



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**ANALISIS DAN PERBANDINGAN BASIS DATA NOSQL
COUCHDB DAN MONGODB DENGAN METODE STATISTIK**

TUGAS AKHIR

YAZID ILYAS BAIHAQI

0110220012

**PROGRAM STUDI TEKNIK INFORMATIK
SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI
AGUSTUS 2024**



**STT TERPADU
NURUL FIKRI**

SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**ANALISIS DAN PERBANDINGAN BASIS DATA NOSQL
COUCHDB DAN MONGODB DENGAN METODE STATISTIK**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar S.Kom

YAZID ILYAS BAIHAQI

0110220012

STT - NF

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

AGUSTUS 2024

HALAMAN PERNYATAAN ORISINALITAS

Skripsi/Tugas Akhir ini adalah hasil karya penulis, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Yazid Ilyas Baihaqi

NIM : 0110220012

Depok, 20 Juni 2024



Yazid Ilyas Baihaqi

STT - NF

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama : Yazid Ilyas Baihaqi

NIM : 0110220012


Program Studi : Teknik Informatika

Judul Skripsi : ANALISIS DAN PERBANDINGAN BASIS DATA NOSQL COUCHDB
DAN MONGODB DENGAN METODE STATISTIK


Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian
persyaratan yang diperlukan untuk memperoleh gelar Sarjana komputer pada Program
Studi Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

DEWAN PENGUJI

Pembimbing


(Amad Rio Adriansyah, S.Si., M.Si.)

Penguji


(Tilani Nabarian, S.Kom., M.T.I.)

Ditetapkan di : Senin

Tanggal : 22 Juli 2024

STT - NF

KATA PENGANTAR

Dengan mengucapkan Alhamdulillah segala puji dan syukur peneliti panjatkan atas kehadiran Allah SWT, berkat rahmat dan hidayah-Nya penyusunan skripsi ini dengan berjudul “Analisis Dan Perbandingan Basis Data NoSQL CouchDB Dan MongoDB Dengan Metode Statistik” ini dapat diselesaikan dengan tujuan untuk memenuhi salah satu persyaratan dalam menyelesaikan pendidikan pada Jurusan Teknik Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.

Dalam pengerjaan skripsi ini dipenuhi banyak sekali hambatan yang dihadapi termasuk hambatan ekonomi maupun mental. Namun berkat rahmat Allah SWT. Penyusun peneliti ini dapat menyelesaikan skripsi ini, dengan penuh kerendahan hati, pada kesempatan ini penulis mengucapkan terimakasih kepada :

1. Ahmad Rio Adriansyah, S.Si., M.Si. Selaku dosen pembimbing yang telah membantu dalam pengerjaan skripsi ini.
2. Tifani Nabarian, S.Kom., M.T.I. Selaku dosen penguji dan juga KAPRODI yang telah melakukan revisi dan memberikan ulasan mengenai skripsi ini.
3. Kepada keluarga dan orang tua penulis yang berada di sisi penulis dan memberikan bantuan kepada penulis ketika mengerjakan skripsi ini
4. Dan juga semuanya juga yang telah menemani penulis dalam periode pengerjaan skripsi ini.

STT - NF

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama : Yazid Ilyas Baihaqi.....

NIM : 0110220012.....

Program Studi : Teknik Informatika.....

Jenis karya : Skripsi / Tugas Akhir

demikian demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty - Free Right)** atas karya ilmiah saya yang berjudul :

ANALISIS DAN PERBANDINGAN BASIS DATA NOSQL COUCHDB DAN MONGODB DENGAN METODE STATISTIK

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

STT - NF Dibuat di : Depok
Pada tanggal : 20 Juli 2024

Yang Menyatakan



(Yazid Ilyas Baihaqi)

ABSTRAK

(300 kata)

Nama : Yazid Ilyas Baihaqi
NIM : 0110220012
Program Studi : Teknik Informatika
Judul : Analisis Dan Perbandingan Basis Data NoSQL CouchDB Dan MongoDB Dengan Metode Statistik

Tugas akhir ini membahas tentang analisis dan perbandingan basis data CouchDB dengan MongoDB. Karena waktu sendiri sangat penting dalam data seperti pengolahan dan penataan data. Dan bukan hanya dalam perusahaan besar, kepentingan pengolahan basis data juga dibutuhkan dalam lingkup kecil seperti organisasi kecil. Oleh karena itu, penelitian dibutuhkan untuk membandingkan kedua database NoSQL untuk mendapatkan perbedaan kecepatan respon atau kecepatan proses basis data masing masing. Metode yang digunakan untuk membandingkan basis data dapat dilakukan saat melakukan operasi CRUD. Dalam operasi ini, setiap basis data akan melakukan operasi *Create*, *Read*, *Update* dan *Delete*. Lalu waktu yang digunakan untuk basis data memproses data tersebut akan dibandingkan dan dievaluasi. Hasil dari penelitian analisis kedua basis data tersebut ini didapatkan bahwa MongoDB memiliki proses data lebih cepat dibanding CouchDB. Hal ini dibuktikan oleh waktu yang dibutuhkan CouchDB dalam proses *Insert*, *Select* dan *Delete*. Tetapi, dalam hal *Update* tidak dapat ditentukan dengan hanya waktu respon walaupun hasil menunjukkan bahwa hampir ada kesamaan dalam kecepatan yang dibutuhkan. Dikarenakan CouchDB tidak memiliki fungsi *update* melainkan memasukan data baru ke sistem basis data tersebut dengan ID relasi yang sama hingga menyebabkan data yang disimpan menjadi dua kali lebih besar dari sebelumnya. Dengan menggunakan sample 1000 mendapatkan hasil rata-rata MongoDB *Insert* 253ms, *Update* 586ms, *Select* 166ms, dan *Delete* 42ms. Sedangkan hasil rata-rata CouchDB *Insert* 635ms,

Update 865ms, *Select* 5023ms, dan *Delete* 1348ms dengan keterangan bahwa setelah *update* data CouchDB menjadi dua dikarenakan ketidakadanya fungsi *update* tersebut.

Kata kunci : CouchDB, MongoDB, Basis Data, NoSQL



STT - NF

ABSTRACT

Nama : Yazid Ilyas Baihaqi

NIM : 0110220012

Study Program : Teknik Informatika

*Title : Analysis And Comparason Between CouchDB And MongoDB
Using Statistical Method*

The focus of final research is about the analysis and comparason between CouchDB and MongoDB by using statistical method. Because of the advancing technology in these time, the factor of time in processin data has become more important than before. Not only in big companies, the importance of processing database is needed in small organization or small group. With that in mind the needed of comparason between to NoSQL database which will be discussed in this research that is MongoDB and CouchDB to achieve a comparason between those two. The methode that will be used in this research that is when using CRUD operation. In this operation, the database will execute operation Create, Read, Update and Delete. After that, the researcher will retrieve the time it took and evaluate the data. The result of this research using a statistic as a method, it's concluded from the data MongoDB have faster processing speed than CouchDB. It's proven on the time it takes for CouchDB to do Insert, Select and Delete. However, in terms of Update it can't be concluded despite the time it took for both of them almost the same in some cases. Because CouchDB itself doesn't have any function Update so instead they added the data with ID as relation that made the stored that becomes two times larger from before. With 1000 sample we get the average of MongoDB Insert 253ms, Update 586ms, Select 166ms, and Delete 42ms. With averages of CouchDB Insert 635ms, Update 865ms, Select 5023ms, and Delete 1348ms. Despite in difference the two have things in common and that is the usage of no SQL document type database, but interms of response time MongoDB still far beyond CouchDB.

Key words : CouchDB, Database, MongoDB, NoSQL

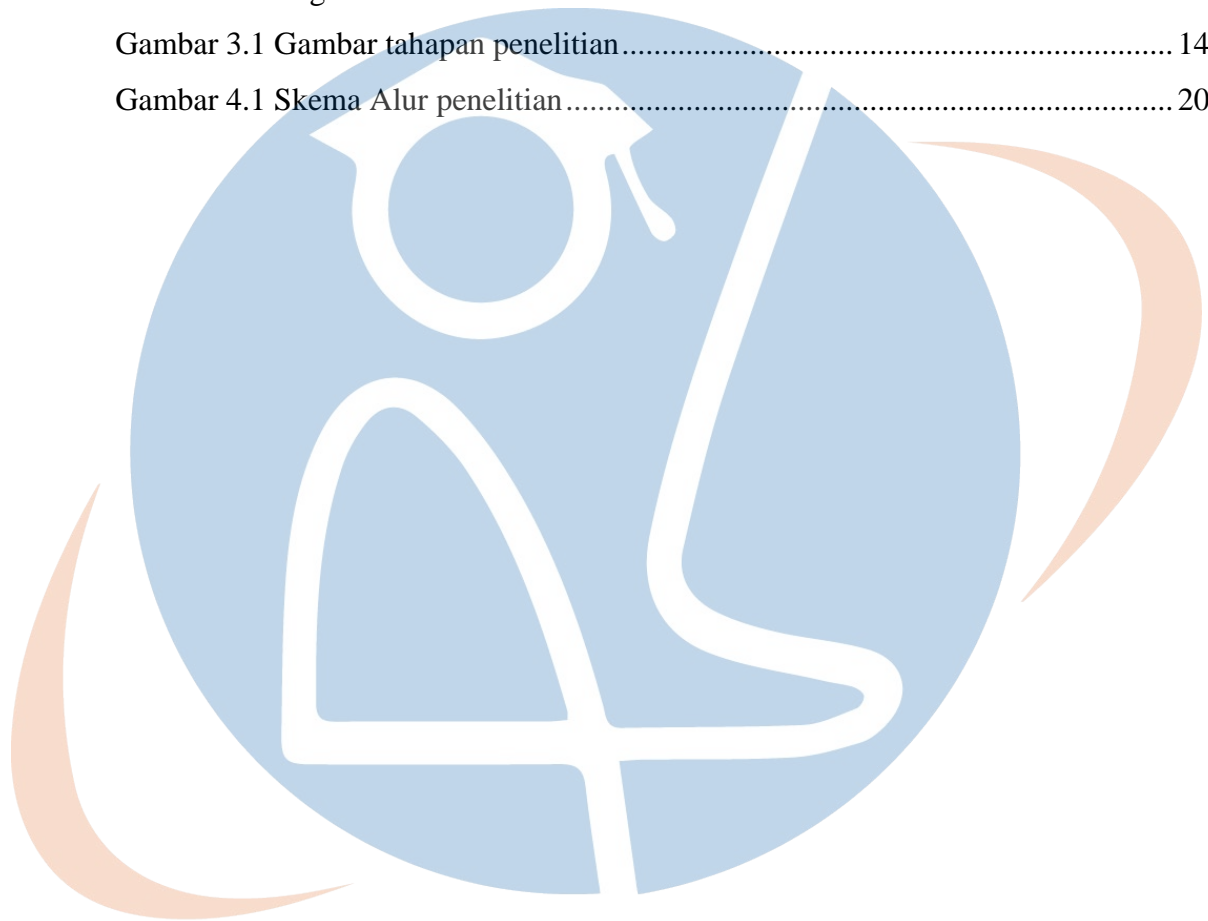
DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS.....	Error! Bookmark not defined.
HALAMAN PENGESAHAN.....	Error! Bookmark not defined.
KATA PENGANTAR	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS	Error! Bookmark not defined.
ABSTRAK	vii
ABSTRACT	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
DAFTAR BAGAN	xiv
BAB 1 PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan.....	3
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Penulisan.....	3
BAB 2 KAJIAN LITERATUR.....	6
2.1 Basis Data.....	6
2.2 SQL	6
2.3 NoSQL.....	7
2.4 MongoDB dan CouchDB	8

