**Perbandingan Kinerja Data Manipulation Language MongoDB dan PostgreSQL**

**M. Asep Ridwan dan Trisna Fauzi**

1. PENDAHULUAN

*Data merupakan bahan mentah bagi informasi yang dirumuskan sebagai kelompok lambang-lambang tidak acak yang menunjukkan tindakan-tindakan, hal-hal dan sebagainya. Data dibentuk dari lambang grafis seperti \* , $ , & . Data-data disusun untuk mengolah tujuan-tujuan menjadi susunan data, susunan kearsipan dan pusat data/landasan data (****Gordon B Davis).***

Dari pendapat di atas, dapat di ambil kesimpulan bahwa data itu dapat digunakan untuk berbagaimacam hal. Data yang sudah disusun sedemikan rupa dengan ketentuan atau aturan tertentu yang saling berelasi sehingga memudahkan pengguna dalam mengelolanya juga memudahkan memperoleh informasi disebut sebagai Database.

MongoDB adalah salah satu produk database noSQL type document yang menggunakan struktur data JSON untuk menyimpan datanya. MongoDB adalah salah satu database noSQL yang paling populer di internet. MongoDB sering dipakai untuk aplikasi berbasis Cloud, Grid Computing, atau Big Data.

PostgreSQL atau (Post-gress-SQL) adalah sebuah relational database manajemen system (RDBMS) yang di kembangkan oleh tim relawan yang ada di seluruh dunia yang bersifat open source.

Pada paper ini akan di bandingkan kinerja dari MongoDB dan PostgreSQL dimana beberapa aspek akan di ambil untuk pembanding antara kedua aplikasi tersebut.

1. METODE PENELITIAN

Aspek – aspek yang dijadikan acuan perbandingan dari kedua basis data ini adalah aspek Data Manipulation Language, yaitu operasi CRUD (Create, Read, Update, Delete). Perbandingan dari database – database tersebut dipisahkan menjadi 3 bagian, yaitu:

* Perbedaan syntax DML untuk MongoDB dan PostgreSQL.
* Perbedaan desain basisdata untuk MongoDB dan PostgreSQL.
* Perbedaan kinerja CRUD antara MongoDB dan PostgreSQL.

Proses perbandingan dilakukan dengan cara membuat dua buah program menggunakan bahasa pemrograman PHP yang berfungsi sebagai parameter perbandingan. Dengan perbedaan utama pada kedua aplikasi ini adalah pada basisdata yang dipergunakan yaitu dengan menggunakan MongoDB dan Postgresql. Adapun perangkat keras yang dipergunakan adalah :

* Processor: Intel Core i5-5200U up to 2,7GHz
* RAM: 4 GB DDR3L
* Harddisk: 500 GB

perangkat lunak yang digunakan adalah :

* Operating system window 10
* MongoDB 3.4
* postgreSQL

Adapun data yang dipergunakan untuk perbandingan yaitu data dosen yang mencakup nidn, nama, dan alamat. Pengujian dilakukan dengan melakukan CRUD (Create, Read, Update, Delete) secara berulang pada jumlah data 100, 200, 400, 800, 1600 dan 3200.

1. HASIL DAN PEMBAHASAN
2. Perbedaan syntax DML untuk MongoDB dan PostgreSQL.

3.1.1. MongoDB

1. Create
2. Membuat database

|  |
| --- |
| use namadatabase |

1. Membuat user

|  |
| --- |
| db.createUser({  user:”nama\_user”,  pwd:”password”,  roles:[“readWrite”,”dbAdmin”]}) |

1. Membuat collection

|  |
| --- |
| db.createCollection (‘nama\_collection’) |

1. Membuat dokument

|  |
| --- |
| db.nama\_collection.insert() |

1. Read
2. Menampilkan database

|  |
| --- |
| show dbs |

1. Menampilkan collection

|  |
| --- |
| show collections |

1. Menampilkan dokument

|  |
| --- |
| db.nama\_collection.find()  atau  db.nama\_collection.find().pretty() |

