology in agriculture is still left behind for example in terms of observation soil moisture. Based on concept of *Wireless Sensor Network* (WSN) we can make a suitable network architecture to solve the problem. *Wireless Sensor Network* (WSN) is one of the emerging technologies which combine sensors and tiny embedded devices over a wireless communication medium. The aim of this study is how to make a reliable communication for the system. To accomplish our objectives, we implemented a *lighweight protocol* as a *web transfer protocol* named *Constrained Application Protocol* (CoAP) for the system. To collect the soil moisture data we integrated *Microcontroller Wemos D1* and sensor *YL-69* to perform as a sensor node, it sends sensing data to the server with *bluetooth module HC-05*. Then servers and clients will exchange data wirelessly using the CoAP protocol. To see the performance of the CoAP protocol, we tested the data exchanging that occurs between server and client in several scenarios. The test result revealed there is a significant increase of delay in payload size more than 64 bytes because of process *blockwise* mechanism in CoAP protocol. In addition, distance can also increase the delay in data transmission process.

Keywords: soil moisture, *Sensor node*, *Wireless Sensor Network* (WSN), *Constrained Application Protocol* (CoAP)

1. PENDAHULUAN

Negara kita Indonesia merupakan sebuah

negara agraris, dimana sebagian besar penduduknya bermata pencaharian sebagai petani. Berdasarkan data yang diperoleh dari Badan Pusat Statistik, jumlah penduduk yang

bekerja di bidang pertanian mencapai 34 % dari total angkatan kerja di Indonesia atau sekitar 40,83 juta jiwa (Ritonga, 2014). Namun jumlah sumber daya manusia tersebut tidak menjamin Indonesia memiliki kualitas sektor pertanian yang baik. Menurut data *Global Food Security Index* (GFSI) ketahanan pangan negara Indonesia berada pada peringkat 71 dari 113 negara (GFSI, 2016). Banyak faktor yang dapat mempengaruhi kondisi tersebut seperti kurangnya penggunaan teknologi dalam bidang pertanian. Dibandingkan dengan negara lain yang memiliki nilai ketahanan pangan lebih baik seperti Jepang, yang telah menggunakan teknologi dalam bidang pertaniannya dimana hampir semua proses pertanian telah menggunakan mesin. Hal ini membuktikan bahwa penerapan teknologi mampu meningkatkan produktivitas dan kualitas sektor pertanian suatu negara.

Salah satu faktor lingkungan yang perlu diperhatikan dalam meningkatkan sektor pertanian itu sendiri ialah tanah. Tanah adalah suatu media tanam dari berbagai jenis tumbuhan. Selain faktor kesuburan tanah itu sendiri, kelembaban tanah sering juga menjadi faktor penentu dari keberhasilan tumbuhnya tanaman disamping faktor lain seperti kandungan mineral tanah (Ihsan, dkk., 2012). Sehingga kondisi kelembaban tanah harus dijaga dan diperhatikan sesuai dengan kebutuhan setiap tanaman. Keterbatasan petani dalam mengawasi keadaan kelembaban tanah saat ini masih menjadi suatu kendala dikarenakan banyak lokasi area perkebunan atau pertanian yang terletak jauh dari rumah. Perlu adanya suatu sistem pengamatan kelembaban tanah dimana petani dapat melakukan pemantauan secara jarak jauh sehingga dapat menghemat waktu dan tenaga untuk mengetahui perubahan kelembaban suatu tanah.

