

Implementasi Peningkatan Performa pada Aplikasi Bikunku = Implementation of Performance Improvement in Bikunku Application

Antonius Anggito Arissaputro, author

Deskripsi Lengkap: <https://lib.ui.ac.id/detail?id=9999920541839&lokasi=lokal>

Abstrak

Bikunku merupakan suatu aplikasi penunjang informasi bis kuning (bikun); sarana transportasi di lingkungan Universitas Indonesia; yang dibuat oleh Alamsyah et al. melalui penelitiannya pada tahun 2022. Salah satu fitur penting yang ditawarkan dalam aplikasi ini adalah tracking lokasi bikun yang sedang beroperasi secara real-time. Meskipun secara keseluruhan sistem ini dapat menjalankan tugas-nya melakukan tracking lokasi bikun, sistem yang menggunakan frontend React, komunikasi WebSocket, dan database PostgreSQL ini kurang optimal untuk menangani data lokasi yang dikirim oleh driver bikun dan dibaca oleh ribuan sivitas akademik Universitas Indonesia.

Penelitian ini bertujuan melakukan improvement berupa peningkatan performa pada sistem tracking bikun pada aplikasi Bikunku yang meliputi pengukuran end-to-end response time, throughput, CPU usage, memory usage, FPS dan rendering speed. Penelitian terdiri dari pengembangan dari sisi backend protokol komunikasi menggunakan gRPC dengan Kafka, sisi database menggunakan Firebase Cloud Firestore sebagai Realtime Database pada backend, dan sisi frontend menggunakan Flutter sebagai aplikasi mobile. Implementasi frontend Flutter pada akhirnya diinkorporasikan dengan pilihan implementasi backend dengan hasil performa yang terbaik.

Dengan pengembangan menggunakan teknologi tersebut, improvement yang dilakukan pada penelitian ini meliputi perubahan cara penyaluran data lokasi tracking dan database dari sistem polling menjadi sistem pushing, perubahan arsitektur sistem yang lebih performant, digunakannya protokol komunikasi yang lebih unggul, perbaikan implementasi pada autentikasi driver sebagai salah satu penyebab bottleneck, perubahan aplikasi menjadi mobile app, dan digunakannya implementasi maps yang lebih siap untuk production. Pengetesan dilakukan dengan load testing membandingkan performa masing-masing sistem implementasi baru terhadap sistem terdahulu Alamsyah. Skenario pengetesan sistem backend dan database meliputi request pengiriman lokasi oleh 11 driver bikun, disertai 1800 concurrent user di sisi client untuk membaca simulasi data lokasi terbaru bikun yang beroperasi. Sementara pada sisi frontend dilakukan pengetesan dengan resource profiling dan observasi terhadap simulasi bikun yang berjalan di frontend.

Hasil pengetesan menunjukkan sistem gRPC-Kafka lebih cepat sebesar 96% dengan throughput yang dihasilkan lebih baik 216% dibandingkan dengan sistem Alamsyah, sistem Firebase lebih cepat sebesar 91% dengan throughput yang dihasilkan lebih baik 127% dibandingkan sistem Alamsyah. Dari hasil yang didapat, sistem gRPC-Kafka kemudian dipilih sebagai sistem backend yang diinkorporasikan dengan implementasi Flutter.

Hasil pengetesan sistem Flutter dengan gRPC-Kafka mendapatkan hasil yang lebih baik pada OS Android di semua skenario dengan peningkatan performa metrik CPU usage, max CPU usage, memory usage, dan rendering speed sebesar 27%, 18%, 42%, dan 33%. Sistem Flutter dengan gRPC-Kafka mayoritas lebih baik pada OS iOS di semua skenario, yakni pada metrik memory usage, FPS, dan rendering speed dengan peningkatan performa sebesar 40%, 28%, dan 16%. Sementara dari segi CPU usage dan max CPU usage,

sistem Alamsyah lebih baik sebesar 9% dan 0,3% dari sistem Flutter dengan gRPC- Kafka.

.....

Bikunku is an information system application for bikun, a bus transportation facility within the University of Indonesia environment, developed by Alamsyah et al. through their research in 2022. One of the important features offered in this application is real-time tracking of the bikuns' location. Although the overall system is capable of tracking the bikun's location, the system, which utilizes React for the frontend, WebSocket for communication, and PostgreSQL for the database, it is not optimized to handle the location data sent by bikun drivers and accessed by thousands of University of Indonesia stakeholders.

This research aims to improve the performance of the bikun tracking system in the Bikunku application, including measuring end-to-end response time, throughput, CPU usage, memory usage, FPS, and rendering speed. The research consists of the development of backend communication protocol using gRPC with Kafka, development of backend using Firebase Cloud Firestore as Realtime Database, and development of the frontend mobile application using Flutter. The implementation of Flutter frontend is then integrated with the best performing backend implementation.

With the development using technologies listed, improvements made in this research include changing the method of location tracking data delivery and database from a polling system to a pushing system, modifying the system architecture for better performance, utilizing superior communication protocol, improving the implementation of driver authentication as one of the bottleneck causes, transforming the application into a mobile app, and utilizing a more production-ready maps implementation. Load testing is conducted to compare the performance of each new implementation with Alamsyah's previous system. The testing scenarios for the backend and database systems involve location request submissions by 11 bikun drivers, along with 1800 concurrent users on the client side to simulate reading the latest location data of operating bikuns. On the frontend side, testing is performed using resource profiling and observation of the running bikun simulations.

The test results show that the gRPC-Kafka system is 96% faster with a 216% better throughput compared to Alamsyah's system, while the Firebase system is 91% faster with a 127% better throughput compared to Alamsyah's system. Based on these findings, the gRPC-Kafka system is then chosen as the backend system to be integrated with the Flutter implementation.

The testing results of Flutter with gRPC-Kafka demonstrate better performance on Android OS in all scenarios, with improvements in CPU usage, max CPU usage, memory usage, and rendering speed by 27%, 18%, 42%, and 33%, respectively. The Flutter with gRPC-Kafka system performs better on iOS OS in all scenarios for the majority, which are in terms of memory usage, FPS, and rendering speed, with performance improvements of 40%, 28%, and 16%, respectively. However, in terms of CPU usage and max CPU usage, the Alamsyah system performs better by 9% and 0.3% to the Flutter with gRPC-Kafka system.