

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

*Internet of Thing (IoT)* dewasa ini semakin populer dan menjadi perhatian berbagai pihak, pada tahun 2010 Ericsson memperkirakan akan ada 50 juta perangkat yang terhubung dengan internet[1]. Penurunan biaya komputasi sebesar 60 kali lipat, penurunan biaya *bandwidth* 40 kali lipat dan penurunan biaya sensor dan alat cerdas lainnya yang hampir 2 kali lipat dalam dekade terakhir ini [2] juga turut mempopulerkan *Internet of Thing*.

*Internet of Thing* sendiri secara sederhana bukanlah hanya *internet connected thing* namun juga secara berkala mengirim data/nilai dari sensor maupun perangkat pintar lainnya ke *IoT server*. Dalam mengirimkan data tersebut dibutuhkan protokol yang ringan, cepat dan pastinya harus cocok dan mampu diterapkan di *constrained device* yang merupakan perangkat mikrokontroler yang tidak memiliki komputasi sekompleks komputer *desktop* sehingga dapat optimal untuk transportasi data.

Nilai dari sensor atau perangkat pintar nantinya akan dikirimkan terus menerus secara berkala namun hal tersebut dapat memberatkan terhadap *server* (*server* yang digunakan untuk menampung data dari *sensor*) karena *server* harus melayani *request* dalam jumlah yang banyak dan membutuhkan *resource* memori yang besar, hal ini jika terjadi secara terus menerus akan membuat *server* menjadi *down*. Di sisi lainnya klien yang akan mengakses data *sensor* tersebut melalui *web browser* harus melakukan *polling* (pengambilan data) secara terus menerus juga, sehingga akan membuat proses pada *browser* memakan banyak *resource* memori yang besar juga untuk proses *polling* untuk itulah digunakan protokol MQTT (*Message Queuing Telemetry Transport*) untuk menggantikan *http request* yang ada dikarenakan lebih *lightweight* atau ringan dalam hal *resource* yang dibutuhkan dan dapat menangani *post request* (dalam MQTT *post request* disebut dengan istilah *publish*) lalu dari sisi klien nya juga lebih hemat *resource* karena tidak perlu

melakukan *polling* secara terus menerus digantikan menggunakan metode *subscribe*, di mana ketika ada data baru *server* akan mengirimkan data tersebut terhadap klien yang melakukan *subscribe* sehingga ketika tidak terdapat data baru maka klien tidak perlu melakukan *polling* secara berkala.[3]

Di sisi lainnya data yang telah dikirimkan melalui protokol MQTT perlu untuk disimpan ke dalam *database* untuk nantinya dapat digunakan untuk keperluan analisis dan pengambilan keputusan berdasarkan data historis, untuk dapat mencapai hal tersebut dibutuhkan sistem *database* khusus yang mampu menangani proses *write* secara *continue* berkala terus menerus, karena tingkat *write* yang tinggi tersebut basis data relasional biasa seperti mysql kurang cocok digunakan dan *Time Series Database* (TSDB) menjadi solusi yang lebih rasional untuk data sensor yang merupakan data berdasarkan waktu (*time series/historis*),[4] dari alasan tersebut dipilihlah *time series database* InfluxDB yang secara teknis mengungguli Cassandra DB sebanyak 4.5 kali lipat dalam kecepatan *write throughput*, dan menggunakan *resource* 10.8 kali lebih hemat penggunaan *storage* berkat *compression* nya[5]. InfluxDB juga mengungguli beberapa *time series database* lainnya dalam berbagai aspek.[6]

Dari beberapa alasan di atas, maka penelitian ini membahas bagaimana membuat sistem terintegrasi untuk *internet of thing* yang ringan, efisien dan maksimal untuk sistem *internet of thing* dengan menggunakan protokol MQTT dan TSDB, sedangkan untuk mengintegrasikan MQTT dengan InfluxDB menggunakan *flow based programming* dari Node-Red kemudian untuk mempermudah menganalisis data dari sensor tadi dapat divisualisasikan dalam bentuk grafik, tabel maupun *chart* dengan menggunakan Grafana Server maupun Node-Red *Dashboard*.