2.5	Node.JS.....	9
2.6	Teori Perbandingan Basis Data	10
2.7	Metode Perbandingan Basis Data.....	10
BAB 3 METODOLOGI PENELITIAN.....		14
3.1	Tahapan Penelitian	14
3.2	Rancangan Penelitian	15
3.2.1	Jenis Penelitian.....	15
3.2.2	Metode Analisis Data.....	15
3.2.3	Metode Pengumpulan Data.....	16
3.2.4	Metode Pengujian	17
3.2.5	Metode Implementasi dan Evaluasi	17
3.2.6	Lingkungan Pengembangan.....	17
BAB 4 IMPLEMENTASI DAN EVALUASI.....		19
4.1	Rancangan Penelitian	19
4.2	Hasil Penelitian.....	21
4.2.1	Struktur Desain Sistem Basis Data	21
4.2.2	Implementasi Kode Dalam Analisis.....	22
4.2.3	Data Hasil Implementasi.....	26
4.2.4	Analisis Data Implementasi	27
4.3	Evaluasi Hasil Penelitian.....	29
BAB 5 KESIMPULAN DAN SARAN		31
5.1	Kesimpulan.....	31
5.2	Saran.....	31
DAFTAR PUSTAKA		33

DAFTAR GAMBAR

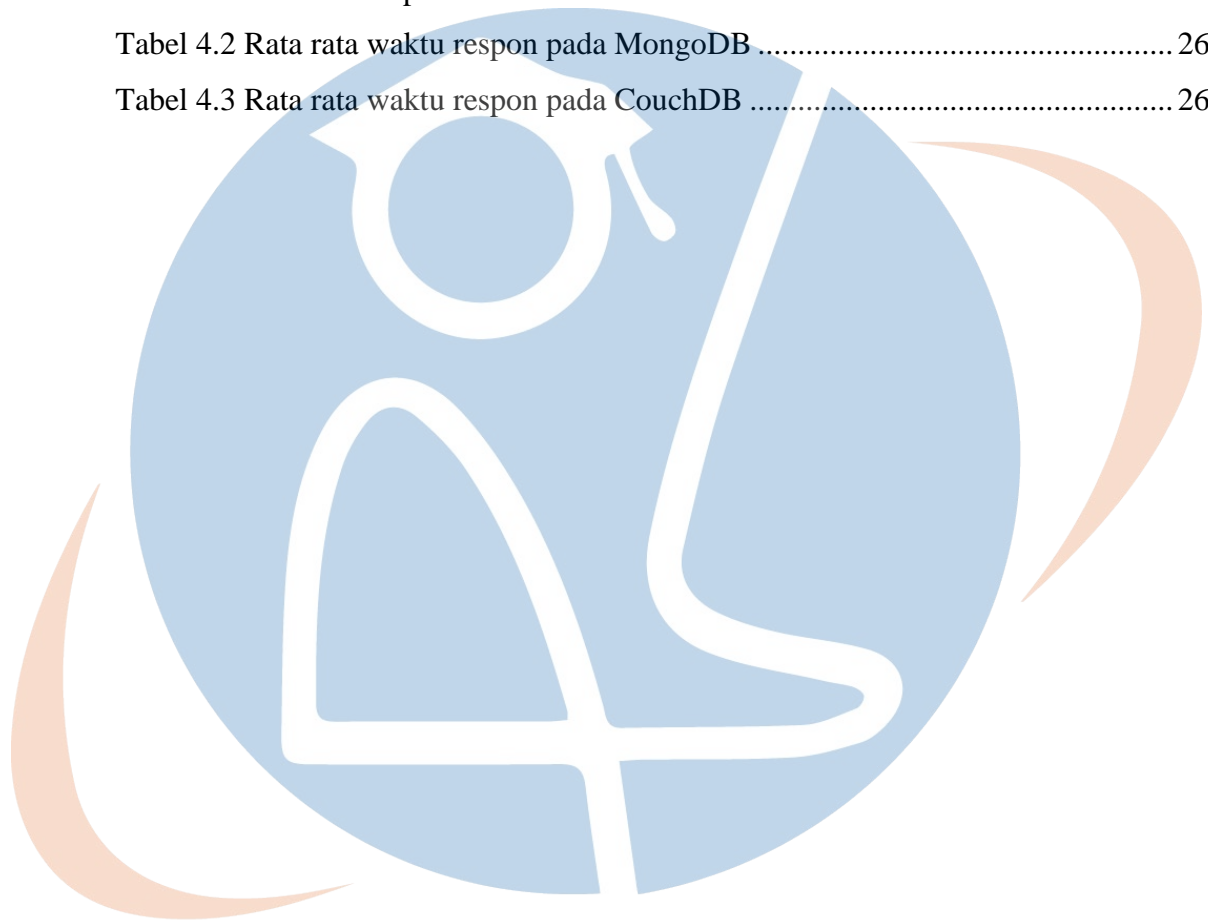
Gambar 1.1 Gambar hasil penelusuran basis data NoSQL yang terkini	1
Gambar 2.1 Logo CouchDB	9
Gambar 2.2 Logo MongoDB	9
Gambar 2.3 Logo Node.JS	10
Gambar 3.1 Gambar tahapan penelitian	14
Gambar 4.1 Skema Alur penelitian	20



STT - NF

DAFTAR TABEL

Tabel 2.1 Perbandingan NoSQL dan SQL.....	7
Tabel 2.2 Perbandingan MongoDB dan CouchDB.....	8
Tabel 2.3 Penelitian Terkait	11
Tabel 4.1 Tabel Persiapan Untuk Melakukan Alur Penelitian.....	20
Tabel 4.2 Rata rata waktu respon pada MongoDB	26
Tabel 4.3 Rata rata waktu respon pada CouchDB	26



STT - NF

DAFTAR BAGAN

Bagan 4.1 Perbandingan fungsi <i>insert</i> dalam ms	27
Bagan 4.2 Perbandingan fungsi <i>update</i> dalam ms	28
Bagan 4.3 Perbandingan fungsi <i>select</i> dalam ms	28
Bagan 4.4 Perbandingan fungsi <i>delete</i> dalam ms.....	29



STT - NF

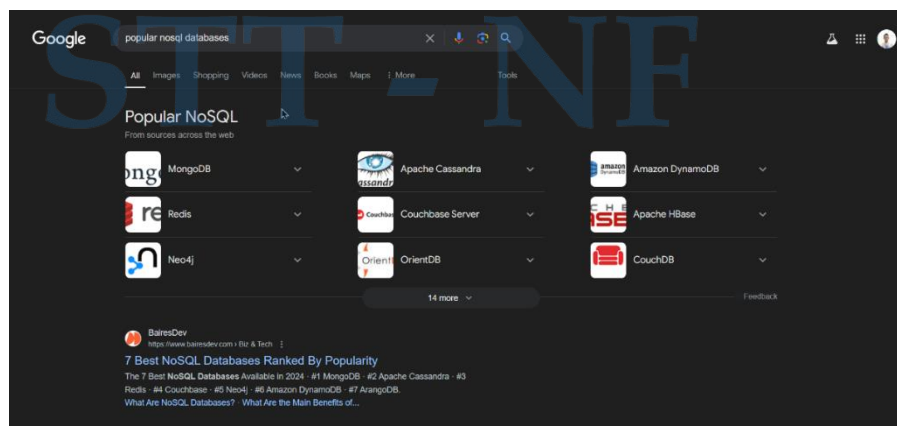
BAB 1

PENDAHULUAN

1.1 Latar belakang

Dalam pemrograman aplikasi, *database* atau basis data merupakan komponen dalam aplikasi. Tanpa basis data, aplikasi tidak memiliki tempat untuk menyimpan data yang ingin digunakan atau diproses. Performa basis data juga menentukan hasil dari data yang di proses, jika sistem basis data tersebut tidak kokoh atau dengan mudah rusak maka data yang ada didalam tersebut dapat rusak atau hilang. Dan jika basis data tersebut memiliki performa yang lambat seperti dalam pengambilan data atau pemasukan data yang dapat mengurangi kecepatan aplikasi tersebut. [1], [2], [3]

