

IMPLEMENTASI FLUTTERFLOW DAN FIREBASE FIRESTORE DALAM PENGEMBANGAN APLIKASI MANAJEMEN KEUANGAN BERBASIS NO-CODE

Kristin Impana Manik, Rizky Wahyudi, Fadilla Amanah, Muhammad Yazid Noor, Adidtya Perdana

Ilmu Komputer, Universitas Negeri Medan
Jl. William Iskandar Ps. V, Kenangan Baru, Kec. Percut Sei Tuan,
Kabupaten Deli Serdang, Sumatera Utara, Indonesia
kristinmanik.4233250033@mhs.unimed.ac.id

ABSTRAK

Penelitian ini bertujuan untuk mengevaluasi penggunaan platform no-code dalam membantu individu, terutama generasi Z, untuk mengelola keuangan mereka. Permasalahan utama yang menjadi fokus adalah rendahnya literasi keuangan dan perilaku konsumtif di kalangan generasi Z, yang diperparah dengan keterbatasan akses terhadap aplikasi manajemen keuangan yang sesuai dengan kebutuhan mereka. Hal ini menyebabkan kesulitan dalam pengelolaan keuangan pribadi dan berdampak pada kesejahteraan finansial jangka panjang. Metode yang digunakan dalam penelitian ini adalah Rapid Application Development (RAD), yang memungkinkan pengembangan aplikasi secara cepat melalui iterasi desain dan pengujian. FlutterFlow digunakan sebagai alat utama untuk merancang antarmuka dan logika aplikasi tanpa perlu menulis kode, sementara Firebase Firestore digunakan untuk penyimpanan data secara real-time. Hasil pengujian melalui metode Blackbox Testing menunjukkan bahwa aplikasi yang dikembangkan memenuhi fungsi yang diharapkan dengan tingkat keberhasilan 100% dari 9 fitur yang diuji. Pengujian membuktikan bahwa implementasi no-code berhasil menghasilkan aplikasi yang fungsional, responsif, dan dapat diakses oleh pengguna tanpa latar belakang teknis, sehingga berpotensi meningkatkan literasi keuangan dan keterampilan pengelolaan keuangan pada generasi Z.

Kata kunci : *FlutterFlow, Firebase Firestore, No-Code, Pengembangan Aplikasi, Manajemen Keuangan*

1. PENDAHULUAN

Kemampuan mengelola keuangan di era digital ini menjadi semakin krusial terutama bagi generasi Z yang tumbuh besar di tengah kemajuan teknologi. Namun permasalahan pengelolaan keuangan seringkali menjadi tantangan yang dipicu oleh kurangnya literasi keuangan serta perilaku konsumtif [1].

Data dari Otoritas Jasa Keuangan (OJK) menunjukkan bahwa tingkat literasi keuangan di Indonesia masih tergolong rendah, terutama di kalangan generasi muda. Hal ini berdampak pada kurangnya kemampuan untuk membuat keputusan keuangan yang tepat dan mengakibatkan masalah finansial jangka panjang seperti tingginya tingkat utang, ketidakmampuan menabung, dan ketidaksiapan menghadapi keadaan darurat finansial.

Permasalahan utama yang menjadi akar penelitian ini adalah kesenjangan antara kebutuhan masyarakat khususnya generasi Z akan aplikasi manajemen keuangan yang intuitif dan keterbatasan kemampuan teknis mereka untuk mengembangkan aplikasi tersebut. Banyak pengguna memiliki ide-ide inovatif tetapi terkendala oleh kurangnya pengetahuan pemrograman. Fenomena ini memunculkan kebutuhan akan solusi yang mudah diakses dan dipahami, seperti aplikasi manajemen keuangan berbasis no-code. Platform seperti FlutterFlow dengan kemudahan penggunaannya menawarkan potensi besar dalam membantu individu mengelola keuangan mereka. Integrasi dengan Firebase Firestore memungkinkan penyimpanan data yang aman dan real-time.

Pentingnya penelitian ini terletak pada kemampuan teknologi no-code untuk memberdayakan lebih banyak orang dalam menciptakan solusi digital tanpa memerlukan keahlian teknis yang kompleks. Menurut Ergin (2023), pengembangan aplikasi kompleks melalui pendekatan no-code dapat dilakukan secara intuitif. Hal ini menunjukkan bagaimana platform no-code dapat merevolusi pendidikan dan memberdayakan lebih banyak orang untuk menjadi pengembang perangkat lunak [2].

Menurut Ilhamsyah & Fathurrahman (2024), Firebase Firestore telah terbukti menjadi alat yang andal dalam mengelola data secara real-time pada perancangan suatu aplikasi [3].

