

PENGEMBANGAN APLIKASI POINT OF SALE BERBASIS ANDROID DENGAN NOTIFIKASI STOK MINIMUM MENGGUNAKAN FIREBASE CLOUD MESSAGING

Faris Fadhil Dhiaulhaq¹, Rini Mayasari², Sofi Defiyanti³

^{1,2,3} Universitas Singaperbangsa Karawang, Jl. HS. Ronggo Waluyo, Puserjaya, Telukjambe Timur, Karawang, Jawa Barat, 41361.

Keywords:

Point of Sale, Flutter, Firebase Cloud Messaging, Manajemen Stok, Android.

Correspondent Email:

faris.fdhiaulhaq@gmail.com

Abstrak. Digitalisasi proses transaksi dan pengelolaan stok menjadi kebutuhan penting bagi usaha mikro, kecil, dan menengah (UMKM) untuk meningkatkan efisiensi operasional. Penelitian ini bertujuan untuk mengembangkan aplikasi Point of Sale (POS) berbasis Android yang mendukung transaksi penjualan, pengelolaan produk dan stok, pelaporan penjualan, serta notifikasi stok minimum secara proaktif. Metode penelitian yang digunakan adalah Software Development Life Cycle (SDLC) model Waterfall yang meliputi tahap analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Aplikasi dikembangkan menggunakan Flutter dengan integrasi Firebase Authentication, Cloud Firestore, Cloud Functions, dan Firebase Cloud Messaging (FCM). Hasil pengujian Black Box Testing terhadap 24 skenario menunjukkan seluruh fungsi sistem berjalan sesuai spesifikasi. Fitur notifikasi stok minimum berhasil mengirimkan push notification secara otomatis saat stok mencapai batas minimum, baik ketika aplikasi aktif, berjalan di latar belakang, maupun perangkat dalam keadaan terkunci. Hasil User Acceptance Testing memperoleh tingkat penerimaan sebesar 98,10% dari kasir dan 100% dari pemilik toko. Dengan demikian, aplikasi yang dikembangkan mampu mendukung operasional toko secara lebih efektif melalui pencatatan transaksi, pengelolaan stok, dan pemantauan stok secara proaktif.



Copyright © 2026 The Author(s), Published by Jurnal Informatika dan Teknik Elektro Terapan. This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Abstract. The digitalization of transaction processing and inventory management has become essential for improving the operational efficiency of micro, small, and medium enterprises (MSMEs). This study aims to develop an Android-based Point of Sale (POS) application that supports sales transactions, product and inventory management, sales reporting, and proactive minimum stock notifications. The study employed the Software Development Life Cycle (SDLC) Waterfall model, consisting of requirements analysis, system design, implementation, testing, and maintenance. The application was developed using Flutter integrated with Firebase Authentication, Cloud Firestore, Cloud Functions, and Firebase Cloud Messaging (FCM). Black Box Testing on 24 test scenarios demonstrated that all system functions operated according to the specified requirements. The minimum stock notification feature successfully delivered automatic push notifications when inventory reached the predefined threshold while the application was active, running in the background, or the device was locked. User Acceptance Testing achieved acceptance rates of 98.10% from cashiers and 100% from the store owner. Therefore, the proposed application effectively supports store operations through transaction management, inventory management, and proactive inventory monitoring.

1. PENDAHULUAN

Perkembangan teknologi informasi telah mendorong transformasi digital pada berbagai sektor usaha, termasuk usaha mikro, kecil, dan menengah (UMKM). Salah satu bentuk transformasi tersebut adalah penerapan sistem *Point of Sale* (POS) yang tidak hanya berfungsi sebagai media pencatatan transaksi, tetapi juga mendukung pengelolaan stok dan penyajian informasi penjualan secara terintegrasi. Implementasi sistem POS mampu meningkatkan efisiensi proses bisnis melalui digitalisasi transaksi, mengurangi kesalahan pencatatan, serta menyediakan informasi yang lebih akurat untuk mendukung pengambilan keputusan oleh pemilik usaha [1], [2], [8]. Selain itu, aplikasi POS berbasis mobile memberikan fleksibilitas yang lebih tinggi karena dapat diakses kapan saja dan di mana saja sehingga sesuai dengan karakteristik operasional UMKM [2], [6], [9].

