



PENERAPAN FIREBASE REALTIME DATABASE PADA APLIKASI MOBILE MENGGUNAKAN PLATFORM KODULAR

Sanusi Amir

Institut Teknologi Informasi dan Bisnis Stikom Ambon
Corresponding Author:: ucilinkun9@mail.com

Article Info

Article history:

Received: 25/09/2025
Revised: 1/10/2025
Accepted: 10/10/2025
Published: 15/10/2025

Keywords:

Firestore Database
Kodular
Mobile Application
CRUD
No-Code Development

ABSTRACT

Aplikasi mobile telah mendorong kebutuhan akan sistem database yang efisien dan real-time. Penelitian ini bertujuan untuk menerapkan Firebase Realtime Database pada aplikasi mobile berbasis Android menggunakan platform Kodular sebagai alternatif pengembangan aplikasi tanpa coding (no-code). Metode yang digunakan adalah eksperimental dengan mengembangkan aplikasi CRUD (Create, Read, Update, Delete). Hasil penelitian menunjukkan bahwa Kodular dapat terintegrasi dengan baik dengan Firebase Realtime Database, memungkinkan operasi CRUD berjalan secara real-time dengan response time rata-rata 1.2 detik. Aplikasi berhasil menyimpan dan menampilkan data secara sinkron di cloud database. Kombinasi Kodular dan Firebase Realtime Database merupakan solusi yang efektif untuk pengembangan aplikasi mobile yang cepat, efisien, dan tidak memerlukan pengetahuan pemrograman mendalam, cocok untuk prototyping dan implementasi sistem informasi skala kecil hingga menengah.

1. INTRODUCTION

Kebutuhan aplikasi mobile berbasis android menjadi salah satu platform yang paling banyak digunakan oleh organisasi dalam mengelola dan mengakses data, tentunya dengan dukungan sistem database yang dapat menyediakan data secara real-time.

Firebase Realtime Database merupakan database yang tersimpan di cloud dan support multiplatform seperti Android, iOS dan Web. Data pada firebase akan disimpan dalam struktur JSON (Java Script Object Notation). Database firebase akan melakukan sinkronisasi secara otomatis terhadap aplikasi client yang terhubung kepadanya. Aplikasi multiplatform yang menggunakan SDK Android, iOS dan JavaScript akan menerima update data terbaru secara otomatis pada saat aplikasi terhubung ke server firebase [1].

Firebase sendiri memiliki keunggulan dengan memiliki fitur lengkap seperti authentication, firestore database, realtime database, google cloud messaging bahkan sampai dengan fitur machine learning. Firebase sendiri secara keseluruhan menyediakan kemudahan dengan cara membuat database server dengan no sql. Oleh sebab itu, pemanfaatan firebase dirasa perlu untuk mengembangkan aplikasi berbasis mobile, dengan memiliki keunggulan tanpa harus membuat server untuk penyimpanan database [2].

Kodular merupakan platform yang mudah digunakan untuk membuat aplikasi mobile dengan antarmuka drag-and-drop, sehingga mengurangi

kebutuhan untuk menulis kode secara manual. Dengan menggunakan Kodular, pengembang dapat dengan mudah membuat dan mempublikasikan aplikasi mobile yang kuat dan canggih [3].

Sistem operasi android merupakan sebuah sistem operasi *mobile* besutan Google yang berjalan diatas kernel *linux* dan bersifat *Open Source*. Struktur aplikasi *android* menggunakan bahasa pemrograman *Java* ataupun *Kotlin*, lalu dengan bahasa pemrograman tersebut di kompilasi sehingga menghasilkan sebuah file dengan ekstensi *.apk* yang mana file inilah yang bisa di pasang pada sistem operasi *android* [4].

Untuk perangkat seluler yang menjalankan Linux, *Android* adalah sistem operasi yang meliputi sistem operasi, middleware, dan aplikasi. *Android* menyediakan platform gratis bagi pengembang untuk membangun aplikasi mereka. *Android* adalah sistem operasi yang mengerakkan lebih dari satu miliar smartphone dan tablet [5].

Penelitian ini fokus pada menerapkan Firebase Realtime Database menggunakan platform Kodular untuk membuat aplikasi sederhana manajemen data mahasiswa program studi bisnis digital ITB-Stikom Ambon. Aplikasi ini mencakup operasi CRUD (Create, Read, Update, Delete) dengan field data yang terdiri dari NIM, nama mahasiswa, angkatan, no handphone, no handphone orang tua, sekolah asal.

Kombinasi kedua teknologi ini diharapkan dapat menghasilkan aplikasi mobile yang mampu

memenuhi kebutuhan organisasi, salah satunya sebagai media dalam menampilkan filter data berdasarkan sekolah asal sebagai informasi penting dalam mendukung program sosialisasi kampus.

