

Nama : Fauzan Abdurrahman

NPM : 1806065

Pemrograman Perangkat Bergerak

---

---

**Perbandingan Framework Pemrograman Mobile**

| <b>Keterangan</b> | <b>Flutter</b>  | <b>React Native</b>   |
|-------------------|---|---|
| Pengertian        | Flutter adalah perangkat UI Google untuk membuat aplikasi cantik yang dikompilasi secara asli untuk seluler, web, dan desktop dari satu basis kode  | React Native adalah sebuah framework untuk membangun aplikasi native menggunakan React  |
| Pembuat           | Google  | Facebook  |
| Tahun didirikan   | 2018  | 2015  |
| Bahasa Pemograman | Dart  | Javascript  |
| Popularitas       | 81.200 stars di Github pada bulan Desember 2019.  | 83.200 stars di Github pada bulan Desember 2019.  |
| Pengguna          | Xianyu app by Alibaba, Hamilton, Google Ads app, dll  | Instagram, Facebook, Skype, Tesla, dll  |
| UI                | <p>Flutter hanya memiliki satu basis kode. Aplikasi yang kita buat akan terlihat dan berperilaku serupa di iOS dan Android.</p> <p>Adanya Material Design dan Cupertino Widget membuat Flutter juga dapat meniru desain platform itu sendiri (iOS atau Android)</p> | <p>Komponen UI aplikasi akan terlihat seperti yang asli. Misalnya, tombol pada perangkat iOS terlihat seperti tombol iOS asli. Begitu juga dengan Android.</p> <p>Fakta bahwa React Native menggunakan komponen asli akan memberikan dampak setelah pembaruan OS OS apa pun, komponen aplikasi kita akan langsung ditingkatkan juga</p> |

## Kelebihan Flutter

### 1. Hot Reload

Flutter menawarkan pengembangan aplikasi yang lebih dinamis dan lebih cepat. Ini adalah salah satu hal terbesar tentang Flutter, yang dihargai oleh setiap perusahaan pengembang aplikasi mobile terkemuka.

Pengembang dapat membuat perubahan pada basis kode saat itu juga, dan melihatnya secara langsung dalam aplikasi. Ini adalah apa yang disebut fitur Hot Reload. Biasanya diperlukan waktu (mili-) detik untuk perubahan ditampilkan.

Fitur ini membantu tim menambahkan fitur, memperbaiki bug, dan bereksperimen dengan ide-ide baru dalam sekejap.

### 2. Sekali Ngoding, 2 atau Lebih Target Platform

Flutter tidak hanya ditujukan untuk Android dan iOS. Sekarang Flutter telah mencapai ke desktop dan website.

Itu artinya, developer akan mampu membuat satu kode program saja yang bisa dijalankan di berbagai platform.

### 3. Fast App

Flutter berfungsi dengan lancar dan cepat – tanpa pernah menggantung atau memotong saat scrolling.

Flutter menggunakan Skia Grafik Library. Berkat ini, UI digambar ulang setiap kali tampilan berubah.

Sebagian besar pekerjaan dilakukan pada GPU (unit pemrosesan grafis); itu sebabnya Flutter UI halus dan menghasilkan 60fps (frames per detik).

Namun, tetap harus berhati-hati selama pengembangan agar tidak menyebabkan *redrawing* elemen-elemen tampilan yang datanya tidak berubah.

*Redrawing* ulang seluruh tampilan dapat memengaruhi kinerja dan kecepatan aplikasi, terutama jika kita harus sering memuat ulang tampilan.

Misalnya dalam aplikasi *stopwatch*.

### 4. Desain yang Disukai Pengguna

Flutter tidak bergantung pada komponen sistem asli. Sebaliknya, Flutter memiliki serangkaian widget khusus, yang dibuat dan dikelola oleh Framework Graphics Engine.

Aplikasi Flutter memiliki UI yang sangat ramah dengan pengguna.

Keuntungan penting bagi Flutter di atas React Native yaitu berasal dari perhatian detail visual.

Flutter dibuat untuk memudahkan kita membuat widget sendiri, atau hanya menyesuaikan widget yang ada.

