

Perbandingan Kinerja *Local Database* Pada Aplikasi *Mobile* Dengan Flutter

Firman Ali Eka Atmojo^{a1}, I Made Widiartha^{a2}

^{a1}Informatics Department, Udayana University
South Kuta, Badung, Bali, Indonesia, 80361

¹firmanali22@gmail.com

²madewidiartha@unud.ac.id

Abstract

Android is currently a popular mobile operating system used all over the world. There are many programming languages that can be used to build android applications, one of which is Flutter. Flutter is a Software Development Kit that supports multi platform apps with the Dart programming language developed by Google. Two database management Android supported systems using Flutter are sqflite and Hive. Flutter SDK provides installable packages to install for developers and implements an application that stores its data in this local database. In this study, it is necessary to reveal a comparison of the performance of these two databases. For this reason, an Android application to be used as a test that includes basic data operations that are widely used in this database is implemented in this study. The test results clearly show that in each test with a drastically increased number of data samples Hive provides better performance than sqflite for each type of basic operation.

Keywords: *Android, Flutter, database, sqflite, Hive, comparison, performance*

1. Pendahuluan

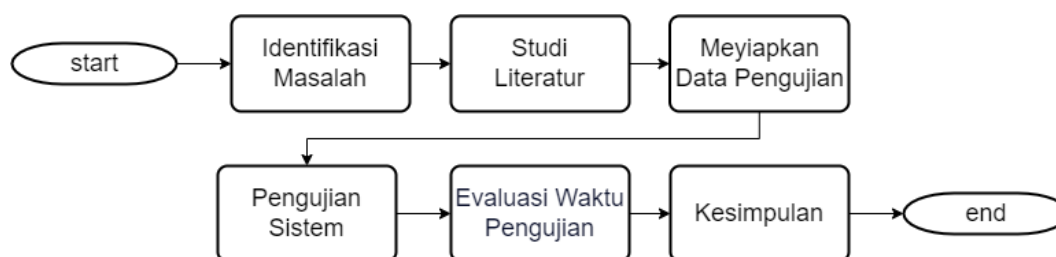
Perkembangan aplikasi seluler saat ini semakin pesat, jutaan aplikasi tersedia di 3 marketplace besar yaitu App Store Apple, Play Store Google dan Microsoft Store [8]. Aplikasi seluler adalah aplikasi perangkat lunak yang dirancang untuk berjalan di perangkat seluler seperti smartphone, tablet, atau jam pintar [15]. Saat ini, ada dua metode pengembangan untuk membuat aplikasi seluler yaitu *native* dan *multi platform*. Flutter adalah SDK (*Software Development Kit*) yang mendukung aplikasi *multi platform* dengan bahasa pemrograman Dart yang dikembangkan oleh Google dengan berbagai dukungan ke perangkat seperti android, ios, web, windows, linux, mac dan perangkat tertanam [7]. Akses mudah ke dan pengelolaan data merupakan hal penting untuk mendukung fitur di banyak aplikasi seluler [10]. *Local database* banyak digunakan pengembang untuk mengelola data yang responsif dan handal pada aplikasi mereka. Terdapat dua jenis basis data yang banyak digunakan untuk menyimpan data yaitu basis data relasional dan basis data non-relasional. Hive merupakan salah satu basis data non-relasional, Hive sangat berkinerja karena memiliki sedikit overhead dibandingkan dengan basis data relasional [4]. SQL merupakan salah satu basis data relasional. SQLite sering digunakan sebagai format dokumen pada disk aplikasi seperti sistem kontrol versi, alat analisis keuangan, katalog media dan suite pengeditan, program penyimpanan catatan, dan sebagainya [5].

