



Rancang Bangun Aplikasi Desktop Pendeteksi Duplikat Data NIK dan NKK pada File Excel Menggunakan Metode *Exact Matching* Berbasis *Flutter*

Ardhi Feisal Wijayanto^{1*}, Rina Firliana², Dwi Harini³

¹⁻³ Sistem Informasi, Universitas Nusantara PGRI Kediri, Indonesia

*Penulis Korespondensi: ardhifeisalwijayanto@gmail.com

Abstract. The management of UMKM assistance recipient data at the Cooperatives and Micro Enterprises Office of Kediri Regency is still carried out manually using Microsoft Excel, resulting in frequent data duplication in the NIK and NKK columns. Excel's limitation of accurately processing only up to 15 digits is a serious obstacle because NIK consists of 16 digits, risking digit truncation that causes data inaccuracy and inaccurate aid distribution. This study aims to design and build a Flutter-based desktop application capable of detecting duplicate NIK or NKK data automatically, accurately, and informatively. The method used is Research and Development (R&D) with the Sugiyono development model simplified into eight stages, along with the Exact Matching algorithm based on HashMap which has $O(n)$ time complexity. System testing was conducted using the Black Box Testing method through ten test scenarios covering file validation, NIK/NKK duplicate detection, result export, and application reset. The results showed that the application successfully detected all duplicate data with 100% accuracy and was able to process Excel files containing 5,000 rows of data in less than 1 second, much faster than the manual method which takes 2 to 3 hours. All ten Black Box Testing scenarios were declared successful (100% pass rate). The implication of this research is the availability of an appropriate technical solution that can be directly used by staff to accelerate data validation, and can serve as a reference for developing similar systems in other government institutions facing similar data duplication problems.

Keywords: Duplicate Data; Exact Matching; Flutter; NIK; NKK.

Abstrak. Pengelolaan data penerima bantuan UMKM di Dinas Koperasi dan Usaha Mikro Kabupaten Kediri masih dilakukan secara manual menggunakan Microsoft Excel, sehingga sering terjadi duplikasi data pada kolom NIK dan NKK. Keterbatasan Excel yang hanya mampu memproses data numerik hingga 15 digit secara akurat menjadi kendala serius karena NIK terdiri dari 16 digit, sehingga berisiko terjadi pemotongan digit yang menyebabkan ketidakakuratan data dan ketidaktepatan sasaran bantuan. Penelitian ini bertujuan merancang dan membangun aplikasi desktop berbasis Flutter yang mampu mendeteksi data duplikat NIK atau NKK secara otomatis, akurat, dan informatif. Metode yang digunakan adalah Research and Development (R&D) dengan model pengembangan Sugiyono yang disederhanakan menjadi delapan tahapan, serta algoritma *Exact Matching* berbasis HashMap yang memiliki kompleksitas waktu $O(n)$. Pengujian sistem dilakukan dengan metode *Black Box Testing* melalui sepuluh skenario uji yang mencakup validasi file, deteksi duplikat NIK/NKK, ekspor hasil, dan reset aplikasi. Hasil penelitian menunjukkan bahwa aplikasi berhasil mendeteksi seluruh data duplikat dengan akurasi 100% dan mampu memproses file Excel berisi 5.000 baris data dalam waktu kurang dari 1 detik, jauh lebih cepat dibandingkan metode manual yang membutuhkan waktu 2 hingga 3 jam. Seluruh skenario pengujian black box dinyatakan berhasil (100% pass rate). Implikasi dari penelitian ini adalah tersedianya solusi teknis tepat guna yang dapat langsung digunakan oleh staf Dinas untuk mempercepat validasi data, serta dapat menjadi referensi bagi pengembangan sistem serupa di instansi pemerintah lain dengan permasalahan duplikasi data yang sejenis.

Kata kunci: Data Duplikat; Flutter; NIK; NKK; Pencocokan Tepat.

1. LATAR BELAKANG

Perkembangan teknologi informasi menuntut setiap instansi pemerintah untuk mengelola data secara akurat dan efisien, terutama data administrasi yang menjadi dasar pengambilan keputusan dan pelaksanaan program layanan publik (Arif Sidik. et al., 2022). Data kependudukan, khususnya Nomor Induk Kependudukan (NIK) dan Nomor Kartu

Keluarga (NKK), bersifat unik dan digunakan sebagai identitas utama dalam berbagai program pemerintah. Kualitas pengelolaan data ini sangat menentukan keakuratan informasi dan keadilan dalam pelaksanaan kebijakan, mengingat kesalahan data dapat berakibat fatal terhadap distribusi bantuan dan pelayanan publik (Muhammad et al., 2024). Oleh karena itu, setiap instansi yang mengelola data kependudukan dituntut untuk memiliki sistem yang mampu menjaga integritas dan konsistensi data secara berkelanjutan.

Dalam praktiknya, pengelolaan data kependudukan pada Dinas Koperasi dan Usaha Mikro Kabupaten Kediri (Diskopusmik) masih menghadapi permasalahan duplikasi data pada file *Excel* yang digunakan untuk pendataan penerima bantuan UMKM. Duplikasi NIK atau NKK terjadi akibat proses *input* yang berulang, penggabungan data dari berbagai sumber, dan belum adanya mekanisme validasi otomatis. Akibatnya, staf Diskopusmik harus menghabiskan waktu dua hingga tiga jam untuk memeriksa duplikat dalam file yang berisi lebih dari lima ribu baris data. Data ganda berpotensi menyebabkan ketidaktepatan sasaran bantuan, ketidakakuratan laporan administrasi, serta menurunnya akuntabilitas instansi (Nurdiansyah et al., 2022). Keterbatasan *Microsoft Excel* dalam memproses data numerik panjang seperti NIK 16 digit menjadi kendala teknis, karena *Excel* hanya akurat hingga 15 digit sehingga digit ke-16 berisiko berubah menjadi nol (Microsoft, 2024). Kondisi ini semakin memperparah permasalahan duplikasi karena data yang sebenarnya berbeda dapat terbaca sama, atau sebaliknya, data yang sama dapat terbaca berbeda akibat kesalahan format dan pembacaan digit (Pratama & Saparingga, 2021).

Duplikasi NIK dan NKK tidak hanya menjadi masalah teknis administratif, tetapi juga berdampak langsung pada kualitas data penerima bantuan UMKM. Dampak paling signifikan adalah terjadinya inkonsistensi data di mana satu individu dapat terdaftar dengan alamat, nama, atau informasi yang berbeda pada dua entri yang berbeda. Hal ini menyebabkan redundansi informasi yang memperbesar volume penyimpanan data tidak perlu dan menyulitkan proses analisis data agregat untuk perencanaan program bantuan. Selain itu, dalam konteks penyaluran bantuan UMKM, duplikasi NIK atau NKK menyebabkan risiko penyaluran bantuan ganda kepada satu penerima yang sama, sementara penerima lain yang berhak justru tidak mendapatkan bantuan karena datanya tidak terdaftar secara valid.

Penelitian terdahulu telah mengupayakan berbagai solusi, namun masing-masing memiliki keterbatasan. (Ismail Setiawan & Aisyah Mutia Dawis, 2023) memanfaatkan fitur bawaan *Excel* secara manual tanpa otomatisasi. (Hanifah et al., 2025) menghasilkan rekomendasi kebijakan tanpa implementasi teknis. (Mita et al., 2024) mengembangkan validasi NIK berbasis web yang bergantung pada internet. (Machfud, 2024) menggunakan *database*

server terpusat dengan konfigurasi kompleks. (Mayola et al., 2024) menerapkan *similarity* teks yang tidak sesuai untuk identifier numerik deterministik. (Andrea Liui & Putra Priyanto, 2025) berfokus pada keamanan data *post-publication*, belum menyentuh validasi integritas data pada fase *pre-use*. Perbedaan penelitian ini dengan penelitian sebelumnya adalah solusi *offline* berbasis algoritma deterministik untuk duplikasi identifier kependudukan, penanganan eksplisit keterbatasan presisi digit *spreadsheet* pada identifier 16 digit dan paradigma *in-memory processing* tanpa *database* eksternal.

