

Perbandingan Flutter dan Native Development dalam Pengembangan Aplikasi Mobile



- Faizal Deshta Nugraha (2340506065)
- Oktario Mufti Yudha (2320506044)
- Hasan Fadhli Robbi (2340506080)
- Khaniq Naufal (2340506071)
- Naufal Dzaky Kusuma (2340506066)

DAFTAR ISI

DAFTAR ISI.....	2
BAB I Pendahuluan.....	3
BAB II Tinjauan Pustaka.....	4
A. Pengembangan Aplikasi Mobile.....	4
B. Cross-Platform dan Flutter.....	4
C. Studi Terkait.....	5
BAB III Pembahasan.....	5
BAB IV Kesimpulan dan Saran.....	7
A. Kesimpulan.....	7
B. Saran.....	7
Daftar Pustaka.....	8

BAB I

Pendahuluan

Pada era digital saat ini, pengembangan aplikasi mobile menjadi sangat penting. Dua platform dominan, Android dan iOS, masing-masing menguasai pasar global sekitar 72,23% dan 27,39% per April 2025. Hal ini berarti aplikasi umumnya harus tersedia di kedua sistem operasi tersebut. Pendekatan tradisional adalah pengembangan native, yaitu membangun aplikasi secara terpisah dengan bahasa dan alat khusus tiap platform (misalnya Kotlin/Java di Android Studio untuk Android, Swift/Objective-C di Xcode untuk iOS). Pendekatan ini menghasilkan performa dan integrasi maksimal dengan perangkat keras, namun membutuhkan dua basis kode terpisah.

Sebagai alternatif, muncul kerangka kerja cross-platform yang memungkinkan satu basis kode untuk dijalankan di banyak OS. Flutter adalah toolkit UI open-source besutan Google yang memungkinkan pembuatan aplikasi lintas platform dengan satu kode saja. Menurut Lovrić dkk. (2023), Flutter memungkinkan satu kode yang dapat digunakan di Android dan iOS, sehingga menghemat waktu dan usaha pengembangan.

Tulisan ini bertujuan membandingkan Flutter dengan pengembangan native (Android Studio dan Xcode). Kajian dilakukan melalui tinjauan pustaka internasional, identifikasi kelebihan/kekurangan, dan studi kasus penerapan Flutter.

Makalah ini tersusun sebagai berikut. BAB II membahas tinjauan pustaka tentang pengembangan aplikasi mobile, konsep Flutter, dan penelitian terkait. BAB III menyajikan pembahasan perbandingan teknis kelebihan dan kekurangan kedua pendekatan, serta contoh studi kasus. BAB IV berisi kesimpulan dan saran.

BAB II

Tinjauan Pustaka

A. Pengembangan Aplikasi Mobile

Pengembangan aplikasi mobile melibatkan pembuatan perangkat lunak yang dijalankan pada perangkat smartphone atau tablet. Statistik terbaru menunjukkan Android dan iOS mendominasi pasar OS mobile global, sehingga pengembang umumnya fokus pada kedua platform tersebut. Dalam pengembangan native, aplikasi ditulis dengan bahasa dan lingkungan yang spesifik untuk masing-masing platform (contoh: Java/Kotlin di Android Studio, Swift/ObjC di Xcode untuk iOS). Pengembangan native memberikan performa optimal dan akses penuh ke fitur perangkat keras, namun juga memerlukan tim dan usaha ganda karena mengelola dua kode terpisah. Pengembangan native seringkali dianggap lebih memakan waktu dan biaya dibanding cross-platform.