Terdapat beberapa penelitian yang membandingkan performa antara kedua basis data tersebut diantaranya yaitu, Erwin [1] dengan membandingkan kinerja *database* NoSQL dan *database* lainnya SQL untuk pemesanan tiket online, pengujian dilakukan dengan mengirim perintah *insert*, *delete* dan *select* menggunakan 100, 500, 1000 dan 5000 *record* data yang masing masing diuji sebanyak 4 kali dan diambil nilai rata-rata. Hasil penelitian ini *database* NoSQL lebih cepat dalam menulis data dibawah 500, *database* NoSQL juga lebih cepat dalam menghapus data dari *database* SQL dan untuk membaca *database* NoSQL unggul dalam membaca data dengan 1 tabel dari *database* SQL. Pada penelitian Abdullah [9] yang membandingkan dua jenis *database* pada android yaitu sqflite dan firebase, penelitian dilakukan dengan mengevaluasi beberapa eksperimen yang mencakup operasi data yang paling banyak digunakan pada *database*. Hasil penelitian ini menunjukkan bahwa SQLite memberikan kinerja yang lebih baik dibandingkan dengan Firebase kecuali menghapus data. Dan pada penelitian Rakhmawati [13] membandingkan basis data pada Aplikasi Egovbench, aplikasi yang

memantau situs web atau media sosial pemerintah daerah Indonesia setiap hari. Penelitian ini menguji perbandingan antara basis data relasional dan basis data non-relasional berdasarkan metrik yang dipilih. Hasil perbandingan menunjukkan bahwa basis data MySQL memiliki keunggulan kueri yang kompleks pengolahan dan penggunaan basis data yang terkecil ruang penyimpanan. Basis data MongoDB memiliki keunggulan rendah volume transfer data. Basis data Couchbase memiliki keuntungan dari pemrosesan kueri yang singkat dan langsung dengan angka yang tinggi.

Dari hasil penelitian-penelitian diatas penelitian ini dilakukan dengan tujuan untuk mengetahui kinerja *local database* SQL dan NoSQL pada aplikasi yang dibangun menggunakan flutter sebagai penyimpanan datanya. Basis data yang digunakan dalam pengujian yaitu sqflite untuk basis data SQL dan Hive untuk basis data NoSQL. Dilakukan beberapa pengujian menggunakan beberapa jumlah sampel data dalam melakukan query database pada aplikasi yang dijalankan di perangkat android dengan unit testing,tujuannya adalah untuk memeriksa kecepatan waktu dalam melakukan operasi basis data seperti *create, read, update* dan *delete*. Operasi dasar ini dipilih dikarenakan dalam konteks fungsi manajemen basis data operasi dasar CRUD merupakan fungsi yang dapat menggambarkan kemampuan database dalam manajemen data didalam media penyimpanan [8]. Pada Bagian kedua akan menjelaskan metode penelitian yang digunakan. Bagian ketiga akan membahas pengujian yang disiapkan, prosedurnya, dan lingkungan pengujiannya serta hasil dari pengujian yang telah dilakukan.

2. Metode Penelitian



Gambar 1. Alur penelitian

Dalam penelitian untuk membandingkan local database antara SQL dengan sqflite dan NoSQL dengan Hive pada aplikasi yang dibangun menggunakan flutter ini dibagi menjadi 6 tahapan seperti yang dapat dilihat pada Gambar 1. Berikut penjelasan dari masing-masing tahapan tersebut:

- a. Identifikasi Masalah
Tahapan awal penelitian untuk mengidentifikasi latar belakang persoalan yang akan diangkat. Tahapan ini akan sangat menentukan tahapan studi literatur, data yang diperlukan, dan metode yang akan digunakan.
- b. Studi Literatur
Tahapan mencari dan mempelajari literatur dari jurnal, penelitian sebelumnya, dan buku yang berkaitan dengan persoalan dan metode yang digunakan.
- c. Menyiapkan Data Pengujian
Untuk membandingkan kinerja local database pada penelitian ini data yang digunakan adalah data *dummy* yang diperoleh secara otomatis dengan sebuah fungsi yang mengacak kumpulan huruf menjadi kumpulan huruf baru dengan jumlah tertentu yang disimpan pada sebuah variabel.
- d. Pengujian Sistem
Tahapan pengujian pada penelitian ini dengan menjalankan beberapa pengujian menggunakan beberapa jumlah sampel data dengan unit testing aplikasi pada perangkat riil ponsel cerdas android.
- e. Evaluasi Waktu Pengujian
Tahapan evaluasi waktu dilakukan untuk mendapatkan perbandingan hasil waktu pengujian dalam menarik dan merumuskan kesimpulan yang akan sangat berperan penting untuk keberlanjutan dari penelitian.
- f. Kesimpulan
Pada tahap terakhir yaitu kesimpulan, berisi tentang hasil keseluruhan analisis data yang didapat dalam penelitian

