

## BASIS DATA APACHE CASSANDRA SEBAGAI MEDIA PENYIMPANAN DATA SENSOR DAN BACKEND INTERNET OF THINGS (IOT)

**Rizki Ariyani**

Ilmu Komputer dan Teknologi Informasi, [rizkiariyani@staff.gunadarma.ac.id](mailto:rizkiariyani@staff.gunadarma.ac.id), Universitas Gunadarma

### ABSTRAK

Millions of data at every meeting are questioned through social media, application websites, mobile applications, and internet usage that uses objects around us to be able to interact with each other known as the Internet of Things (IoT). IoT requires a database with very high performance and a high level of availability. NoSQL is one database that can store diverse data. Currently there are many NoSQL databases with different implementation and storage characteristics. This brings challenges in the selection of NoSQL databases used as IoT data storage media. In several studies that have been carried out by several researchers who use IoT data storage media using the Apache Cassandra NoSQL Database. The results of literature studies and literature reviews show that Cassandra is the best apache that can be used as a data storage sensor, and backend Internet of Things (IoT).

**Keywords:** *Cassandra, Internet of Things (IoT), NoSQL*

### Abstrak

Jutaan data setiap harinya dipertukarkan melalui media sosial, aplikasi *website*, aplikasi *mobile*, maupun penggunaan internet yang melibatkan benda-benda di sekitar kita untuk dapat saling berkomunikasi satu dengan lainnya yang dikenal dengan *Internet of Things* (IoT). IoT membutuhkan *database* dengan performa yang sangat mumpuni dan tingkat *availability* yang tinggi. NoSQL merupakan salah satu *database* yang dapat menyimpan data yang beragam. Saat ini terdapat banyak NoSQL *database* dengan mekanisme implementasi dan karakteristik penyimpanan yang berbeda-beda. Hal tersebut membawa tantangan dalam pemilihan NoSQL *database* yang digunakan sebagai media penyimpanan data IoT. Pada beberapa penelitian yang telah dilakukan oleh beberapa peneliti mengusulkan media penyimpanan data IoT dengan menggunakan NoSQL Basis Data Apache Cassandra. Hasil studi literatur dan telaah pustaka menunjukkan bahwa Cassandra adalah apache terbaik yang dapat digunakan sebagai media penyimpanan data sensor, dan *backend Internet of Things* (IoT).

**Kata Kunci:** *Cassandra, Internet of Things (IoT), NoSQL*

### 1. PENDAHULUAN

Di era milenial saat ini, internet merupakan kebutuhan pokok bagi para pengguna baik orangtua sampai anak-anak tingkat sekolah dasar. Internet digunakan untuk dapat saling bertukar informasi, bertukar data, dan berkomunikasi secara *realtime*. Internet dapat diakses dimana saja, kapan saja tanpa ada batas jarak dan waktu. Internet yang melibatkan benda-benda disekitar kita untuk dapat saling berkomunikasi satu dengan yang lain, dikenal dengan istilah *Internet of Things* (IoT). Pada IoT data yang dipertukarkan tidak sedikit sehingga dibutuhkan *database* dengan performa yang sangat baik serta ketersediaan tinggi tanpa mengganggu kinerja (*availability*) yang tinggi. Saat ini terdapat dua jenis *database* yaitu *Relational Database* dan *Not Only SQL* (NoSQL) *Database*.