Semakin berkembangnya zaman, maka semakin berkembangnya teknologi secara pesat. Seperti contoh teknologi basis data, sekarang terdapat dua tipe basis data yaitu NoSQL dan SQL. SQL atau *Structured Query Language* sendiri menggunakan konsep basis data relasi yang terstruktur seperti table dan memiliki keunggulan yaitu mudah dalam memanipulasi data. Sedangkan menurut Vaish pada tahun 2013, NoSQL adalah istilah umum yang merujuk pada penyimpanan data yang tidak mengikuti aturan dari NoSQL tidak menggunakan basis data berupa relasi tetapi menggunakan metode sendiri yaitu *key-value*, *document-oriented*, *graph*, dan masih banyak lagi. [3], [4], [5]



Gambar 1.1 Gambar hasil penelusuran basis data NoSQL yang terkini

Contoh dari basis data NoSQL yang menggunakan document-oriented adalah CouchDB dan MongoDB. CouchDB dan MongoDB sendiri merupakan salah satu basis data NoSQL yang sering digunakan atau sering dipakai, alasan dipilihnya MongoDB dan CouchDB sendiri dikarenakan sekilas kedua basis data tersebut memiliki kemiripan seperti fungsi dalam *select* maupun dalam *insert* dan juga CouchDB merupakan salah satu alternatif dari MongoDB. Seperti yang dijelaskan dalam gambar 1.1, MongoDB dan CouchDB merupakan salah satu dari basis data NoSQL yang populer. Dan alasan digunakannya CouchDB karena CouchDB tersebut yang memiliki kemiripan dengan MongoDB. MongoDB Seorang developer harus menentukan basis data yang tepat untuk kebutuhannya. Jika aplikasi tersebut membutuhkan penyimpanan yang besar maka developer tersebut sebaiknya menggunakan SQL. Sedangkan jika aplikasi tersebut memproses dan memasukan data yang banyak dalam waktu yang cepat maka developer sebaiknya menggunakan NoSQL karena struktur datanya berupa *document oriented*. Tetapi, NoSQL dan SQL juga memiliki kekurangan masing masing seperti NoSQL rentan filenya rusak dan SQL memiliki waktu pemrosesan yang lama. Seperti halnya kedua basis data tersebut, basis data tipe NoSQL seperti CouchDB dan MongoDB memiliki kegunaan masing masing. Yang dipertanyakan adalah manakah yang harus dipakai dalam keadaan menginput, mengambil, dan juga keamanan data. Pentingnya menentukan perbandingan kedua database tersebut adalah untuk mendapatkan siapa yang lebih unggul dalam hal waktu pemrosesan. Karena waktu sendiri sangat penting dalam data seperti pengolahan dan penataan data. Dan bukan hanya dalam perusahaan besar, kepentingan pengolahan basis data juga dibutuhkan dalam lingkup kecil seperti organisasi kecil. Oleh karena itu, penelitian dibutuhkan untuk membandingkan kedua database NoSQL untuk mendapatkan perbedaan kecepatan respon atau kecepatan proses basis data masing masing. [6]

1.2 Rumusan Masalah

1. Bagaimana cara membandingkan performa kecepatan antara dua buah basis data CouchDB dengan MongoDB?
2. Basis data manakah yang lebih cepat performanya dengan dibandingkan melalui waktu respon?

1.3 Tujuan

Tujuan dari penelitian ini adalah:

1. Mendapatkan metode untuk membandingkan kedua basis data yang dapat membantu pembaca atau peneliti menentukan basis data yang akan digunakan untuk pengolahan data.
2. Mendapatkan perbandingan performa kecepatan secara waktu respon.

1.4 Manfaat Penelitian

Manfaat dari penelitian ini adalah mendapatkan hasil analisis kedua basis data tersebut yang dapat digunakan untuk referensi menentukan basis data yang akan digunakan baik untuk pengolahan data informasi maupun untuk penelitian selanjutnya atau penggunaan dalam aplikasi.

1.5 Batasan Masalah

1. Penelitian hanya menguji coba MongoDB dan CouchDB secara lokal.
2. Bahasa pemrograman yang akan digunakan adalah Node.JS dan *front-end* yang akan digunakan adalah Express.
3. Data yang akan dipakai merupakan data informasi *dummy*.

1.6 Sistematika Penulisan

Dalam penyusunan penelitian ini, untuk mengarah pada pokok permasalahan dan mengatur penyusunan bab. Penyajian laporan penelitian ini menggunakan sistematika penulisan sebagai berikut:

1. Bagian Awal Skripsi

Bagian awal penelitian ini memuat halaman sampul depan, halaman judul, halaman pernyataan orisinalitas, halaman pengesahan, halaman kata pengantar, halaman pernyataan persetujuan publikasi, halaman abstrak bahasa Indonesia dan Inggris, dan halaman daftar isi, gambar, tabel dan bagan

2. Bagian Utama Skripsi

Bagian utama ini terdiri atas bab dan sub bab dari sebagai berikut:

BAB I PENDAHULUAN

Dalam bab ini, peneliti membahas tentang Latar Belakang penelitian, Rumusan Masalah, Tujuan, Manfaat Penelitian, dan Sistematika Penulisan.

BAB II KAJIAN LITERATUR

Dalam bab ini peneliti menulis tentang teori teori dan kajian yang telah di kutip dari jurnal penelitian dan juga buku yang dikaji. Dalam bab ini juga meliputi :

- A. Penjelasan teori teori yang akan dibahas atau digunakan
- B. Teori perbandingan

BAB III METODOLOGI PENELITIAN

Dalam bab ini berisi tentang metodologi dan juga analisis tentang penelitian yang akan dilakukan oleh peneliti. Di bab ini akan juga membahas tentang alat dan metode yang akan digunakan untuk membandingkan penelitian tersebut. Isi dari bab ini berisi dengan sub bab yang meliputi :

- A. Tahapan Penelitian
- B. Rancangan Penelitian

BAB IV METODOLOGI PENELITIAN

Dalam bab ini membahas tentang implementasi yang telah di rancan sebelumnya, serta hasil dan evaluasi akan dibahas secara detail atau berupa table di bab ini.

BAB V KESIMPULAN DAN SARAN

Bab ini akan membahas tentang kesimpulan yang didapatkan pada bab sebelumnya dan membahas tentang tujuan dari penelitian ini.



STT - NF

BAB 2

KAJIAN LITERATUR

2.1 Basis Data

Pengertian dari basis data adalah suatu koleksi yang diatur serupa berisi informasi atau data yang disimpan dalam hardware suatu komputer. Basis data sendiri biasanya dikontrol oleh *Database Management System* (DBMS). Sistem basis data biasanya disebut ketika data dan DBMS digunakan secara bersamaan yang dihubungkan oleh suatu aplikasi. Basis data banyak digunakan dalam berbagai aplikasi seperti bisnis, pendidikan, penelitian, dan masih banyak bidang lainnya. [7]

Tipe basis data sendiri beragam seperti basis data relasional, basis data berorientasi objek, gudang data, grafik data dan masih banyak lagi. Akan tetapi dari semua tipe basis data tersebut memiliki kesamaan yaitu sebagian menggunakan *row* atau *columns* dan sebagian tidak menggunakan desain tersebut. Maka klasifikasi basis data dibagi menjadi dua yaitu NoSQL atau Not only SQL dan SQL. Tipe basis data sangat penting dalam pengaruh aplikasi, karena aplikasi sangat bergantung pada sistem basis data yang memiliki peran penting dalam menjaga data data dan juga memproses data tersebut tanpa ada kesalahan sekalipun [4], [8]

2.2 SQL

SQL atau *Structured Query Language* adalah bahas pemrograman standar yang digunakan untuk manajemen basis data relasional dan melakukan banyak operasi terhadap data yang ada didalam basis data tersebut. Basis data SQL dapat mudah ditemukan di beberapa aplikasi, karena SQL yang menggunakan relasi tabel dan model relasi merupakan salah satu cara yang paling mudah untuk mengkoleksi data yang ada di tabel tersebut. [9]

Basis data yang menggunakan SQL atau tabel relasional adalah lain MySQL, PostgreSQL dan juga Microsoft SQL Server. Dalam SQL sendiri memiliki kelebihan yaitu dalam kapasitas dan keamanan. SQL memiliki kapabilitas dalam penambahan kapasitas yang lebih banyak, dikarenakan tipe

atau model data relasi tersebut. Dan juga karena menggunakan model relasi, semakin jarang terjadinya error atau data yang rusak dalam basis data tersebut.

2.3 NoSQL

Menurut Gaurav Vaish, NoSQL adalah istilah umum yang merujuk pada penyimpanan data yang tidak mengikuti aturan dari Relational Basis data. Awal mula terdirinya NoSQL dimulai pada tahun 1998 yang ditemukan oleh Carlo Strozzi. Karena berkembangnya teknologi dan juga aplikasi yang semakin menyatu pada kehidupan manusia seperti sosial media, *e-commerce*, penyimpanan *cloud*, dan masih banyak lagi. Integrasinya aplikasi tersebut membuat semakin bertambahnya data yang disimpan dan juga data yang akan di proses yang pada akhirnya mengarah ke berkembangnya NoSQL. [4]

Yang membedakan NoSQL dengan SQL yaitu struktur datanya. SQL lebih menggunakan struktur tabel atau relasional sedangkan NoSQL menggunakan document atau JSON. Sistem NoSQL sendiri tidak hanya menggunakan struktur document atau JSON, banyak sekali jenis jenis sistem yang tersedia antara lain *graph-based*, *column-based*, dan juga *key-value* model data atau juga dikenal sebagai JSON (*JavaScript Object Notation*). NoSQL sendiri memiliki kelebihan khusus seperti sebagian besar basis data NoSQL *open-source* yakni dapat digunakan atau dilihat secara gratis oleh user lainnya. [9], [10], [11]