Wireless Sensor Network (WSN) merupakan suatu arsitektur jaringan yang cocok untuk dikembangkan pada sistem pengamatan ini. WSN terdiri dari sejumlah perangkat kecil, yang dilengkapi dengan unit sensor, mikroprosesor, dan komunikasi antar node secara wireless. Tentu saja, hal semacam itu membutuhkan protokol komunikasi yang memadai dan mampu mengatur jaringan secara otomatis (Krcro, 2009). *Constrained Application Protocol* (CoAP) merupakan suatu protokol di dalam WSN yang dikembangkan untuk menggantikan peran protokol HTTP dalam lingkup *Internet of Things* (IoT). CoAP memiliki

ukuran paket data yang lebih kecil dibandingkan dengan HTTP sehingga dapat menghemat penggunaan sumber daya dalam jaringan dan meningkatkan proses transmisi data yang berdampak pada perpindahan arus informasi menjadi lebih cepat. Dengan kata lain protokol ini cocok diimplementasikan untuk membangun suatu sistem pengamatan.

Pada penelitian sebelumnya yang berjudul “Remote Patient Monitoring System Based Coap in Wireless Sensor Networks” merancang suatu sistem monitoring terhadap pasien menggunakan protokol CoAP. Dimana pada penelitian menggunakan aplikasi Cooja simulator yang beroperasi pada Contiki OS untuk membuat simulasi pertukaran data dari sensor menuju database server menggunakan protokol CoAP yang kemudian data tersebut diakses oleh *client* dalam kasus ini ialah dokter untuk melakukan penanganan selanjutnya terhadap pasien berdasarkan hasil diagnosa yang dikirimkan oleh sensor (Pandesswaran, dkk., 2016).

Oleh karena itu dalam penelitian ini, akan merancang suatu sistem pengamatan kelembaban tanah dengan mengimplementasikan protokol CoAP untuk komunikasi antar *server* dan *client*. Di dalam sistem terdapat suatu node sensor yang terhubung dengan mikrokontroler untuk mengoleksi data kelembaban. Kemudian data tersebut akan dikirim melalui *bluetooth* ke *server*, selanjutnya *client* dan *server* akan melakukan pertukaran data yang berisi informasi kelembaban tanah secara nirkabel. Beberapa skenario pengujian akan dilakukan di dalam penelitian ini untuk melihat hasil implementasi protokol CoAP dan kinerjanya.

2. TINJAUAN PUSTAKA

2.1 Kelembaban Tanah

Kelembaban tanah adalah air yang tersimpan pada sebagian atau seluruh pori-pori tanah (Hardjowigeno, dkk., 2005). Dari seluruh air hujan di daerah tropis, sekitar 75% persen dari air hujan tersebut masuk ke dalam tanah dalam bentuk kelembaban tanah, pada tanah tidak jenuh dan sebagai air tanah pada tanah jenuh atau tanah berbatu. Sesuai dengan tujuan studi yang mengacu pada tanah bagian permukaan yang merupakan bagian dari profil tanah, maka kelembaban permukaan tanah (*surface soil*) tidak lepas pengaruhnya dari

kelembaban tanah pada lapisan di bawahnya (*sub soil*). Komponen tanah yang mempengaruhi kelembaban tanah adalah ketersediaan air di dalam tanah itu sendiri. Beberapa faktor yang mempengaruhi kelembaban tanah adalah tekstur tanah, struktur tanah, kandungan bahan organik, dan kedalaman solum tanah.

2.2 Constrained Application Protocol (CoAP)

Constrained Application Protocol adalah suatu *web transfer* protokol khusus untuk penggunaan dengan node terbatas dan jaringan yang dibatasi (Shelby, dkk., 2014). CoAP dikembangkan oleh *International Engineering Task Force* (IETF) merupakan suatu protokol layer aplikasi dan termasuk ke dalam standar RFC 7252. CoAP yang juga merupakan suatu *lightweight protocol* dimaksudkan untuk digunakan sebagai pengganti HTTP untuk menjadi protokol pada layer aplikasi di dalam IoT (*Internet of Things*). CoAP menyediakan model interaksi *request/response* antara aplikasi dan endpoint, dikarenakan CoAP memiliki beberapa fitur yang menyerupai HTTP. Tidak seperti HTTP yang beroperasi pada TCP, CoAP beroperasi pada UDP untuk menghindari congestion control yang kompleks. CoAP memiliki *Representational State Transfer* (REST) arsitektur, sehingga menyediakan URI dan metode seperti GET, POST, PUT, dan DELETE. Untuk mengimbangi dari kekurangan UDP, CoAP memiliki sebuah mekanisme retransmisi. Untuk mengatasi kelemahan dalam keterbatasan sumber daya, CoAP perlu mengoptimalkan panjang datagram untuk menyediakan komunikasi yang handal. Terdapat empat tipe pesan yang digunakan pada CoAP untuk melakukan pertukaran data antara *client* dan *server*, berikut tipe – tipe pesan tersebut :