Tujuan penelitian ini adalah merancang dan membangun aplikasi desktop berbasis *Flutter* yang mampu mendeteksi duplikat NIK atau NKK pada file *Excel* secara otomatis dan akurat menggunakan algoritma *exact matching* berbasis *HashMap* dengan kompleksitas $O(n)$. Pendekatan *Research and Development (R&D)* dipilih untuk memastikan bahwa sistem yang dikembangkan benar-benar sesuai dengan kebutuhan pengguna melalui tahapan analisis, perancangan, implementasi, dan pengujian dengan metode *Black Box Testing* (Efriani et al., 2025). Kontribusi utama penelitian ini adalah penyediaan solusi teknis konkret yang dapat langsung digunakan oleh staf Diskopusmik untuk mempercepat validasi data, meningkatkan akurasi deteksi, dan menghasilkan laporan terstruktur. Penelitian ini juga diharapkan dapat menjadi referensi bagi instansi pemerintah lain yang menghadapi permasalahan serupa.

2. KAJIAN TEORITIS

Data Ganda (Duplikasi Data)

Data ganda atau duplikasi data adalah kondisi di mana satu entitas data yang sama muncul lebih dari satu kali dalam satu kumpulan data, baik secara keseluruhan maupun pada kolom-kolom tertentu yang menjadi identitas unik (Wijaya, 2025). Duplikasi data dapat terjadi karena beberapa faktor, antara lain proses *input* data yang dilakukan berulang oleh operator yang berbeda, penggabungan data dari berbagai sumber tanpa normalisasi, belum adanya mekanisme validasi otomatis, serta kesalahan manusia dalam proses entri data (Maulidah et al., 2025). Dampak dari duplikasi data sangat signifikan terhadap kualitas pelayanan publik, yaitu menyebabkan ketidakakuratan laporan administrasi, ketidaktepatan sasaran bantuan, peningkatan beban kerja petugas, serta menurunnya tingkat kepercayaan masyarakat terhadap akuntabilitas instansi (Nurdiansyah et al., 2022).

Exact Matching

Exact matching merupakan metode pencocokan data yang membandingkan dua atau lebih nilai data secara utuh dan persis (*exact*) tanpa toleransi terhadap perbedaan sekecil apapun

(Gunawan & Aji Sudarsono, 2022). Keunggulan utama *Exact matching* dibandingkan metode pencocokan lainnya seperti *fuzzy matching* atau *probabilistic matching* adalah kesederhanaan implementasi, kepastian hasil (deterministik), serta efisiensi komputasi pada dataset berukuran besar (Anggia Utami & A. Saehan, 2022). Kompleksitas waktu algoritma *exact matching* dapat mencapai $O(n)$ dengan memanfaatkan struktur data *hash table* atau *set*, di mana n adalah jumlah baris data yang diproses (RIVAL FAKHRI AMRULLAH, 2023). Dengan kompleksitas linear tersebut, *exact matching* mampu memproses ribuan hingga puluhan ribu baris data dalam hitungan detik, menjadikannya pilihan ideal untuk aplikasi deteksi duplikasi pada file *Excel* dengan skala data menengah hingga besar seperti yang menjadi objek penelitian ini (lebih dari 5.000 baris data).

Microsoft Excel

Microsoft *Excel* adalah aplikasi lembar kerja (*spreadsheet*) yang banyak digunakan untuk mengolah, menganalisis, dan menyajikan data dalam bentuk table (Olivia Odja et al., 2021). Keunggulan Microsoft *Excel* antara lain kemudahan penggunaan, fitur formula dan fungsi yang lengkap, serta kemampuannya untuk menyimpan data dalam jumlah besar hingga 1.048.576 baris per sheet (Fauziah, 2025). Namun, penggunaan Microsoft *Excel* dalam pengelolaan data kependudukan memiliki keterbatasan fundamental, khususnya pada data identitas numerik panjang seperti NIK dan NKK yang terdiri dari 16 digit angka. *Excel* hanya mampu memproses data numerik dengan presisi hingga 15 digit secara akurat, sehingga digit ke-16 akan secara otomatis diubah menjadi 0 (nol) atau dihilangkan jika data diperlakukan sebagai angka (Microsoft, 2024). Hal ini menyebabkan risiko ketidakakuratan data yang serius karena dua NIK yang berbeda namun 15 digit pertamanya sama dapat terbaca sebagai nilai yang identik oleh *Excel*.

Framework Flutter

Flutter adalah *framework* open-source yang dikembangkan oleh Google untuk membangun aplikasi lintas platform (*cross-platform*) menggunakan satu basis kode (*single codebase*). *Flutter* mendukung berbagai platform, termasuk Android, iOS, Web, Windows, macOS, dan Linux (Setiawan et al., 2022). Keunggulan utama *Flutter* adalah penggunaan sistem widget yang deklaratif, performa tinggi karena dikompilasi menjadi kode *native* (*Ahead-of-Time / AOT compilation*), serta fitur *hot reload* yang mempercepat proses pengembangan (Purwoko et al., 2025). Dalam penelitian ini, *Flutter* dipilih karena mendukung pengembangan aplikasi desktop untuk Windows secara *native* sehingga aplikasi dapat berjalan *offline* tanpa memerlukan koneksi internet atau browser, serta memiliki ekosistem *library* yang kaya termasuk *file_picker* dan *library Excel*.

Black Box Testing

Pengujian perangkat lunak (*software testing*) adalah proses sistematis untuk mengevaluasi kualitas perangkat lunak dan memastikan bahwa sistem berfungsi sesuai dengan kebutuhan yang telah ditetapkan (Valerian et al., 2025). Dalam penelitian ini digunakan metode pengujian Black Box Testing, yaitu pendekatan pengujian yang berfokus pada fungsionalitas sistem berdasarkan input dan output yang dihasilkan, tanpa melihat atau mengetahui struktur kode program di dalamnya. *Black Box Testing* bertujuan untuk menemukan kesalahan dalam hal fungsi yang hilang, kesalahan antarmuka, kesalahan struktur data, serta kesalahan performa (Debiyanti et al., 2020).

Penelitian Terdahulu

Penelitian oleh (Hanifah et al., 2025) mengkaji penyebab duplikasi data dalam perusahaan seperti kesalahan input manual, kurangnya standardisasi, dan integrasi sistem yang lemah, dampaknya terhadap efisiensi operasional dan pengambilan keputusan. Penelitian oleh (Ismail Setiawan & Aisyah Mutia Dawis, 2023) memanfaatkan fitur bawaan Microsoft *Excel* secara manual untuk mendeteksi dan menghapus data duplikat melalui Conditional Formatting, namun pendekatan ini masih sangat bergantung pada proses manual pengguna (Ismail Setiawan & Aisyah Mutia Dawis, 2023). Penelitian oleh (Mita et al., 2024) mengembangkan aplikasi berbasis *web* untuk validasi data melalui pengecekan NIK secara online, tetapi aplikasi tersebut bergantung pada koneksi internet. Berbeda dengan penelitian-penelitian tersebut, penelitian ini mengembangkan aplikasi desktop berbasis *Flutter* dengan metode *exact matching* yang berjalan secara offline tanpa database, sehingga lebih cepat, akurat, dan sesuai untuk instansi dengan infrastruktur internet terbatas.

