

Bab I Pendahuluan

1.1 Latar Belakang

Pada dasarnya bangunan memiliki fungsi sebagai tempat tinggal. Namun seiring berkembangnya teknologi, adanya pemikiran untuk meningkatkan kenyamanan bangunan dengan cara meningkatkan fungsionalitas komponen dari bangunan, salah satunya dengan otomatisasi berbagai komponen yang ada pada bangunan, misalnya ventilasi, lampu, AC, pintu dan lainnya. Diperkirakan dimasa depan teknologi ini menjadi tren dan sangat berperan dalam kehidupan, untuk itu perlu adanya pembaharuan dan peningkatan kualitas, khususnya dalam hal efektifitas dan efisiensi yang menjadi tantangan di waktu yang akan datang.

Internet of Things (IoT) merupakan sejarah baru dalam perkembangan internet, *Internet of Things* (IoT) memiliki konsep dimana berbagai komponen dapat saling terhubung melalui internet. Penerapan internet of things antara lain *smart city* dan *smart building*. *Smart city* adalah konsep teknologi IoT yang diterapkan pada skala besar, sedangkan *Smart Building* adalah konsep teknologi IoT yang diterapkan pada skala kecil, seperti dalam bangunan. Bentuk hasil dari IoT pada *smart building* adalah bagaimana berbagai komponen dalam bangunan dapat saling berinteraksi dan bersifat *automation*, misalnya *air conditioner*(AC) akan menyala ketika suhu udara panas, lampu menyala ketika intensitas cahaya berkurang dan lain sebagainya.

Dalam implementasinya, *Smart building* menggunakan teknologi *Machine-to-Machine* dengan menggunakan *middleware platform*, dalam hal ini adalah OpenMTC yang dibantu dengan protokol komunikasi standarnya HTTP. Namun, mengingat diperlukannya peningkatan kualitas komunikasi pada *smart building*, maka dikembangkannya OpenMTC dengan protokol yang lebih moderen, dalam penelitian ini digunakan protokol CoAP. Protokol CoAP sendiri merupakan protokol yang memiliki kelebihan berupa *low overhead*, *multicast* dan *simplicity*. Sebelumnya protokol CoAP telah diterapkan pada *platform* Contiki, untuk penghematan energi dalam menghubungkan *constrained devices*.

1.2 Perumusan Masalah

Berdasarkan latar belakang tersebut, didapatkan beberapa rumusan masalah pada penelitian ini, antara lain :

1. Bagaimana mengimplementasikan protokol CoAP pada OpenMTC?
2. Bagaimana kinerja protokol CoAP berbasis OpenMTC dengan parameter *overhead*, *delay*, dan *throughput*?
3. Bagaimana perbandingan kinerja yang dihasilkan protokol CoAP jika dibandingkan dengan protokol HTTP dengan *overhead*, *delay* dan *throughput*?

1.3 Tujuan

Adapun tujuan yang ingin dicapai dari penelitian ini adalah

1. Melakukan implementasi protokol CoAP pada *platform* OpenMTC versi 2.
2. Menguji dan menganalisis kinerja penggunaan protokol CoAP dalam OpenMTC dengan parameter *overhead*, *delay*, dan *throughput*.
3. Membandingkan kinerja protokol CoAP dengan protokol HTTP menggunakan parameter *overhead*, *delay*, dan *throughput*.

1.4 Batasan Masalah

Adapun batasan masalah dari penelitian ini sebagai berikut :

1. Penelitian mengenai kinerja protokol CoAP berbasis *Platform* OpenMTC versi 2 ditinjau dari sisi *overhead*, *delay*, dan *throughput*.
2. Protokol CoAP berperan pada bagian antara GSCL OpenMTC dan *Device Application*
3. Data sensor menggunakan data *temperature* dan *lighting* yang dibangkitkan menggunakan *generator*.
4. Membandingkan kinerja (*overhead*, *delay*, dan *throughput*) antara protokol CoAP dan protokol HTTP berbasis OpenMTC.
5. Fungsionalitas protokol CoAP yang diterapkan hanya berupa pengiriman data.

1.5 Hipotesa

Pada *smart building* berbasis *platform* OpenMTC, Protokol CoAP memiliki kinerja yang lebih baik dalam hal pengiriman data antara *Machine to Machine* dengan memberikan *response time* dan *overhead* lebih rendah dibandingkan dengan Protokol HTTP dalam penggunaan jaringan LAN maupun WLAN. Protokol CoAP telah dirancang untuk memenuhi arsitektur REST yang cocok untuk *constrained environment*, dengan sifat *multicast*, *low overhead*, dan *simplicity*. Hal tersebut dikarenakan protokol CoAP menggunakan UDP dan *binary header* dalam transmisi pakatnya sehingga meminimalisir kemungkinan terjadinya penurunan kinerja dalam pengiriman paket.[12]

1.6 Metodologi penyelesaian masalah

Penelitian dikerjakan dengan metodologi sebagai berikut :

1. Studi literatur

Pada tahap ini dilakukan pencarian dan pengumpulan berbagai referensi pendukung mengenai CoAP, OpenMTC, node.js, JSON dan berbagai hal pendukung lainnya. Pengumpulan referensi berasal dari jurnal, buku, internet dan makalah ilmiah.

2. Perancangan sistem

Melakukan perancangan sistem yang akan dibangun. Perancangan sistem dilakukan dengan menentukan berbagai material yang diperlukan, menentukan titik permasalahan yang akan dipecahkan, dan menggambarkan secara keseluruhan bagaimana sistem akan bekerja.

3. Implementasi sistem

Pada tahap ini dilakukan pengimplementasian protokol CoAP pada *platform* OpenMTC dengan melakukan konfigurasi dan pemrograman untuk menghubungkan sensor *gateway* dan GSCL. Konfigurasi dan pemrograman dilakukan pada bagian GIP OpenMTC menggunakan JSON dan Node.js. Selain itu dilakukan pembangunan *hardware*, beserta sensor virtual yang akan digunakan untuk simulasi.

4. Pengujian sistem

Pengujian akan dilakukan untuk mengukur kinerja dari protokol CoAP berbasis Open MTC serta protokol pembandingnya yaitu HTTP. Pengujian menggunakan sensor virtual dengan jumlah tertentu yang dihubungkan dengan *gateway*(GSCL) melalui protokol CoAP. Pengujian dilakukan dengan menangkap transmisi paket.

5. Analisis hasil dan penulisan laporan

Melakukan analisis terhadap hasil pengujian kinerja protokol CoAP dan mengambil kesimpulan dari pengujian tersebut. Selanjutnya melakukan penulisan laporan penelitian.