1. *Confirmable* (CON), merupakan pesan yang berisi *request* dan memerlukan *Acknowledgment*.
2. *Non-Confirmable* (NON), merupakan pesan yang digunakan berulang secara teratur tanpa memerlukan *Acknowledgment*.
3. *Acknowledgment* (ACK), merupakan pesan yang berisi *response*.
4. *Reset* (RST), merupakan pesan yang digunakan ketika pesan CON tidak diterima dengan benar atau terdapat konteks yang hilang.

2.3 Mikrokontroler Wemos D1

Wemos D1 merupakan sebuah mikrokontroler pengembangan berbasis modul mikrokontroler ESP8266. Mikrokontroler ini diproduksi oleh produsen Cina yang berbasis di Shanghai. Ai-Thinker membuat ESP-01 dengan menggunakan lisensi oleh Espressif. Perangkat ini memiliki dimensi ukuran 68.6 mm x 53.4 mm, beroperasi pada tegangan 3.3V. Modul ini memiliki digital I/O pin sebanyak 11 buah dan 1 analog input pin. Modul ini memungkinkan mikrokontroler untuk terhubung dengan jaringan WiFi dan membuat koneksi TCP/IP hanya dengan menggunakan *command*. Dengan *clock speed* mencapai 80/160 MHz chip ini dibekali dengan 4MB *flash memory*, mendukung format IEEE 802.11 b/g/n sehingga tidak menyebabkan gangguan bagi yang lain (Putri, 2017).

2.4 Sensor YL-69

Sensor kelembaban tanah YL-69 merupakan suatu sensor sederhana yang dapat membaca jumlah atau kadar intensitas air yang ada di dalam tanah (Pamungkas, dkk., 2011). Sensor ini terdiri dari dua probe untuk melewati arus melalui tanah, kemudian membaca resistansinya untuk mendapatkan nilai tingkat kelembaban. Semakin banyak air membuat tanah lebih mudah menghantarkan listrik (resistansi kecil), sedangkan tanah yang kering sangat sulit menghantarkan listrik (resistansi besar). Sensor YL-69 memiliki spesifikasi tegangan input sebesar 3.3V atau 5V, dengan tegangan output sebesar 0-4.2V dan arus sebesar 35mA. Modul sensor ini memiliki 4-pin, yaitu GND (untuk *ground*), VCC (3.3 - 5V), AO (keluaran analog yang akan dibaca oleh Arduino), dan DO (dapat diatur sensitivitasnya menggunakan knob pengatur, dan menghasilkan logika digital *HIGH/LOW* pada level kelembaban tertentu). Sensor ini memiliki *value range* ADC sebesar 1024 bit dimulai dari 0 – 1023 bit.

2.5 Modul Bluetooth HC-05

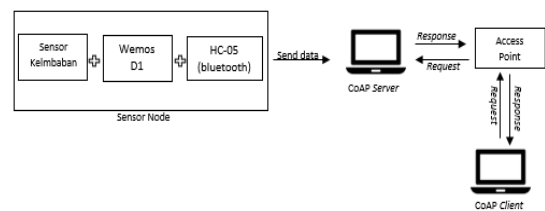
Bluetooth merupakan sebuah standar untuk komunikasi nirkabel. Standar teknologi *Bluetooth* telah banyak diterapkan pada berbagai bidang komunikasi seperti perangkat komunikasi personal, komunikasi jaringan nirkabel, dan berbagai sistem transmisi (Jayantilal, 2014). Modul *Bluetooth* HC-05 beroperasi pada tegangan 3.3V, dimana untuk modul jenis lain biasanya beroperasi pada tegangan 5V sampai 12V. *Bluetooth* beroperasi