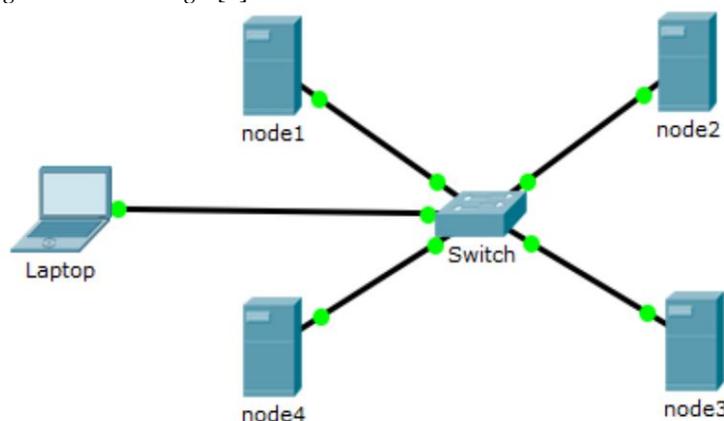
*Relational Database* memiliki beberapa masalah dan keterbatasan terkait skalabilitas dan *availability*. Sedangkan NoSQL *Database* menawarkan solusi atas masalah dan keterbatasan tersebut dalam sistem *Relational Database*. NoSQL *Database* cocok untuk digunakan sebagai media penyimpanan data sensor IoT mengingat *availability* merupakan faktor penting agar sistem dapat berjalan secara *realtime*. Ada banyak contoh NoSQL *Database* yang banyak digunakan saat ini seperti Apache Cassandra, MongoDB, Neo4J, Apache HBase, Aerospike, CouchDB dan masih banyak lagi yang lainnya [1]. Basis data Apache

Cassandra memiliki kelebihan seperti skalabilitas dan ketersediaan tinggi tanpa mengganggu kinerja. Dukungan Cassandra untuk mereplikasi di beberapa pusat data adalah yang terbaik di kelasnya, *Column-based database*, dapat mengolah data super besar yang telah digunakan oleh eBay, GitHub, Instagram, Netflix, dan lebih dari 1500 perusahaan menggunakan Cassandra sebagai media penyimpanan datanya.

Media penyimpanan data sensor dan *backend Internet of Things (IoT)* dengan menggunakan basis data apache Cassandra berdasarkan studi literatur dan telaah pustaka yang telah dilakukan. Dalam melaksanakan studi literatur digunakan sumber primer (*primary source*) agar memperoleh hasil yang otentik. Sumber literatur primer yang ditulis oleh seorang yang melihat, mengalami, atau mengerjakan sendiri. Bahan literatur berupa jurnal, *proceedings*, buku harian (*autobiography*), tesis, disertasi, dan laporan penelitian. Kemudian di dukung dengan beberapa sumber sekunder (*secondary source*) adalah tulisan tentang penelitian orang lain, tinjauan, ringkasan, kritikan, dan tulisan-tulisan serupa mengenai hal-hal yang tidak langsung disaksikan atau dialami sendiri oleh penulis. Berikut ini adalah pembahasan dan hasil dari masing-masing penelitian yang telah dilakukan oleh para peneliti.

## 2. TINJAUAN PUSTAKA

Studi literatur pertama adalah penelitian yang dilakukan oleh Ibrahim et al (2010) menguji performa antara HBase dan Cassandra menggunakan beberapa parameter uji yaitu *Throughput*, *Latency*, *Runtime database*, *CPU Usage* dan *Memory Usage* dari *database server* agar menambah pertimbangan dalam penentuan *database* mana yang lebih baik di antara HBase dan Cassandra ketika digunakan sebagai media penyimpanan data sensor IoT. Penelitian tersebut mengambil tolak ukur dari penelitian Abramova et al (2014) yang menjelaskan bahwa *Latency*, *Throughput* serta *Execution Time (Runtime)* merupakan parameter yang digunakan untuk menilai performa dari suatu *database*. Tujuan penelitian ingin mengetahui performa yang paling baik antara HBase dan Cassandra. Terdapat dua jenis skenario pengujian yang dilakukan untuk menguji performa sistem *database*, yaitu berdasarkan banyaknya baris data yang dimasukkan (50.000, 100.000, 250.000 dan 500.000 baris data) dan berdasarkan jumlah *node* yang digunakan (1, 2, 3 dan 4 *node database server*). Parameter uji yang digunakan *Throughput*, *Latency*, *Runtime*, *Memory Usage* dan *CPU Usage* [2].