2.1 Basis Data

Basis data merupakan sekumpulan tabel data yang terkumpul sehingga menghasilkan satu kesatuan sebuah bangunan data. Basis data relasional menyimpan data dalam bentuk struktur tabel terdiri dari baris dan kolom [6]. Basis data relasional handal untuk model data yang konsisten [9]. Terdapat beberapa basis data relasional pada flutter yaitu sqflite, Floor dan Drift. Basis data non-relational atau biasa disebut NoSQL paling cocok untuk digunakan di aplikasi yang melibatkan sejumlah besar data, dan data berupa data yang tidak terstruktur, terstruktur, atau semi-terstruktur [14]. Basis data relasional yang dapat digunakan pada flutter seperti Hive, ObjectBox dan Shared Preferences.

2.2 Flutter

Flutter merupakan sebuah SDK yang digunakan untuk mengembangkan aplikasi multiplatform yang dibuat oleh Google. Flutter dikembangkan untuk membuat aplikasi yang mempunyai performa yang tinggi dari basis kode tunggal. Flutter menggunakan bahasa pemrograman dart sehingga dapat dengan mudah untuk dipelajari. Selain itu, Flutter juga menyediakan kerangka fungsional reaktif, mesin render 2D, widget yang siap untuk digunakan, dan tools yang digunakan untuk membantu dalam melakukan pengembangan aplikasi [11]. Aplikasi populer yang dibagikan di perpustakaan pengembangan Flutter adalah The New York Times, Alibaba, Iklan Google, eBay, Groupon, Tencent, Square, Asisten Google, Baidu, Philips Hue, Groupon [3].

2.3 Dart

Dart merupakan bahasa pemrograman yang dibuat oleh Google. Dart adalah bahasa yang dioptimalkan untuk klien untuk mengembangkan aplikasi cepat di platform apapun, tujuannya adalah untuk menawarkan bahasa pemrograman paling produktif untuk pengembangan multi-platform, dipasangkan dengan platform runtime eksekusi yang fleksibel untuk kerangka kerja aplikasi [2]. Dart merupakan Bahasa pemrograman yang mendukung adanya pendefinisian fungsi di luar kelas atau sering disebut dengan *top level function*. Dalam Dart, kode program utama disimpan di dalam fungsi main() sama halnya seperti C/C++ [12].

2.4 Data Pengujian

Untuk representasi data, model sampel dirancang sebagai propertinya dengan datanya jenis tercantum dalam Tabel 1. Properti "id" mewakili kunci utama model dan itu adalah ditetapkan secara bertahap dari mulai 1 hingga jumlah sampel. Properti "title" mewakili contoh data String yang dihasilkan dengan membuat data dummy yang di generate otomatis dengan membuat fungsi getRandomString memanfaatkan fungsi Random() dari library math pada Dart yang panjangnya 20 karakter dan berisi huruf termasuk angka dihasilkan untuk "title". Properti "note" mewakili contoh data String yang dihasilkan dengan membuat data dummy yang di generate otomatis dengan membuat fungsi getRandomString memanfaatkan fungsi Random() dari library math pada Dart yang panjangnya 50 karakter dan berisi huruf termasuk angka dihasilkan untuk "note". Properti "createdAt" mewakili contoh data String berupa waktu milisecond yang dihasilkan otomatis dengan fungsi DateTime.now().millisecondsSinceEpoch.toString() yang memanfaatkan fungsi pengambilan waktu pada Dart dan dikonversi kedalam tipe data integer seperti pada sample data Gambar 2.