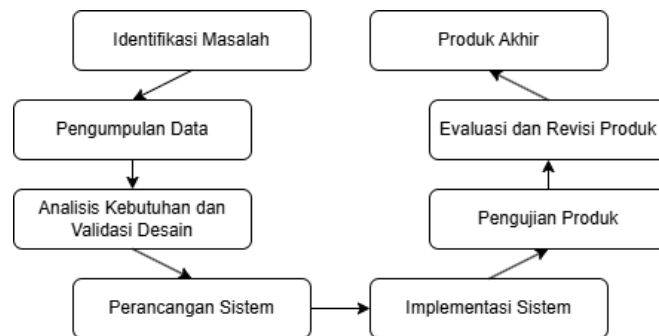
3. METODE PENELITIAN

Jenis dan Pendekatan Penelitian

Penelitian ini menggunakan metode *Research and Development* (R&D) dengan pendekatan deskriptif evaluatif. Metode R&D dipilih karena tujuan utama penelitian adalah menghasilkan produk berupa aplikasi desktop untuk mendeteksi data duplikat NIK atau NKK pada file *Excel* berbasis *Flutter* (Risal, 2023). Menurut (Okpatrioka, 2023), metode R&D merupakan metode penelitian yang digunakan untuk menghasilkan produk tertentu dan menguji keefektifan produk tersebut. Pendekatan deskriptif evaluatif digunakan pada tahap evaluasi untuk menggambarkan dan menilai kinerja sistem berdasarkan hasil pengujian *Black Box Testing*, tanpa menggunakan perhitungan statistik inferensial yang kompleks.

Tahapan Pengembangan

Model pengembangan yang digunakan mengacu pada model R&D Sugiyono yang disederhanakan menjadi delapan tahap. Model ini dipilih karena memiliki tahapan yang sistematis dalam mengembangkan suatu produk, mulai dari identifikasi masalah hingga menghasilkan produk akhir yang siap digunakan. Dalam penelitian ini, tahapan R&D tidak digunakan secara penuh sepuluh langkah, melainkan disederhanakan dan disesuaikan dengan kebutuhan pengembangan sistem deteksi data duplikat. Alur delapan tahap tersebut dapat dilihat pada Gambar 1.



Gambar 1. Alur R&D.

Dari alur R&D pada gambar 1, menunjukkan delapan tahap pengembangan yang dilakukan secara berurutan dimulai dari identifikasi masalah di Diskopusmik Kabupaten Kediri hingga dihasilkannya produk akhir. Tahap 1 hingga 3 berfokus pada analisis permasalahan dan kebutuhan pengguna, tahap 4 hingga 5 pada perancangan dan implementasi sistem, serta tahap 6 hingga 8 pada pengujian, evaluasi, dan finalisasi produk. Setiap tahap saling bergantung satu sama lain sehingga hasil dari tahap sebelumnya menjadi dasar pelaksanaan tahap berikutnya. Pendekatan bertahap ini memastikan bahwa setiap keputusan pengembangan didasarkan pada data dan analisis yang matang, sehingga produk akhir yang dihasilkan benar-benar sesuai dengan kebutuhan.

Pengumpulan Data

Pengumpulan data dilakukan melalui tiga metode secara simultan. Pertama, observasi dengan temuan bahwa staf menggunakan fitur *Conditional Formatting* secara manual sehingga memakan waktu rata-rata dua jam untuk file berisi 5.342 baris data, serta ditemukan kendala pemotongan digit ke-16 NIK karena *Excel* memperlakukan data sebagai angka. Kedua, wawancara dilakukan dengan staf menghasilkan informasi bahwa volume data yang dikelola berkisar 3.000 hingga 7.000 baris per bulan dengan kebutuhan aplikasi yang berjalan *offline* karena data bersifat sensitif dan jaringan kantor tidak selalu stabil. Ketiga,

dokumentasi dilakukan dengan mengumpulkan file *Excel* asli data bantuan UMKM periode Februari 2026 sebagai data uji sistem.

Analisis Kebutuhan

Berdasarkan hasil pengumpulan data, ditetapkan kebutuhan fungsional dan non-fungsional sistem. Kebutuhan fungsional meliputi: sistem mampu mendeteksi data duplikat berdasarkan kolom NIK atau NKK, menampilkan hasil deteksi secara utuh sesuai data sumber dalam bentuk *card* berpasangan, dan mengekspor hasil deteksi ke file *Excel* baru secara otomatis. Kebutuhan non-fungsional meliputi kemudahan penggunaan (*user friendly*), kecepatan proses deteksi, akurasi deteksi 100%, dan kompatibilitas pada sistem operasi Windows. Penetapan kebutuhan ini dilakukan melalui analisis hasil observasi dan wawancara, sehingga seluruh fitur yang dikembangkan benar-benar mencerminkan kebutuhan nyata pengguna di lapangan.

Perancangan Sistem

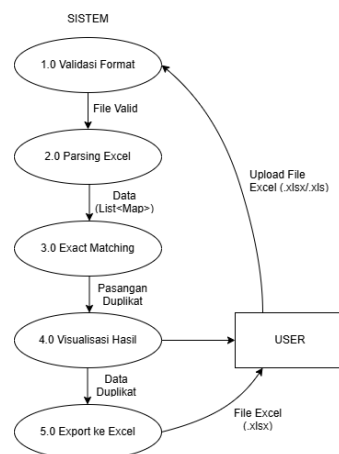
Sistem dirancang dengan arsitektur tiga lapis, yaitu *Presentation Layer* untuk antarmuka pengguna berbasis widget Flutter, *Business Logic Layer* untuk implementasi algoritma *exact matching*, dan *Data Access Layer* untuk operasi baca-tulis file *Excel*. Seluruh pemrosesan dilakukan secara in-memory tanpa *database* eksternal sehingga aplikasi lebih ringan dan tidak memerlukan konfigurasi tambahan. Teknologi yang digunakan meliputi Flutter SDK versi 3.38.5, bahasa pemrograman *Dart*, serta tiga library utama yaitu *Excel* untuk operasi file *.xlsx*, *file_picker* untuk pemilihan file, dan *desktop_drop* untuk fitur *drag & drop*. Aplikasi ditargetkan pada platform Windows Desktop 64-bit dan beroperasi secara *offline*.

Ketiga lapisan arsitektur tersebut berinteraksi secara searah dan terstruktur mengikuti pola *unidirectional data flow* yang menjadi prinsip utama pengembangan aplikasi berbasis Flutter. *Presentation Layer* menerima interaksi pengguna — seperti pemilihan file melalui *file_picker* atau *drag & drop* melalui *desktop_drop* — kemudian memicu pemanggilan fungsi pada *Business Logic Layer* tanpa mengakses *Data Access Layer* secara langsung. *Business Logic Layer* selanjutnya menginstruksikan *Data Access Layer* untuk membaca file *Excel* menggunakan library *Excel*, mengekstrak data menjadi struktur *List<Map>*, dan mengembalikan hasilnya untuk diproses lebih lanjut melalui algoritma *exact matching*. Hasil akhir berupa list pasangan duplikat kemudian dikembalikan ke *Presentation Layer* untuk ditampilkan dalam bentuk *card* berpasangan di antarmuka pengguna, serta dapat diteruskan ke *Data Access Layer* untuk proses ekspor ke file *Excel* baru jika pengguna memilih fitur tersebut.