Penelitian ini akan mencoba merancang dan membangun sistem *IoT* terintegrasi pada Raspberry Pi 3 model B dan VPS (*Virtual Private Server*) dari Amazon AWS EC2 T2.*micro* yang mana masing masing infrastruktur tersebut mempunyai kelebihan dan kekurangan masing masing, seperti Raspberry Pi 3 model B yang lebih ringkas dan murah yang memiliki kemampuan komputasi yang cukup untuk penerapan sistem *IoT* porsi kecil sampai sedang dan bersifat lokal

(diimplementasikan di *edge network*) sedangkan AWS EC2 memiliki komputasi yang lebih baik dibandingkan dengan Raspberry Pi model B sehingga cocok untuk sistem *IoT* yang lebih besar porsi nya dari Raspberry Pi, serta jaminan *uptime*, *dedicated IP* dan beberapa fitur lainnya yang membuat AWS EC2 untuk kebutuhan yang bersifat publik (diimplementasikan di *cloud*) sehingga dari mana saja dapat mengakses dan menggunakan sistem *IoT* dengan mudah asalkan terdapat jaringan internet.

## 1.2 Rumusan Masalah

Dari latar belakang dapat disimpulkan beberapa rumusan masalah:

- a. Bagaimana merancang dan membangun sistem *IoT* yang mengintegrasikan protokol MQTT dengan TSDB untuk menjadi solusi sistem *IoT* terintegrasi yang efisien dan ringan
- b. Bagaimana menyajikan dan memvisualisasikan data *IoT* untuk memudahkan memonitor serta analisis data *IoT*?

## 1.3 Batasan Masalah

Sistem *IoT* Terintegrasi menggunakan *Flow Based Programming* dengan protokol MQTT dan *Time Series* DB dibatasi dengan beberapa batasan sebagai berikut:

- a. Penelitian ini tidak meneliti performa dan keakuratan dari sensor
- b. Penelitian ini dijalankan dengan menggunakan Raspberry Pi 3 dan AWS EC2 sebagai *mqtt broker* dan *IoT server*, ESP8266 sebagai *gateway* dari sensor suhu kelembaban DHT11 dan BME280 sebagai *node* sensornya.
- c. Penelitian ini menggunakan 2 unit ESP8266 untuk mengirimkan data.
- d. *Time Series Database* yang digunakan dalam penelitian ini adalah InfluxDB dari InfluxData.
- e. *MQTT Broker* yang digunakan di penelitian ini adalah EMQ (Erlang MQTT Broker)

## 1.4 Tujuan

Tujuan dari penelitian tugas akhir ini adalah untuk menciptakan sistem *IoT* terintegrasi dengan protokol MQTT dan TSDB secara mandiri sehingga didapatkan sistem *IoT* terintegrasi yang ringan efektif *customizeable*, dapat dianalisis dan mampu digunakan untuk keperluan monitor sensor suhu dan kelembaban sebagai contohnya.

## 1.5 Manfaat

Manfaat dari penelitian ini adalah sebagai berikut:

1. Memperoleh sistem *IoT* terintegrasi dengan protokol MQTT dan TSDB yang mudah serta efektif untuk memproses dan menyimpan data dari sensor maupun perangkat pintar lainnya
2. Memperoleh kontrol penuh terhadap sistem *IoT* dan *IoT data*
3. Memperoleh *dashboard* yang *realtime* dan *graph & chart* yang dapat diatur, difilter disederhanakan dan dapat dikustomisasi secara bebas untuk memudahkan memonitor dan analisis.

## 1.6 Sistematika Penulisan

Sistematika penulisan dalam penulisan laporan penelitian tugas akhir ini menggunakan sistematika sebagai berikut:

### **BAB I: PENDAHULUAN**

Pendahuluan mencakup latar belakang penelitian tugas akhir, rumusan masalah yang ada, batasan batasan yang membahas permasalahan yang akan dibahas, dan tujuan serta manfaat yang bisa didapat dari penelitian tugas akhir ini serta metodologi penelitian dan sistematika penulisan laporan.

### **BAB II: LANDASAN TEORI**

Landasan teori merupakan teori dan pustaka yang menjadi dasar penelitian tugas akhir ini termasuk di dalamnya penelitian terdahulu dengan topik yang sama dan dasar teori tentang *internet of thing* serta *software* maupun *hardware* terkait lainnya.

**BAB III: PERANCANGAN**

Perancangan memuat deskripsi tugas akhir, model yang diusulkan serta terkait *life cycle* pembuatan sistem yang berisi analisis kebutuhan, desain sistem serta desain pengujian.

**BAB IV: IMPLEMENTASI DAN PENGUJIAN**

Implementasi dan pengujian memuat kelanjutan dari *life cycle* yang berupa implementasi dan pemrograman, proses pengujian dan hasil dari pengujian sistem.

**BAB V: PENUTUP**

Penutup mencakup kesimpulan dan hasil penelitian tugas akhir, saran serta beberapa penelitian yang dapat dilanjutkan (*further research*).