Table 1. Properti data model dan tipe datanya

Properti	Tipe Data
id	int
title	String
note	String
createdAt	int

```
{
  "id": 1,
  "title": "z61czmQgX5yvyuTuQD1i",
  "note": "eQweXAdmwJfDtIODQ1gB9SFRy3Yz7A3rykrNcZz5XPITKTZ9tv",
  "createdAt": 1665967250674
}
```

Gambar 2. Sampel data pengujian

3. Hasil dan Diskusi

3.1 Pengujian Sistem

Pada pengujian dalam penelitian kinerja *local database* ini akan dilakukan pada perangkat android yang sama untuk mengevaluasi beberapa eksperimen pada sqfite dan Hive *database*. Aplikasi yang dikembangkan berupa unit testing yang dijalankan pada perangkat riil ponsel cerdas Xiaomi Redmi Note 10 Pro dengan spesifikasi perangkat tercantum dalam Tabel 2. Penelitian ini menargetkan Android versi terbaru dengan teknologi memori baru agar dapat memberikan informasi yang *up-to-date* terkait perbandingan performa basis data dan sistem operasi. Perangkat yang digunakan untuk pengujian menggunakan Android versi terbaru Android 12.0 dengan kecepatan memori tulis sampai dengan 200 MB/s dan baca sampai dengan 400 MB/s.

Table 2. Spesifikasi Perangkat Android

Android Version	API Level	RAM	Processor	Memory
Android 12.0 (Snow Cone)	31	8 GB	ARM x64 2.3GHz	UFF 2.2 Write up to 200MB/s Read up to 400MB/s

3.2 Tahapan Pengujian

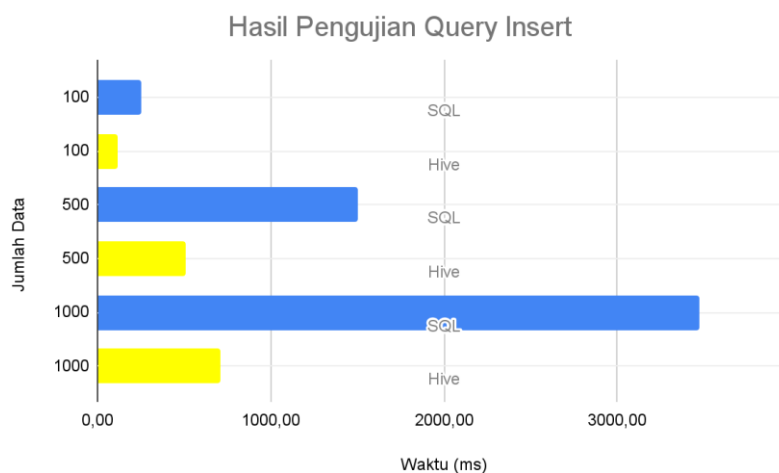
Lima pengujian yang akan dievaluasi pada sqflite dan Hive dirancang untuk membandingkan kinerja basis data dalam hal operasi data yang paling umum create, read, update, dan delete. Operasi dasar ini dipilih dikarenakan dalam konteks fungsi manajemen basis data operasi dasar CRUD merupakan fungsi yang dapat menggambarkan kemampuan database dalam manajemen data didalam media penyimpanan. Adapun tahapan pengujian ini (1) memasukkan data ke dalam *database*, (2) mengambil seluruh data yang tersedia di *database*, (3) memperbaharui data yang ada di *database*, (4) penghapusan data berdasarkan id dari *database* dan (5) penghapusan seluruh data dalam *database*. Daftar kueri yang digunakan untuk mengevaluasi percobaan tercantum dalam Tabel 3. Demi mengevaluasi percobaan, dan Aplikasi Android diimplementasikan menggunakan bahasa pemrograman flutter yang mengevaluasi desain percobaan dan mengungkapkan hasil percobaan dari setiap percobaan pada kedua basis data. Masing-masing dirancang setiap percobaan dan hasil percobaannya dibahas pada sub bagian berikut.