pada spektrum frekuensi antara 2.4 sampai 2.485 GHz. Jarak yang dapat dicakup oleh *bluetooth* berkisar dari 1-10 meter. *Bluetooth* memiliki kecepatan pengiriman hingga 2.1Mbps, dan memiliki *Radio Frequency* (RF) transmit power sebesar +4dBm. *Bluetooth* HC-05 mempunyai *default baudrate* yakni 38400, dan mendukung ukuran baudrate lainnya seperti 9600, 19200, 57600, 115200, 230400 dan 460800. Modul *Bluetooth* ini dapat diatur sebagai mode *master* atau *slave*. Mode *master* akan mengatur fungsi *Bluetooth* untuk dapat menerima dan mengirim data. Sedangkan pada mode *slave*, *Bluetooth* hanya dapat berfungsi untuk menerima data saja.

3. METODOLOGI

3.1 Perancangan Sistem

Pada perancangan perangkat keras akan terlebih dahulu menghubungkan mikrokontroler dengan sensor kelembaban tanah dan modul *bluetooth*. Kemudian untuk menjalankan fungsinya mikrokontroler akan diprogram menggunakan Arduino IDE. Selanjutnya akan dibangun suatu *server* CoAP dengan menggunakan *framework* Twisted dengan *library* CoAP yang berbasis bahasa pemrograman Pyhton. Pada sisi *server* akan menerima data dari sensor melalui komunikasi *bluetooth*, data tersebut akan dijadikan sebagai *payload*. Untuk melakukan komunikasi pada protokol CoAP, *server* dan *client* akan terhubung pada satu jaringan WiFi yang sama melalui sebuah access point, sebelumnya pada *browser client* telah ditambahkan suatu plugin yakni Copper agar dapat melakukan *request* pada *server* CoAP. *Request* tersebut akan direspon oleh *server* dengan mengirimkan *payload* yang berisi data hasil pengamatan. Diagram blok pada Gambar 1. akan menggambarkan keseluruhan sistem.

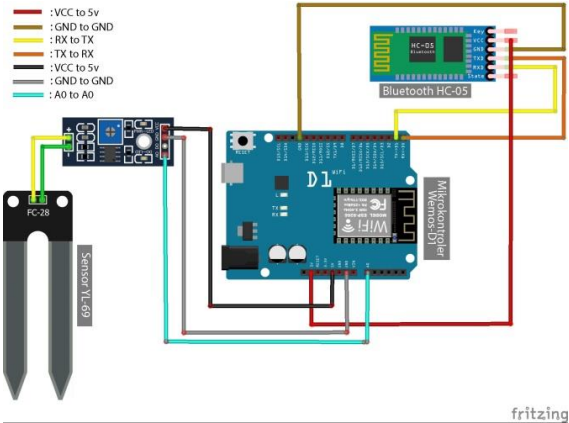


Gambar 1. Diagram Blok Sistem

3.2 Implementasi Sensor Node

Sensor node disini merupakan bagian dari sistem yang berfungsi untuk mengoleksi data

atau pengamatan terhadap kondisi kelembaban tanah. Sensor node pada sistem ini terdiri dari Mikrokontroler Wemos D1, Sensor kelembaban tanah YL-69, dan modul *bluetooth* HC-05. Rancangan sensor node pada sistem dapat dilihat pada Gambar 2.