Penelitian ini bertujuan untuk mengevaluasi implementasi FlutterFlow dan Firebase Firestore dalam pengembangan aplikasi manajemen keuangan berbasis no-code. Dengan pendekatan ini diharapkan dapat ditemukan model pengembangan aplikasi yang tidak hanya efisien tetapi juga mampu memenuhi kebutuhan spesifik pengguna. Penelitian ini juga diharapkan dapat memberikan kontribusi terhadap literatur terkait teknologi no-code serta menjadi panduan praktis bagi pengembang maupun individu non-teknis dalam memanfaatkan FlutterFlow dan Firebase Firestore untuk menciptakan solusi digital yang relevan. Hal ini sejalan dengan temuan Rudyanto & Basri (2024) yang menyatakan bahwa penggunaan platform no-code memungkinkan pembuatan aplikasi yang fungsional tanpa memerlukan kemampuan pemrograman khusus [4].

2. TINJAUAN PUSTAKA

2.1. Manajemen Keuangan

Manajemen keuangan merupakan perencanaan, mengorganisir, mengatur, dan mengawasi sumber daya keuangan organisasi untuk mencapai tujuan yang telah ditetapkan. Teknik manajemen yang efektif sangat penting untuk menjaga stabilitas operasional dan memfasilitasi pengambilan keputusan yang tepat [5].

Manajemen keuangan yang efektif dan efisien mencakup perencanaan keuangan, pengendalian pengeluaran, dan pengambilan keputusan terkait keuangan [6].

Perilaku manajemen keuangan merupakan aspek krusial dalam ilmu keuangan yang mencakup tanggung jawab individu dalam mengatur keuangan dan aset secara efektif [7].

Teknologi semakin berperan dalam manajemen keuangan di era digital. Digitalisasi membantu mencatat transaksi dengan lebih efektif dan meminimalkan kesalahan. Laporan keuangan juga menjadi lebih akurat. Selain pencatatan, manajemen keuangan yang baik memerlukan analisis serta perencanaan berbasis data [5].

Manajemen keuangan digital menawarkan berbagai manfaat, termasuk kemudahan akses, efisiensi, dan otomatisasi proses keuangan. Individu dan bisnis dapat melacak transaksi serta mengelola anggaran lebih baik melalui aplikasi digital [8].

2.2. No-Code Development

No-code development memungkinkan konsumen untuk membuat aplikasi perangkat lunak tanpa memerlukan keahlian pemrograman. Platform tanpa kode memungkinkan pengguna, termasuk yang non-teknis, membuat aplikasi digital melalui antarmuka grafis dengan fitur seret dan letakkan tanpa perlu menulis kode [9].

Terdapat dua jenis platform dalam kategori ini, yaitu Platform Pengembangan No-Code (NCDP) dan Platform Pengembangan Low-Code (LCDP)[10]. NCDP dirancang untuk menyederhanakan dan mempercepat pengembangan aplikasi tanpa membutuhkan keahlian pemrograman khusus sehingga memungkinkan developer menghemat waktu secara lebih efisien [11].

Low-code dan no-code adalah platform pengembangan berbasis visual yang mempercepat pembuatan perangkat lunak atau situs web. Meski mengurangi waktu pengkodean, waktu untuk perencanaan, desain, dan pengujian tetap memerlukan durasi yang sama. Platform ini berasal dari alat pengembangan aplikasi cepat sebelumnya seperti MS Excel dan MS Access [12].

2.3. FlutterFlow

FlutterFlow merupakan alat pengembangan aplikasi visual yang digunakan untuk membuat aplikasi berbasis Flutter. Flutter sendiri merupakan SDK yang dikembangkan oleh Google untuk membangun aplikasi mobile [13].

FlutterFlow adalah platform pengembangan berbasis GUI untuk pembuatan aplikasi mobile berbasis Flutter. Alat ini memungkinkan pengembang membangun berbagai fitur aplikasi yang kompleks dengan cara visual [3].

Antarmuka drag-and-drop FlutterFlow mempercepat pengembangan tetapi memiliki kekurangan seperti akses kode sumber yang terbatas, kontrol kode, dan penyesuaian yang lebih terbatas. Karena keterbatasannya, FlutterFlow [14].

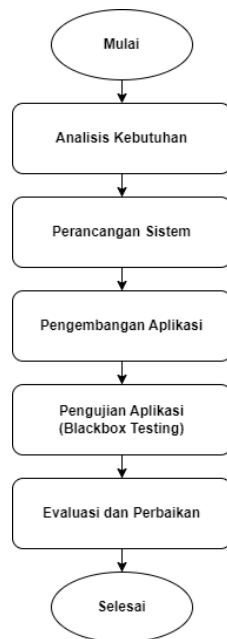
2.4. Firebase

Platform Firebase dirancang oleh Google untuk membantu para pengembang dalam membuat dan mengembangkan aplikasi. Di antara banyak layanan yang disediakan Firebase adalah backend sebagai layanan dan basis data real-time. Penyimpanan data real-time dan pembaruan data otomatis adalah fitur utama Firebase Realtime Databases (FRD). Bekerja secara offline dengan cache lokal dan memiliki sinkronisasi otomatis ketika kembali online adalah manfaat tambahan [15].