## 5. Tampilan Sama, Meski di Platform yang berbeda

Aplikasi yang kita bangun menggunakan Flutter akan berjalan sama, baik di Android ataupun di iOS.

Ini memberikan kita pekerjaan yang lebih mudah dan relatif singkat.

## Kekurangan Flutter

### 1. Jumlah Komunitas Developer Flutter

Dalam hal bahasa pemrograman, Dart tidak digunakan secara luas seperti JavaScript, setidaknya untuk saat ini.

Komunitas membutuhkan waktu untuk mendidik penggunanya dan untuk mendapatkan lebih banyak pengalaman.

Flutter tumbuh secara eksponensial. Plus, ada kegembiraan tersendiri ketika menggunakan toolkit ini.

Data mengenai pengguna Flutter :

- Tren tag stackoverflow menunjukkan peningkatan minat pengembang di Flutter.
- Di Udemy, ada 50+ kursus Flutter (dengan sekitar 213.000 peserta kursus).
- Ada juga 35 kursus Dart (dengan sekitar 187.500 peserta kursus (25 Juni 2019)).
- Flutter memiliki angka sekitar 287.000 siswa di 59 kursus.
- Flutter memiliki 80.600+ bintang di Github sementara React Native memiliki 83.000+
- Singkatnya, Flutter memang memiliki komunitas yang lebih kecil dan kurang berpengalaman saat ini, dan Dart lebih disukai daripada JavaScript.

### 2# Dokumentasi dan Support Mengesankan, Tetapi Masih Belum Cukup

Dukungan Google untuk Flutter sangat mengesankan, tetapi Flutter masih cukup baru. Ini berarti kita tidak dapat selalu menemukan *problem solving* yang kita butuhkan di dokumentasi yang ada.

Sehingga kita perlu membangun penyelesaian *problem solving* sendiri. Pastinya ini akan membutuhkan waktu.

Pada saat penulisan, Flutter tidak memiliki dukungan untuk platform CI seperti Travis atau Jenkins. Jadi, untuk mencapai pembangunan, pengujian, dan penyebaran otomatis, pengembang perlu menggunakan dan memelihara skrip khusus seperti [ini](#).

Perlu dicatat bahwa :

- Ada sistem CI / CD baru untuk aplikasi Flutter yang diumumkan di Flutter Live 2018
- Pada Januari 2019, Bitrise mengumumkan Flutter CI berfitur lengkap.

#### 4. Risiko Platform

Meskipun Flutter adalah *open source*, jika Google memutuskan untuk menarik dukungan untuk proyek tersebut, itu akan berarti bencana.

Sejak Google merilis Flutter versi Beta, ia hanya meningkatkan usahanya, seperti yang diilustrasikan oleh peran Flutter yang menonjol selama Google IO '19, di samping acara Flutter Live baru-baru ini.

#### 5. Ukuran Aplikasi

Aplikasi yang dibuat menggunakan Flutter lebih besar daripada yang asli. Namun, tim Flutter sedang berupaya mengurangi ukuran aplikasi yang dibuat dengan Flutter.

## **Kelebihan React Native**

### **1. Menggunakan Bahasa yang Populer**

React Native menggunakan JavaScript: bahasa pemrograman yang banyak pengembang tahu dengan baik (sedangkan Dart masih belum begitu banyak dikenal atau digunakan).

Jika seorang pengembang yang lebih suka bahasa pemrograman yang diketik secara statis, kita dapat menggunakan TypeScript (subset JavaScript) di React Native.

### **2. Kematangan Relative**

Rilis asli React Native lebih dari 5 tahun yang lalu, sehingga tim Facebook memiliki banyak waktu untuk menstabilkan API, serta fokus pada memperbaiki masalah dan menyelesaikan masalah.

### **3. Mudah Dipelajari**

Keuntungan ini sangat ditargetkan untuk React Developer.

Jika kita memiliki latar belakang dalam pengembangan web dan sudah menggunakan React yang populer, kita dapat dengan mudah bekerja dengan React Native, tanpa harus mempelajari dokumentasi yang baru.

### **4. Sekali Ngoding, 2 atau Lebih Target Platform**

React Native pun bisa membuat aplikasi ke dalam 2 target (iOS dan Androd) hanya dengan sekali coding. Ini akan memudahkan kita dan mempersingkat waktu pengerjaan project.