2. MATERIALS AND METHODS

2.1 Desain Penelitian

Penelitian ini menggunakan metode eksperimental dengan pendekatan development research. Tahapan penelitian mengikuti model Rapid Application Development (RAD) yang terdiri dari: requirement planning, design workshop, dan implementation.

2.2 Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian ini adalah:

Hardware:

1. Laptop dengan spesifikasi: Intel Core i5, RAM 8GB, Storage 256GB SSD
2. Smartphone Android versi 8.0 (Oreo) atau lebih tinggi untuk testing

Software:

1. Platform Kodular Creator (web-based)
2. Google Firebase Console
3. Web Browser (Google Chrome)
4. Firebase Realtime Database
5. Kodular Companion (untuk testing di smartphone)

Data:

Data mahasiswa sample dengan field: NIM, nama mahasiswa, angkatan, asal sekolah, nomor hp, nomor hp orang tua.

2.3 Prosedur Penelitian

Prosedur penelitian dilakukan dengan tahapan sebagai berikut:

2.3.1 Tahap Persiapan

1. Membuat akun Firebase dan membuat project baru
2. Mengaktifkan Firebase Realtime Database
3. Mengkonfigurasi Firebase security rules
4. Membuat akun Kodular dan memulai project baru

2.3.2 Tahap Design Interface

1. Merancang user interface aplikasi di Kodular
2. Menambahkan komponen-komponen UI: TextBox, Button, ListView
3. Mengatur layout dan styling aplikasi

2.3.3 Tahap Implementasi Firebase

1. Menambahkan Firebase Database component di Kodular
2. Mengkonfigurasi Firebase URL dan Project ID
3. Mengimplementasikan fungsi-fungsi CRUD:
 1. Create: Menambah data mahasiswa baru
 2. Read: Menampilkan daftar mahasiswa
 3. Update: Mengubah data mahasiswa
 4. Delete: Menghapus data mahasiswa

2.3.4 Tahap Testing

1. Testing fungsionalitas CRUD
2. Testing sinkronisasi real-time
3. Pengukuran response time

2.3.5 Tahap Evaluasi

1. Analisis hasil testing
2. Evaluasi performa sistem

2.4 Variabel Penelitian

Variabel yang diukur dalam penelitian ini:

1. Variabel Input: Data mahasiswa (NIM, nama mahasiswa, angkatan, asal sekolah, nomor hp, nomor hp orang tua.)
2. Variabel Proses: Operasi CRUD, Koneksi Firebase, Sinkronisasi data
3. Variabel Output: Response time, Status keberhasilan operasi, Data yang tersimpan di Firebase

2.5 Teknik Analisis Data

Analisis data dilakukan dengan:

1. Observasi langsung terhadap fungsionalitas aplikasi
2. Pengukuran response time untuk setiap operasi CRUD
3. Analisis deskriptif terhadap hasil implementasi

3. RESULTS AND DISCUSSION

3.1 Konfigurasi Firebase Realtime Database

3.1.1 Setup Firebase Project

Langkah pertama adalah membuat project di Firebase Console dengan nama "ProdiBD". Setelah project dibuat, Firebase Realtime Database diaktifkan dengan memilih menu "Realtime Database" dan klik "Create Database". Database diinisialisasi dengan mode "test mode" untuk development.

Struktur Database di Firebase:

NIM: ["NIM", "nama mahasiswa", "angkatan", "asal sekolah", "nomor hp", "nomor hp orang tua"]

Firebase Database Rules:

```
{
  "rules": {
    ".read": true,
    ".write": true
  }
}
```

3.1.2 Integrasi Firebase dengan Kodular

Di Kodular Creator, komponen Firebase Database ditambahkan dari kategori "Google" → "Firebase Database". Konfigurasi yang dilakukan: Firebase Component Properties:

1. Firebase URL: <https://crudbd1-default-rtdb.firebaseio.com/>
2. Project Bucket: ProdiBD
3. Firebase Token: (diambil dari Firebase Console)
AlzaSyDi8zRTnYP9a8VsHgyrDLd7PFj_FlpB5Iw