Firebase menawarkan berbagai alat untuk menganalisis data, melaporkan dan memperbaiki aplikasi, serta mendukung strategi pemasaran dan uji coba produk [16].

3. METODE PENELITIAN

Penelitian ini menggunakan metode Rapid Application Development (RAD) sebagai pendekatan pengembangan aplikasi. Metode ini dipilih karena memungkinkan pengembangan yang cepat melalui iterasi desain dan pengujian. Alat utama yang digunakan dalam penelitian ini adalah FlutterFlow, sebuah platform no-code yang memungkinkan pembuatan antarmuka dan logika aplikasi tanpa perlu menulis kode secara manual. Untuk pengelolaan basis data, penelitian ini memanfaatkan Firebase Firestore yaitu layanan penyimpanan berbasis cloud yang mendukung sinkronisasi data secara real-time. Pengujian dilakukan menggunakan metode Blackbox Testing untuk mengevaluasi fungsionalitas aplikasi berdasarkan skenario uji yang telah ditetapkan. Analisis data dilakukan secara deskriptif kualitatif, dengan menjelaskan bagaimana implementasi FlutterFlow dan Firebase Firestore dalam pengembangan aplikasi manajemen keuangan berbasis no-code serta efektivitas penggunaannya bagi individu tanpa latar belakang teknis. Adapun tahapan dari penelitian ini adalah sebagai berikut.



Gambar 1. Diagram alir penelitian

3.1. Analisis Kebutuhan

Tahapan analisis kebutuhan diawali dengan identifikasi fitur utama yang diperlukan dalam aplikasi manajemen keuangan berbasis no-code. Penelitian ini berfokus pada pengembangan aplikasi yang memungkinkan pengguna mencatat pemasukan dan pengeluaran, melihat riwayat transaksi, serta mendapatkan ringkasan laporan keuangan dalam bentuk grafik atau tabel. Selain itu, dipertimbangkan pula kebutuhan integrasi dengan Firebase Firestore untuk menyimpan data pengguna secara real-time.

3.2. Perancangan Sistem

Setelah kebutuhan sistem diidentifikasi, tahap selanjutnya adalah perancangan sistem yang mencakup desain antarmuka pengguna dan model basis data. Desain awal dibuat menggunakan Canva, yang kemudian direplikasi ke FlutterFlow untuk memastikan kesesuaian tampilan dan fungsionalitas. Struktur basis data di Firebase Firestore dirancang dengan model koleksi dan dokumen, di mana setiap pengguna memiliki koleksi tersendiri untuk menyimpan catatan transaksi mereka. Selain itu, aturan keamanan Firebase Firestore dikonfigurasi untuk memastikan bahwa setiap pengguna hanya dapat mengakses datanya sendiri.

3.3. Pengembangan Aplikasi

Proses pengembangan Implementasi dimulai dengan pembuatan halaman utama, formulir pencatatan transaksi, serta halaman ringkasan laporan keuangan. Setiap komponen diuji secara bertahap untuk memastikan interaksi yang sesuai antara pengguna dan sistem. Firebase Firestore digunakan untuk menyimpan setiap transaksi secara real-time, sehingga pengguna dapat mengakses dan memperbarui data kapan saja. Selain itu, jika

diperlukan, fitur autentikasi Firebase dapat diterapkan untuk membatasi akses hanya kepada pengguna yang terdaftar.

3.4. Pengujian Aplikasi

Pengujian dilakukan dengan metode Blackbox Testing untuk memastikan bahwa setiap fitur dalam aplikasi berfungsi sesuai dengan spesifikasi. Pengujian mencakup validasi input transaksi, tampilan riwayat keuangan, serta akurasi data yang tersimpan di Firebase Firestore. Skenario uji dirancang untuk mengidentifikasi potensi kesalahan, seperti data yang tidak tersimpan dengan benar atau tampilan yang tidak sesuai dengan desain awal. Jika ditemukan masalah dalam proses pengujian, perbaikan dilakukan sebelum aplikasi diuji ulang untuk memastikan stabilitas dan kehandalan sistem.