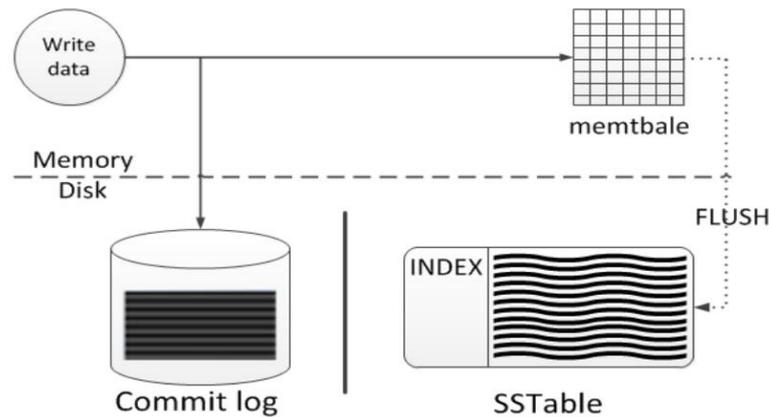


Gambar 1. Topologi Jaringan  
[Sumber: Ibrahim et al, 2010]

Hasil penelitian adalah Cassandra memiliki nilai *Throughput* yang lebih besar daripada HBase pada arsitektur terdistribusi. Sedangkan HBase memiliki nilai *Throughput* yang lebih besar daripada Cassandra pada arsitektur *Standalone mode*. Perbandingan *Latency* antara Cassandra dan HBase, Cassandra memiliki nilai *Latency* yang lebih kecil dibandingkan dengan HBase, baik pada arsitektur terdistribusi maupun *Standalone mode* sehingga semakin banyak *node* yang digunakan pada suatu arsitektur sistem *database*, maka nilai *Latency* akan semakin besar. Perbandingan *Runtime* antara HBase dan Cassandra, Cassandra memiliki performa yang lebih baik daripada HBase karena memiliki nilai *Runtime* yang lebih kecil pada seluruh skenario pengujian. Dapat disimpulkan nilai *Throughput* yang besar dan nilai *Latency* yang kecil akan membuat nilai *Runtime* menjadi kecil atau proses transaksi data membutuhkan waktu yang singkat. *Memory Usage* antara HBase dan Cassandra, HBase menggunakan *resource* yang lebih kecil daripada Cassandra karena memiliki persentase *Memory Usage* yang lebih kecil pada seluruh skenario pengujian. Hasil perbandingan *CPU Usage* antara HBase dan Cassandra, HBase menggunakan *resource* yang lebih

kecil dibanding Cassandra pada arsitektur terdistribusi. Sedangkan Cassandra menggunakan *resource* yang lebih kecil daripada HBase pada arsitektur *Standalone mode*.

Studi literatur kedua penelitian yang dilakukan oleh Guohua dan Qingyun (2017), merancang struktur penyimpanan yang sesuai untuk menyimpan histori data sensor dari kendaraan transportasi dengan menggunakan pemilihan basis data Cassandra dalam kondisi volume data yang besar. Penelitian ini mempelajari kelayakan dan cara untuk merancang struktur penyimpanan yang sesuai untuk menyimpan data sensor historis dari kendaraan transportasi dengan menggunakan pemilihan basis data Cassandra dalam kondisi volume data yang besar. Berdasarkan karakteristik struktur basis data Cassandra, mempertimbangkan masalah umum penyimpanan dan pencarian dengan volume data yang besar, untuk merancang dan mempraktikkan struktur penyimpanan data.



Gambar 2. Proses Penulisan Data Cassandra  
[Sumber: Guohua & Qingyun, 2017]

Menggunakan dua jenis basis data yaitu MySQL basis data relasional dan basis data Cassandra NoSQL sebagai sistem pendukung untuk merancang model data mereka sendiri dan menyimpan volume *database historical sensing* yang sama, kemudian melakukan uji *query* dalam fungsi yang sama [3].