B. Cross-Platform dan Flutter

Untuk mengatasi duplikasi kerja di pengembangan native, muncul kerangka kerja cross-platform. Flutter adalah salah satunya, yaitu toolkit UI dari Google yang mendukung pengembangan aplikasi Android dan iOS menggunakan satu basis kode (berbahasa Dart). Dengan Flutter, aplikasi dibuat sekali dan dapat dijalankan di banyak platform, sehingga mengurangi waktu dan usaha pengembangan. Lovrić dkk. (2023) menekankan bahwa aplikasi dikembangkan sekali dan dapat diterapkan pada berbagai platform tanpa modifikasi kode, serta waktu pengembangan dan biaya dapat ditekan drastis. Sejak diperkenalkan akhir 2017, popularitas Flutter tumbuh pesat. Salah satu studi menyebutkan Flutter kini merupakan framework lintas platform paling populer dalam pengembangan aplikasi mobile. Keunggulan lain Flutter termasuk hot reload untuk percepatan iterasi desain, kumpulan widget UI yang kaya, dan dukungan kuat dari komunitas Google.

C. Studi Terkait

Beberapa studi membahas perbandingan Flutter dan native. Olsson (2020) membandingkan aplikasi Flutter dengan aplikasi native (Kotlin untuk Android, Swift untuk iOS). Ia menemukan aplikasi Flutter memiliki kinerja CPU yang hampir sebanding dengan aplikasi native, tetapi animasi dan efek grafis Flutter masih belum setara native. Flutter

juga menggunakan jauh lebih sedikit baris kode untuk aplikasi sederhana. Penelitian Andersson (2022) pada aplikasi Android menunjukkan hasil serupa: aplikasi Flutter unggul dalam pemrosesan file (lebih cepat) tetapi kurang optimal dalam rendering animasi dibanding aplikasi Android native. Secara keseluruhan, keduanya menunjukkan bahwa performa Flutter kini mendekati native, sehingga Flutter tidak dapat diremehkan meski bersifat cross-platform. Lovrić dkk. (2023) juga melakukan studi dengan membuat prototipe aplikasi konseling kanker menggunakan Flutter. Hasil studi ini menegaskan kemudahan pengembangan lintas platform dengan Flutter; sebuah aplikasi dibuat sekali dan dijalankan di Android dan iOS sekaligus. Studi-studi tersebut memberikan landasan teori bahwa Flutter banyak diminati karena efisiensi pengembangan, meski native masih unggul dalam aspek tertentu.

BAB III

Pembahasan

Pada bab ini dijelaskan kelebihan dan kekurangan masing-masing pendekatan serta contoh studi kasus penerapan Flutter. Ringkasan perbandingan disajikan sebagai berikut:

- **Kelebihan Flutter:** Satu basis kode untuk Android dan iOS, sehingga mempermudah pemeliharaan dan mempercepat pengembangan. Sebagai contoh, Alibaba melaporkan Flutter memperpendek waktu pengembangan fitur baru dari 1 bulan menjadi 2 minggu. Flutter juga umumnya membutuhkan lebih sedikit baris kode untuk aplikasi sederhana dibanding native. Performa CPU aplikasi Flutter kini dapat bersaing dengan aplikasi native pada banyak operasi (misalnya pemrosesan data dan akses basis data). Selain itu, Flutter menyediakan beragam widget UI, fitur hot reload, serta komunitas dan dokumentasi aktif dari Google.
- **Kekurangan Flutter:** Animasi dan penggunaan GPU kompleks kurang optimal dibanding native. Hal ini terlihat dari studi di mana animasi Flutter tertinggal di belakang aplikasi native dengan OpenGL atau framework grafis asli. Selain itu, paket pustaka pihak ketiga (library) Flutter masih terus berkembang, sehingga beberapa fungsi khusus perangkat mungkin memerlukan kerja lebih. Ukuran file (APK/IPA) aplikasi Flutter cenderung lebih besar, karena menyertakan mesin

runtime dan aset UI-nya.