3.5. Evaluasi dan Penyempurnaan

Tahapan Setelah pengujian selesai, tahap evaluasi dilakukan dengan menilai efektivitas FlutterFlow dalam membantu individu tanpa latar belakang teknis mengembangkan aplikasi manajemen keuangan. Evaluasi ini mencakup kemudahan dalam membangun antarmuka, fleksibilitas integrasi dengan Firebase Firestore, serta performa aplikasi dalam menangani transaksi pengguna. Jika ditemukan kekurangan, dilakukan penyempurnaan berdasarkan hasil analisis dan umpan balik dari pengguna uji coba. Dengan pendekatan ini, penelitian bertujuan untuk memberikan gambaran menyeluruh tentang bagaimana FlutterFlow dan Firebase Firestore dapat dimanfaatkan secara optimal dalam pengembangan aplikasi berbasis no-code.

4. HASIL DAN PEMBAHASAN

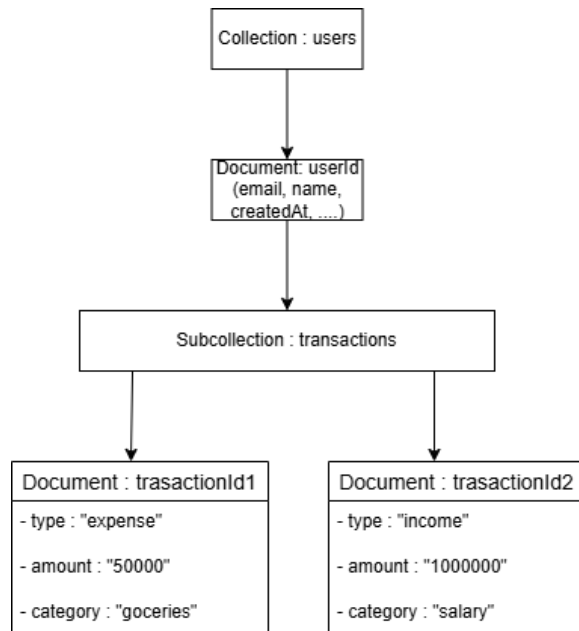
4.1. Analisis Kebutuhan

Pada tahap analisis kebutuhan, dilakukan identifikasi fitur utama yang harus dimiliki oleh aplikasi manajemen keuangan berbasis no-code yang dikembangkan menggunakan FlutterFlow dan Firebase Firestore. Beberapa fitur utama yang harus tersedia dalam aplikasi ini antara lain pencatatan pemasukan dan pengeluaran, pembuatan kategori transaksi, serta visualisasi data keuangan dalam bentuk grafik. Selain fitur utama, kebutuhan teknis juga menjadi perhatian, khususnya integrasi Firebase sebagai basis data yang memungkinkan penyimpanan dan sinkronisasi data pengguna secara real-time. Identifikasi kebutuhan ini menjadi dasar dalam perancangan dan pengembangan aplikasi agar dapat memenuhi kebutuhan pengguna secara optimal.

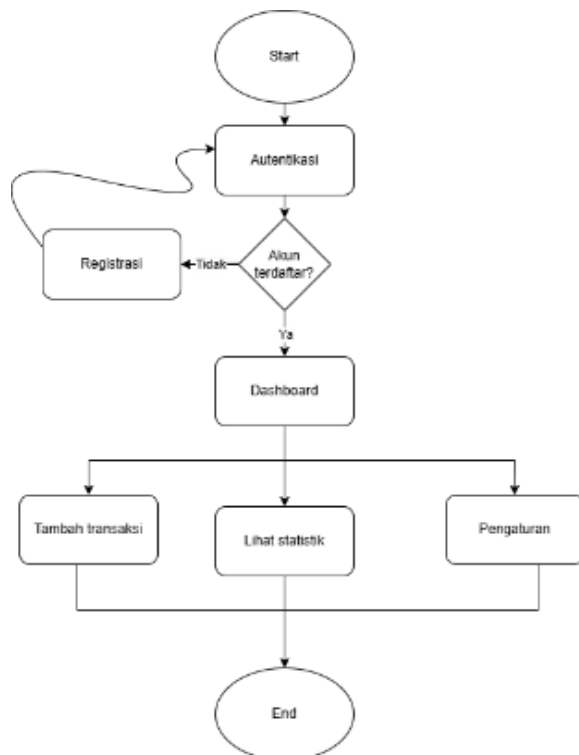
4.2. Perancangan Sistem

Setelah kebutuhan sistem teridentifikasi, tahap selanjutnya adalah perancangan sistem yang mencakup desain antarmuka pengguna dan struktur basis data. Desain antarmuka dibuat menggunakan FlutterFlow dengan pendekatan yang berfokus pada pengalaman pengguna (User Experience/UX) agar

mudah diakses. Sementara itu, perancangan basis data dilakukan dengan menggunakan Firebase Firestore, yang diatur dalam bentuk koleksi dan dokumen untuk menyimpan data transaksi pengguna. Skema basis data dirancang sedemikian rupa agar memungkinkan akses yang cepat serta sinkronisasi data secara real-time, sehingga pengguna dapat melihat perubahan data keuangan mereka secara instan.