1. Update

|  |
| --- |
| db.nama\_colletion.update() |

1. Delete
2. Menghapus database

|  |
| --- |
| db.dropDatabase() |

1. Menghapus collection

|  |
| --- |
| db.nama\_collection.drop() |

1. Menghapus dokument

|  |
| --- |
| db.nama\_collection.remove() |

* + 1. PostgreSQL

1. Create

Membuat database

|  |
| --- |
| CREATE TABLE public.namaDatabase |

1. Read

Dengan Select

|  |
| --- |
| SELECT \* FROM public.namaTabel |

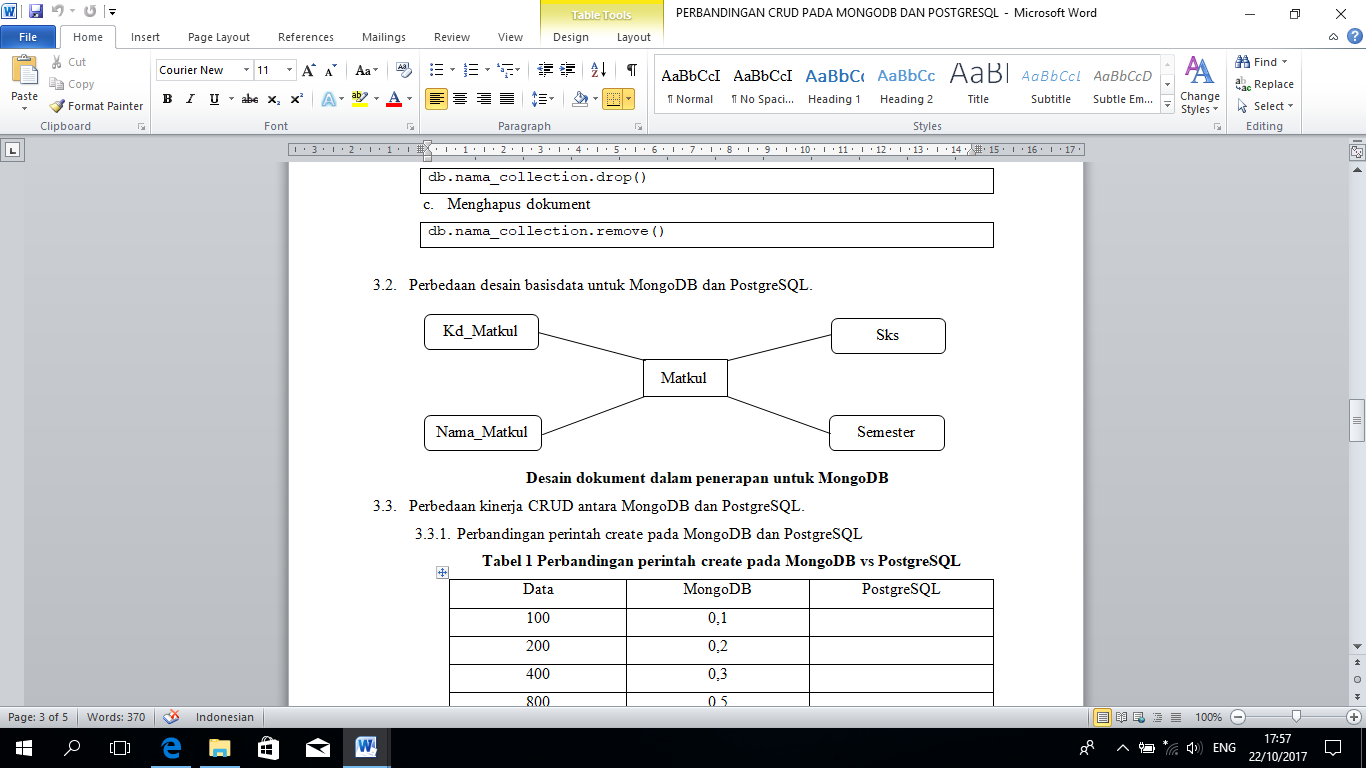
1. Update

|  |
| --- |
| UPDATE public.namaTabel  SET namaColom=value  WHERE <condition> |

1. Delete

|  |
| --- |