Tabel 2.1 Perbandingan NoSQL dan SQL

NoSQL	SQL
Tidak menggunakan design relasi	Menggunakan design relasi
Memiliki skema yang dinamis untuk data yang tidak teratur	Memiliki data yang teratur dan skema yang telah di definisikan
Berupa dokumen, grafik, dan juga kolom	Berupa tabel
Sangat berskala secara horizontal	Sangat berskala secara vertikal

2.4 MongoDB dan CouchDB

MongoDB adalah suatu tipe basis data yang berbasis dokumen dan basis data NoSQL. NoSQL atau *Not OnlySQL*, merupakan basis data yang didesain untuk mengelolah data secara besar. Dalam MongoDB, lebih dikenal menggunakan model dokumen dibanding dengan SQL yang lain yaitu row. Dikarenakan format berupa JSON, mempermudah memformat atau mengambil data dengan cepat. [4], [5], [6]

CouchDB tidak jauh berbeda dengan MongoDB, memiliki desain orientasi dokumen dan tipe basis data NoSQL. CouchDB sendiri adalah *single-node* yang memiliki fungsi sama terhadap basis data lain. CouchDB memiliki keseriusan dalam keutuhan data dengan *multi-node* kluster yang menyimpan data secara bersamaan dan berlebihan. Hal ini memiliki kelebihan yaitu data selalu ada ketika dibutuhkan untuk mengambil atau menyimpan. [9]

Tabel 2.2 Perbandingan MongoDB dan CouchDB

MongoDB	CouchDB
Menyimpan data dalam format BSON	Menyimpan data dalam format JSON
Sistem basis data menyimpan berupa koleksi	Sistem basis data menyimpan berupa Dokumen
Memfokuskan konsistensi	Memfokuskan ketersediaan
Mengikuti <i>Map/Reduce</i> dalam pembuatan kolesi dan bahasa query berbasis objek	Mengikuti metode <i>Map/Reduce query</i>
Menggunakan antarmuka TCP/IP	Menggunakan antarmuka HTTP/REST
Tidak dapat berjalan di perangkat genggam	Dapat berjalan di Apple iOS dan Android



Gambar 2.1 Logo CouchDB



Gambar 2.2 Logo MongoDB

2.5 Node.JS

Node.JS adalah runtime JavaScript berbasis event asynchronous, dirancangnya untuk aplikasi yang skalabel yang berbasis website. Node.Js sendiri merupakan *open source* dan *cross platform*. Dalam Node.JS kita dapat menjalankan banyak koneksi secara bersamaan. Setiap koneksi dapat memanggil callback secara bersamaan, tetapi jika tidak ada pekerjaan atau fungsi yang berjalan maka Node.js akan tertidur. Hal ini yang membedakan Node.js dengan model environment lainnya yang menggunakan *Thread OS*.

Selain itu, proses *deadlock* tidak akan terjadi pada user dikarenakan Node.js tidak melakukan I/O secara langsung. Karena itu pemblokiran tidak akan terjadi kecuali jika user melakukan I/O metode dari pustaka standar Node.JS. Alasan menggunakan Node.JS adalah *library* atau *package* yang

sangat berguna, salah satunya adalah Morgan yang dapat digunakan sebagai logger aplikasi dan juga Express.Js sebagai framework atau API. [12], [13]



Gambar 2.3 Logo Node.JS

2.6 Teori Perbandingan Basis Data

Dalam segi dasar, karena struktur basis data yang berbeda dalam perbandingan NoSQL dan MySQL sangat terlihat. Seperti pada SQL yang menggunakan tabel sedangkan NoSQL menggunakan *document*. Akan tetapi dalam perbandingan NoSQL dengan NoSQL lain, yang dilihat tidak hanya konsep struktur dari basis data tersebut melainkan dilihat dari fitur atau efisiensi yang ada di dalam basis data masing masing.

Dikarenakan adanya perbedaan dalam fitur dan juga fungsi, perbandingan dilakukan dalam pengujian statistik. Yaitu perbandingan seberapa lama waktu yang dibutuhkan ketika menginput dan juga mengoutput suatu API atau perintah. MongoDB memiliki *server-side script* berupa JavaScript sedangkan CouchDB sendiri memiliki fungsi dan *timer* di JavaScript dan UDFs di Java, Python, dan SQL++.

2.7 Metode Perbandingan Basis Data

Metode yang digunakan untuk membandingkan basis data dapat dilakukan saat melakukan operasi CRUD. Dalam operasi ini, setiap basis data akan melakukan operasi *Create, Read, Update* dan *Delete*. Lalu waktu yang digunakan untuk basis data memproses data tersebut akan dibandingkan dan dianalisa dan operasi ini akan dilakukan dalam lingkungan Node.JS berbasis JavaScript dengan Postman sebagai memanggil API tersebut lalu mendapatkan waktu respon seberapa lama dokumen atau data di proses. [10]

Dalam metode pengujian, dilakukan adalah *statistical testing*. Menurut pendapat dari Sugiyono, menyatakan bahwa teknik analisis data merupakan langkah - langkah untuk mendapatkan hasil data yang diinginkan, mengolah dengan cara sistematis data yang didapatkan dari hasil wawancara, survey lapangan, dan dokumentasi, dengan cara mengelompokan data ke dalam kategori tertentu, dan mengelompokan ke unit-unit yang sudah di sesuaikan. Metode analisis statistik data bisa dimanfaatkan, dipilih dan disesuaikan dengan tujuan penelitian tersebut.

Statistical testing ini sangat berguna dalam perbandingan dua basis data yang akan di uji coba dalam metode analisis yang telah disebutkan. Perbandingan dilakukan ketika mendapatkan rata rata dalam data tersebut, seperti contoh membandingkan waktu rata rata pada fungsi *read* MongoDB dengan CouchDB. [14]

Tabel 2.3 Penelitian Terkait

No	Nama dan Tahun	Judul	Topik	Subjek	Hasil
1	Niteshwar Datt Bhardwaj, 2016	<i>Comparative Study of CouchDB and MongoDB – NoSQL Document Oriented Databases</i>	Basis data	Developer	Analisis CouchDB dan MongoDB
2	Akhmad Aan Said Daroini, 2016	PERBANDINGAN PENGGUNAAN NOSQL MONGODB DAN MYSQL PADA BASIS DATA FORUM KOMUNIKASI	Basis data	Developer Back-end	MongoDB memiliki keunggulan dalam efisiensi sedangkan MySQL memiliki keunggulan dalam kapasitas

3.	Kharisma Syahputri, 2023	<i>Peran Database Dalam Sistem Informasi Manajemen</i>	Basis data	Developer	Kecepatan basis data sangat berpengaruh dalam sistim informasi manajemen
4.	Figueiredo, 2016	<i>Managing knowledge – the importance of databases in the scientific production</i>	Basis data dalam penggunaan ilmiah	Peneliti	Basis data sangat dibutuhkan dalam penelitian. Walaupun masih dalam manual seperti dalam validasi data.

Berikut menjelaskan lebih rinci mengenai masing masing penelitian terkait yang ditunjukkan pada tabel 2.3 :

1. Nitesh Daat Bhardwaj melakukan penelitian tersebut dalam tujuan untuk mendapatkan perbandingan performa dan reliabilitas dari kedua sistem basis data tersebut. Penelitian ini memfokuskan komparasi basis data menggunakan bahasa pemrograman Java dan juga alat berupa Apache JMeter karena banyak sekali basis data NoSQL tetapi memiliki kelebihan dan kekurangan masing masing. Kesimpulan yang didapatkan adalah MongoDB memiliki respon lebih cepat dibanding CouchDB dalam lingkungan Apache. [10]
2. Akhmad Aan Said Daroni, Wiyli Yustanti. Melakukan sebuah penelitian berjudul Perbandingan Penggunaan NoSQL MongoDB dan MySQL Pada Basis Data Forum Komunikasi, pada tahun 2016. Dalam penelitian disebutkan bahwa banyak teknologi basis data yang dapat memudahkan penggunanya dalam mengakses data. Salah satu teknologi yang terbesar adalah NoSQL dan SQL. SQL atau *Structured Query Language* adalah sebuah basis data yang menggunakan relasi dengan struktur dasar yang

menggunakan tabel. Keunggulan dari SQL sendiri yaitu data yang telah di masukan atau di olah telah terstruktur. Sedangkan NoSQL sendiri kebalikannya dari SQL tidak menggunakan tabel yang terstruktur namun menggunakan *document-oriented* dan juga didasarkan oleh *key-value*. Tujuan dari penelitian itu untuk mendapatkan informasi manakan dari kedua tipe basis data yang lebih cepat dalam skenario menangani transaksi atau keluar masuknya data dalam jumlah yang besar. Kesimpulan yang didapatkan dalam penelitian tersebut adalah MySQL unggul dalam pengambilan data tetapi MongoDB memiliki kecepatan lebih dalam memasukan, memperbaharui, dan menghapus data. [4]