Tabel 1. Eksperimen Perbandingan *Query Time-Consuming* oleh MySQL dan Cassandra

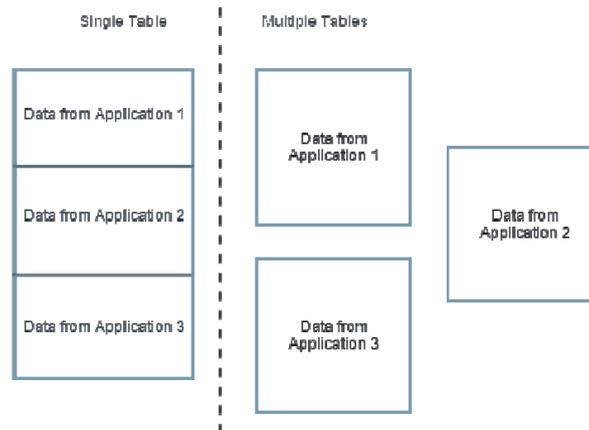
| <i>Time Quantum</i> | MySQL/SQL  | Cassandra/CQL |
|---------------------|------------|---------------|
| 30 menit            | 0.77 detik | 0.0179 detik  |
| 2 jam               | 0.06 detik | 0.0158 detik  |
| 6 jam               | 0.04 detik | 0.0193 detik  |
| 12 jam              | 0.04 detik | 0.0023 detik  |
| 18 jam              | 0.04 detik | 0.0024 detik  |
| 24 jam              | 0.05 detik | 0.0027 detik  |

[Sumber: Guohua & Qingyun, 2017]

Dalam kondisi yang sama, melalui perbandingan, analisis dan uji antara Cassandra dan MySQL. Hasilnya menunjukkan bahwa memilih basis data Cassandra dan merancang model penyimpanan yang sesuai untuk tujuan denormalisasi dan permintaan permintaan untuk menyimpan sejumlah *sence data* yang besar. Hal ini tidak hanya dapat menjamin respon permintaan sistem, tetapi juga memecahkan masalah penyimpanan sejumlah besar data sensor historis yang digunakan dalam *platform Internet of Things*.

Studi literatur ketiga penelitian yang dilakukan oleh Duarte dan Bernardino (2016), mengevaluasi permintaan waktu pemrosesan Cassandra menggunakan pertanyaan dari lingkungan waktu nyata IoT dan membandingkannya dengan berbagai jenis arsitektur data. Fokus utamanya adalah untuk menyelidiki apakah Cassandra dapat memberikan kinerja yang baik dalam sistem IoT. Tujuan dari penelitian pada studi literatur ketiga ini adalah menemukan mesin basis data yang lebih baik untuk mendukung karakteristik sistem *Internet of Things* (IoT). Hal tersebut sangat penting untuk menganalisis basis data yang ada dan mengujinya dalam konteks praktis. Sehingga memutuskan untuk menggunakan salah satu *database* paling populer yaitu Cassandra, untuk mengevaluasinya pada lingkungan IoT, mengevaluasi permintaan waktu pemrosesan Cassandra menggunakan pertanyaan dari lingkungan waktu nyata IoT dan membandingkannya dengan berbagai jenis arsitektur data. Fokus utama dari penelitian ini adalah untuk menyelidiki apakah Cassandra dapat memberikan kinerja yang baik dalam sistem IoT. Ada dua cara untuk mengatur *database*

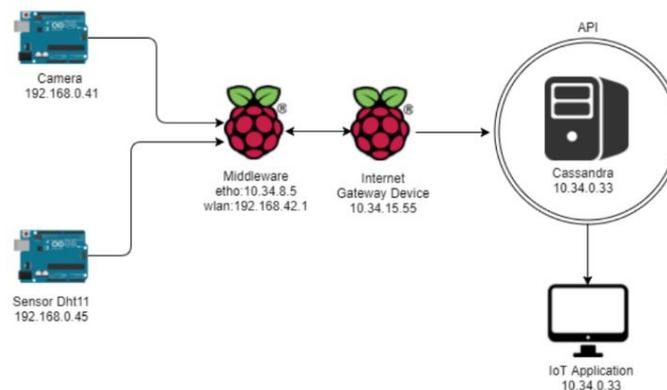
yang akan relevan dalam hal implementasi yaitu *single table* (tabel tunggal) dan *multiple table* (tabel terpisah).