Gambar 2. Rangkaian Sensor Node

Pada gambar rangkaian diatas pin-pin antara sensor, mikrokontroler, dan modul *bluetooth* saling terhubung menggunakan kabel jumper. Pada sensor YL-69 terdapat 3 pin yang digunakan yaitu pin A0, VCC, dan GND. Sedangkan pada modul *bluetooth* HC-05 terdapat 4 pin yang digunakan yaitu pin RX, TX, VCC, dan GND. Untuk memperjelas hubungan pin-pin antar modul dapat dilihat pada Tabel 1 dan Tabel 2.

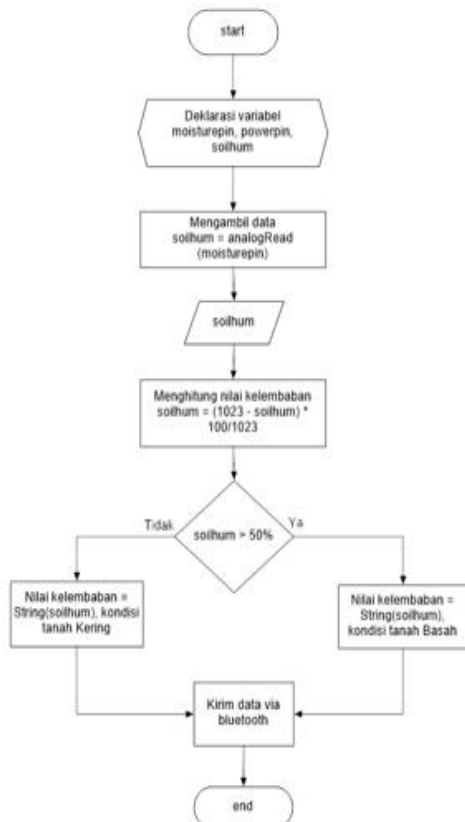
Tabel 1. Keterangan Pin Sensor

Sensor YL-69	Mikrokontroler Wemos D1
VCC	5V
GND	GND
A0	A0

Tabel 2. Keterangan Pin Modul Bluetooth

Modul Bluetooth HC-05	Mikrokontroler Wemos D1
VCC	5V
GND	GND
RX	TX
TX	RX

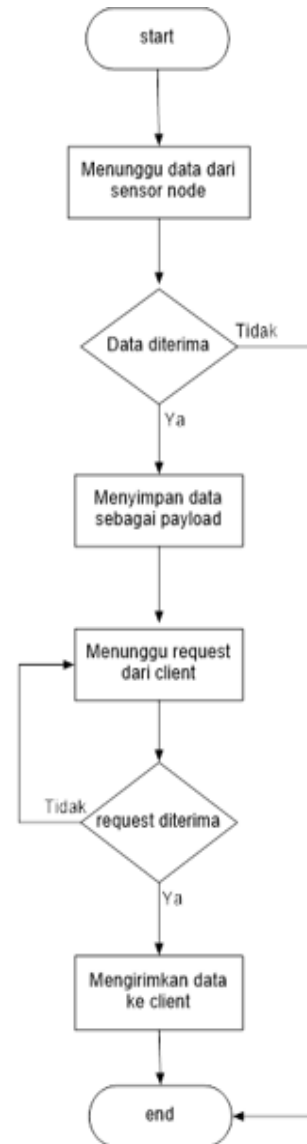
Selain perancangan hardware pada sensor node, pada bagian ini juga dijelaskan bagaimana perancangan aliran data yang ada pada hardware tersebut. Aliran data pada sensor node dapat dilihat pada Gambar 3.