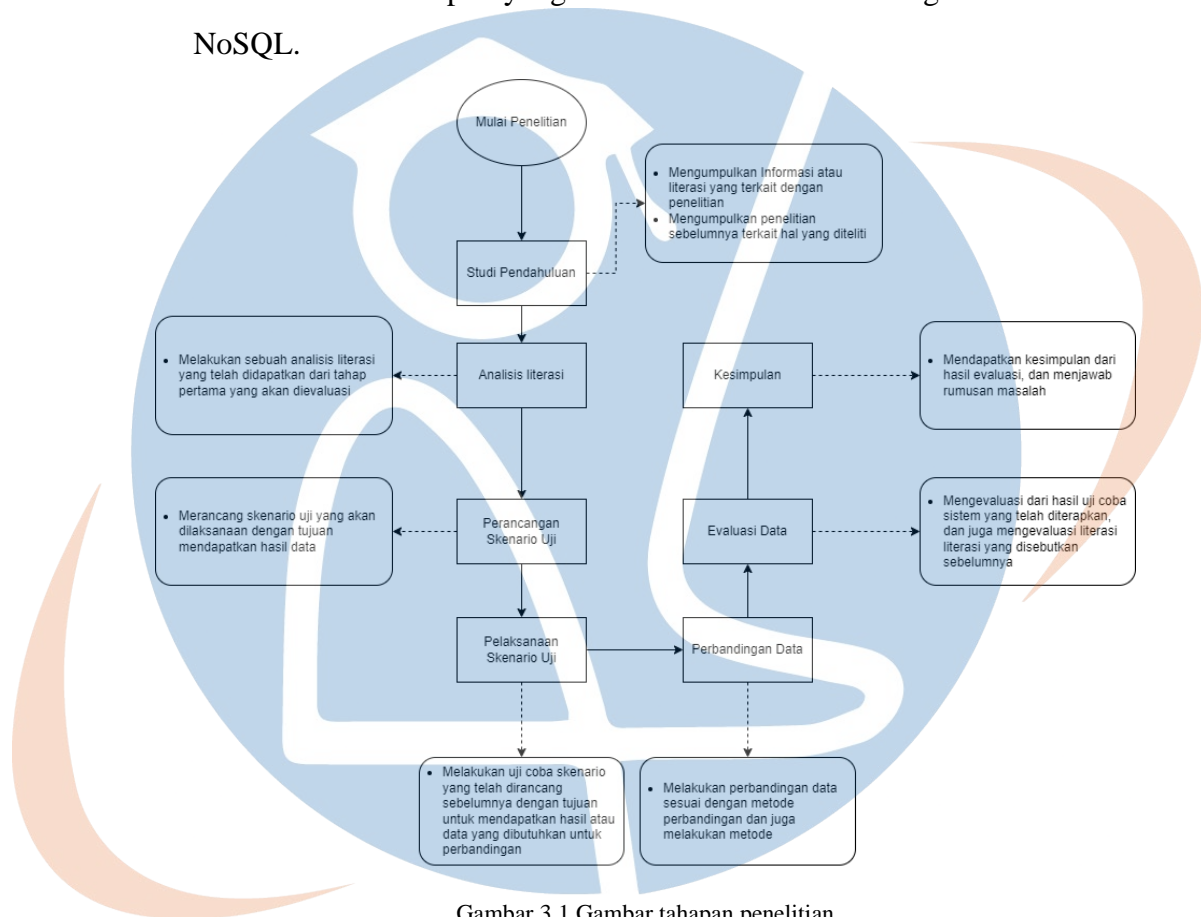
3. Kharisma Syahputri, melakukan penelitian dengan judul Peran Database Dalam Sistem Informasi. Tujuan ini menentukan peran pengaplikasian basis data dalam sistem informasi yang dipakai keseharian dapat memberikan dampak pada aplikasi sistem informasi tersebut. Yang mendapatkan hasil yaitu Sistem Informasi Manajemen (SIM) mendapatkan penambahan kualitas yang sangat meningkat dibandingkan sebelumnya setelah pengaplikasian basis data. Hal yang mengefek kualitas tersebut adalah peningkatan akurasi, waktu, dan juga reabilitas dalam pengambilan data. Basis data sendiri yang menyediakan integirtas data yang memberikan keamanan dalam data dari kerusakan dan akses yang tidak sah, fleksibilitas dalam akses informasi, dan juga memungkinkan kolaborasi atau penggunaan secara bersama dalam aplikasi tersebut. [8]
4. Figueiredo M.S.N, dan Pereira, A.M. melakukan sebuah penelitian dengan judul *Managing knowledge – the importance of databases in the scientific production*. Tujuan dari penelitian ini adalah untuk mendapatkan review dalam tema manajemen perilmuan dan keterlibatan basis data dalam produksi bidang ilmu tersebut. Dengan kesimpulan bahwa basis data yang berhubungan dengan ilmu sangat berdampak pada penggunaan manajemen ilmu seperti basis data. Manajemen tersebut dapat memiliki dampak positif yaitu berupa optimisasinya produktifitas tersebut. [1]

BAB 3

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Berikut adalah tahapan yang akan dilakukan untuk menganalisis basis data NoSQL.



Gambar 3.1 Gambar tahapan penelitian

STT - NF

3.2 Rancangan Penelitian

Dalam penelitian menggunakan pendekatan penelitian perbandingan. Penelitian perbandingan ini dilakukan dengan cara membandingkan dua basis data yang telah disebutkan sebelumnya. Dalam bagian ini ditambahkan kerangka penelitian yang menjelaskan rancangan penelitian dari latar belakang sampai kesimpulan

3.2.1 Jenis Penelitian

Dalam penelitian ini akan dilakukan penelitian komparatif dengan pendekatan kuantitatif. Menurut para ahli yaitu Sugiyono pada tahun 2017, metode komparatif adalah penelitian yang bermaksud membandingkan nilai satu atau lebih variabel mandiri pada dua atau lebih populasi, sampel atau waktu yang berbeda atau gabungan semuanya. Penelitian komparatif tingkat kesulitannya lebih tinggi daripada deskriptif. Atau rumusan komparatif adalah menyediakan suatu variable dan sample yang berbeda yang akan di bandingkan kedua variabel tersebut. [15]

Alasan digunakannya penelitian komparatif dengan pendekatan kuantitatif tersebut adalah data yang didapatkan berupa angka dan dilakukan komparatif untuk mendapatkan kesimpulan dari penelitian tersebut.

3.2.2 Metode Analisis Data

Analisis yang dilakukan adalah analisis kuantitatif dan analisis dokumen. Alasan digunakannya analisa berikut dikarenakan data yang didapatkan berupa waktu atau angka. Analisis dokumen juga digunakan yaitu menggunakan buku atau dokumentasi sebagai acuan dari hasil penelitian.

Data yang didapatkan berupa dari uji implementasi pemrograman, dalam hal itu analisis kuantitatif sangat cocok untuk metode analisis dalam penelitian ini. Dokumen juga sangat berpengaruh dalam evaluasi atau

kesimpulannya dengan metode komparatif yaitu membandingkan kedua data yang telah di proses atau olah yang akan dievaluasikan kembali untuk mendapatkan hasil yang ingin didapatkan.

3.2.3 Metode Pengumpulan Data

Metode pengumpulan data yang dilakukan adalah berikut

A. Studi literatur

Dalam pengumpulan data dilakukan sebuah analisis dokumen atau *study literature* untuk mendapatkan kesimpulan yang telah dilakukan peneliti sebelumnya. Menurut M. Nazir sendiri, studi literasi atau studi perpustakaan adalah buku, literatur, catatan dan bahkan jurnal yang telah dikumpulkan dan ditelaah kembali dengan tujuan untuk menghubungkan masalah yang dipecahkan. Studi literasi ini juga berguna untuk mencari kekurangan atau kelebihan yang akan di analisis dan evaluasi pada tahap akhir yang dimana akan dibandingkan dari kedua hasil literasi tersebut. Studi literatur ini juga bertujuan untuk mendapatkan teori-teori yang relevan dengan permasalahan yang sedang diteliti.

B. Dokumentasi

Menurut Sugiyono dokumentasi adalah suatu cara yang digunakan untuk mendapatkan data dan informasi dalam berupa buku, arsip, dokumen, tulisan angka, dan gambar dengan keterangan yang dapat mendukung penelitian. Dokumentasi yang dimaksud untuk penelitian ini yaitu mengumpulkan data hasil uji coba dan diurutkan dengan keterangan masing masing yang dimana data tersebut akan di evaluasi dan dibandingkan.

3.2.4 Metode Pengujian

Dalam metode pengujian, dilakukan adalah *statistical testing*. Menurut pendapat dari Sugiyono, menyatakan bahwa teknik analisis data merupakan langkah - langkah untuk mendapatkan hasil data yang diinginkan, mengolah dengan cara sistematis data yang didapatkan dari hasil wawancara, survey lapangan, dan dokumentasi, dengan cara mengelompokkan data ke dalam kategori tertentu, dan mengelompokkan ke unit-unit yang sudah di sesuaikan. Metode analisis statistik data bisa dimanfaatkan, dipilih dan disesuaikan dengan tujuan penelitian tersebut.

Statistical testing ini sangat berguna dalam perbandingan dua basis data yang akan di uji coba dalam metode analisis yang telah disebutkan. Perbandingan dilakukan ketika mendapatkan rata rata dalam data tersebut, seperti contoh membandingkan waktu rata rata pada fungsi *read* MongoDB dengan CouchDB. [14]

Rumus yang akan digunakan untuk mendapatkan rata rata dari hasil data tersebut adalah

$$\frac{\text{Waktu respon yang didapatkan}}{\text{Jumlah uji coba yang dilakukan}} = \text{Waktu rata rata yang dibutuhkan}$$

3.2.5 Metode Implementasi dan Evaluasi

Dalam penerapan metode implementasi akan dilakukan sebuah implementasian kode dari yang telah dirancang. Dalam penerapan ini kode akan ditulis dalam bahasa JavaScript dengan environment Node.JS. Evaluasi akan dilakukan sesuai dengan metode kuantitatif komparatif yang membandingkan dua data yang telah disediakan atau disiapkan.

3.2.6 Lingkungan Pengembangan

Penelitian akan dilakukan pada laptop localhost sebagai hardware. Adapun alat dan software yang akan digunakan pada penelitian ini yaitu :

1. Laptop HP

- a. CPU : AMD Ryzen 3 Mobile 5300U
- b. RAM : 8 GB
- c. OS : Windows 11 64-bit

2. Postman

Postman merupakan aplikasi yang dapat digunakan dalam pembuatan atau pengembangan API.

3. Node.JS

Sebagai environment yang akan digunakan untuk menjalankan API menjalankan *Create, Read, Update, dan Delete*.

Pengembangan dan pengaplikasian akan dilakukan secara lokal atau menggunakan hardware sendiri. Pilihan ini dilakukan karena mengurangi error terhadap internet atau error terhadap hosting jika menggunakan suatu provider atau service hosting secara online.