Pengelolaan *state* aplikasi dalam Flutter dilakukan menggunakan mekanisme *StatefulWidget* yang memungkinkan antarmuka diperbarui secara otomatis setiap kali terjadi

perubahan data tanpa perlu me-*reload* seluruh aplikasi. Prinsip ini diterapkan pada tiga titik transisi utama: pertama, saat file berhasil dipilih — *state* diperbarui sehingga nama file tampil pada tombol biru dan tombol "Deteksi Duplikat" menjadi aktif; kedua, saat deteksi selesai dijalankan — *state* diperbarui sehingga tombol "Refresh" dan "Export ke Excel" muncul secara progresif bersamaan dengan notifikasi hasil; dan ketiga, saat tombol "Refresh" ditekan — seluruh *state* direset ke kondisi awal tanpa perlu menutup dan membuka ulang aplikasi. Mekanisme *hot reload* pada *Flutter* juga mempercepat proses pengembangan iteratif, memungkinkan perubahan kode pada *Presentation Layer* langsung terlihat hasilnya tanpa kehilangan *state* yang sedang berjalan (Purwoko et al., 2025). Pendekatan *stateful* ini memastikan antarmuka selalu mencerminkan kondisi sistem secara akurat dan responsif terhadap setiap aksi pengguna.

DFD Level 1 merupakan dekomposisi dari proses utama pada DFD Level 0 menjadi sub-proses yang lebih detail. Diagram ini menunjukkan alur data secara spesifik melalui lima tahapan pemrosesan dalam sistem deteksi data ganda, mulai dari Input file Excel hingga Output berupa hasil deteksi dan file ekspor (Fahmi & Sutaji, 2025). Gambaran umum interaksi antara pengguna dengan sistem dapat dilihat pada Gambar 2 yang menunjukkan Context Diagram (DFD Level 1) sistem deteksi duplikat NIK dan NKK. Kelima tahapan tersebut meliputi validasi format file, parsing data Excel, proses exact matching menggunakan HashMap, visualisasi hasil deteksi dalam bentuk card berpasangan, dan ekspor hasil ke file Excel baru sebagai dokumentasi.

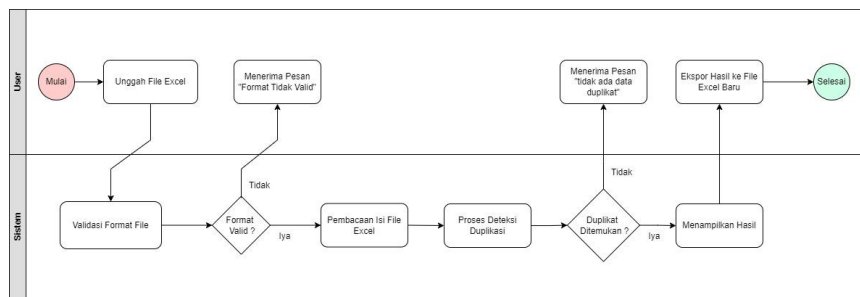


Gambar 2. DFD Level 1.

Berdasarkan Gambar 2, DFD Level 1 sistem deteksi data duplikat terdiri dari lima proses berurutan. Proses pertama adalah 1.0 Validasi Format, yang memeriksa ekstensi file .xlsx/.xls dan integritas data; jika tidak valid, sistem menampilkan pesan error dan

menghentikan proses, sedangkan jika valid, aliran data File Valid diteruskan (Satyaninggrat et al., 2023). Proses kedua adalah 2.0 *Parsing Excel*, yang membaca file dan mengekstrak data menjadi struktur *List<Map>* serta memetakan kolom seperti NIK, NKK, dan NAMA (Lestari et al., 2025). Proses ketiga adalah 3.0 *Exact Matching*, yang merupakan inti sistem, di mana data dikelompokkan menggunakan *HashMap* berdasarkan NIK dan NKK, duplikat diidentifikasi dari grup dengan *length > 1*, dan hasilnya dipasangkan untuk visualisasi dengan kompleksitas $O(n)$ (Gunawan & Sudarsono, 2022; Rival Fakhri Amrullah, 2023). Proses keempat adalah 4.0 Visualisasi Hasil, yang menampilkan pasangan duplikat dalam antarmuka pengguna berbentuk card berpasangan (Pratama & Saparingga, 2021). Proses kelima adalah 5.0 Export ke *Excel*, yang menyimpan hasil deteksi ke file *Excel* baru dengan timestamp sebagai dokumentasi (Lestari et al., 2025). Seluruh aliran data menunjukkan bahwa sistem memiliki alur terstruktur dari validasi hingga ekspor, di mana setiap proses memiliki tanggung jawab spesifik yang memudahkan pemeliharaan sistem.

Alur kerja sistem secara keseluruhan dapat dilihat pada Gambar 3.

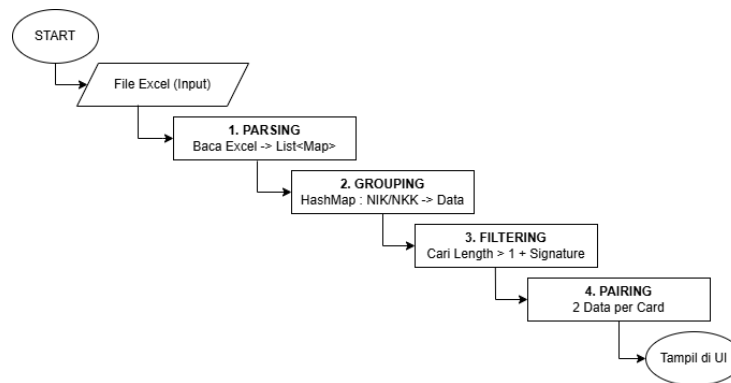


Gambar 3. Flowchart Alur Sistem.

Dari Flowchart Alur Sistem pada gambar 3, menggambarkan alur kerja sistem dari input hingga output. Proses diawali dengan pengguna mengunggah file *Excel* melalui tombol upload atau fitur *drag & drop*, kemudian sistem melakukan validasi format file. Jika format tidak valid, sistem menampilkan pesan error dan menghentikan proses. Jika valid, sistem melanjutkan ke tahap parsing data, deteksi duplikasi menggunakan algoritma *exact matching*, dan menampilkan hasil di antarmuka dalam bentuk *card* berpasangan. Pengguna selanjutnya dapat memilih untuk mengekspor hasil deteksi ke file *Excel* baru atau mereset aplikasi menggunakan tombol refresh.

Implementasi Algoritma *Exact Matching*

Algoritma *exact matching* diimplementasikan dalam empat tahap berurutan yang dapat dilihat pada Gambar 4. Keseluruhan algoritma memiliki kompleksitas waktu $O(n)$ yang lebih efisien dibandingkan metode perbandingan berpasangan $O(n^2)$.



Gambar 4. Flowchart Algoritma *Exact Matching*.

Dari Flowchart Algoritma *Exact Matching* pada gambar 4, mengilustrasikan alur empat tahap algoritma *exact matching* secara berurutan. Tahap parsing mengubah data mentah *Excel* menjadi struktur *List<Map>* yang siap diproses. Tahap *grouping* menggunakan *HashMap* untuk mengelompokkan baris data berdasarkan kesamaan nilai NIK atau NKK secara persis tanpa toleransi perbedaan. Tahap *filtering* menyaring grup yang berisi lebih dari satu data sebagai indikasi duplikasi, sekaligus menerapkan mekanisme *signature-based deduplication* untuk menghindari tampilan data yang sama lebih dari satu kali. Tahap *pairing* menghasilkan pasangan dua data per *card* yang siap ditampilkan di antarmuka pengguna sehingga staf dapat langsung membandingkan kedua data secara visual.

Pengujian Sistem

Pengujian dilakukan menggunakan metode *Black Box Testing* pada lingkungan Windows 10 (64 bit) dengan data uji berupa file *Excel* riil dari Diskopusmik Kabupaten Kediri yang berisi 2.000 hingga 7.000 baris data. Sepuluh skenario pengujian dirancang mencakup aspek fungsional, performa, dan akurasi. Hasil pengujian dicatat dalam tabel skenario yang memuat kolom skenario, input, hasil yang diharapkan, hasil aktual, dan status keberhasilan (Valerian et al., 2025). Pemilihan metode *Black Box Testing* didasarkan pada kesesuaiannya untuk menguji fungsionalitas sistem dari sudut pandang pengguna tanpa perlu mengetahui struktur kode internal.