### **5. Fast Refresh**

Pada dasarnya fitur ini sama dengan Flutter.

Hot Reload mempercepat proses pengembangan dengan memungkinkan pengembang untuk menyuntikkan kode baru langsung ke aplikasi yang sedang berjalan.

Jadi, pengembang dapat melihat perubahan secara instan, tanpa membangun kembali aplikasi.

Hot Reload juga mempertahankan status aplikasi, menghindari risiko kehilangannya saat memuat ulang penuh (manfaat penting dalam konteks kerangka kerja berbasis negara) – mempercepat proses pengembangan aplikasi lebih jauh.

Untuk meningkatkan pengalaman pengembang dengan pemuatan Hot Reload, tim React Native memasukkan dalam versi 0.61 fitur baru yang disebut penyegaran cepat yang menyatukan pemuatan Hot dan Reload.

Ini lebih tahan terhadap kesalahan ketik dan kesalahan dibandingkan dengan versi sebelumnya. Anda dapat membaca lebih lanjut tentang penyegaran cepat di [sini](#).

## **Kekurangan React Native**

### **1. Bukan Native Asli**

Seperti halnya solusi lintas-platform, pengalaman atau kinerja UI tidak akan sama dengan di aplikasi asli (bawaan OS), hanya cukup dekat dengan mereka.

Tapi tetap saja, lebih mudah untuk mencapai “native feeling” dengan React Native daripada dengan Flutter.

Jika kita ingin aplikasi Flutter memiliki komponen asli, itu akan membutuhkan pekerjaan tambahan.

### **2. Aplikasi Lebih Besar Daripada Native Asli**

Aplikasi yang ditulis dalam React Native harus dapat menjalankan kode Javascript (JavaScript Virtual Machine). Android tidak memiliki fungsi ini secara default.

Artinya aplikasi harus menyertakan librray yang mendukung kode JavaScript, dan itu akan menghasilkan aplikasi yang lebih besar daripada Android aslinya.

Aplikasi iOS yang dibuat dengan React Native tidak jadi masalah, tetapi mereka biasanya lebih besar dari yang Native.

### **3. UI Rapuh**

Fakta bahwa React Native menggunakan komponen native, memberi kita keyakinan bahwa, setiap pembaruan OS OS, komponen aplikasi Anda akan langsung ditingkatkan juga.

Ini dapat merusak UI aplikasi tetapi jarang terjadi.

Yang lebih buruk, pembaruan dapat menjadi lebih berbahaya jika mereka menyebabkan perubahan tertentu pada API di Native.

### **4. Kebebasan Memilih Pengembang**

Setelah pengembang membuat proyek baru, mereka kemudian perlu memutuskan paket navigasi mana – serta manajemen negara global mana – yang akan digunakan.

### **5. Lebih Sedikit Komponen Out The Box**

React Native hanya mendukung komponen dasar out the box (banyak di antaranya adaptif dengan platform out the box, seperti tombol, indikator pemuatan, atau penggeser).

Ada repo luar dengan banyak komponen tambahan untuk React Native.

Pengembang dapat menggunakannya dalam suatu proyek tetapi itu membutuhkan usaha dan waktu tambahan.

Di sisi lain, jika melihat Flutter vs React Native, justru Flutter dirancang untuk mendukung Desain Bahan out the box, sehingga kerangka kerja mendukung lebih banyak widget.

Ini menghemat waktu.

Pengembang yang menggunakan Flutter dapat membuat sebagian besar tampilan dengan widget yang mudah disesuaikan dan lintas platform.

### **Kesimpulan**

Saya memilih Flutter

## **Daftar Pustaka**

Administrator. (2020, Mei 24). *Flutter vs React Native: Performa Keduanya di Tahun 2020*. Retrieved from Ilmu Coding: <https://ilmucoding.com/flutter-vs-react-native/>

Facebook. (n.d.). *React Native*. Retrieved from React Native: <https://reactnative.dev/>

Google. (n.d.). *Flutter*. Retrieved from Flutter: <https://flutter.dev/>