STT - NF

BAB 4 IMPLEMENTASI DAN EVALUASI

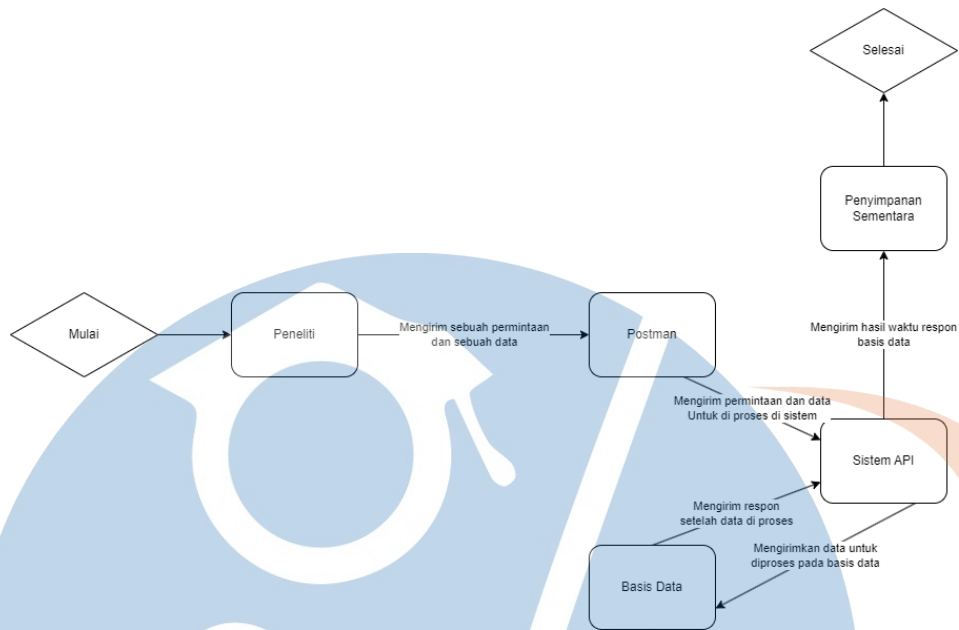
4.1 Rancangan Penelitian

Dalam penelitian ini, data yang diambil adalah berupa waktu respon dari database ke API atau klien. Diawali dengan merancang suatu sistem atau API untuk menerima permintaan user ketika user meminta atau mengeksekusi API tersebut. Sistem aplikasi menerima permintaan seperti *insert*, *update*, *insert*, dan *delete*. Yang dimana peneliti akan mengirim permintaan dan juga jumlah data yang ingin dieksekusi oleh sistem aplikasi tersebut.

Sistem aplikasi akan mengirim sebuah *query* yang akan di eksekusi oleh basis data. Dimana basis data tersebut menjalankan *query* yang telah diterima, hasil dari sistem basis data tersebut akan dikirimkan ke sistem aplikasi dan sistem aplikasi mengirim waktu respon yakni waktu diantara ketika peneliti mengirim sebuah API atau perintah dan ketika peneliti menerima respon dari sistem aplikasi tersebut.

Hasil yang telah dikumpulkan melalui percobaan secara berkala tersebut akan dikumpulkan dan dianalisis sesuai dengan metode perbandingan yang telah ditentukan sebelumnya. Maka dari ini tidak terdapat sebuah prototipe atau foto terkait sistem aplikasi dikarenakan sistem aplikasi tersebut menggunakan kode dan aplikasi yang digunakan untuk meminta API atau menguji API adalah aplikasi Postman.

STT - NF



Gambar 4.1 Skema Alur penelitian

Dalam penelitian juga perlu disiapkan beberapa hal sebelum melakukan seperti contoh dalam pendapatan waktu respon *insert* perlu disiapkan sebuah url dalam Postman dan juga sebuah file berisi data *dummy*

Tabel 4.1 Tabel Persiapan Untuk Melakukan Alur Penelitian

Fungsi	URL	File
<i>Insert</i>	http://127.0.0.1:3000/{basis_data}/insert	File <i>dummy</i> untuk di <i>insert</i>
<i>Update</i>	http://127.0.0.1:3000/{basis_data}/update	Data yang ingin di <i>update</i>
<i>Select</i>	http://127.0.0.1:3000/{basis_data}/select	-
<i>Delete</i>	http://127.0.0.1:3000/{basis_data}/delete	-

4.2 Hasil Penelitian

Dalam Bab ini akan membahas hasil penelitian analisis CouchDB dan MongoDB yang telah dilakukan dalam environment Node.JS. Hasil pengujian yang telah dilakukan

4.2.1 Struktur Desain Sistem Basis Data

Struktur desain sistem basis data yang akan digunakan dalam analisa dua basis data tersebut adalah sebagai berikut:

```
{
  userID: Integer unique,
  nama: String,
  kategori: Integer,
  updates: String
}
```

Dalam skema tersebut userID akan digunakan sebagai ID dengan data *mockup* yang telah disiapkan. Skema tersebut digunakan untuk skema dummy MongoDB dan juga CouchDB. Dalam pengaplikasian skema, digunakan kode untuk memasukan data ke basis data sesuai dengan desain yang telah ditentukan.

Dalam pembentukan skema MongoDB, digunakan kode

```
const dataSchema = new mongoose.Schema({
  id: Schema.Types.UUID,
  nama: String,
  kategori: Number,
  userId: String,
  updates: String
})
```

Kode skema CouchDB

```
couchdb.insert({
  nama: e.name,
  kategori: e.category,
  userId: e.id,
  updates: ""
}, e.id)
```

Yang membedakan CouchDB dengan MongoDB skema. Dalam MongoDB skema harus dibuat terlebih dahulu sebelum memasukan basis data, skema tersebut sebagai *template* data yang akan dimasukan ke tabel tersebut.

4.2.2 Implementasi Kode Dalam Analisis

Dalam implementasi kode, digunakan Node.JS sebagai environment untuk menjalankan kode tersebut. Dalam mendapatkan waktu respon API, digunakan *response-time package* dari *Node.JS Package Manager* (NPM). Kode sendiri yang akan digunakan untuk mendapatkan hasil respon tersebut adalah.

Digunakan untuk mendapatkan waktu respon pada app tersebut.

```
app.use(responseTime())
```



```
app.get("/mongo/insert", async function(req, res) {
  var data = []
  req.body.forEach(e => {
    data.push({
      nama: e.name,
      kategori: e.category,
      userId: e.id,
      updates: ""
    })
  });
  await dataModel.insertMany(data).then(() => {
    console.log("Data inserted")
  }).catch((err) => {
    console.log(err)
  })
  res.send(req.body)
})

app.get("/mongo/update", async function(req, res) {
  for (const data of req.body) {
    const respond = await
dataModel.updateMany({kategori:data.category},{updates:dat
a.updates}).then(()=>{
    console.log("Update done category : " +
data.category)
  }).catch((err)=>{
    console.log(err)
  })
    console.log(respond)
  }
  res.send(req.body)
})
```

```

app.get("/mongo/select", async function (req, res) {
  dataModel.find().select().then((data) => {
    res.send(data)
  })
})

app.get("/mongo/delete", async function(req, res){
  dataModel.deleteMany({}).then((data) =>{
    res.send(data)
  })
})

```

Di implementasikan untuk API *Insert, Select, Update, dan Delete* MongoDB.

```

app.get("/couch/insert", async function(req, res){
  req.body.forEach( e => {
    couchdb.insert({
      nama: e.name,
      kategori: e.category,
      userId: e.id,
      updates: ""
    },
    e.id)
  })
  res.send("done")
})

```

STT - NF

```
app.get("/couch/update", async function(req, res){
  const update = req.body
  const doclist = await couchdb.list({include_docs: true})
  doclist.rows.forEach(async (doc) => {
    update.forEach(async (data) => {
      if (data.category == doc.doc.kategori){
        await couchdb.insert({
          _id: doc._id,
          _rev: doc._rev,
          nama: doc.doc.nama,
          kategori: doc.doc.kategori,
          userId: doc.doc.userId,
          updates: "updated"
        }).then((e) => {
          console.log("updated")
        })
        return
      }
    })
  })
  res.send("done")
})

app.get("/couch/select", async function(req, res){
  const doclist = await couchdb.list().then((body) => {
    body.rows.forEach((doc) => {
      console.log(doc);
    })
  });
  res.send("done")
})

app.get("/couch/delete", async function(req, res) {
  const doclist = await couchdb.list().then((body) => {
    body.rows.forEach((doc) => {
      couchdb.destroy(doc.id, doc.value.rev)
    })
  });
  res.send("done")
})
```

Dalam CouchDB, tidak memiliki fungsi *update* tetapi mereka menggunakan fungsi *insert* untuk memasukan data yang baru ke tabel tersebut.

4.2.3 Data Hasil Implementasi

Dari hasil implementasi kode yang telah diterangkan diatas mendapatkan hasil berupa tabel dibawah. Hasil tersebut didapatkan dari menggunakan API yang telah di program menjalankan *insert, update, select* dan *delete*. Dan data *dummy* yang dipakai adalah frekuensi kelipatan 50 yaitu 50, 100, 150, 200, dan 250. Setelah itu dipakai frekuensi kelipatan 250 yaitu 250, 500, 750, 1000, dilakukan juga sebuah pengulangan siklus dengan urut *insert, update, select* dan *delete* dengan siklus 5 kali yang dimana hasil tersebut di proses menjadi sebuah rata rata dari hasil semua percobaan.

Tabel 4.2 Rata rata waktu respon pada MongoDB

Fungsi	Data rata rata MongoDB dalam ms							
	50	100	150	200	250	500	750	1000
<i>Insert</i>	48,2682	72,8502	109,815	84,9598	135,9956	153,585	216,1882	253,4568
<i>Update</i>	52,8956	78,6904	122,8132	142,876	204,5956	295,6096	411,574	585,9716
<i>Select</i>	69,0634	67,9466	72,0004	81,4602	87,7626	95,7498	102,1736	166,073
<i>Delete</i>	12,1716	8,7814	9,4056	19,352	18,8768	28,258	22,7154	42,0874

Dalam data yang ditampilkan, terlihat ada penambahan stabil setiap ditambahkan data walaupun ada ketidak stabilan waktu rata rata seperti pada data 150 menuju 200 yaitu berkurang tetapi terjadi peningkatan stabil setiap penjumlahan data.