Pengelolaan stok merupakan salah satu aspek penting dalam menjaga kelangsungan operasional usaha ritel. Ketersediaan stok yang tidak terpantau dengan baik dapat menyebabkan keterlambatan pengadaan barang hingga terjadinya *stockout* yang berpotensi menghilangkan kesempatan penjualan dan menurunkan kepuasan pelanggan. Integrasi antara sistem POS dan manajemen persediaan terbukti membantu meningkatkan akurasi data stok serta meminimalkan perbedaan antara stok aktual dan stok pada sistem [1], [3], [8]. Namun demikian, sebagian besar aplikasi POS yang digunakan oleh UMKM masih menerapkan mekanisme pemantauan stok secara pasif sehingga pengguna harus membuka aplikasi secara berkala untuk mengetahui kondisi stok terkini [1], [3], [6].

Permasalahan tersebut ditemukan pada Toko Pestisida X di Kabupaten Karawang yang bergerak di bidang penjualan pestisida, pupuk, dan perlengkapan pertanian. Toko telah menggunakan aplikasi POS komersial untuk mencatat transaksi penjualan dan mengelola persediaan barang. Namun, sistem yang digunakan belum menyediakan mekanisme peringatan otomatis ketika stok produk mencapai batas minimum. Akibatnya, pemilik

toko harus memeriksa aplikasi secara manual untuk mengetahui kondisi stok. Dalam praktiknya, pemeriksaan stok tidak selalu dilakukan secara konsisten karena kesibukan operasional sehingga beberapa kali terjadi kehabisan stok yang baru diketahui ketika pelanggan datang untuk membeli produk. Kondisi tersebut menunjukkan bahwa digitalisasi pencatatan transaksi belum sepenuhnya mampu mengatasi permasalahan pengelolaan stok apabila belum didukung mekanisme pemantauan yang bersifat proaktif.

Perkembangan teknologi komputasi awan memungkinkan implementasi sistem pemantauan stok secara *real-time* melalui layanan *Backend as a Service* (BaaS) tanpa memerlukan pengelolaan server secara mandiri. Firebase menyediakan layanan Cloud Firestore sebagai basis data *real-time* untuk menyimpan dan menyinkronkan data transaksi secara terpusat [4]. Selain itu, Cloud Functions memungkinkan eksekusi proses secara otomatis berdasarkan perubahan data (*event-driven*) sehingga sesuai untuk membangun mekanisme pemantauan stok tanpa pengelolaan server secara manual [10], [11]. Firebase Cloud Messaging (FCM) memungkinkan sistem mengirimkan *push notification* secara otomatis ke perangkat Android meskipun aplikasi tidak sedang dibuka [5], [12], [13]. Integrasi layanan-layanan tersebut memberikan peluang untuk membangun aplikasi POS yang tidak hanya mendukung transaksi penjualan, tetapi juga mampu memberikan informasi stok secara proaktif kepada pemilik usaha.

Beberapa penelitian telah mengembangkan aplikasi POS berbasis mobile untuk mendukung operasional UMKM dengan berbagai pendekatan. Candwian dan Septiani [2] mengembangkan aplikasi POS yang mengintegrasikan transaksi penjualan dengan manajemen stok. Wisnu et al. [6], Afriza et al. [3], serta Thoriq et al. [7] mengembangkan aplikasi POS berbasis Flutter dan mobile untuk mendukung pengelolaan transaksi, inventaris, dan pelaporan penjualan. Nurhidayat et al. [14] mengintegrasikan pembayaran QRIS pada aplikasi POS berbasis Android, sedangkan Dayfit et al. [9] menunjukkan bahwa Flutter

mampu menghasilkan aplikasi POS dengan antarmuka yang responsif dan performa yang baik pada perangkat Android.