Table 3. List query SQL yang digunakan dalam pengujian

Pengujian	Kueri SQL
Memasukkan data ke dalam <i>database</i>	<i>insert into note values([note], [title])</i>
Mengambil seluruh data yang tersedia di <i>database</i>	<i>select * from note</i>
Memperbaharui data yang ada di <i>database</i>	<i>update tweet set title=[title], note=[note] where id=[id]</i>
Penghapusan data berdasarkan id dari <i>database</i>	<i>delete from note where id=[id]</i>
Penghapusan seluruh data dalam <i>database</i>	<i>delete from note</i>

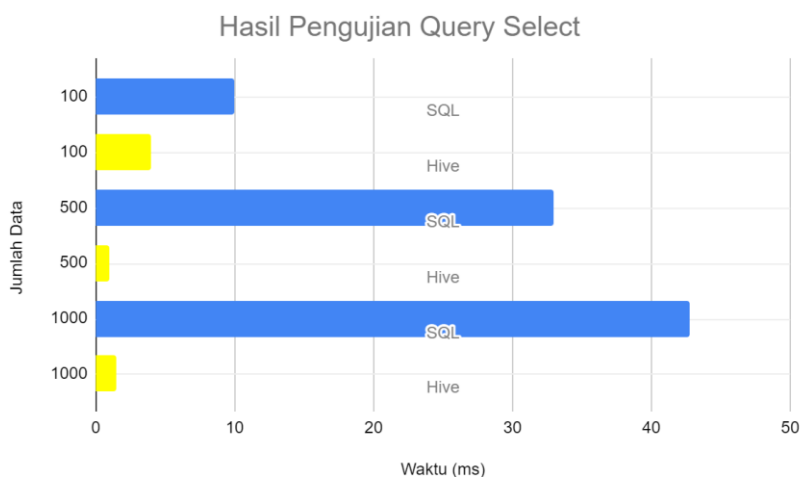
a. Pengujian 1 - Memasukkan Data Ke Dalam Database

Pengujian ini mengevaluasi kinerja basis data dalam hal memasukkan data. Demi perbandingan, jumlah sampel yang berbeda (dari 100 sampel hingga 1000 sampel data) dihasilkan dan dimasukkan ke dalam variabel sebelum pengujian dijalankan kemudian dimasukkan ke dalam setiap basis data, untuk nilai waktu didapatkan dari 4 kali pengujian dan diambil nilai rata-rata sebagai hasil evaluasi disajikan pada Gambar 3. Seperti pada Gambar 3, ketika ukuran data ditingkatkan, kinerja kedua database juga meningkat. Kesimpulan lain yang diperoleh dari hasil eksperimen bahwa Hive memberikan kinerja waktu 2 kali lipat lebih baik dari sqflite. Pada kesimpulan yang didapat dari hasil eksperimen adalah performansi kueri dari kedua basis data meningkat seiring meningkatnya jumlah sampel data, Hive memberikan kinerja yang jauh lebih baik dibandingkan dengan sqflite untuk berbagai ukuran data.



Gambar 3. Hasil pengujian memasukkan data di sqflite dan Hive

b. Pengujian 2 - Mengambil Seluruh Data Yang Tersedia Di Database



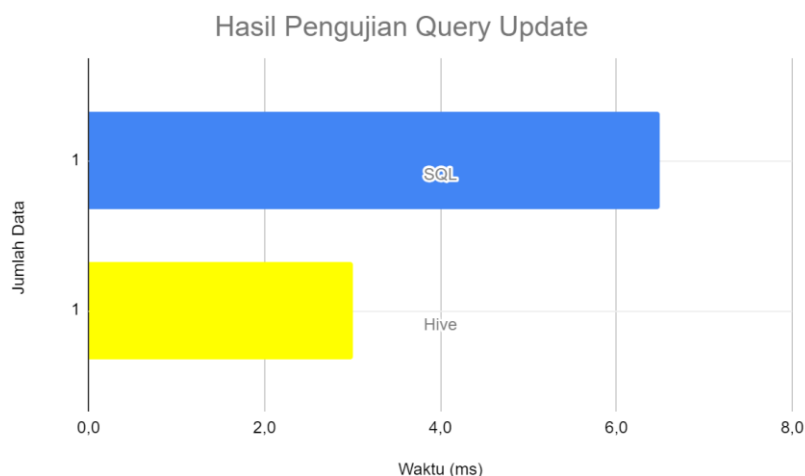
Gambar 4. Hasil pengujian mengambil data di sqflite dan Hive