Gambar 3. Diagram Aliran Data Sensor Node

3.3 Implementasi CoAP Server

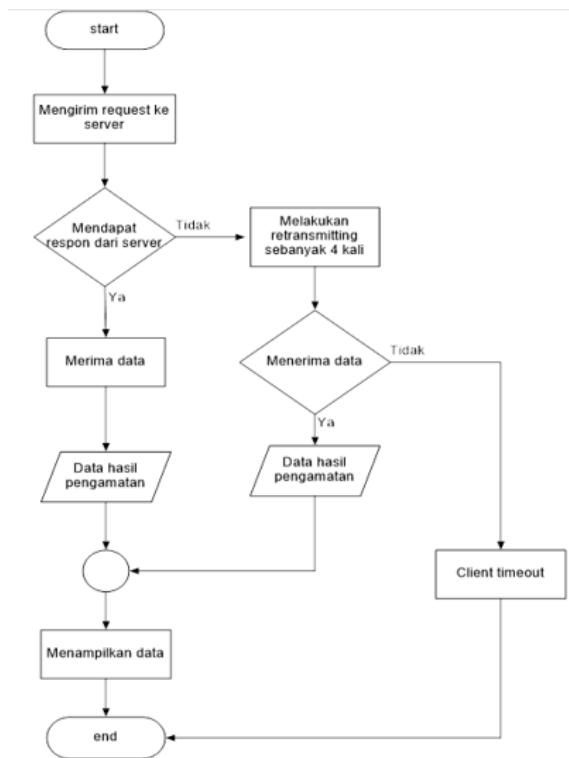
Server pada sistem ini merupakan sebuah Laptop yang berperan sebagai CoAP server, dimana server berfungsi untuk menghubungkan client dengan sensor node. Pada implementasi perangkat lunak untuk server, pemrograman akan dilakukan menggunakan bahasa pemrograman Python. Server ini dibangun menggunakan framework Twisted dengan tambahan sebuah library CoAP yaitu TxThings. Server akan menerima data dari sensor node dan menyimpannya sebagai *payload*, kemudian client akan melakukan *request* ke server selanjutnya server mengirimkan data tersebut ke client. Perancangan aliran data pada server ditunjukkan pada Gambar 4.



Gambar 4. Diagram Aliran Data Server

3.4 Implementasi CoAP Client

Client pada sistem ini merupakan sebuah Laptop yang berperan sebagai CoAP client, dimana client akan mengirimkan *request* kepada server untuk mendapatkan data hasil pengamatan yang dilakukan oleh sensor node. Implementasi perangkat lunak pada client dalam sistem ini akan menggunakan browser Mozilla Firefox dengan tambahan plugin Copper. Plugin tersebut berfungsi agar client dapat menjalankan skema 'Coap URI' untuk berinteraksi dengan server. Setelah data dikirimkan oleh server, data tersebut yang berisi informasi kelembaban tanah akan ditampilkan melalui sebuah interface browser. Perancangan aliran data pada client ditunjukkan pada Gambar 5.



Gambar 5. Diagram Aliran Data Client

4. PENGUJIAN & ANALISIS

Terdapat dua jenis skenario yang akan dilakukan dalam pengujian ini, yaitu skenario pengujian *payload* dan skenario pengujian jarak. Pengujian *payload* dilakukan untuk mengetahui bagaimana performansi kinerja protokol CoAP dalam melakukan pertukaran data antara *client* dan *server* dengan ukuran *payload* yang berbeda. Terdapat perubahan ukuran *payload* sebanyak 3 kali yaitu 16 byte, 64 byte dan 128 byte. Hasil pengujian *payload* tersebut dapat dilihat pada Tabel 3. Pengujian jarak dilakukan untuk mengetahui bagaimana faktor perubahan jarak dapat mempengaruhi proses transmisi data dalam komunikasi nirkabel. Proses pengujian ini dilakukan dengan cara merubah ukuran jarak antara *client* dan *server* dengan access point. Terdapat perubahan ukuran jarak sebanyak 3 kali yaitu 1 m, 5 m dan 10 m. Hasil pengujian jarak tersebut dapat dilihat pada Tabel 4. Parameter *Quality of Service* (QoS) yang akan diukur pada pengujian ini adalah *delay*. *Delay* pada pengujian ini merupakan waktu yang dibutuhkan untuk 1 kali proses *request* dan *response* pada sistem.