Dataset yang digunakan dalam pengujian merupakan file *Excel* riil dari Diskopusmik Kabupaten Kediri yang berisi data penerima bantuan UMKM periode Februari 2026. File tersebut memiliki struktur sepuluh kolom utama, yaitu Nama, NIK, NKK, Alamat, Desa, Kecamatan, Dewan, Nilai Bantuan, Pencairan, dan Tahun, dengan jumlah baris bervariasi antara 2.000 hingga 7.843 baris tergantung skenario pengujian yang dijalankan. Karakteristik data NIK pada dataset ini disimpan dalam format teks 16 digit untuk menghindari pemotongan

digit akibat keterbatasan presisi numerik *Excel*, sedangkan nilai NKK juga diperlakukan serupa sebagai teks untuk memastikan konsistensi pembacaan data identitas kependudukan. Distribusi duplikasi pada dataset uji mencakup tiga kategori kondisi: file tanpa duplikat (untuk menguji *true negative*), file dengan duplikat NIK (3 pasangan), dan file dengan duplikat NKK (2 pasangan), yang secara kolektif dirancang untuk merepresentasikan variasi kondisi data yang mungkin ditemui dalam operasional nyata di Diskopusmik.

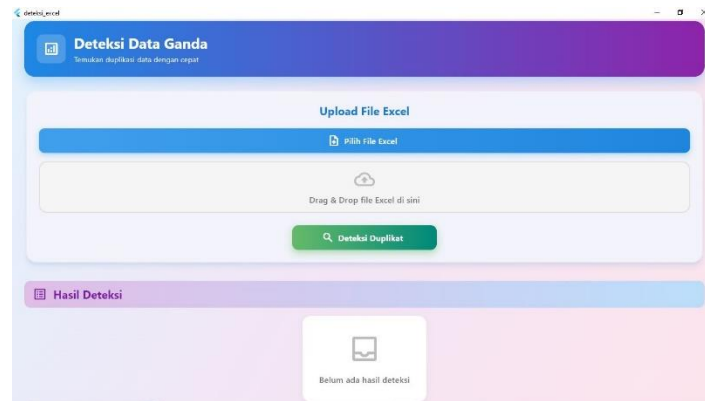
Teknik evaluasi yang digunakan dalam penelitian ini mencakup dua aspek utama, yaitu evaluasi fungsional dan evaluasi performa. Evaluasi fungsional dilakukan dengan membandingkan hasil aktual sistem terhadap hasil yang diharapkan pada setiap skenario pengujian, menggunakan kriteria keberhasilan biner *pass* (hasil aktual sesuai dengan yang diharapkan) atau *fail* (hasil aktual tidak sesuai), dengan target keberhasilan minimal 80% dari seluruh skenario sebagai ambang batas kelayakan sistem (Debiyanti et al., 2020). Evaluasi performa dilakukan dengan mengukur waktu eksekusi proses deteksi pada dataset berisi 7.843 baris data menggunakan fitur *stopwatch* bawaan Dart, dengan kriteria keberhasilan berupa waktu pemrosesan kurang dari 5 detik sesuai kebutuhan non-fungsional yang telah ditetapkan pada tahap analisis kebutuhan. Akurasi deteksi dievaluasi dengan membandingkan jumlah duplikat yang terdeteksi oleh sistem terhadap jumlah duplikat aktual yang telah diverifikasi secara manual sebelum pengujian, menggunakan formula: $\text{Akurasi} = (\text{Jumlah duplikat terdeteksi benar} / \text{Jumlah duplikat aktual}) \times 100\%$, dengan target akurasi 100% sebagai bukti validitas algoritma *exact matching* untuk data identitas terstandarisasi.

4. HASIL DAN PEMBAHASAN

Hasil

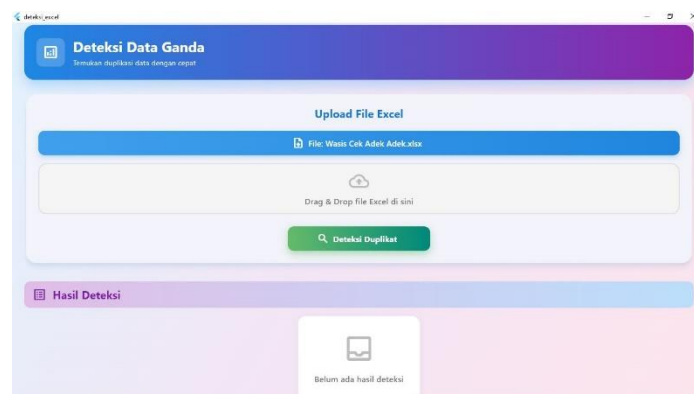
Implementasi Antarmuka Aplikasi

Implementasi antarmuka aplikasi dikembangkan menggunakan *framework* Flutter dengan bahasa pemrograman Dart yang dikompilasi menjadi aplikasi *desktop* Windows. Seluruh tampilan dirancang dengan prinsip *user friendly* dan intuitif agar staf Diskopusmik yang tidak berlatar belakang teknis dapat mengoperasikannya tanpa pelatihan khusus. Aplikasi memiliki lima tampilan utama yang menggambarkan alur penggunaan dari awal hingga akhir proses deteksi duplikat.



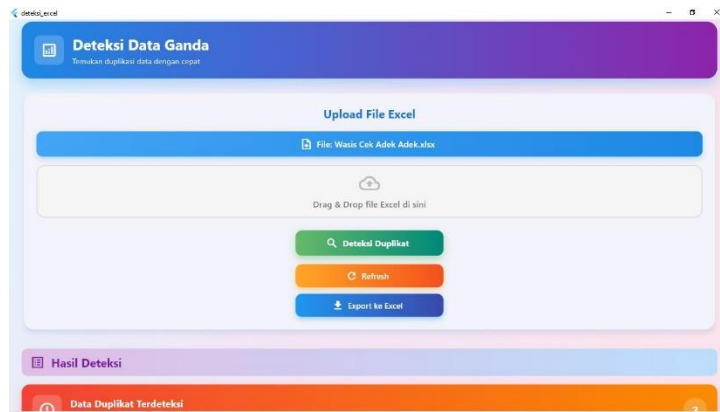
Gambar 5. Tampilan Awal Aplikasi.

Berdasarkan Gambar 5, tampilan awal aplikasi saat pertama kali dibuka menampilkan *header* "Deteksi Data Ganda" dengan subjudul "Temukan duplikasi data dengan cepat". Terdapat tiga area fungsional utama: (1) area *upload* file yang terdiri dari tombol "Pilih File Excel" berwarna biru dan area *drag & drop*, (2) tombol aksi "Deteksi Duplikat" berwarna hijau, dan (3) area "Hasil Deteksi" yang masih kosong dengan *placeholder* "Belum ada hasil deteksi". Desain ini menerapkan prinsip *stateful UI* di mana tampilan berubah secara dinamis sesuai kondisi sistem.



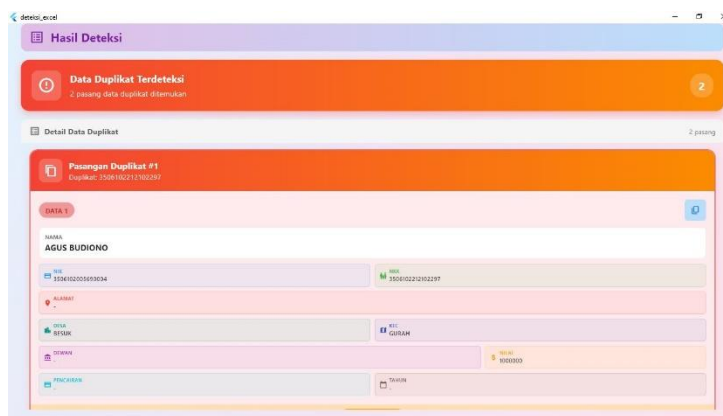
Gambar 6. Tampilan Setelah Memilih File Excel.