Gambar 2. Struktur basis data *Firebase Firestore*

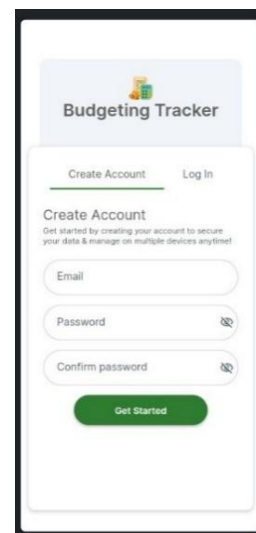


Gambar 3. Diagram alur aplikasi manajemen keuangan

Pada Gambar 2, struktur basis data Firebase Firestore dirancang dengan struktur hierarkis di mana koleksi "users" menyimpan data pengguna, dengan setiap dokumen pengguna memiliki koleksi bersarang "transactions" untuk menyimpan catatan transaksi individu. Setiap transaksi mencakup atribut seperti jenis (pemasukan/pengeluaran), jumlah, kategori, tanggal, dan deskripsi. Model ini memungkinkan query yang efisien dan memastikan setiap pengguna hanya dapat mengakses data mereka sendiri.

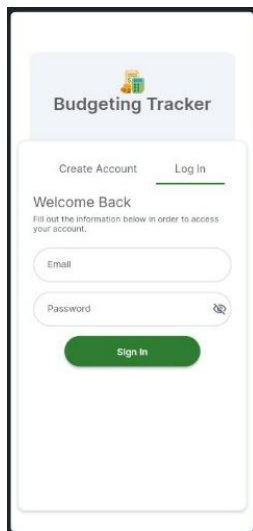
Gambar 3 menampilkan alur kerja aplikasi manajemen keuangan yang dikembangkan. Alur dimulai dengan autentikasi pengguna (login/registrasi), kemudian pengguna diarahkan ke dashboard utama. Dari dashboard, pengguna dapat memilih untuk menambah transaksi baru, melihat statistik keuangan, atau mengatur akun. Alur ini dirancang untuk memastikan navigasi yang intuitif dan pengalaman pengguna yang mulus.

4.3. Pengembangan Aplikasi



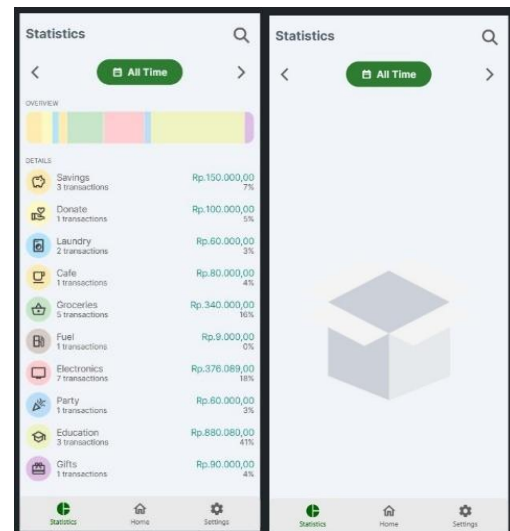
Gambar 4. Tampilan *create account*

Gambar di atas merupakan halaman "Create Account" dari aplikasi Budgeting Tracker. Pengguna dapat membuat akun baru dengan mengisi kolom email, kata sandi, dan konfirmasi kata sandi. Selain itu, kata sandi dapat ditampilkan atau disembunyikan dengan menggunakan ikon mata pada kolom kata sandi. Pengguna dapat menyelesaikan prosedur pendaftaran dengan mengklik tombol hijau "Get Started" setelah semua informasi dimasukkan.



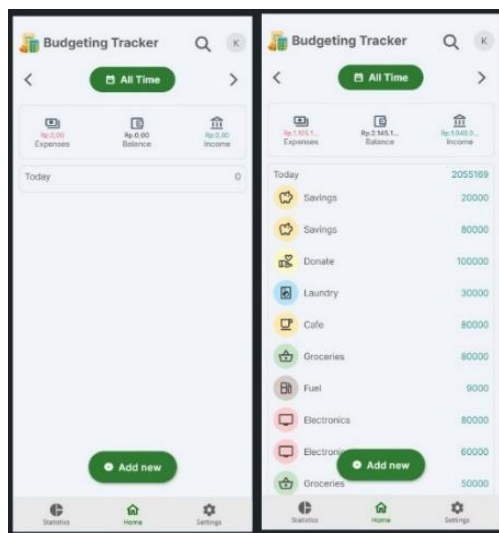
Gambar 5. Tampilan Login

Gambar 5 merupakan tampilan dari tab login, dimana pengguna yang sudah pernah mendaftar atau melakukan register pada website ini dapat memilih menu 'log in' seperti gambar di atas. Menu login ini digunakan oleh pengguna untuk mengakses akun mereka dengan memasukkan email dan password pada kolom yang tersedia. Selain itu, kata sandi dapat ditampilkan atau disembunyikan dengan menggunakan ikon mata pada kolom kata sandi. Setelah informasi dimasukkan secara akurat, pengguna dapat masuk ke akun mereka dengan mengklik tombol hijau "Sign in".