Gambar 3. Rencana Lapisan Data Arsitektur [Sumber: Duarte & Bernardino, 2016]

Sebuah tabel tunggal dengan semua data, yang kemudian akan difilter dan ditangani ketika diperlukan, atau beberapa tabel untuk setiap aplikasi spesifik yang mengirim acara. Dari sudut pandang teoritis, tampaknya cara terbaik untuk mengatur data adalah melalui pembuatan tabel per aplikasi karena akan menghasilkan tabel yang lebih kecil dibandingkan dengan tabel terpusat yang menyimpan semuanya jauh lebih cepat karena mereka memiliki catatan yang jauh lebih sedikit. Penelitian ini tidak menggunakan alat *benchmark* karena ingin memperkirakan pengujian yang dilakukan dengan skenario dunia nyata (Duarte & Bernardino, 2016). Pendekatan ini menjamin bahwa hasil lebih akurat dan dapat direplikasi dalam lingkungan produksi. Hasilnya menunjukkan bahwa tabel tunggal memiliki kinerja terburuk dan implementasi terbaik adalah dengan tabel terpisah, yang tidak hanya memberikan kinerja yang lebih baik, tetapi juga memberikan kemandirian data yang lebih baik. Dalam IoT, data diproduksi terus menerus oleh setiap aplikasi, yang berarti bahwa tabel yang berbeda juga akan menjadi pilihan yang baik, memberikan cara yang independen untuk setiap aplikasi untuk menyimpan datanya dan dapat menskalakan tanpa mengorbankan kinerja.

Terakhir studi literatur keempat penelitian yang dilakukan oleh Kurniawan et al (2019), membahas tentang media penyimpanan data IoT dengan menggunakan NoSQL Cassandra karena mekanisme implementasi dan karakteristik penyimpanannya berbeda dengan NoSQL MongoDB. Pengujian dilakukan dari segi fungsionalitas pada Cassandra dalam menyimpan data dari *node* sensor, serta dari segi kinerja Cassandra dan MongoDB dalam melakukan operasi *insert* data *string* dan data gambar dengan menggunakan parameter *runtime*, *throughput*, *CPU usage*, *memory usage* dan *disk I/O*. Dari hasil pengujian fungsionalitas, Cassandra dapat menyimpan data yang heterogen dari *node* sensor [4]. Penelitian terhadap media penyimpanan data IoT dengan membandingkan Cassandra dan MongoDB ditinjau dari parameter *runtime*, *throughput*, *CPU usage*, *memory usage* dan *disk I/O* dalam melakukan operasi *insert* data. Data yang digunakan pada saat proses *insert* menggunakan data *string* dan data gambar. Selain itu, Cassandra yang diusulkan akan dilakukan pengujian fungsional dalam hal menyimpan dan menampilkan data IoT.



Gambar 4. Tapologi Penelitian

[Sumber: Kurniawan et al, 2019]