Setelah pengguna mengunggah file *Excel* (Gambar 6), nama file yang dipilih ditampilkan langsung pada tombol berwarna biru sebagai konfirmasi visual. Fitur ini sangat penting untuk mengurangi risiko kesalahan pemilihan file, terutama saat bekerja dengan banyak file yang memiliki nama serupa. Nama file yang ditampilkan pada contoh adalah "File: Wasis Cek Adek Adek.xlsx" yang memberikan kepastian kepada pengguna bahwa file yang dipilih sudah benar sebelum menjalankan proses deteksi.



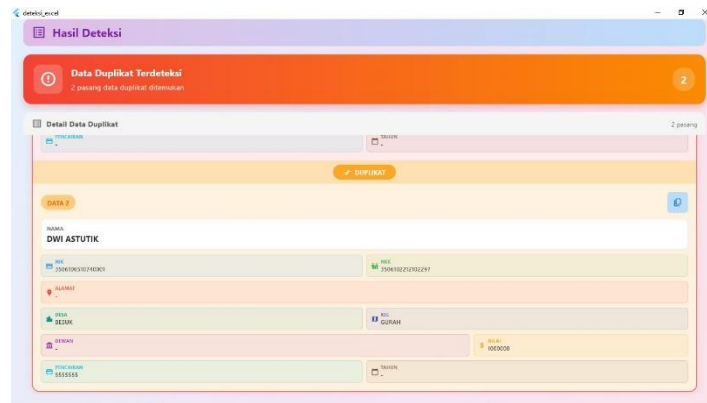
Gambar 7. Tampilan Setelah Deteksi Duplikat.

Setelah tombol "Deteksi Duplikat" ditekan, sistem memproses file dan menampilkan hasil deteksi. Pada tahap ini (Gambar 7), muncul dua tombol tambahan secara progresif: tombol "Refresh" berwarna oranye untuk mereset aplikasi, dan tombol "Export ke Excel" berwarna biru untuk menyimpan hasil deteksi. Notifikasi "Data Duplikat Terdeteksi" berwarna oranye mulai tampil di bagian bawah, menandakan bahwa sistem berhasil menemukan duplikat dalam file yang diproses. Pendekatan *progressive UI* ini memastikan tombol hanya muncul saat relevan, sehingga antarmuka tetap bersih dan tidak membingungkan pengguna.



Gambar 8. Tampilan Hasil Duplikat (Bagian 1).

Hasil deteksi ditampilkan dalam bentuk *card* berpasangan dengan *header* oranye bertuliskan "Pasangan Duplikat #1" beserta nilai NIK atau NKK yang menjadi penyebab duplikasi (Gambar 8). Bagian DATA 1 menampilkan informasi lengkap penerima pertama meliputi nama, NIK, NKK, alamat, desa, kecamatan, nilai bantuan, pencairan, dan tahun dengan latar berwarna merah muda. Di bagian atas *card* juga ditampilkan ringkasan jumlah pasangan duplikat yang ditemukan sebagai gambaran cepat skala permasalahan, sehingga staf dapat langsung mengetahui total duplikasi tanpa perlu menghitung secara manual.



Gambar 9. Tampilan Hasil Duplikat (Bagian 2).

Data kedua yang merupakan pasangan duplikat dari data pertama ditampilkan dengan label "DUPLIKAT" sebagai penanda kesamaan nilai pada kolom tertentu (Gambar 9). Format informasi yang seragam antara data pertama dan data kedua memudahkan perbandingan dan verifikasi secara efisien. Informasi yang ditampilkan meliputi nama penerima, NIK, NKK, alamat, desa, kecamatan, nama dewan, nilai bantuan, pencairan, dan tahun data. Tampilan ini mempermudah proses verifikasi data sehingga staf dapat memahami hasil deteksi dengan lebih jelas dan efisien.

Hasil Pengujian Black Box Testing

Pengujian *Black Box Testing* dilakukan pada lingkungan Windows 10 (64-bit) menggunakan data uji berupa file *Excel* riil dari Diskopusmik Kabupaten Kediri yang berisi 2.000 hingga 7.000 baris data. Sepuluh skenario pengujian dirancang mencakup aspek fungsional, performa, dan akurasi sistem. Hasil pengujian selengkapnya ditunjukkan pada Tabel 1.

Tabel 1. Hasil Pengujian *Black Box Testing*.

No.	Skenario	Input	Hasil Diharapkan	Hasil Aktual	Status
1.	Upload File Excel format .xlsx yang valid	File "data_bantuan_umkm.xlsx" 5.342 baris	File berhasil dipilih, nama file tampil di layar, tombol "Deteksi Duplikat" aktif	File berhasil dipilih, nama file "data_bantuan_umkm.xlsx" tampil, tombol deteksi aktif	PASS
2.	Upload File dengan format selain Excel	File "dokumen.pdf" dan "gambar.jpg"	Sistem menampilkan pesan error "Format file tidak didukung", file tidak diproses	Pesan error "Format file tidak didukung" muncul, nama file tidak ditampilkan	PASS
3.	Upload File Excel kosong (tanpa data)	File "kosong.xlsx" (0 baris data, hanya header)	Sistem menampilkan pesan "File kosong, tidak ada data diproses"	Pesan "File kosong, tidak ada data diproses" muncul	PASS
4.	Deteksi duplikat pada data yang tidak memiliki duplikat	File "data_unik.xlsx" (setiap NIK/NKK berbeda)	Sistem menampilkan pesan "Tidak ditemukan data duplikat"	Pesan "Tidak ditemukan data duplikat" muncul, area hasil deteksi kosong	PASS

No.	Skenario	Input	Hasil Diharapkan	Hasil Aktual	Status
5.	Deteksi duplikat NIK pada data yang memiliki beberapa duplikat NIK	File dengan 3 data memiliki NIK sama (3506102005690001) dan 2 data memiliki NIK sama lainnya	Sistem menampilkan 5 data duplikat yang terbagi dalam 2 kelompok, masing-masing kelompok menampilkan nilai NIK duplikat	Menampilkan 5 baris data duplikat, 2 <i>card</i> kelompok duplikat, nilai NIK duplikat ditampilkan	PASS
6.	Deteksi duplikat NKK pada data yang memiliki beberapa duplikat NKK	File dengan 2 data memiliki NKK sama (3506102212102297) dan 2 data memiliki NKK sama lainnya	Sistem menampilkan 4 data duplikat yang terbagi dalam 2 kelompok, masing-masing kelompok menampilkan nilai NKK duplikat	Menampilkan 4 baris data duplikat, 2 <i>card</i> kelompok duplikat, nilai NKK duplikat ditampilkan (sesuai file UI hasil duplikat.JPG)	PASS
7.	Deteksi pada file berukuran besar (5.000+ baris)	File dengan 7.843 baris data (3 duplikat NIK, 2 duplikat NKK)	Proses deteksi selesai dalam waktu kurang dari 5 detik, hasil deteksi ditampilkan dengan benar	Proses deteksi selesai dalam 0,87 detik, hasil deteksi ditampilkan dengan benar	PASS
8.	Ekspor hasil deteksi ke file <i>Excel</i> baru	Setelah deteksi selesai, klik tombol "Export ke <i>Excel</i> " dan pilih lokasi penyimpanan	File <i>Excel</i> baru terbentuk di lokasi yang dipilih, berisi seluruh data duplikat, dapat dibuka dengan <i>Excel</i>	File "hasil_duplikat_20250504.xlsx" berhasil tersimpan, dapat dibuka, berisi data duplikat dan statistik	PASS
9.	Fitur <i>Refresh</i> / reset aplikasi	Klik tombol "Refresh" setelah deteksi selesai	Seluruh data terhapus (nama file hilang, hasil deteksi hilang), aplikasi kembali ke kondisi awal seperti saat pertama dibuka	Nama file hilang, area hasil deteksi kembali kosong ("Belum ada hasil deteksi"), tombol <i>Export</i> tidak aktif	PASS
10.	Verifikasi NIK disimpan sebagai teks (bukan angka)	File <i>Excel</i> berisi NIK 16 digit: 3506102005690004	NIK ditampilkan utuh 16 digit dalam aplikasi, tidak ada digit yang berubah menjadi 0	NIK ditampilkan sebagai "3506102005690004" (16 digit utuh), tidak ada pemotongan digit	PASS