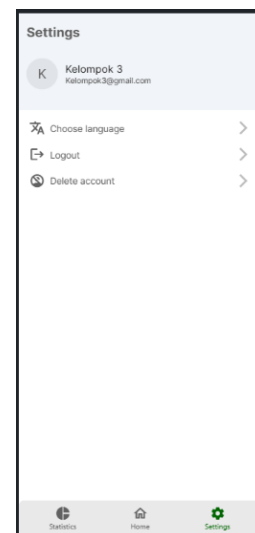
Gambar 7. Tampilan tab Statistics

Gambar di atas menampilkan tab "Statistik" pada aplikasi Budgeting Tracker. Halaman statistik masih kosong di tampilan kiri, yang menggambarkan kondisi awal sebelum ada catatan keuangan. Setelah beberapa data keuangan dimasukkan, statistik akan ditampilkan pada tampilan sebelah kanan. Angka-angka ini mencakup berbagai kategori pengeluaran, termasuk groceries(belanjaan), laundry(cucian), donasi, tabungan, dan banyak lagi yang disertai dengan jumlah transaksi dan jumlah total yang dibelanjakan. Selain itu, distribusi anggaran pengguna dirangkum secara visual oleh tampilan grafik di bagian atas.



Gambar 6. Tampilan tab Home

Gambar di atas menunjukkan tab "home" dimana gambar sebelah kiri menunjukkan tampilan dimana saldo, pengeluaran dan pemasukannya masih Rp 0.00 yang menggambarkan kondisi awal sebelum ada catatan keuangan. Sementara itu tampilan sebelah kanan menunjukkan daftar saldo, pengeluaran dan pemasukan setelah beberapa data dimasukan. Pada menu home juga terdapat tombol "add new" untuk menambahkan catatan keuangan.



Gambar 8. Tampilan menu settings

Gambar diatas merupakan tampilan dari "Settings" pada aplikasi Budgeting Tracker. Pengguna dapat melihat detail akun, termasuk nama dan alamat email pada halaman ini. Ada juga sejumlah opsi pengaturan, termasuk "Choose language" untuk mengubah bahasa, 'Logout' untuk keluar dari akun, dan "Delete account" untuk menghapus akun.

4.4. Pengujian Aplikasi

Pengujian dilakukan menggunakan metode Black Box Testing untuk mengevaluasi fungsi utama aplikasi yang dikembangkan dengan FlutterFlow. Metode ini berfokus pada validasi keluaran berdasarkan berbagai skenario tanpa melihat struktur internal kode. Pengujian mencakup beberapa fitur penting, seperti pembuatan akun, autentikasi pengguna

(login dan reset password), pengelolaan data keuangan (pemasukan dan pengeluaran), serta fitur tambahan seperti menampilkan statistik dan pengaturan akun (logout dan penghapusan akun). Setiap skenario diuji dengan memasukkan data yang relevan dan mengevaluasi apakah hasil yang diperoleh sesuai dengan ekspektasi. Hasil pengujian disajikan dalam Tabel 1 berikut:

Tabel 1. Hasil *Blackbox Testing*

No	Fitur	Skenario Pengujian	Ekspektasi Output	Hasil	Keterangan
1	Pembuatan Akun	Pengguna mengisi formulir pendaftaran	Akun berhasil dibuat dan masuk ke dashboard	✓	Berhasil sesuai ekspektasi
2	Login	Pengguna memasukkan kredensial yang benar	Berhasil login ke halaman utama	✓	Fitur berjalan dengan baik
3	Login (Invalid Credentials)	Pengguna memasukkan email/password salah	Pesan error "Email atau password salah"	✓	Berhasil sesuai ekspektasi
4	Membuat Daftar Pengeluaran	Pengguna menambahkan pengeluaran baru	Data tersimpan dan tampil di daftar	✓	Fitur berjalan dengan baik
5	Membuat Daftar Pemasukkan	Pengguna menambahkan pemasukkan baru	Data tersimpan dan tampil di daftar	✓	Berhasil sesuai ekspektasi
6	Menampilkan Statistik	Pengguna melihat statistik pengeluaran	Statistik ditampilkan dengan benar	✓	Fitur berjalan dengan baik
7	Reset Password	Pengguna melakukan reset password	Link reset dikirim ke email	✓	Berhasil sesuai ekspektasi
8	Logout	Pengguna melakukan logout	Kembali ke halaman login	✓	Fitur berjalan dengan baik
9	Delete Account	Pengguna menghapus akun	Akun terhapus dan tidak bisa login lagi	✓	Berhasil sesuai ekspektasi