| DELETE FROM public.namaDatabase |

1. Perbedaan desain basisdata untuk MongoDB dan PostgreSQL.



**Gambar 1 Desain dokument dalam penerapan untuk MongoDB**

|  |
| --- |
| Data dosen |
| nidn  nama  alamat |

**Gambar 2 ERD dari 1 tabel dalam penerapan untuk PostgreSQL**

1. Perbedaan kinerja CRUD antara MongoDB dan PostgreSQL dalam waktu detik.
2. Perbandingan perintah create pada MongoDB dan PostgreSQL.

Syntax MongoDB

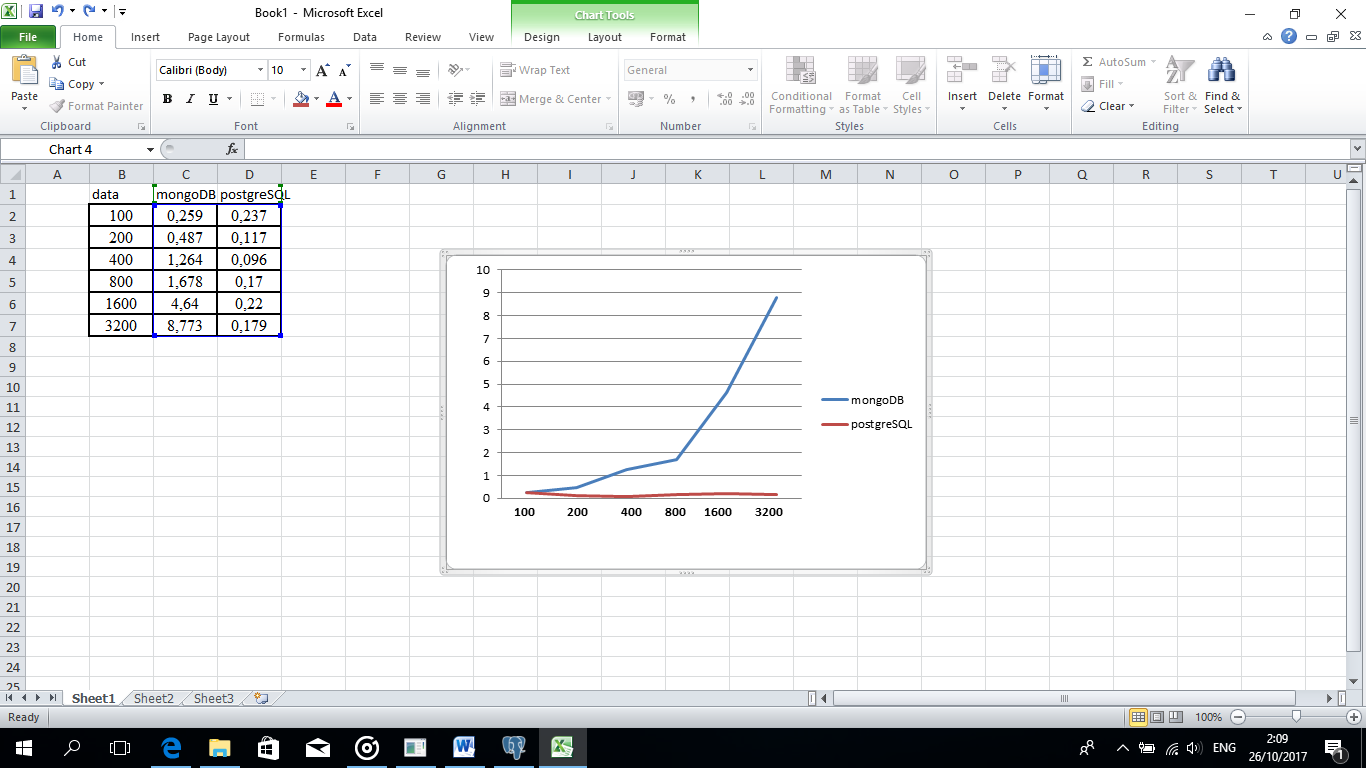
|  |
| --- |
| for (var i = 1; i <= /\*banyak\_data\*/; i++) db.nama\_collection.save({nidn:1,nama:”Jarwo”,alamat:”Jakarta”}) |