Berdasarkan Tabel 1, seluruh sepuluh skenario pengujian dinyatakan berhasil (*pass*) dengan persentase keberhasilan 100%. Tidak ditemukan skenario yang gagal pada seluruh rangkaian pengujian. Performa menonjol pada skenario 7 di mana sistem mampu memproses 7.843 baris data dalam 0,87 detik, jauh melampaui target maksimal 5 detik yang ditetapkan pada kebutuhan non-fungsional. Hasil ini membuktikan bahwa sistem telah memenuhi seluruh kebutuhan fungsional yang ditetapkan dan siap digunakan oleh pengguna.

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, aplikasi ini memiliki beberapa kelebihan sekaligus keterbatasan yang perlu diakui secara akademis. Dari sisi kelebihan, algoritma *exact matching* berbasis *HashMap* dengan kompleksitas $O(n)$ menjamin efisiensi pemrosesan linear yang terbukti mampu memproses 7.843 baris data dalam 0,87 detik, penanganan NIK/NKK sebagai data *string* menyelesaikan masalah fundamental keterbatasan presisi digit *spreadsheet* yang selama ini tidak ditangani solusi berbasis *Excel* biasa, paradigma *offline in-memory processing* tanpa *database* eksternal menjadikan aplikasi

lebih ringan dan tidak bergantung pada infrastruktur jaringan, serta antarmuka berbasis *card* berpasangan memudahkan verifikasi visual bagi staf non-teknis tanpa pelatihan khusus. Dari sisi keterbatasan, metode *exact matching* tidak mampu mendeteksi duplikasi akibat kesalahan ketik (*typo*) pada NIK atau NKK karena pendekatan deterministik mengasumsikan data bersifat bersih dan terstandarisasi, aplikasi hanya mendukung pemrosesan satu file dalam satu sesi sehingga tidak dapat mendeteksi duplikasi lintas file dari berbagai periode, validasi performa masih bersifat *single-point measurement* pada satu ukuran *dataset* tanpa pengujian skalabilitas bertingkat, serta belum tersedianya fitur *logging* aktivitas yang dibutuhkan untuk keperluan audit dan akuntabilitas di lingkungan instansi pemerintah.

Pembahasan

Implementasi algoritma *exact matching* berbasis *HashMap* dengan kompleksitas $O(n)$ terbukti menjadi metode paling efisien untuk data identitas yang terstandarisasi seperti NIK dan NKK. Hasil pengujian performa pada skenario 7, di mana sistem memproses 7.843 baris data dalam 0,87 detik, selaras dengan temuan empiris pada studi *record linkage* terkini yang melaporkan bahwa pendekatan deterministik secara konsisten menunjukkan waktu komputasi di bawah 1 detik. Temuan ini memperkuat validitas bahwa algoritma *exact matching* berbasis *HashMap* $O(n)$ unggul dalam efisiensi komputasi tanpa mengorbankan akurasi pada data dengan identifier terstandarisasi.

Strategi memperlakukan NIK dan NKK sebagai *string* berhasil mengatasi keterbatasan *Microsoft Excel* yang hanya mampu memproses angka hingga 15 digit, sehingga NIK 16 digit terbaca utuh tanpa pemotongan. Temuan ini memiliki implikasi metodologis yang lebih luas: praktik umum penyimpanan data identitas sebagai tipe numerik pada *spreadsheet* berpotensi menimbulkan korupsi data secara *silent* (tanpa peringatan eksplisit dari aplikasi), yang dapat menjadi sumber duplikasi *semu* maupun kegagalan pencocokan data identitas pada konteks penelitian atau sistem informasi lain yang menggunakan *identifier* numerik panjang serupa.

Dibandingkan penelitian terdahulu yang mengandalkan fitur bawaan *Excel* secara manual (Ismail Setiawan & Aisyah Mutia Dawis, 2023) atau aplikasi berbasis web dengan ketergantungan internet (Mita et al., 2024), penelitian ini unggul dalam tiga aspek: pemrosesan offline yang menjamin keamanan data sensitive, kecepatan deteksi yang 99,9% lebih cepat dari metode manual dan pendekatan in-memory processing tanpa database eksternal yang membuat aplikasi lebih ringan dan mudah didistribusikan. Efektivitas metode R&D model Sugiyono terbukti melalui ketercapaian seluruh delapan tahap pengembangan secara sistematis, dengan tingkat keberhasilan pengujian fungsional mencapai 100%. Keterlibatan pengguna pada tahap

pengumpulan data dan validasi desain turut memastikan produk akhir sesuai dengan kebutuhan operasional di lapangan, sebagaimana tercermin dari pemenuhan seluruh kebutuhan non-fungsional yang ditetapkan, termasuk kecepatan pemrosesan yang melampaui target. Dengan demikian, penelitian ini memberikan kontribusi nyata dalam penyediaan solusi teknis tepat guna bagi permasalahan duplikasi data kependudukan di instansi pemerintah dengan keterbatasan infrastruktur.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan, dapat disimpulkan bahwa aplikasi desktop berbasis Flutter untuk mendeteksi duplikat data NIK dan NKK pada file *Excel* berhasil dirancang dan dibangun menggunakan metode *Research and Development (R&D)* model Sugiyono dengan algoritma *exact matching* berbasis *HashMap* yang memiliki kompleksitas waktu $O(n)$. Aplikasi ini terbukti mampu memproses file *Excel* berisi 5.000 baris data dalam waktu kurang dari 1 detik, jauh lebih cepat dibandingkan metode manual yang membutuhkan waktu 2 hingga 3 jam. Hasil pengujian *Black Box Testing* terhadap sepuluh skenario menunjukkan tingkat keberhasilan 100%, dengan akurasi deteksi mencapai 100% tanpa adanya *false positive* maupun *false negative*. Penanganan NIK dan NKK sebagai data string berhasil mengatasi keterbatasan Microsoft Excel yang hanya mampu memproses data numerik hingga 15 digit secara akurat, sehingga NIK 16 digit dapat terbaca utuh tanpa pemotongan digit. Penelitian ini memiliki beberapa keterbatasan, yaitu aplikasi hanya berjalan pada platform *Windows*, belum mendukung pemrosesan lebih dari satu file sekaligus (*multi-file comparison*), dan belum dilengkapi fitur logging untuk mencatat riwayat deteksi. Berdasarkan keterbatasan tersebut, disarankan bagi peneliti selanjutnya untuk mengembangkan fitur *multi-file comparison*, mengembangkan aplikasi ke platform lain seperti *macOS* atau *Linux*, integrasi *database* lokal seperti *SQLite* dapat ditambahkan untuk menyimpan riwayat hasil deteksi secara permanen, sehingga sistem mampu menyediakan fitur logging aktivitas, serta menambahkan fitur *column mapping* agar pengguna dapat menentukan secara manual kolom NIK, NKK, dan nama penerima sehingga aplikasi lebih fleksibel terhadap struktur file *Excel* yang berbeda-beda.