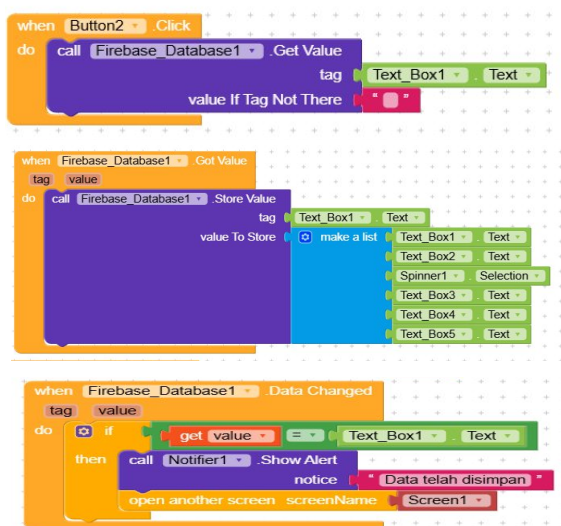
3.2 Desain dan Implementasi



Gambar 1. Tampilan menu utama



Gambar 2. Tampilan isian data

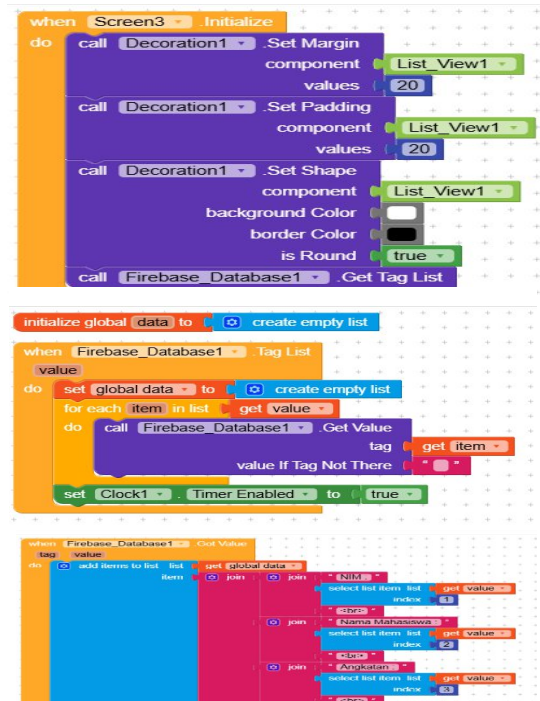


Gambar 3. blok programming input data

Setelah menambahkan data mahasiswa dengan NIM "0005", Nama "Rina", Angkatan "2025", Asal sekolah "Sma5", Nomor HP "0821474781", Nomor HP orang tua "0821474782". data tersimpan di Firebase dengan struktur:



Gambar 4. hasil inputan data di firebase



Gambar 5. blok programming tampilan data

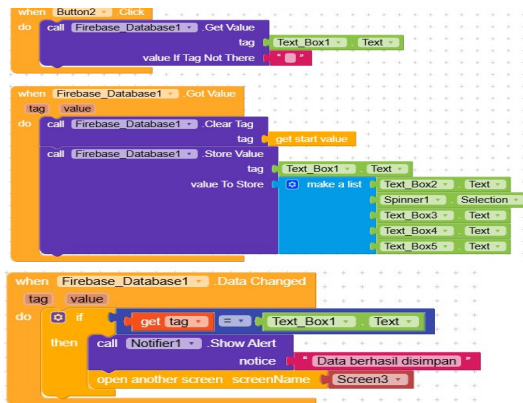


Gambar 6. Hasil tampil data dari firebase



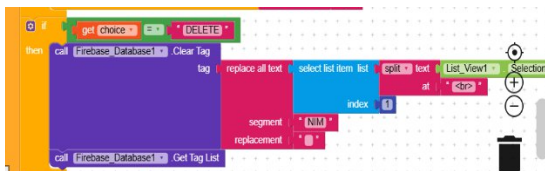
Gambar 7. Hasil pemilihan data

Pemilihan data pada list akan menampilkan rincian data dan menu pilihan edit dan delete. Jika pemilihan edit maka akan ditampilkan screen edit data dan proses update data di firebase



Gambar 8. blok programming penyimpanan hasil update data ke firebase

Jika pemilihan delete maka data akan di hapus dari daftar data firebase



Gambar 9. blok programming hapus data dari firebase

3.3 Analisis Performa

3.3.1 Response Time

Pengukuran response time dilakukan untuk setiap operasi CRUD dengan 10 kali percobaan:

Operasi	Rata-rata Response Time	Min	Max
CREATE	1.2 detik	0.9s	1.5s
READ	0.8 detik	0.6s	1.1s
UPDATE	1.1 detik	0.8s	1.4s
DELETE	1.0 detik	0.7s	1.3s

Analisis:

1. Response time paling cepat adalah operasi READ (0.8s) karena hanya mengambil data tanpa validasi
2. CREATE memiliki response time paling lama (1.2s) karena melibatkan proses validasi dan penulisan data baru

3. Semua operasi memiliki response time < 2 detik, menunjukkan performa yang baik untuk aplikasi mobile

3.3.2 Data Consistency

Testing konsistensi data dilakukan dengan:

1. Menambahkan 20 data mahasiswa secara berurutan
2. Memverifikasi semua data tersimpan dengan benar di Firebase
3. Menampilkan data di aplikasi dan membandingkan dengan data di Firebase Console

Hasil: 100% data consistency - semua data yang dikirim tersimpan dengan benar tanpa ada data loss atau corruption.