- **Kelebihan Native (Android/iOS):** Performa maksimal dan integrasi penuh dengan perangkat keras (misalnya pemrosesan grafis dan animasi) menghasilkan pengalaman pengguna halus. Ekosistem library dan alat pengembangan yang sudah matang, serta panduan desain Material (Android) atau Human Interface (iOS) disediakan lengkap. Akses API terbaru platform lebih cepat tersedia.
- **Kekurangan Native:** Pengembangan harus dilakukan dua kali (terpisah untuk Android dan iOS), sehingga memerlukan tim lebih besar dan sumber daya yang lebih banyak. Biaya dan waktu pengembangan biasanya lebih tinggi. Pemeliharaan aplikasi menjadi lebih rumit karena perubahan kode perlu dilakukan dua kali di basis kode berbeda.

Contoh studi kasus penerapan Flutter menunjukkan keuntungan nyata di lapangan. Misalnya, Alibaba (e-commerce besar) menyatakan Flutter membuat pengembangan fitur baru lebih cepat dua kali lipat. Demikian pula, Lovrić dkk. (2023) menunjukkan pembuatan prototipe aplikasi konseling kanker lebih efisien karena cukup dengan satu kode Flutter untuk lintas platform. Selain itu, banyak perusahaan global (seperti Alibaba, Google Ads, dan Nubank) menggunakan Flutter untuk mempercepat waktu ke pasar aplikasi mereka.

BAB IV

Kesimpulan dan Saran

A. Kesimpulan

Berdasarkan tinjauan pustaka dan pembahasan di atas, dapat disimpulkan bahwa Flutter menawarkan jalur pengembangan lintas platform yang efisien. Dengan satu basis kode yang sama untuk Android dan iOS, Flutter mempercepat pengembangan dan mempermudah pemeliharaan aplikasi. Studi menunjukkan kinerja aplikasi Flutter dalam aspek CPU kini mendekati aplikasi native. Namun demikian, keunggulan native masih terlihat pada pengolahan grafis dan animasi kompleks; aplikasi native umumnya lebih

unggul di area tersebut. Singkatnya, Flutter sangat cocok digunakan untuk aplikasi berskala kecil hingga menengah yang mengutamakan waktu-ke-pasar dan efisiensi pengembangan, sedangkan aplikasi dengan kebutuhan performa puncak atau fitur platform khusus sebaiknya mempertimbangkan pengembangan native.

B. Saran

Pengembang sebaiknya mempertimbangkan kebutuhan proyek secara menyeluruh saat memilih antara Flutter atau native. Jika waktu dan biaya pengembangan lebih diutamakan, Flutter merupakan pilihan efektif karena “satu kali pengembangan untuk banyak platform”. Jika aplikasi memerlukan performa maksimal atau integrasi fitur khusus, pengembangan native mungkin lebih sesuai. Untuk penelitian selanjutnya, direkomendasikan eksplorasi lebih lanjut aspek lain seperti keamanan, skalabilitas, dan pengalaman pengguna (UX) secara empiris. Mengingat Flutter terus berkembang cepat, studi tambahan sangat penting untuk memverifikasi hasil saat ini dan memperbarui perbandingan ketika versi baru rilis.

Daftar Pustaka

Andersson, H. (2022). *A Comparison of the Performance of an Android Application Developed in Native and Cross-Platform Using the Native Android SDK and Flutter*. Bachelor's thesis, Blekinge Institute of Technology, Sweden.

Lovrić, L., Fischer, M., Röderer, N., & Wunsch, A. (2023). *Evaluation of the cross-platform framework Flutter using the example of a cancer counselling app*. In Proceedings of the 9th International Conference on ICT for Ageing Well and e-Health (ICT4AWE).

Olsson, M. (2020). *A Comparison of Performance and Looks Between Flutter and Native Applications: When to Prefer Flutter over Native in Mobile Application Development*. Bachelor's thesis, Blekinge Institute of Technology, Sweden.

StatCounter. (2025). *Mobile Operating System Market Share Worldwide – April 2025*. Diakses dari <https://gs.statcounter.com/os-market-share/mobile/worldwide>. (dipublikasi 2025)

Zammetti, F. (2019). *A new adventurer in mobile development: Flutter*. Diakses dari <https://medium.com/flutter>. (artikel pendukung)