Tabel 3. Hasil Skenario Pengujian *Payload*

No.	<i>Payload</i> (byte)	<i>Delay</i> (s)
1	16	0.002936653
2	64	0.003275341
3	128	0.049077139

Pada hasil pengujian diatas terlihat bahwa terdapat kenaikan nilai *delay* ketika terdapat perubahan besaran *payload*. Untuk membuktikan hal tersebut maka dilakukan perhitungan sebagai berikut.

$$X1 = 0.002936653 \text{ s (Delay untuk payload 16 byte)}$$

$$X2 = 0.003275341 \text{ s (Delay untuk payload 64 byte)}$$

$$X3 = 0.049077139 \text{ s (Delay untuk payload 128 byte)}$$

$$\begin{aligned} \text{Delay}X2 - X1 &= 0.003275341 - 0.002936653 \\ &= 0.000338688 \end{aligned}$$

$$\begin{aligned} \text{Delay}X3 - X2 &= 0.049077139 - 0.003275341 \\ &= 0.045801798 \end{aligned}$$

Dari hasil perhitungan diatas dapat diketahui terjadi kenaikan nilai *delay* akibat perubahan *payload* dari 16 ke 64 byte sebesar 0.000338688s. Sedangkan terjadi kenaikan nilai *delay* yang sangat signifikan pada ukuran *payload* 64 dan 128 byte sebesar 0.045801798s. Hal ini terjadi dikarenakan terdapat mekanisme *blockwise* dalam protokol CoAP untuk mentransmisikan data yang berukuran lebih dari 64 byte.

Tabel 4. Hasil Skenario Pengujian Jarak

No	Jarak	Delay pada payload 16 byte (sec)	Delay pada payload 64 byte (sec)	Delay pada payload 128 byte (sec)
1	1 meter	0.002936653	0.003275341	0.049077139
2	5 meter	0.003096938	0.003306577	0.059808438
3	10 meter	0.003268172	0.003352752	0.13649748

Pada hasil pengujian ini dapat dilihat bahwa terjadi juga kenaikan nilai *delay* ketika terdapat perubahan ukuran jarak. Untuk membuktikan hal tersebut maka dilakukan perhitungan sebagai berikut.

- 1) Pada perhitungan ini akan menghitung nilai *delay* berdasarkan perubahan *payload* dari 16 ke 64 byte pada jarak 5 meter.

$$X1 = 0.003096938 \text{ s (Delay untuk payload 16 byte)}$$

$$X2 = 0.003306577 \text{ s (Delay untuk payload 64 byte)}$$

$$\text{Delay}X2 - X1 = 0.003306577 - 0.003096938$$

$$= 0.000209639$$

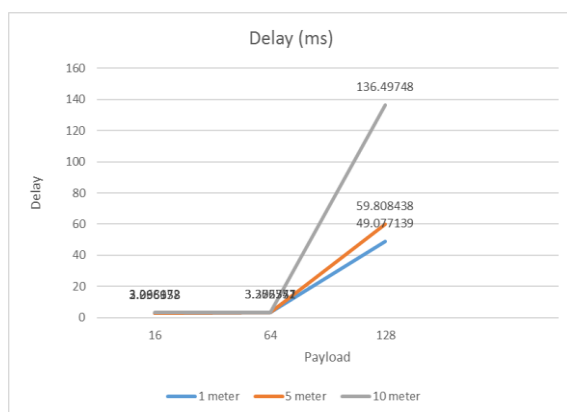
- 2) Pada perhitungan ini akan menghitung nilai *delay* berdasarkan perubahan jarak dari 1 ke 5 meter pada *payload* 64 byte.