Setiap fitur dari aplikasi manajemen keuangan yang dibuat menggunakan FlutterFlow dan Firebase Firestore telah melalui pengujian yang sukses dengan tingkat keberhasilan 100%, sesuai dengan hasil pengujian Black Box Testing yang ditunjukkan pada Tabel 1. Ada 9 macam skenario yang berbeda yang disertakan dalam pengujian, mulai dari kemampuan yang sederhana seperti login dan registrasi hingga kemampuan yang lebih canggih seperti manajemen transaksi dan penyajian data statistik. Setiap fitur bekerja seperti yang diharapkan, menunjukkan bahwa metode no-code yang dikombinasikan dengan FlutterFlow dan Firebase dapat membuat aplikasi yang dapat diandalkan dan berguna tanpa perlu menulis kode secara manual.

4.5. Evaluasi dan Penyempurnaan

Setelah pengujian dilakukan, evaluasi efektivitas FlutterFlow dalam mengembangkan aplikasi manajemen keuangan dilakukan dengan mempertimbangkan berbagai aspek, seperti kemudahan dalam membangun antarmuka, fleksibilitas integrasi dengan Firebase Firestore, serta performa aplikasi dalam menangani transaksi pengguna. Hasil pengujian menunjukkan bahwa FlutterFlow mempermudah proses pengembangan bagi individu tanpa latar belakang teknis, terutama dengan fitur drag-and-drop yang intuitif. Selain itu, integrasi dengan Firebase Firestore berjalan dengan baik dalam menyimpan serta menampilkan data pemasukan dan pengeluaran secara real-time. Namun, terdapat beberapa kendala, seperti keterbatasan dalam

kustomisasi logika bisnis yang lebih kompleks serta keterlambatan sinkronisasi data pada beberapa skenario uji coba.

Untuk mengatasi kekurangan yang ditemukan, dilakukan beberapa penyempurnaan, seperti optimasi pengaturan database agar proses pembaruan data menjadi lebih responsif dan penyesuaian desain antarmuka untuk meningkatkan pengalaman pengguna. Selain itu, fitur tambahan, seperti konfirmasi sebelum menghapus akun, ditambahkan untuk meningkatkan keamanan aplikasi. Dengan pendekatan ini, penelitian ini memberikan gambaran menyeluruh tentang bagaimana FlutterFlow dan Firebase Firestore dapat dimanfaatkan dalam pengembangan aplikasi berbasis no-code, serta batasan yang perlu diperhatikan agar aplikasi dapat berfungsi dengan optimal.

5. KESIMPULAN DAN SARAN

Dengan tingkat keberhasilan 100% dari 9 fitur yang diuji menggunakan pendekatan Black Box Testing, menunjukkan bahwa FlutterFlow dan Firebase Firestore sangat efektif jika digunakan dalam pengembangan sistem manajemen keuangan tanpa kode. Sementara integrasi dengan Firebase Firestore memastikan penyimpanan dan sinkronisasi data keuangan secara real-time, platform FlutterFlow telah menunjukkan kemampuannya dalam memfasilitasi proses desain dan pengembangan aplikasi, memungkinkan individu tanpa latar belakang teknis untuk membuat aplikasi secara intuitif. Metode tanpa kode secara keseluruhan telah menghasilkan perangkat

lunak yang berfungsi untuk memenuhi kebutuhan manajemen keuangan Generasi Z, meskipun masih ada beberapa kesulitan dalam mengadaptasi logika bisnis yang kompleks. Disarankan agar pengembang yang menggunakan pendekatan RAD lebih fokus pada umpan balik dan iterasi pengguna, membuat dokumentasi yang lebih menyeluruh, dan menyelidiki potensi integrasi dengan teknologi lain yang dapat meningkatkan fungsionalitas aplikasi dan menyesuaikannya dengan kebutuhan khusus industri manajemen keuangan.

