

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Penelitian mengenai react native dalam studi kasus membangun aplikasi reminder mutaba'ah yaumiyah yang dilakukan oleh Sugara, Syauqi Rahmat (2018). Hasil aplikasi yang dihasilkan memiliki kualitas yang baik dari aplikasi yang dibangun sebelumnya tanpa react native.

Penelitian yang dilakukan oleh Aji, Muhammad Darma (2017) tentang analisis perbandingan antara Yii dengan Codeigniter. Hasil dari penelitian mendapatkan informasi mengenai komparasi dari kedua framework tersebut.

Penelitian yang dilakukan oleh Aji, Dwi Rizal Fathoni (2018) tentang Perbandingan Phonegap dan React native sebagai framework pengembangan aplikasi mobile. Hasil yang dihasilkan dari penelitian tersebut adalah informasi mengenai komparasi dari kedua framework tersebut.

Dari penelitian yang telah dilakukan oleh peneliti-peneliti sebelumnya terdapat perbedaan yang bisa dilihat pada tabel 2.1.

Tabel 2. 1 Tabel Perbandingan Penelitian

| No | Nama Penulis | Judul | Metode/ Teknologi | Masalah | Hasil |
|----|---|--|----------------------|--|---|
| | Sugara, Syauqi Rahmat (2018) | Eksplorasi framework React native dalam pembangun | React native | Aplikasi masih belum memiliki kualitas yang baik | Aplikasi dibangun dengan React native |

| No | Nama Penulis | Judul | Metode/ Teknologi | Masalah | Hasil |
|----|----------------------------|---|-----------------------------|---|---|
| | | an aplikasi berbasis mobile (Studi kasus: aplikasi reminder mutaba'ah amal yaumiyah) | | | |
| 2 | Aji, Muhammad Darma (2017) | Analisis perbandingan antara Yii framework dengan codeigniter framework untuk developer web | Code Igniter dan Yii | Framework apa yang lebih unggul antara Code Igniter dan Yii | Informasi hasil komparasi antara Code Igniter dan Yii |
| 3 | Aji, Dwi Wijonarko, Rizal | Perbandingan Phonegap dan React | Javascript dan React native | Framework apa yang lebih unggul Phonegap | Informasi hasil komparasi antara |

| No | Nama Penulis | Judul | Metode/ Teknologi | Masalah | Hasil |
|----|----------------|---|-------------------|-------------------------------|---------------------------|
| | Fathoni (2018) | native sebagai framework pengembangan aplikasi mobile | | dan React native dengan hasil | Phonegap dan React native |

2.2. Dasar Teori

2.2.1. Flutter

Flutter merupakan sebuah SDK untuk pengembangan aplikasi mobile yang dikembangkan oleh Google untuk membangun aplikasi yang memiliki kinerja tinggi serta dapat dipublikasi ke platform Android dan iOS dari codebase tunggal.

Flutter dapat dengan mudah dipelajari karna menggunakan bahasa pemrograman Dart yang pastinya terasa familier jika sudah terbiasa menggunakan bahasa pemrograman Java atau Javascript. Selain itu Flutter juga menyertakan kerangka reactive-functional, mesin render 2D, widget siap pakai, dan tools untuk pengembangan.

Ada begitu banyak kerangka yang bisa digunakan untuk mengembangkan aplikasi lintas platform, seperti React Native, Nativescript, dan Fuse. Namun yang membedakannya adalah, Flutter tidak menggunakan Webview maupun widget bawaan, Flutter punya mesin render sendiri untuk menampilkan widget-nya, hal

ini menguntungkan developer yang ingin memiliki tampilan UI unik yang konsisten pada semua perangkat karena tidak bergantung pada widget bawaan OEM.

Kinerja Flutter yang tinggi ini tentunya didukung oleh berbagai teknologi terbaik. Flutter dibuat dengan C, C++, Dart, Skia untuk mesin render 2D, Mojo IPC, dan Blink untuk sistem render.

Cara kerja Flutter pada platform Android yaitu, kode C/C++ dikompilasi menggunakan Android NDK, sebagian besar kerangka dan kode aplikasi dijalankan dalam bentuk kode native yang dikompilasi oleh Dart compiler. Sedangkan pada platform iOS, kode dikompilasi dengan LLVM dan aplikasi dijalankan dengan kumpulan instruksi native tanpa interpreter.