Hasil pengujian kinerja *database* dari sisi *backend data storage* dihasilkan nilai dari parameter *runtime* pada saat melakukan operasi *insert* menggunakan data *string*, MongoDB lebih unggul dibandingkan dengan Cassandra. Sedangkan pada saat melakukan operasi *insert* dengan data gambar, Cassandra lebih unggul dibandingkan dengan MongoDB. Hasil *throughput* MongoDB memiliki nilai *throughput* lebih cepat dari pada Cassandra saat melakukan operasi *insert* menggunakan data *string*. Sedangkan dengan data gambar, Cassandra yang memiliki nilai *throughput* lebih cepat dari pada MongoDB. Ditinjau dari parameter *memory usage*, Cassandra memiliki penggunaan memori yang efisien pada saat melakukan operasi *insert* menggunakan data *string* sedangkan pada saat melakukan operasi *insert* dengan data gambar MongoDB memiliki penggunaan memori yang efisien. Ditinjau dari parameter *CPU usage*, MongoDB memiliki penggunaan CPU yang efisien pada saat melakukan operasi *insert* data dengan menggunakan data *string*. Kebalikannya MongoDB memiliki penggunaan CPU yang efisien operasi *insert* dengan data gambar. Parameter terakhir *Disk I/O* kecepatan *hardisk* dalam menulis data dipengaruhi oleh besar kecil dari ukuran data yang ditransaksikan. Pada saat melakukan operasi *insert* data dengan data *string*, nilai dari *disk I/O* terbaik didapatkan pada MongoDB. Sedangkan pada saat melakukan operasi *insert* data dengan data gambar, nilai dari *disk I/O* terbaik didapatkan oleh Cassandra.

### 3. KESIMPULAN DAN SARAN

Studi literatur pertama menunjukkan Apache Cassandra memiliki performa transaksi data yang lebih baik daripada Apache HBase. Sehingga Cassandra merupakan sistem *database* yang paling baik untuk digunakan sebagai media penyimpanan data sensor IoT karena memiliki performa transaksi data yang lebih baik daripada HBase, karena suatu sistem IoT membutuhkan kecepatan yang baik dalam proses transaksi datanya. Jumlah *node database server* yang digunakan serta banyak baris data yang dimasukkan merupakan faktor-faktor yang mempengaruhi performa sistem *database* HBase maupun Cassandra. Apabila jumlah *node* yang digunakan semakin banyak dan jumlah data yang dimasukkan semakin besar, maka nilai *Throughput* akan semakin kecil dan nilai *Latency* akan semakin besar, sehingga akan menyebabkan nilai *Runtime* akan semakin besar atau proses transaksi data membutuhkan waktu yang semakin lama.

Pada studi literatur kedua memberikan titik referensi untuk menggunakan basis data Cassandra bertujuan pada tuntutan yang berbeda untuk merancang model data yang sesuai. Dengan populernya *Internet of Things* (IoT), *platform* pemantauan jaringan untuk keamanan dari bahan yang berbahaya merupakan bagian dari IoT. Masalah antara penyimpanan sejumlah besar data historis dan kontradiksi akses cepat akan membatasi aplikasi *platform* untuk ditingkatkan. Sedangkan *platform* sistem basis data sepenuhnya relasional tradisional untuk berpikir mode arsitektur penyimpanan data, menggunakan skema penggabungan beberapa teknologi basis data, menggunakan penyimpanan *database historical sensing* dengan Cassandra terbukti menjadi pilihan yang tepat. Berdasarkan pada karakteristik sistem basis data Cassandra, dikombinasikan dengan persyaratan penyimpanan dan permintaan data historis penginderaan, dengan merancang model penyimpanan data yang masuk akal, tidak hanya memastikan sejumlah data historis yang akan disimpan, tetapi juga secara efektif meningkatkan kecepatan respon *query* data.

Hasil penelitian studi literatur ketiga adalah setelah menguji dua arsitektur yang berbeda untuk lapisan data yang menyediakan dua pendekatan yang berbeda saat berurusan dengan data. *Database NoSQL* Cassandra telah dievaluasi dan diterapkan dalam *platform Internet of Things*. Cassandra dapat digunakan pada *platform* IoT sebagai sistem basis data utama karena memuat karakteristik yang diperlukan untuk menangani persyaratan keseluruhan *platform* ini. Dataset yang digunakan bisa lebih besar dan lebih heterogen, meskipun hasilnya menunjukkan perbedaan antara kedua pendekatan. Namun demikian, tes dengan dataset lebih besar dan dengan variasi data yang lebih besar diperlukan untuk memahami apakah skalabilitas merupakan masalah. Saran dimasa depan agar pengujian serupa dapat dilakukan dengan *sharding*, yang merupakan pembagian data secara horizontal yang meningkatkan kinerja keseluruhan *query*. Tujuan utamanya adalah untuk membagi aplikasi dengan *shard*, memberikan pendekatan yang mirip dengan tabel terpisah dan juga ingin mendistribusikan sistem, mengujinya untuk ketersediaan yang lebih baik.