Pengujian ini mengevaluasi kinerja basis data dalam hal mengambil seluruh data. Sebagai perbandingan, jumlah data yang berbeda (dari 100 sampel hingga 1000 sampel) adalah dihasilkan dan dimasukkan ke dalam variabel sebelum pengujian dijalankan kemudian dimasukkan ke dalam setiap basis data, untuk nilai waktu didapatkan dari 4 kali pengujian dan diambil nilai rata-rata yang disajikan pada Gambar 4. Ketika ukuran data ditingkatkan, kinerja sqflite juga meningkat namun pada Hive memiliki performa waktu yang tinggi pada data dengan jumlah sampel yang banyak. Pada kesimpulan yang didapat dari hasil eksperimen adalah performansi waktu kueri dari kedua basis data berbanding jauh dengan meningkatnya jumlah

sampel data, dan hive memiliki performa yang tinggi karena datanya tersimpan pada *cache memory*.

c. Pengujian 3 - Memperbaharui Data Yang Ada Di Database

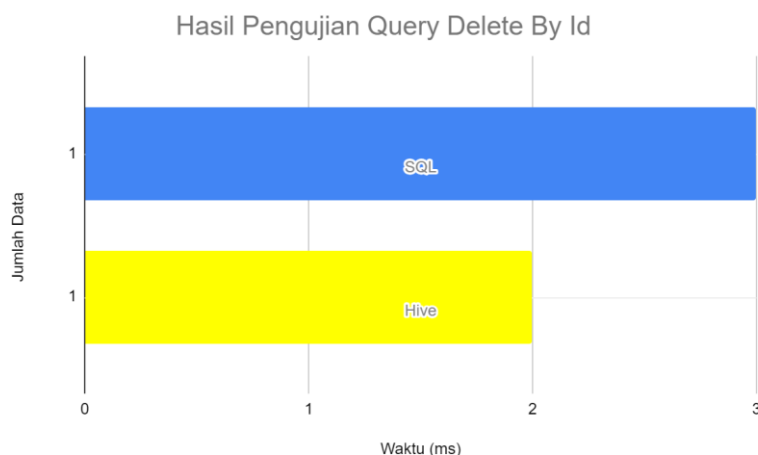
Pengujian ini mengevaluasi kinerja basis data dalam hal memperbaharui data yang sudah ada. Demi mengidentifikasi kinerja kueri seluruh data secara keseluruhan, kueri record dipilih secara khusus untuk tidak dimasukkan ke dalam basis data, untuk nilai waktu didapatkan dari 4 kali pengujian dan diambil nilai rata-rata. Seperti yang dapat dilakukan dari Gambar 5 yang menyajikan hasil pengujian, cukup jelas bahwa Hive memiliki performa waktu yang lebih baik dalam hal menulis data. Hasilnya masuk akal karena pada hive data akan sesegera mungkin ditulis ulang.



Gambar 5. Hasil pengujian memperbaharui data di sqllite dan Hive

d. Pengujian 4 - Penghapusan Data Berdasarkan Id Dari Database

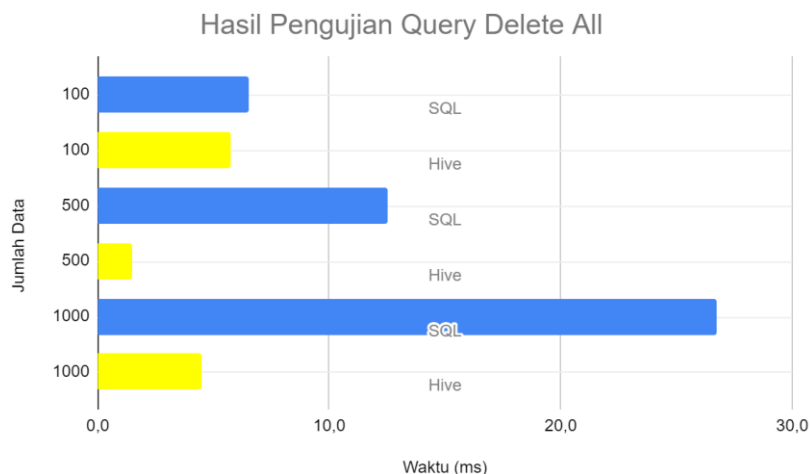
Pengujian ini mengevaluasi kinerja basis data dalam hal menghapus data berdasarkan id. Demi mengidentifikasi kinerja kueri seluruh data secara keseluruhan, kueri sampel dipilih secara khusus untuk tidak dimasukkan ke dalam basis data, untuk nilai waktu didapatkan dari 4 kali pengujian dan diambil nilai rata-rata. Seperti yang dapat dilakukan dari Gambar 6 yang menyajikan hasil pengujian, Waktu yang diperlukan Hive dalam pengujian ini 2 ms yang lebih unggul dibanding sqllite