$Y1 = 0.003275341s$ (*Delay* untuk jarak 1 meter)

$Y2 = 0.003306577s$ (*Delay* untuk jarak 5 meter)

$$\begin{aligned} \text{Delay } Y2 - y1 &= 0.003306577 - 0.003275341 \\ &= 0.000031236 \end{aligned}$$

Berdasarkan hasil perhitungan di atas dapat diketahui bahwa pengaruh perubahan *payload* menyebabkan kenaikan nilai *delay* yang lebih besar dibandingkan dengan pengaruh perubahan jarak. Dimana dari hasil perhitungan selisih kenaikan nilai *delay* akibat penambahan jumlah *payload* pada perhitungan pertama lebih signifikan dibandingkan dengan penambahan jarak pada perhitungan kedua. Grafik hasil pengujian ini dapat dilihat pada Gambar 6.



Gambar 6. Grafik Kenaikan Delay

5. KESIMPULAN

Pada penelitian implementasi protokol CoAP pada sistem pengamatan kelembaban tanah ini dapat diketahui bagaimana membangun sebuah komunikasi protokol CoAP pada sisi *server* dan *client*, proses komunikasi dimulai dengan *request* dengan tipe pesan Confirmable Message (CON) yang dikirimkan oleh *client* kepada *server* selanjutnya *server* merespon dengan mengirimkan *payload* yang berisi informasi kelembaban tanah dengan tipe pesan Acknowledgement Message (ACK). Informasi yang telah didapat kemudian ditampilkan melalui *browser* pada laptop *client*. Pada pengujian penelitian ini, dapat diketahui bagaimana perubahan ukuran *payload* dapat memberikan pengaruh yang lebih besar

dibandingkan dengan perubahan jarak. Hal ini dapat dilihat pada performansi protokol CoAP melalui sebuah parameter *delay* pada hasil pengujian. Selain itu dapat juga dilihat bagaimana mekanisme *blockwise* yang ada protokol pada CoAP mempengaruhi proses transmisi untuk *payload* dengan ukuran lebih dari 64 byte, sehingga waktu yang dibutuhkan menjadi lebih lama dengan adanya mekanisme *blockwise*.

DAFTAR PUSTAKA

- GFSI. 2016. Performance of Countries Based on Their 2016 Food Security Score. [Online]. Tersedia di : <http://foodsecurityindex.eiu.com/Index> [Diakses 16 April 2017]
- HARDJOWIGENO, S. & RAYES, L., 2005. Karakteristik, Kondisi, dan Permasalahan Tanah Sawah di Indonesia. Dalam: Tanah Sawah. Malang: Bayumedia.
- IHSAN, H., KOMARUDIN, C., IRAWAN. R. & LASOL., 2012. Pengukuran Kelembaban Tanah Dengan Kadar Air Yang Bervariasi Menggunakan Sensor Kelembaban Tanah SEN5007 dan VH400. Volume V, pp. 1-6.
- JAYANTILAL, H. S., 2014. Interfacing of AT Command based HC-05 Serial Bluetooth Module with Minicom in Linux. *International Journal for Scientific Research & Development*, II(03), pp. 329-332.
- KRCO, K., 2009. *Bluetooth Based Wireless Sensor Networks –Implementation Issues and Solutions*.
- PANDESSWARAN, SURENDER & KARTHIK, 2016. Remote Patient Monitoring System Based Coap in Wireless Sensor. *Sensor Networks and Data Communications*, V(3).
- PAMUNGKAS, H., PUSPITA, E. & TAUFIQURAHMAN, 2011. Alat Monitoring Kelembaban Tanah Dalam Pot Berbasis Mikrokontroler.
- PUTRI, M., 2017. Mengenal Wemos D1 Mini Dalam Dunia IoT. [Online]. Tersedia di: <http://ilmuti.org/2017/02/23/mengenal-wemos-d1-mini-dalam-dunia-iot/> [Diakses 25 Juli 2017]
- RITONGA, R. 2014. Badan Pusat Statistik – ILO [Online]. Available at :

www.ilo.org/wcmsp5/groups/public/@asia/@ro-bangkok/@ilo/wcms_346599.pdf.
[Diakses 18 April 2017]

SHELBY, Z., HARTKE, K. & BORMANN, C.,
2014. The Constrained Application
Protocol (CoAP) – RFC 7252.