Flutter dapat berjalan pada sistem operasi Android 4.1 atau lebih tinggi dan iOS 8 atau lebih tinggi; dan dapat dijalankan pada perangkat asli maupun Android emulator, serta iOS simulator.

Semua developer pastinya menginginkan kinerja aplikasi yang paling optimal. Flutter menjanjikan kinerja yang sempurna, aplikasi yang dibangun dengan Flutter dapat berjalan secara konstan pada tingkat 60 frame per detik hingga 120fps.

Alasan utama mengapa Flutter tidak menggunakan widget bawaan yaitu, Flutter tidak ingin kinerjanya dibatasi dan bergantung pada kinerja widget bawaan sehingga aplikasi kualitas tinggi dapat dihasilkan.

Flutter sudah dipaketkan dengan kode pengaksesan platform service dan API, seperti sensor, penyimpanan lokal dan lain sebagainya. Namun, untuk menghindari kendala, pengaksesan API untuk platform yang spesifik, dapat digunakan dengan plugin untuk integrasi.

Jika ingin melakukan interoperabilitas dengan kode yang sudah ada seperti Java, Objective-C, atau Swift, bisa diaktifkan dengan melintaskan pesan secara fleksibel dengan PlatformMessageChannel.

2.2.2. React Native

React native adalah framework dari javascript yang digunakan untuk membuat aplikasi mobile pada platform IOS dan android. React native ini dibuat berdasarkan React, tetapi tidak mengacu kepada browser, melainkan ke platform mobile.

seperti reactjs, react native ditulis dari perpaduan javascript dan XML-esque yang disebut JSX. React native ini yang menjembatani native rendering APIs pada Objective-C(untuk IOS) dan Java (untuk Android) sehingga dapat digunakan pada IOS dan Android sekaligus. React native juga bisa digunakan untuk mengakses fungsi dari platform contohnya seperti kamera dan GPS.

2.2.3. APK (Android Package Kit)

APK (Android Package Kit) adalah format file untuk aplikasi yang digunakan pada sistem operasi Android. File APK dikompilasi dengan Android Studio yang merupakan IDE (Integrated Development Environment) resmi untuk membangun perangkat lunak Android.

File APK mencakup semua kode dan aset program perangkat lunak. Pengembang yang membuat Aplikasi untuk digunakan pada perangkat Android harus mengkompilasi aplikasi mereka ke dalam format APK sebelum mengunggah ke Google Play, pasar resmi untuk aplikasi Android. Google Play membutuhkan file APK berukuran kurang dari 100 MB. Untuk membantu pengembang dengan batasan ini, Google Play akan menyimpan hingga dua file ekspansi APK untuk setiap konten tambahan yang diperlukan untuk aplikasi mereka.

File ekspansi APK digunakan untuk menyimpan file media, grafik yang tinggi, atau aset besar lainnya yang akan menyebabkan APK melebihi batasan ukuran. File APK juga dapat didistribusikan secara langsung ke pengguna Android lain untuk instalasi di perangkat pengguna. Pengguna Android dapat memberikan izin kepada perangkat pengguna untuk menginstal aplikasi yang tidak dikenal jika mereka ingin mengakses file APK dari sumber lain dan menginstalnya secara langsung. Pengguna Android mungkin ingin menginstal APK secara langsung jika pengguna sedang menguji versi aplikasi yang belum dirilis, atau karena pembatasan perangkat, tidak dapat mengunduh aplikasi dari Google Play. (<https://developmenteloper.android.com>)

2.2.4. API (Application Programming Interface)

API adalah sebuah teknologi untuk memfasilitasi pertukaran informasi atau data antara dua atau lebih aplikasi perangkat lunak. API adalah antarmuka virtual antara dua fungsi perangkat lunak yang saling bekerja sama, seperti antara sebuah word processor dan spreadsheet. Sebuah API mendefinisikan bagaimana cara

programmer memanfaatkan suatu fitur tertentu dari sebuah komputer. API tersedia untuk sistem windowing, sistem file, sistem basis data dan sistem jaringan.