Penelitian pada studi literatur keempat yaitu mengukur media penyimpanan data IoT dengan membandingkan Cassandra dan MongoDB ditinjau dari parameter *runtime*, *throughput*, *CPU usage*, *memory usage* dan *disk I/O*. Hasilnya dalam melakukan operasi *insert* data yang dapat diambil pada saat data yang dipertukarkan ukurannya besar, maka Cassandra menjadi solusi untuk media penyimpanan datanya. Akan tetapi jika data yang dipertukarkan ukurannya kecil, MongoDB menjadi solusi sebagai media penyimpanan datanya.

Keempat hasil penelitian studi literatur didapatkan kesimpulan bahwa Apache Cassandra memiliki performa transaksi data terbaik dikelas NoSQL dibandingkan dengan apache lain dikelasnya, seperti HBase dan MongoDB. Cassandra merupakan sistem *database* yang paling baik untuk digunakan sebagai penyimpanan media sensor *Internet of Things* (IoT) karena suatu sistem IoT membutuhkan kecepatan yang baik dalam proses transaksi data, ketersediaan tinggi tanpa mengganggu kinerja (*availability*) yang tinggi, sehingga Cassandra menjadi solusi untuk media penyimpanan data yang besar. Saran dimasa depan agar pengujian serupa dapat dilakukan dengan pembagian data yang lebih banyak dan besar dengan jaringan yang lebih luas lagi. Karena tantangan dimasa depan, data yang harus disimpan dan diproses untuk pertukaran data akan semakin banyak dan kompleks. Tujuannya untuk melihat apakah Cassandra tetap menjadi apache terbaik untuk *Internet of Things* dimasa yang akan datang atau ada apache lain dengan ketersediaan yang lebih baik dibanding Cassandra.

#### DAFTAR PUSTAKA

- [1] Zafar R, Yafi E, Zuhairi MF, Dao H. "Big Data: The NoSQL and RDBMS review", in *Proceedings of the 1st International Conference on Information and Communication Technology*, pp 120-126. 2016.
- [2] Ibrahim DM, Primananda R, Data M. "Perbandingan Performa Database Apache HBase dan Apache Cassandra Sebagai Media Penyimpanan Data Sensor Internet of Thing", *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol.2, No.8, pp 2943-2949. 2018.
- [3] Guohua Y, Qingyun D. "Research on Application of Sensing Data Storage BASED on Cassandra Database", in *Proceedings of the 2017 2nd International Conference on Communication and Information Systems*, Pages 355–360. 2017.
- [4] Kurniawan AK, Pramukantoro ES, Trisnawan PH. "Perbandingan Kinerja Cassandra dan MongoDB sebagai Backend IoT Data Storage", *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Vol.3, No.1, Pages 364-371. 2019.
- [5] Abramova V, Bernardino J, Furtado P. "Testing Cloud Benchmark Scalability with Cassandra", *2014 IEEE World Congress on Services*, Pages 434-441. 2014.
- [6] Duarte A, Bernardino J. "Cassandra for Internet of Things: An Experimental Evaluation", in *Proceedings of the International Conference on Internet of Things and Big data (IoTBD 2016)*, Pages 49-56. 2016.
- [7] Pramukantoro ES, Yahya W, Arganata G, Bhawiyuga A, Basuki A. "Topic Based IoT Data Storage Framework For Heterogeneous Sensor Data", *2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*. 2017.
- [8] Sukaridhoto Sitrusta. *Bermain dengan Internet of Things & Bigdata*. Politeknik Elektronika Negeri Surabaya (Pens), Surabaya, Indonesia. 2016.