Syntax PostgreSQL

|  |
| --- |
| do $$  begin  for i in 1..100 loop  INSERT INTO public."dosen"(nidn, nama,alamat)  VALUES (i, 'Jarwo', 'Jakarta’);  end loop;  end;  $$; |

**Tabel 1 Perbandingan perintah create pada MongoDB vs PostgreSQL**

|  |  |  |
| --- | --- | --- |
| Data | MongoDB | PostgreSQL |
| 100 | 0,259 s | 0,058 s |
| 200 | 0,487 s | 0,065 s |
| 400 | 1,264 s | 0,118 s |
| 800 | 1,678 s | 0,233 s |
| 1600 | 4,640 s | 0,415 s |
| 3200 | 8,773 s | 0,867 s |



**Gambar 3 Perbandingan perintah Create MongoDB dan PostgreSQL**

1. Perbandingan perintah read pada MongoDB dan PostgreSQL

Syntax MongoDB

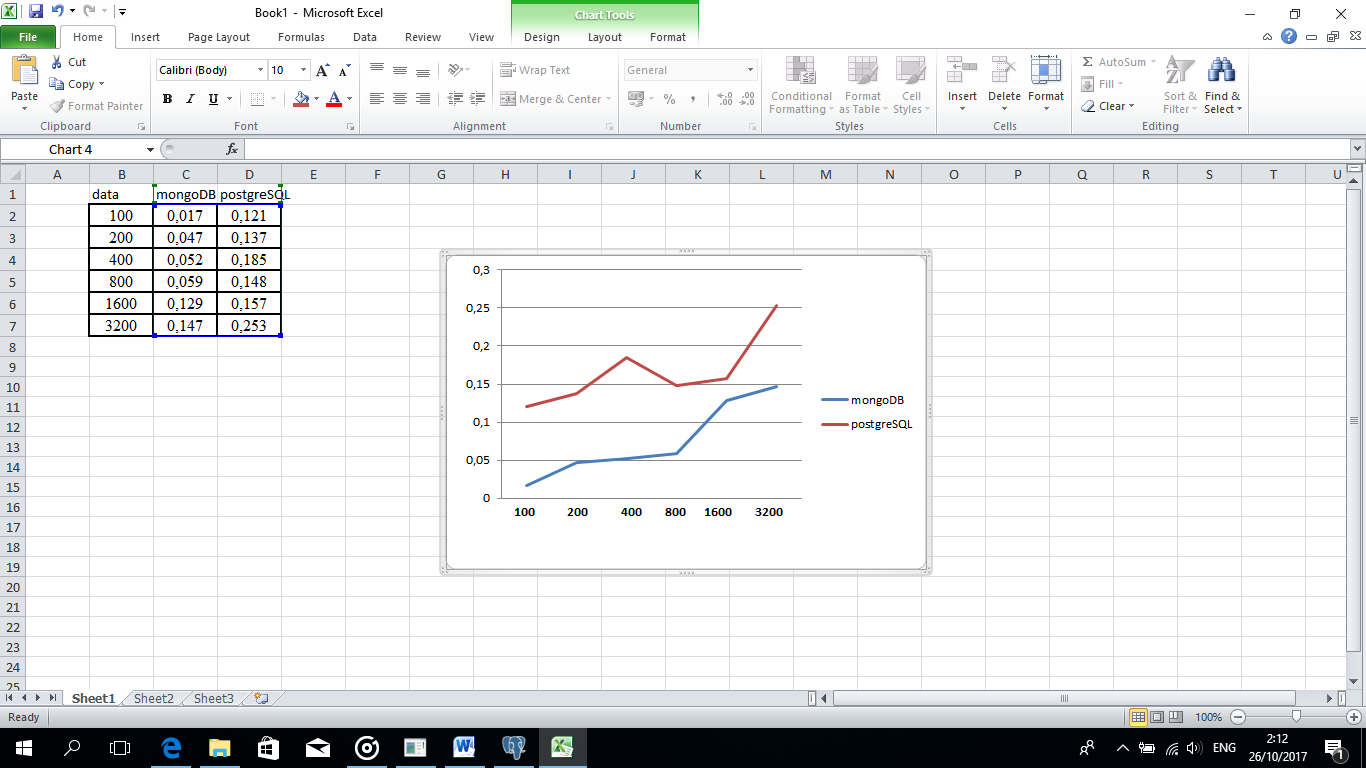
|  |
| --- |
| db.nama\_collection.find() |