Gambar 6. Hasil pengujian penghapusan suatu data di sqllite dan Hive

e. Pengujian 5 - Penghapusan Seluruh Data Dalam Database

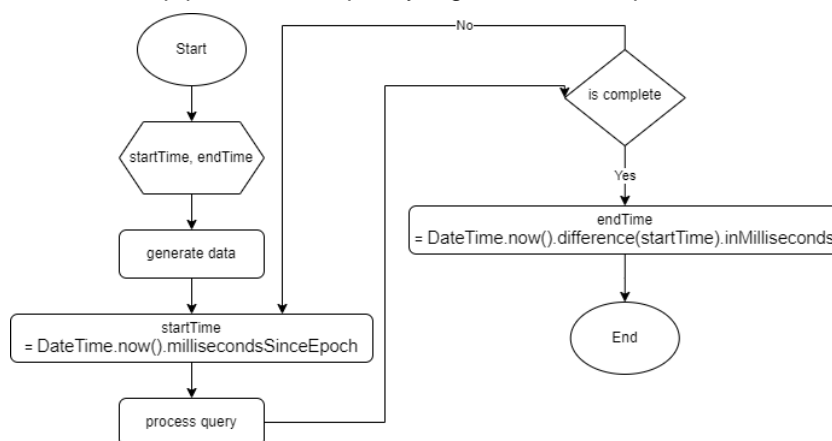
Pengujian ini mengevaluasi kinerja basis data dalam hal menghapus data. Demi perbandingan, jumlah data yang berbeda (dari 100 sampel hingga 1000 sampel) adalah dihasilkan dan dimasukkan ke dalam variabel sebelum pengujian dijalankan kemudian dimasukkan ke dalam setiap basis data, untuk nilai waktu didapatkan dari 4 kali pengujian dan diambil nilai rata-rata yang disajikan. Seperti yang dapat dilakukan dari Gambar 7 yang menyajikan hasil pengujian, Hive memberikan kinerja yang jauh lebih baik dibandingkan dengan sqflite untuk berbagai ukuran data dengan waktu kurang dari 10 ms dan performa waktu meningkat seiring dengan sampel data yang semakin banyak.



Gambar 7. Hasil pengujian penghapusan seluruh data di sqflite dan Hive

3.3 Evaluasi Waktu Pengujian

Waktu yang diambil untuk setiap percobaan dihitung waktu sebelum dengan metode `DateTime.now().millisecondsSinceEpoch`, kemudian menggunakan metode perbandingan waktu antara waktu mulai dengan waktu selesai `DateTime.now().difference(startQueryTime).inMilliseconds` dari bahasa pemrograman Dart untuk menghitung perbedaan waktu dalam milidetik antara sebelum dan kemudian sesudah setiap percobaan seperti yang diilustrasikan pada Gambar 8.



Gambar 8. Flowchart proses perhitungan waktu yang diambil untuk setiap percobaan

Dari hasil pengujian yang sudah dilakukan dan didapatkan nilai waktu setiap pengujian dengan menghitung waktu sebelum dan sesudah pengujian antara basis data SQL dengan sqflite dan NoSQL dengan Hive sudah didapatkan hasil keputusan perbandingan performa waktu antara kedua basis data, Hive yang lebih unggul untuk setiap pengujian dengan performa waktu yang lebih baik dibanding sqflite.