4. CONCLUSION

Hasil implementasi menunjukkan bahwa integrasi Firebase Realtime Database dengan platform Kodular berjalan dengan baik. Beberapa poin penting dalam kesimpulan:

4.1 Kemudahan Implementasi

Platform Kodular dengan visual blocks programming sangat mempermudah implementasi Firebase. Tanpa perlu menulis kode Java atau Kotlin, developer dapat membuat aplikasi yang fully functional [6].

4.2 Performa Real-time Synchronization

Response time rata-rata 1.2 detik untuk operasi CRUD menunjukkan performa yang baik.

4.3 Struktur Data JSON

Firebase menggunakan struktur NoSQL berbasis JSON yang lebih fleksibel dibandingkan SQL tradisional. Dalam implementasi ini, struktur hierarchical dengan "mahasiswa" sebagai parent node dan NIM sebagai child key memudahkan operasi CRUD dan query data.

4.4 Keamanan Data

Implementasi Firebase Database Rules penting untuk mengamankan data. Dalam penelitian ini, rules dibuat dalam "test mode" untuk development. Untuk production, rules perlu dikonfigurasi lebih ketat dengan autentikasi user dan validasi data yang lebih comprehensive.

4.5 Keterbatasan Platform

Meskipun Kodular mempermudah development, terdapat beberapa keterbatasan:

1. Customization:
Terbatas pada komponen yang disediakan oleh Kodular
2. Complex Logic:

Untuk logika yang sangat kompleks, blocks programming bisa menjadi kurang efisien

3. Performance:

Aplikasi yang dihasilkan mungkin tidak seoptimal aplikasi native yang dikembangkan dengan Android Studio

4. Platform Lock-in:

Aplikasi terikat pada ekosistem Kodular

Namun, untuk prototyping dan aplikasi dengan kompleksitas low to medium, kombinasi Kodular dan Firebase merupakan solusi yang sangat efektif. Selain itu untuk sistem informasi skala kecil hingga menengah seperti aplikasi manajemen data mahasiswa, Kodular menawarkan time-to-market yang jauh lebih cepat dengan hasil yang acceptable.

ACKNOWLEDGEMENTS

Penulis mengucapkan terima kasih kepada:

1. Rektor ITB - Stikom Ambon yang telah menyediakan fasilitas dan dukungan untuk penelitian ini.
2. Rekan-rekan peneliti yang telah memberikan support dan diskusi konstruktif.
3. Semua pihak yang telah membantu dalam penyelesaian penelitian ini

REFERENCES

- [1] I.K.G. Sudiarta, I.N.E. Indrayana, and I.W. Suasawa, "Membangun Struktur Realtime Database Firebase", Jurnal Ilmu Komputer, VOL. XI, No. 2
- [2] R.A. Setyawan, "Penerapan Firebase Realtime Database Pada Aplikasi Catatan Harian Diabetes Melitus", FAHMA – Jurnal Informatika Komputer, Bisnis dan Manajemen, Vol.22, No.1, pp. 1-9, Januari. 2024.
- [3] M. Alda, E. D. Siregar, A. Sitepu, and A. G. Silalahi, "Perancangan Aplikasi Quiz Pemrograman berbasis Android Menggunakan Web Kodular", Remik: Riset dan E-Jurnal Manajemen Informatika Komputer, Vol. 9, No.1, Januari. 2025, doi: 10.33395/remik.v9i1.14436.
- [4] T. Kurniawan, Samsudin, and Triase "Implementasi Layanan Firebase pada Pengembangan Aplikasi Sewa Sarana Olahraga Berbasis Android", Jurnal Informatika Universitas Pamulang Penerbit: Program Studi Teknik Informatika Universitas Pamulang, Vol. 6, No.1,Maret. 2021, doi: 10.32493/informatika.v6i1.10270.
- [5] W.C. Purnama, F. Annas, H. A. Musril, and G. Darmawati, "Perancangan Media Pembelajaran Pai Berbasis Android Menggunakan Kodular Kelas X Di Sma N 1 Iv Koto", JATI (Jurnal Mahasiswa Teknik Informatika), Vol. 7, No.2, April. 2023.
- [6] M. Alda, B. S. Wanandi, H. Bancin, and M. A. Panjaitan, "Implementasi Aplikasi Pencatatan Data Magang Mahasiswa Berbasis Mobile Menggunakan Kodular Menggunakan Metode Waterfall", Bulletin Of Computer Science Research, Vol. 4, No.1, pp.34-39, Desember. 2023, doi:10.47065/bulletincsr.v4i1.317