Syntax PostgreSQL

|  |
| --- |
| SELECT \* FROM public."dosen" |

**Tabel 2 Perbandingan perintah read pada MongoDB vs PostgreSQL**

|  |  |  |
| --- | --- | --- |
| Data | MongoDB | PostgreSQL |
| 100 | 0,017 s | 0,159 s |
| 200 | 0,047 s | 0,393 s |
| 400 | 0,052 s | 0,801 s |
| 800 | 0,059 s | 1,563 s |
| 1600 | 0,129 s | 3,189 s |
| 3200 | 0,147 s | 4,418 s |



**Gambar 4 Perbandingan perintah Read MongoDB dan PostgreSQL**

1. Perbandingan perintah update pada MongoDB dan PostgreSQL

Syntax MongoDB

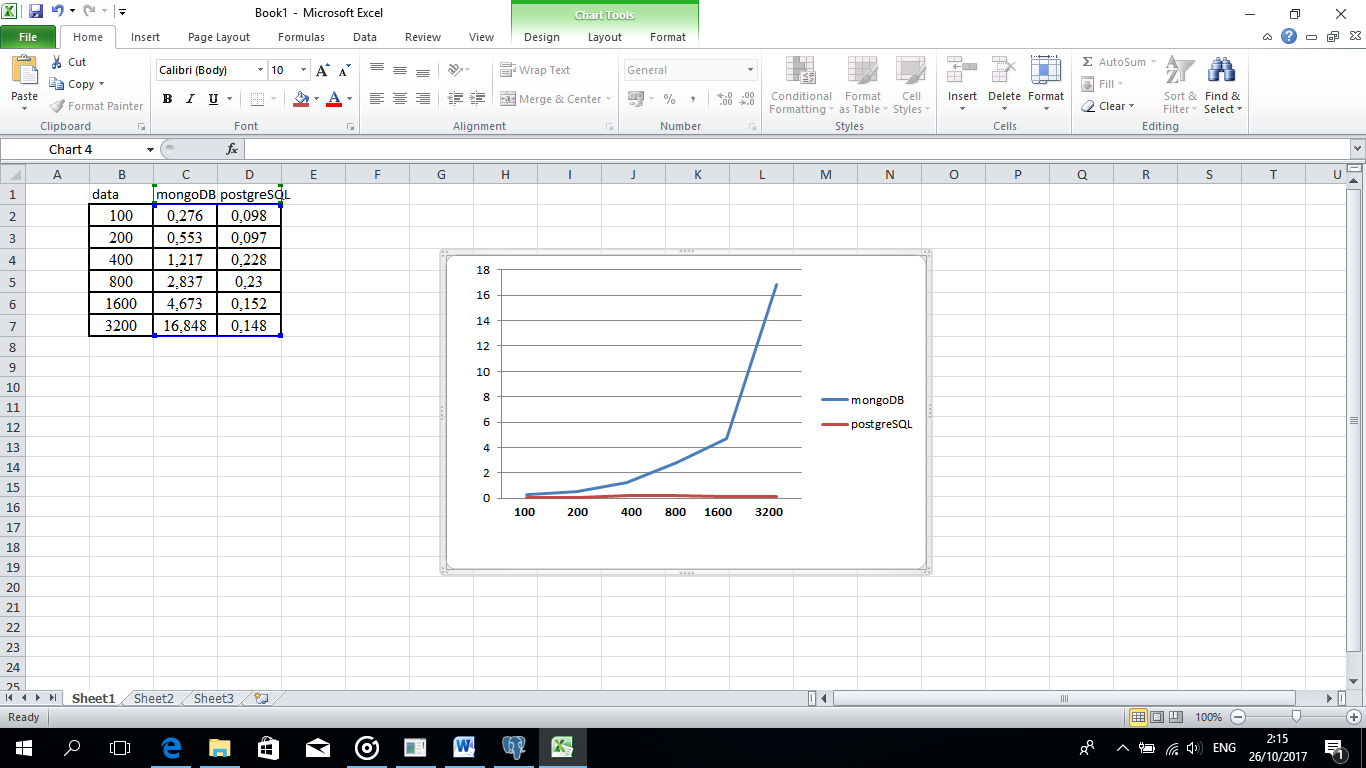
|  |
| --- |
| for (var i = 1; i <= /\*banyak\_data\*/; i++) db.nama\_collection.update({nama=”Jarwo Kuat”},{status:”Succesfully”}) |