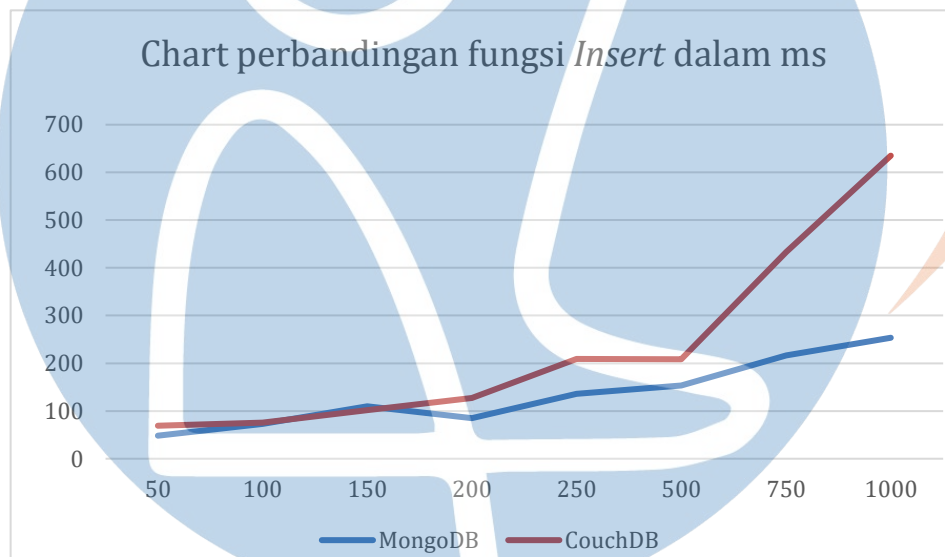
Tabel 4.3 Rata rata waktu respon pada CouchDB

Fungsi	Data rata rata CouchDB dalam ms							
	50	100	150	200	250	500	750	1000
<i>Insert</i>	69,1464	75,6728	101,4412	127,2742	209,1602	208,551	433,1946	635,1086
<i>Update</i>	71,4754	119,3084	140,5712	124,9522	187,299	466,552	490,8618	865,9378
<i>Select</i>	212,766	381,4492	602,985	551,3026	1017,254	2347,339	2896,594	5023,78
<i>Delete</i>	113,4568	161,3818	192,0652	206,7798	352,0138	706,4534	886,842	1348,706

Dalam uji coba basis data CouchDB. Terjadi peningkatan waktu respon yang tetap setiap peningkatan data. Kekurangan dari kode atau dalam pengekseskuan tersebut adalah CouchDB tidak memiliki fungsi *update*. Hanya dapat melakukan *insert* ke basis data dengan ID yang telah ada sebagai *parent ID*. Hal tersebut tidak dapat membuat menghapuskan data sebelumnya dan diganti ke data yang baru.

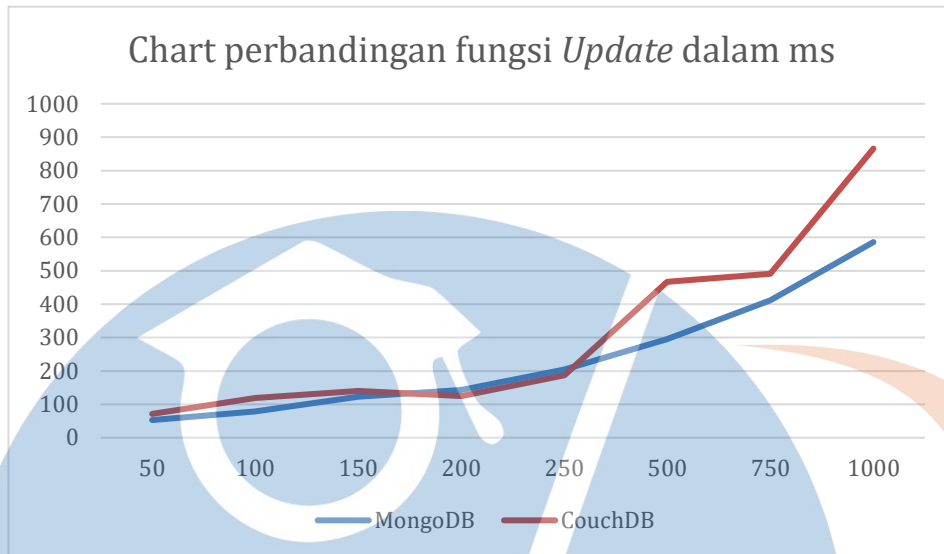
4.2.4 Analisis Data Implementasi

Bagan 4.1 Perbandingan fungsi *insert* dalam ms



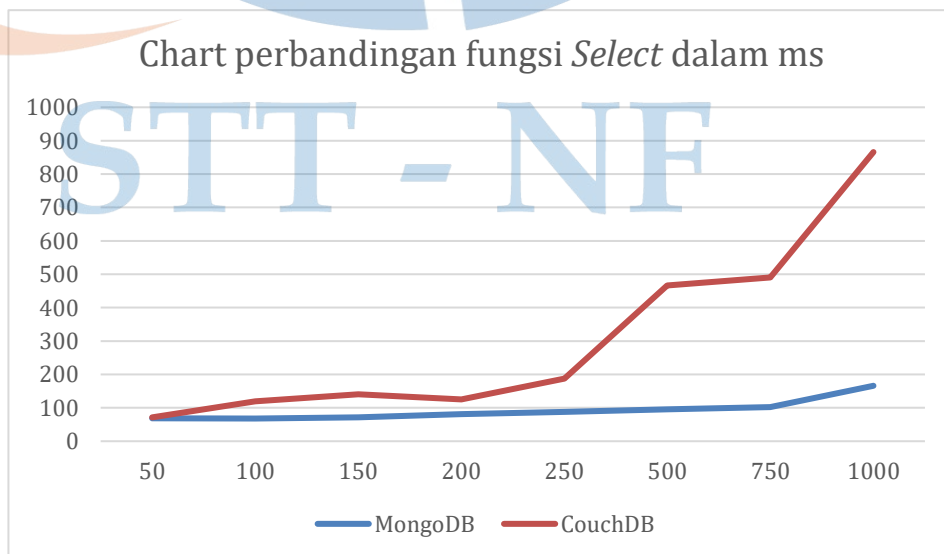
Dalam bagan dijelaskan bahwa terjadi penambahan waktu secara eksponen terhadap MongoDB akan tetapi terjadi peningkatan drastis pada basis data CouchDB.

Bagan 4.2 Perbandingan fungsi *update* dalam ms

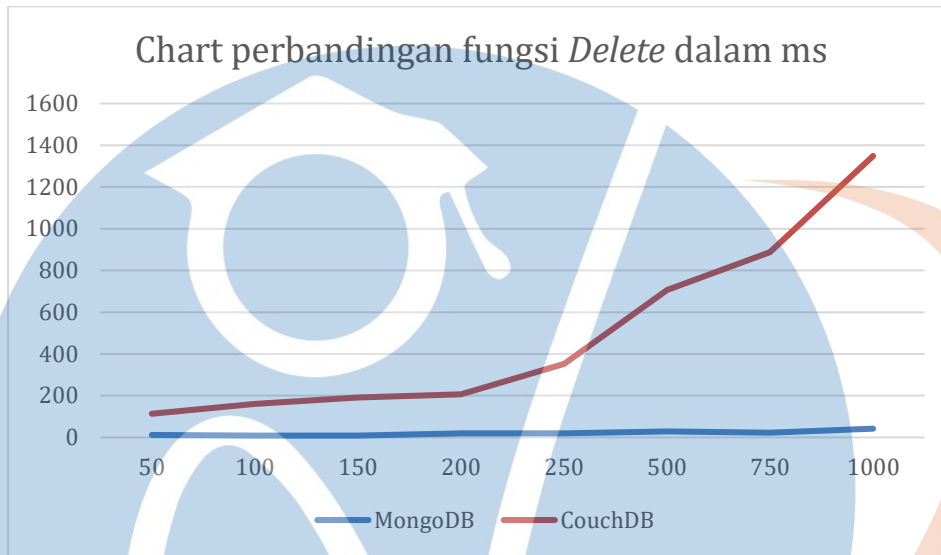


Dari *sample* 50 sampai 250 memiliki kesetaraan dalam waktu, akan tetapi terdapat anomali dari 500 sampai 700 pada CouchDB. Perlu disebutkan bahwa CouchDB tidak memiliki fungsi bawaan *update* melainkan menambahkan data yang baru dengan relasi. Namun dikarenakan CouchDB tidak memiliki fungsi *update*, mereka hanya menambahkan data dengan relasi ID sehingga data menjadi dua kali lebih banyak yang dapat menyebabkan bertambahnya waktu dalam *select* dan *delete*

Bagan 4.3 Perbandingan fungsi *select* dalam ms



Bagan 4.4 Perbandingan fungsi *delete* dalam ms



Evaluasi dari hasil penelitian membuktikan bahwa adanya perbandingan dalam fungsi *insert* dan *update*. Sedangkan dalam basis data CouchDB tidak memiliki fungsi *Update* yang menyebabkan data yang dimasukan tidak diubah melainkan hanya memasukan data baru. Hal ini menyebabkan data yang telah ada tidak dihapus melainkan perlu dihapus secara manual kembali. Ketidak adanya fungsi *update* mendampakkan fungsi lain seperti *select* dan juga *delete*, hal ini juga membuktikan bahwa fitur *multi-cluster* milik CouchDB tidak berjalan sebaik walaupun banyak data sekaligus.