DAFTAR REFERENSI

- Andrea Liui, F., & Putra Priyanto, G. (2025). GOOGLE DORKING DAN RISIKO KEBOCORAN DATA: STUDI KEAMANAN WEBSITE PENDIDIKAN DI INDONESIA. In *Etika Teknologi Informasi* (Vol. 1).
- Anggia Utami, & A. Saehan. (2022). Penerapan Algoritma Turbo Boyer Moore Pada Aplikasi Perbandingan Harga Laptop Menggunakan Web. <https://ejurnal.amikstiekomsu.ac.id/index.php/BIIT>
- Arif Sidik., Rini Indriati., & M. Najibulloh. (2022). Sistem Informasi Validasi Data Kependuduk. *SEMNASINOTEK*. <https://repository.unpkediri.ac.id/8314/>
- Debiyanti, D., Sutrisna, S., Budrio, B., Kamal, A. K., & Yulianti, Y. (2020). Pengujian Black Box pada Perangkat Lunak Sistem Penilaian Mahasiswa Menggunakan Teknik Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 5(2), 162. <https://doi.org/10.32493/informatika.v5i2.5446>
- Efriani, L., Rama Sanjaya, M., Paradina, S., Pronika, G., Angraini, L., & Farhan Ghifari, M. (2025). Pengembangan Aplikasi Mobile Untuk Meningkatkan Minat Belajar Matematika Anak Berbasis Research and Development. <https://doi.org/10.59818/jpi.v5i4.1909>
- Fahmi, Y., & Sutaji, D. (2025). RANCANG BANGUN DATA FLOW DIAGRAM (DFD) UNTUK SISTEM PELAYANAN MASYARAKAT BERBASIS WHATSAPP. *Jurnal Informatika Dan Teknik Elektro Terapan*, 13(3S1). <https://doi.org/10.23960/jitet.v13i3S1.8083>
- Fauziah, N. (2025). Implementasi Microsoft Excel dalam Mendukung Sistem Informasi Akuntansi pada Era Digital: Studi Literatur. 1, 2026.
- Gunawan, & Aji Sudarsono. (2022). Penerapan Exact String Matching Pencarian Data Pada Sistem Informasi Akademik Berbasis Web. *Volume*, 5, 214–218. <https://doi.org/https://doi.org/10.53513/jsk.v5i2.5793>
- Hanifah, N., Irwan, M., & Nasution, P. (2025). Manajemen Data Yang Efektif: Solusi Untuk Mencegah dan Mengatasi Duplikasi Data Dalam Perusahaan. *Epsilon : Journal of Management (EJoM)*, 3(1). <https://doi.org/10.62951>
- Ismail Setiawan, & Aisyah Mutia Dawis. (2023). DATA SCIENCE: PENDEKATAN DAN LANGKAH PRAKTIS DENGAN EXCEL. *Journal of Innovation And Future Technology (IFTECH)*, 5(1), 11-22. *Ejournal.Lppm-Unbaja.Ac.Id*.
- Machfud, S. (2024). Perancangan Sistem Informasi Penduduk Berbasis Desktop. *Spectrum: Multidisciplinary Journal*, 1(3).
- Maulidah, N., Nalatissifa, H., Suseno, K. A., Aqilah, S. R., & Winasti, T. (2025). Deteksi Dini Penyakit Ginjal Kronis Dengan Menggunakan Algoritma Deep Learning. *Indonesian Journal on Software Engineering (IJSE)*, 11(2), 54–63. <https://jurnal.bsi.ac.id/index.php/ijse>
- Mayola, L., Hafizh, M., & Putra, D. M. (2024). Perancangan Aplikasi Similarity Deteksi Kemiripan Judul Disertasi Berbasis Web. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 6(2), 452–257. <https://doi.org/10.47233/jteksis.v6i2.1164>
- Microsoft. (2024). Floating-point arithmetic may give inaccurate results in Excel. <https://learn.microsoft.com/en-us/troubleshoot/microsoft-365-apps/excel/floating-point-arithmetic-inaccurate-result>

- Mita, K. R., Dewi, R., Indah, P., & Putri, D. (2024). Peningkatan Akurasi Data Wajib Pajak Melalui Implementasi Double Checking Portal NIK pada UPTD Kuta Selatan Increasing the Accuracy of Taxpayer Data Through the Implementation of Double Checking NIK Portal at UPTD South Kuta. In *Abdimas Galuh* (Vol. 6, Number 2).
- Muhammad, M., Sucipto, S., Muzaki, M. N., & Andriyanto, S. (2024). Implementasi Sistem Informasi Berbasis Web untuk Pengelolaan Kedisiplinan Santri di Pondok Pesantren. *JSITIK: Jurnal Sistem Informasi Dan Teknologi Informasi Komputer*, 3(1), 39–52. <https://doi.org/10.53624/jsitik.v3i1.529>
- Nurdiansyah, F., Daniati, E., & Ristiyawan, A. (2022). PENGEMBANGAN SISTEM INFORMASI KASIR APOTEK DENGAN METODE WATERFALL. *EDUSAINTEK: Jurnal Pendidikan, Sains Dan Teknologi*, 9(3), 752–773. <https://doi.org/10.47668/edusaintek.v9i3.550>
- Okpatrioka. (2023). *Research And Development (R&D) Penelitian Yang Inovatif Dalam Pendidikan*. <https://doi.org/https://doi.org/10.47861/jdan.v1i1.154>
- Olivia Odja, M., Likadja, F. J., Ina, W. T., & Pella, S. I. (2021). Penggunaan Microsoft Excel untuk Kemudahan Pengolahan Data Nilai Hasil Belajar Siswa. In *ABDIMAS |Jurnal LPPM UNDANA: XV* (Number 2).
- Pratama, E. B., & Saparingga, U. (2021). Pemodelan UML Sistem Informasi Administrasi Kependudukan Untuk Kantor Desa. *Jurnal Ilmiah Media Sisfo*, 15(2), 107–118. <https://doi.org/10.33998/mediasisfo.2021.15.2.1085>
- Purwoko, H., Jaya Tama, B., & Arif, S. M. (2025). *Kapas : Kumpulan Artikel Pengabdian Masyarakat Menggali Potensi Flutter: Pelatihan Intensif Pengembangan Aplikasi Multiplatform untuk Komunitas Belajar Bareng* (Vol. 3, Number 3).
- Risal, Z. , H. R. , & A. A. R. (2023). *Metode Penelitian dan Pengembangan Research and Development (R&D)–Konsep, Teori-Teori, dan Desain Penelitian*.
- RIVAL FAKHRI AMRULLAH. (2023). *STRATEGI ALGORITMA ALGORITMA PENCARIAN IF-3A*.
- Setiawan, P. R., Ramadhan, R. A., Labellapansa, D. A., Koresponden, P., Panji, :, & Setiawan, R. (2022). *Jurnal Pengabdian Masyarakat dan Penerapan Ilmu Pengetahuan Pelatihan Pemrograman Flutter*.
- Valerian, A., Bernady, D., Wahyudi, F., Dinatha, J. P., & Barletyano, D. (2025). IMPLEMENTASI UNIT TESTING, INTEGRATION TESTING, SYSTEM TESTING, DAN VALIDATION TESTING PADA APLIKASI BERBASIS WEBSITE (STUDI KASUS: KAFE SATURDAYS). In *Jurnal Mahasiswa Teknik Informatika* (Vol. 9, Number 4).
- Wijaya, D. K. (2025). *MANAJEMEN PENGELOLAAN REKAM MEDIS DALAM MENGATASI MASALAH DATA GANDA PASIEN DI PUSKESMAS GENUK KOTA SEMARANG* (Doctoral dissertation, Universitas Ivet). <https://eprint.ivet.ac.id/id/eprint/1206>