Syntax PostgreSQL

|  |
| --- |
| UPDATE public."dosen"  SET nidn=?, nama=?, alamat=?  WHERE <condition>; |

**Tabel 3 Perbandingan perintah update pada MongoDB vs PostgreSQL**

|  |  |  |
| --- | --- | --- |
| Data | MongoDB | PostgreSQL |
| 100 | 0,276 s | 0,003 s |
| 200 | 0,553 s | 0,006 s |
| 400 | 1,217 s | 0,06 s |
| 800 | 2,837 s | 0,041 s |
| 1600 | 4,673 s | 0,011 s |
| 3200 | 16,848 s | 0,103 s |



**Gambar 5 Perbandingan perintah Update MongoDB dan PostgreSQL**

1. Perbandingan perintah delete pada MongoDB dan PostgreSQL

Syntax MongoDB

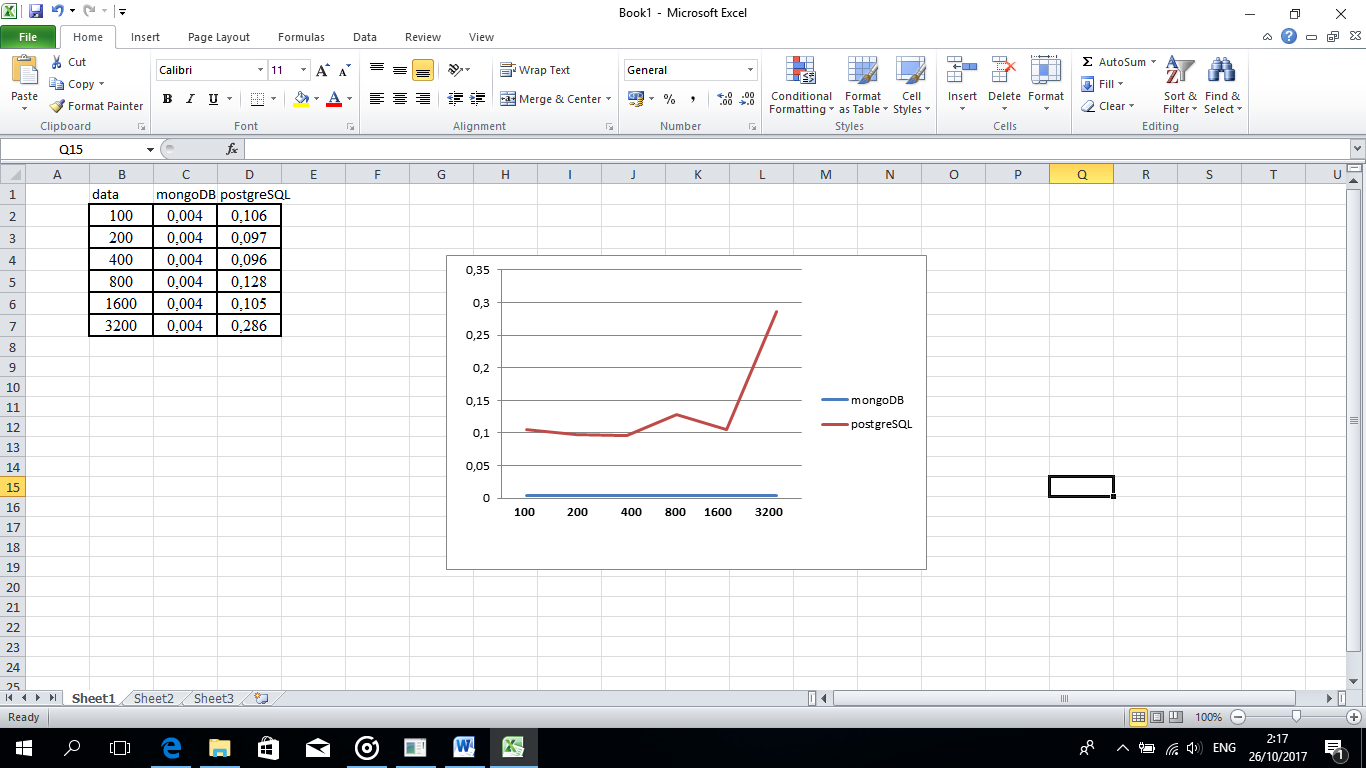
|  |
| --- |
| db.nama\_collection.drop() |

Syntax PostgreSQL

|  |
| --- |
| DELETE FROM public."dosen " |

**Tabel 4 Perbandingan perintah delete pada MongoDB vs PostgreSQL**

|  |  |  |
| --- | --- | --- |
| Data | MongoDB | PostgreSQL |
| 100 | 0,004 s | 0,004 s |
| 200 | 0,004 s | 0,038 s |
| 400 | 0,004 s | 0,004 s |
| 800 | 0,004 s | 0,003 s |
| 1600 | 0,004 s | 0,004 s |
| 3200 | 0,004 s | 0,005 s |



**Gambar 6 Perbandingan perintah Delete MongoDB dan PostgreSQL**

1. KESIMPULAN