4.3 Evaluasi Hasil Penelitian

Dari hasil penelitian yang telah dilakukan, terdapat kelebihan unggul oleh MongoDB dalam waktu respon. Terlihat bahwa MongoDB memiliki proses lebih cepat dibandingkan CouchDB. Walaupun dalam fungsi *Update* memiliki kesetaraan atau kemiripan data MongoDB, terjadi kenaikan dalam waktu yang dibutuhkan CouchDB untuk melakukan sebuah operasi tersebut.

Akan tetapi dikarenakan waktu pengujian keadaan *device* belum tentu stabil seperti adanya aplikasi atau sistem yang berjalan di latar belakang yang berdampak pada hasil waktu respon memiliki ketidakstabilan.



STT - NF

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dalam penelitian ini, ditemukan kesimpulan dari data data yang telah didapatkan dan juga dari literatur yang telah digunakan sebagai referensi:

1. Untuk membandingkan performa, digunakan sebuah modul bernama *response-time* untuk mendapatkan waktu respon yang dibutuhkan suatu fungsi atau kueri. Dengan menggunakan sample 1000 mendapatkan hasil rata-rata MongoDB *Insert* 253ms, *Update* 586ms, *Select* 166ms, dan *Delete* 42ms. Sedangkan hasil rata-rata CouchDB *Insert* 635ms, *Update* 865ms, *Select* 5023ms, dan *Delete* 1348ms dengan keterangan bahwa setelah *update* data CouchDB menjadi dua dikarenakan fungsi *update* tersebut.
2. Dalam segi performa, lebih cepat MongoDB dibanding CouchDB, dikarenakan waktu yang dibutuhkan MongoDB dalam memproses data data tersebut jauh lebih cepat dibanding CouchDB yang semakin banyak data maka semakin lambat waktu responnya.

Kesimpulan yang didapatkan adalah, MongoDB memiliki keunggulan dibanding CouchDB, MongoDB juga memiliki produk *cloud* yaitu MongoDB Atlas. Walaupun CouchDB juga memiliki fitur CouchDB Atlas fitur tersebut masih berupa *localhost* atau masih hanya dapat di akses oleh komputer tersebut. MongoDB Atlas juga sangat mudah untuk di konfigurasi dibandingkan CouchDB yang perlu dihosting atau mengganti *bind ip* sebelum dapat digunakan oleh user lain atau dipasang dalam server.

5.2 Saran

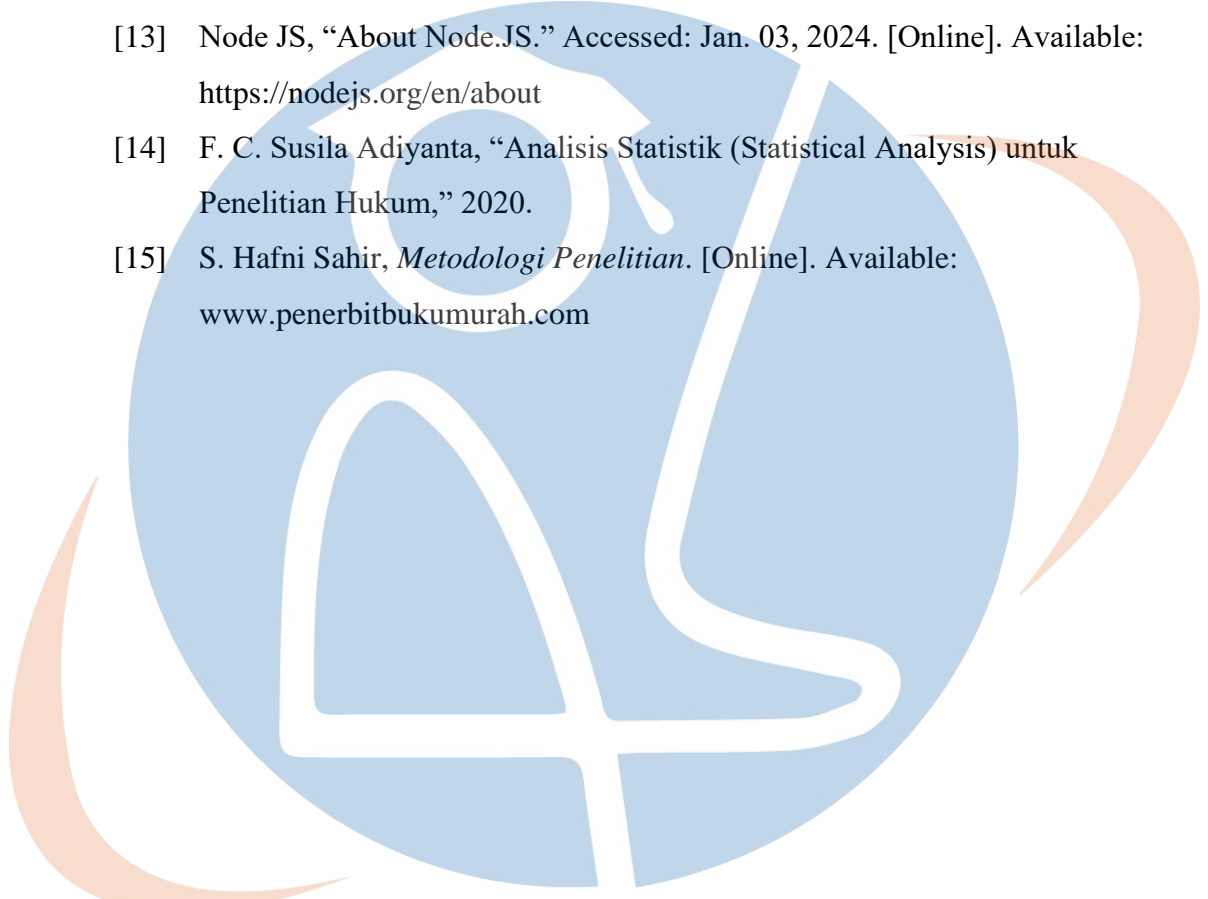
Penelitian ini dapat digunakan sebagai referensi dalam penelitian atau uji coba dalam pengembangan lainnya. Namun ada beberapa saran untuk pembaca dalam penyempurnaan penelitian perbandingan CouchDB dan MongoDB tersebut antara lain :

1. Metode yang digunakan dapat di kembangkan selanjutnya, dan juga penelitian selanjutnya tidak hanya dilakukan dalam satu network atau localhost melainkan menggunakan network lainnya seperti hosting atau *VPS (Virtual Private Server)*. Perlu diperhatikan juga dalam pengujian di VPS, faktor seperti bandwidth atau kecepatan internet dapat mempengaruhi waktu respon yang didapatkan
2. Untuk penelitian selanjutnya diharapkan mencari sumber referensi literasi atau dokumentasi bahkan sumber informasi dari suatu *informant* atau narasumber lainnya, seperti informasi mengenai basis data masing masing terkait performa atau fitur yang dimiliki masing masing basis data.
3. Diluapkan waktu yang lebih banyak untuk analisis atau pengumpulan literasi, dikarenakan waktu dalam penelitian ini sangat pendek oleh karena itu literasi dan informasi yang didapatkan sangat minim.

STT - NF

DAFTAR PUSTAKA

- [1] M. S. N. Figueiredo and A. M. Pereira, "Managing Knowledge – The Importance of Databases in the Scientific Production," *Procedia Manuf*, vol. 12, pp. 166–173, 2017, doi: 10.1016/j.promfg.2017.08.021.
- [2] K. Syahputri, M. Irwan, and P. Nasution, "Peran Database Dalam Sistem Informasi Manajemen," *Jurnal Akuntansi Keuangan dan Bisnis*, vol. 1, no. 2, pp. 54–58, 2023, [Online]. Available: <https://jurnal.ittc.web.id/index.php/jakbs/index>
- [3] A. Meier and M. Kaufmann, "SQL & NoSQL Databases Models, Languages, Consistency Options and Architectures for Big Data Management."
- [4] A. Aan, S. Daroini, and W. Yustanti, "PERBANDINGAN PENGGUNAAN NOSQL MONGODB DAN MYSQL PADA BASIS DATA FORUM KOMUNIKASI," 2016.
- [5] G. Vaish, "Getting Started with NoSQL Your guide to the world and technology of NoSQL," 2013. [Online]. Available: www.it-ebooks.info
- [6] CouchDB, "CouchDB About." Accessed: Jan. 30, 2024. [Online]. Available: <https://couchdb.apache.org/#about>
- [7] Oracle, "'What Is A Database?'" Accessed: Jan. 25, 2024. [Online]. Available: <https://www.oracle.com/id/database/what-is-database>
- [8] K. Syahputri, M. Irwan, and P. Nasution, "Peran Database Dalam Sistem Informasi Manajemen," *Jurnal Akuntansi Keuangan dan Bisnis*, vol. 1, no. 2, pp. 54–58, 2023, [Online]. Available: <https://jurnal.ittc.web.id/index.php/jakbs/index>
- [9] [4] R. Elmasri and S. Navathe, *Fundamentals of Database Systems*. Addison-Wesley, 2011.
- [10] N. Datt Bhardwaj, "Comparative Study of CouchDB and MongoDB- NoSQL Document Oriented Databases," 2016. [Online]. Available: www.ijcaonline.org

- 
- [11] P. K. Ghavami, *Big data governance : modern data management principles for Hadoop, NoSQL & big data analytics.*
- [12] S. Bradshaw, E. Brazil, and K. C. Mongoddb, “The Definitive Guide Powerful and Scalable Data Storage MongoDB The Definitive Guide Powerful and Scalable Data Storage.”
- [13] Node JS, “About Node.JS.” Accessed: Jan. 03, 2024. [Online]. Available: <https://nodejs.org/en/about>
- [14] F. C. Susila Adiyanta, “Analisis Statistik (Statistical Analysis) untuk Penelitian Hukum,” 2020.
- [15] S. Hafni Sahir, *Metodologi Penelitian.* [Online]. Available: www.penerbitbukumurah.com

STT - NF