DAFTAR PUSTAKA

- [1] M. Rahmawati, Lathifaturahmah, Sheni, Aryanti, D. N. Tamara, and I. Schabudin, "PENGARUH MANAJEMEN KEUANGAN PADA GENERASI Z UNTUK KESEJAHTERAAN KEUANGAN JANGKA PANJANG," *Blantika Multidiscip. J.*, vol. 8, no. 1, pp. 2108–2121, 2025.
- [2] H. Ergin, "Developing a Twitter Clone by Using a No-Code Software Platform," *J. Comput. Sci. Coll.*, vol. 39, no. 4, pp. 100–101, 2023, doi: <https://dl.acm.org/doi/abs/10.5555/3637036.3637050>.
- [3] M. Ilhamsyah and M. Fathurrahman, "RANCANG BANGUN MYSMARTFISHFEEDING APLIKASI ANDROID MENGGUNAKAN FLUTTERFLOW DAN FIREBASE FIRESTORE," *JITET (Jurnal Inform. dan Tek. Elektro Ter.)*, vol. 12, no. 3, pp. 4114–4122, 2024.
- [4] A. R. Rudyanto and S. A. P. Basri, "Implementasi Sistem E-Report Berbasis Mobile pada Toko Pakaian Benang Ratu," *Pros. Semin. Nas. Sains Dan Teknol.*, vol. 14, no. 1, pp. 468–475, 2024, doi: <https://doi.org/http://dx.doi.org/10.36499/psnst.v14i1.12281>.
- [5] E. Suprihatin and N. Maudita, "Aplikasi pengelolaan penerimaan keuangan menggunakan appsheet berbasis android," *J. Komput. Bisnis*, vol. 17, no. 2, pp. 1–6, 2025.
- [6] D. P. Ompusunggu and N. Irenetia, "Pentingnya Manajemen Keuangan Bagi Perusahaan," *CEMERLANG J. Manaj. dan Ekon. Bisnis*, vol. 3, no. 2, pp. 140–147, 2023.
- [7] M. A. Handayani, C. Amalia, and T. D. R. Sari, "Pengaruh Pengetahuan Keuangan, Sikap Keuangan dan Kepribadian Terhadap Perilaku Manajemen Keuangan (Studi Kasus Pada Pelaku UMKM Batik di Lampung)," *J. Ekombis Rev. – J. Ilm. Ekon. dan Bisnis*, vol. 10, no. 2, pp. 647 – 660, 2022.
- [8] S. Yolanda, S. Shaddiq, H. Faisal, and I. Kurnianti, "PERAN MANAJEMEN KEUANGAN DIGITAL DALAM," *Indones. Red Crescent Humanit. J.*, vol. 2, no. 1, pp. 23–32, 2023, doi: 10.56744/irchum.v1i2.31.
- [9] R. Soman, "EMPOWERING SMES THROUGH NO-CODE DEVELOPMENT : A TECHNICAL," *Int. J. Res. Comput. Appl. Inf. Technol.*, vol. 8, no. 1, pp. 2108–2121, 2025.
- [10] A. Farisi, D. Novita, and F. P. Sihotang, "Pembuatan Aplikasi Mobile Tanpa Coding Bagi Karyawan PT Dunia Kimia Utama," *J. Pengabd. Kpd. Masy. FORDICATE (INFORMATICS Eng. DEDICATION)*, vol. 2, no. 1, pp. 129–139, 2022.
- [11] A. Kholik, A. Soegiarto, and S. R. Putri, "Perancangan Aplikasi Sistem Peminjaman Laboratorium Berbasis No-Code Development Platforms (NCDPs) Menggunakan Glideapps," *JIM J. Ilm. Multidisiplin*, vol. 3, no. 1, pp. 285–295, 2024.
- [12] S. R. Hedau and P. O. Mandge, "LOW CODE & NO CODE SOFTWARE DEVELOPMENT OF THE FUTURE," no. 06, pp. 1131–1135, 2021.
- [13] M. Dzakiyyudin and V. Frendiana, "Seminar Nasional Teknik Elektro, Sistem Informasi, dan Teknik Informatika," *Pros. Semin. Nas. Tek. Elektro, Sist. Informasi, Dan Tek. Inform.*, vol. 1, no. 1, pp. 593–598, 2024, doi: <https://dl.acm.org/doi/abs/10.5555/3637036.3637050>.
- [14] Riki Chandra and Muhammad Rachmadi, "Perbandingan Efisiensi Antara Pengembangan Dengan Laravel Dan Flutterflow Pada Sistem Informasi Kepegawaian," *J. Sist. Inf. dan Ilmu Komput.*, vol. 2, no. 2, pp. 01–10, 2024, doi: 10.59581/jusiik-widyakarya.v2i2.2892.
- [15] T. Narwastu, B. T. Hanggara, and W. H. P. Nugraha, "Pengembangan Sistem Informasi Penjualan Motor berbasis Web menggunakan Teknologi Firebase (Studi Kasus : Dealer Motor ' Suzuki Katang '), " *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 11, pp. 4775–4782, 2021.
- [16] P. Chougale, V. Yadav, and A. Gaikwad, "Firebase - Overview and usage," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 3, no. 12, pp. 1178–1183, 2021.