Dari berbagai macam pengukuran yang dilakukan pada MongoDB dan PostgreSQL, maka dapat di ambil kesimpulan :

* Untuk design basisdata nya terlihat bahwa MongoDB lebih sederhana di bandingkan PostgreSQL.
* Untuk kinerja operasi sendiri, MongoDB unggul pada proses Read dan Delete, Sedangkan PostgreSQL unggul pada proses Create dan Update.

1. SARAN

Beberapa percobaan kembali dapat di lakukan untuk menghasilkan data yang lebih akurat, karena untuk percobaan nya sendiri mungkin akan berbeda pada perangkat yang berbeda pula, tergantung pada kondisi. Seperti saat melakukan pengujian dengan membuka aplikasi lain maka porses akan lebih lambat, dan waktu untuk pengerjaan proses nya pun mungkin akan berbeda.

UCAPAN TERIMAKASIH

Terimakasih kepada Universitas Siliwangi, khususnya pada pembimbing matakuliah Sistem Terdistribusi yang sudah mendukung penelitian ini.

DAFTAR PUSTAKA

* The PostgreSQL Global Development Group., 1996-2017, tersedia: <https://www.postgresql.org/>, diakses: 21 Oktober 2017
* MongoDB, Inc., 2017, tersedia: <https://www.mongodb.com/>, diakses: 21 Oktober 2017
* <http://php.net/manual/en/book.mongodb.php>

diakses: 22 Oktober 2017