4. Kesimpulan

Flutter secara resmi mendukung *database* sqflite dan Hive dengan menyediakan *library* yang perlu diinstall untuk memfasilitasi pengembang mengimplementasikan aplikasi mereka yang menyimpan datanya di salah satu basis data ini. sqflite sebagai basis data relasional dan Hive sebagai basis data non-relasional didasarkan pada konsistensi yang berbeda model dan menyediakan cara yang berbeda untuk menangani operasi data. Sebagai konsekuensi dari itu, kinerja basis data ini diharapkan akan berbeda. Dengan perbedaan kinerja sqflite dan Hive dalam hal performa waktu untuk menyelesaikan beberapa eksperimen yang mencakup sebagian besar operasi data umum terungkap dalam penelitian ini. Hasil pengujian yang dilakukan dengan menjalankan unit test pada perangkat riil android yang memiliki spesifikasi Android 12 dengan RAM 8 GB dan memori UFF 2.2 yang memiliki kecepatan tulis sampai dengan 200 MB/s dan baca sampai dengan 400 MB/s dengan jelas menunjukkan bahwa pada setiap pengujian dengan jumlah sampel data yang meningkat drastis Hive memberikan kinerja waktu yang lebih baik daripada sqflite untuk setiap jenis operasi dasar.

Referensi

- [1] Apriliyanto, E., Laksmi, C. N., Swardiana Pandu, W. I., & Kusri. "Perbandingan Kinerja Database NoSQL Dengan Database SQL Server Pada Pemesanan Tiket Pesawat Online". *INDONESIAN JOURNAL OF APPLIED INFORMATICS*, vol. 4, no. 2, 2020.
- [2] Dev, D. "Dart overview", 2022. [Online]. Available: <https://dart.dev/overview> [Accessed: 2-10-2022].
- [3] Dev, F. "Flutter Developer Library", 2020. [Online]. Available: <https://flutter.dev/showcase>. [Accessed: 2-10-2022].
- [4] Dev, H. "When to use Hive", 2020. [Online]. Available: https://docs.hivedb.dev/#/best-practices/when_to_use_hive. [Accessed: 1-10-2022].
- [5] Dev, S. "Appropriate Uses For SQLite", [Online]. Available: <https://www.sqlite.org/whentouse.html>. [Accessed: 1-10-2022].
- [6] Fadli, A., Zulfa, M. I., Widhi Nugraha, A. W., Taryana, A., & Aliim, M. S. "Analisis Perbandingan Unjuk Kerja Database SQL dan Database NoSQL Untuk Mendukung Era Big Data". *JURNAL NASIONAL TEKNIK ELEKTRO*, vol. 9, no. 3, 2020.
- [7] Flutter. "Flutter Documentation", 2020. [Online]. Available: <https://docs.flutter.dev/> [Accessed: 2-10-2022].
- [8] Ilhami, M. "Tren dan Peluang Cross-Platform Mobile App untuk Developer Pemula". *KONSTELASI: Konvergensi Teknologi Dan Sistem Informasi*, vol. 1, no. 2. 2021.
- [9] Kabakuş, A. T. "A performance comparison of SQLite and Firebase databases from a practical perspective". *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, vol. 7, no. 1, 2019.
- [10] Lyu, Y., Gui, J., Wan, M., & Halfond, W. G. J. "An empirical study of local database usage in android applications". *IEEE International Conference on Software Maintenance and Evolution*, 2017.
- [11] Muslim., Sari, R. P., & Rahmayuda, S. "Implementasi Framework Flutter Pada Sistem Informasi Perpustakaan Masjid (Studi Kasus: Masjid di Kota Pontianak)". *Coding : Jurnal Komputer dan Aplikasi*, vol. 10, no. 1, 2022.
- [12] Raharjo, B. "Pemrograman Android dengan Fitur Flutter". Bandung: Informatika Bandung. 2019.
- [13] Rakhmawati, N. A., Zuhri, M., Wibowo, R. P., Romadhon, A., Ardiansyah, H., & Khoirul, O. "Benchmarking MySQL and NoSQL Databases on Egovbench Application". *Journal of Information Technology and Its Utilization*, vol. 2, no. 1, 2019.
- [14] Sudhakar, K. "Difference Between Sql and Nosql Databases". *International Journal of Management, IT & Engineering*, vol. 8, no. 6, 2018.
- [15] Wasilewski, K., & Zabierowski, W. "A comparison of java, flutter and kotlin/native technologies for sensor data-driven applications". *Sensors*, vol. 21, no. 10, 2021.