

Flutter - Masa Depan Aplikasi Mobile?



Selama ini pengembangan aplikasi *mobile* terbagi menjadi beberapa cara pengembangan yaitu:

Native: menggunakan bahasa pemrograman dan *framework* yang disediakan oleh vendor sistem operasi, di mana Android menggunakan Kotlin, Java dan c++, sedangkan iOS, menggunakan Swift dan Objective C. Kelebihan cara pengembangan *native* adalah performa aplikasi yang sangat cepat dan dapat mengakses ke kamera, mikrofon, GPS dan *bluetooth* dengan lebih mudah. Sedangkan kekurangannya adalah biaya pengembangan yang mahal dan membutuhkan waktu lebih lama, karena masing-masing *platform* harus dikembangkan oleh *developer* yang menguasai bahasa pemrograman yang berbeda.

Hybrid: menggunakan bahasa pemrograman web (HTML dan Javascript) yang berjalan di dalam Native App, dengan menggunakan WebView Wrapper. Aplikasi yang menggunakan cara pengembangan ini juga dapat mengakses kamera, mikrofon, GPS dan beberapa perangkat lainnya, tetapi tidak semudah Native. Kelebihan cara pengembangan ini adalah proses lebih cepat dan lebih murah karena *developer* hanya mengembangkan satu aplikasi web untuk kemudian 'dibungkus' dalam satu *webview* di Native App. Sedangkan kekurangannya adalah performa yang cenderung kurang responsif dan terlalu banyak mengkonsumsi memori yang akhirnya menyebabkan menurunnya *user experience*. Ada beberapa *framework* yang bermain di area ini, salah satunya Apache Cordova.

Core Native: menggunakan bahasa yang bukan Native tetapi hasil dari kompilasi yang diterjemahkan menjadi Core Native sehingga performanya mendekati Native. Contoh *framework* yang dipakai adalah: React Native dari Facebook, Ionic, Xamarin dari Microsoft, dan Flutter dari Google. Kelebihan cara ini adalah menghemat waktu dan biaya pengembangan, kemampuan mengakses sebagian besar fitur API (Application Programming Interface) Native, dan performa yang mendekati Native. Kekurangannya adalah tidak semua porsi dari kode dapat diterjemahkan ke masing-masing *platform* sehingga *developer* harus membuat *bridging* (dalam Native) yang kadang memakan waktu yang lebih lama [untuk lebih detail] dan ukuran aplikasi yang lebih besar.

Dalam artikel singkat ini akan dibahas sekilas tentang Flutter, sebuah Core Native *framework open source* yang dikembangkan oleh Google sejak 2015 berbasis bahasa pemrograman Dart yang juga dikembangkan oleh Google. Pada akhir 2018, setelah melalui pengembangan selama 3 tahun, Flutter akhirnya mencapai tahap stabil versi 1.0. Salah satu alasan Flutter lebih menarik daripada *framework* Core Native lainnya karena Google berada di belakang *framework* ini, dan aplikasi yang dikembangkan menggunakan Flutter juga dapat berjalan di sistem operasi milik Google yaitu Fuchsia.

Beradaptasi dengan Flutter (dart) tidak begitu sulit jika Anda sudah mengenal bahasa pemrograman Java, Kotlin, Swift, atau bahasa pemrograman berorientasi objek lainnya yang merupakan turunan C. Yang membuat kurang nyaman bagi *developer* yang terbiasa menggunakan *layout tools* seperti *storyboard* pada XCode atau bahasa Markup XML pada android adalah mereka harus terbiasa membuat *layout* dengan kode. Dalam hal kecepatan di beberapa *benchmark online*, Flutter dirasa lebih cepat dan responsif dibandingkan React Native, tetapi tidak lebih cepat dibandingkan Native. Fitur Hot Reload sangat memudahkan *developer* dalam mengembangkan aplikasi. Seperti halnya React Native, aplikasi tidak perlu dibangun ulang setiap kali *developer* melakukan perubahan di kode. Walaupun XCode dan Android studio sudah mempunyai fitur Incremental Build, Hot Reload tetap lebih cepat.

Kelebihan Flutter sampai saat ini adalah "*single codebase run anywhere*", pengembangan aplikasi lebih cepat dan biaya lebih murah, performa aplikasi mendekati Native. Sedangkan kekurangannya adalah *library* pendukung pihak ke 3 masih sangat minim, *framework* masih sangat baru sehingga rentan sekali akan perubahan besar di versi berikutnya (seperti Swift 1 ke Swift 2) dan belum ada konvensi *design pattern* yang dapat dijadikan acuan karena komunitas masih sangat baru sehingga "*best practice*" nya belum terdefinisi dengan jelas. Hingga saat ini ada beberapa nama besar yang sudah mengadopsi Flutter sebagai *framework* utama mereka, salah satunya adalah Alibaba dengan Xianyu app nya.

Flutter bukan merupakan pengganti dari bahasa utama yang saat ini umum digunakan untuk pengembangan aplikasi *mobile*, tetapi tidak menutup kemungkinan di tahun - tahun berikutnya seiring dengan banyaknya nama - nama besar yang ikut andil, Flutter akan menjadi satu *framework* yang akan menjadi standar dalam pengembangan aplikasi *mobile*. Untuk saat ini Trimegah belum menggunakan Flutter sebagai *framework* untuk membangun aplikasi *mobile*, tetapi tidak tertutup kemungkinan Flutter akan menjadi *framework* utama ke depannya.

Referensi :

- https://flutter.io/?gclid=EAIaIQobChMIl-yJ0PrE4AIVUB0rCh1KIw_aEAAYASAAEgK-L_D_BwE
- <https://www.androidpolice.com/2018/12/05/google-announces-flutter-1-0-the-first-stable-release-of-its-cross-platform-mobile-development-toolkit/>
- <https://medium.com/airbnb-engineering/sunsetting-react-native-1868ba28e30a>

PT Trimegah Sekuritas Indonesia Tbk adalah Perusahaan Efek Anggota Bursa Efek Indonesia. PT Trimegah Sekuritas Indonesia Tbk dan PT Trimegah Asset Management telah memiliki izin usaha, terdaftar dan diawasi oleh Otoritas Jasa Keuangan.