Berdasarkan penelitian-penelitian tersebut, pengembangan aplikasi POS umumnya berfokus pada digitalisasi transaksi, pengelolaan produk, pelaporan penjualan, maupun integrasi metode pembayaran. Meskipun beberapa penelitian telah menyediakan fitur manajemen stok, mekanisme pemantauan masih bersifat pasif sehingga pemilik usaha tetap harus melakukan pemeriksaan aplikasi secara manual untuk mengetahui kondisi stok. Implementasi mekanisme notifikasi stok minimum secara proaktif menggunakan integrasi Cloud Firestore, Cloud Functions, dan Firebase Cloud Messaging sebagai *early warning system* bagi pemilik toko juga masih sangat terbatas. Akibatnya, potensi keterlambatan pengadaan barang tetap dapat terjadi ketika stok kritis tidak segera diketahui oleh pemilik usaha. Oleh karena itu, diperlukan pengembangan aplikasi POS yang mampu mengintegrasikan fungsi transaksi, pengelolaan stok, serta mekanisme notifikasi otomatis ketika stok mencapai batas minimum.

Penelitian ini bertujuan mengembangkan aplikasi *Point of Sale* berbasis Android menggunakan framework Flutter dengan dukungan Firebase Authentication, Cloud Firestore, Cloud Functions, dan Firebase Cloud Messaging (FCM). Sistem yang dikembangkan menerapkan mekanisme login *multi-role* bagi kasir dan pemilik toko, menyediakan fitur transaksi penjualan, pengelolaan produk dan stok, laporan penjualan, serta notifikasi stok minimum yang dikirimkan secara otomatis ketika jumlah stok mencapai batas yang telah ditentukan. Kontribusi utama penelitian ini adalah implementasi mekanisme pemantauan stok secara proaktif melalui integrasi Cloud Functions dan Firebase Cloud Messaging sehingga pemilik toko dapat memperoleh informasi stok minimum secara cepat tanpa harus melakukan pemeriksaan aplikasi secara berkala.

2. TINJAUAN PUSTAKA

2.1 Point Of Sales (POS)

Point of Sale (POS) merupakan sistem informasi yang digunakan untuk mendukung proses transaksi penjualan sekaligus membantu pengelolaan operasional usaha, seperti pencatatan produk, pengelolaan stok, dan penyusunan laporan penjualan. Implementasi sistem POS mampu meningkatkan efisiensi proses bisnis melalui digitalisasi transaksi, meminimalkan kesalahan pencatatan, serta menyediakan informasi yang lebih akurat bagi pemilik usaha dalam mengambil keputusan [1], [2], [8]. Integrasi POS dengan sistem inventori juga membantu menjaga konsistensi data stok antara transaksi penjualan dan kondisi persediaan aktual sehingga mendukung operasional usaha yang lebih efektif [3], [15]. Pada sektor UMKM, aplikasi POS berbasis mobile menjadi solusi yang banyak digunakan karena mudah diakses, fleksibel, dan sesuai dengan kebutuhan operasional sehari-hari [2], [4], [6].

2.2 Flutter

Flutter merupakan framework yang dikembangkan oleh Google untuk membangun aplikasi lintas platform menggunakan bahasa pemrograman Dart melalui satu basis kode (single codebase). Flutter mampu menghasilkan antarmuka pengguna yang responsif dengan performa tinggi melalui mekanisme hot reload, sehingga mempercepat proses pengembangan sekaligus mempermudah pemeliharaan aplikasi [6], [8], [9]. Berbagai penelitian telah menunjukkan bahwa Flutter sesuai digunakan dalam pengembangan aplikasi POS berbasis Android karena mampu menghasilkan antarmuka yang konsisten, responsif, dan mudah dikembangkan untuk kebutuhan operasional UMKM [6], [8], [9].

2.3 Firebase dan Firebase Cloud Messaging

Firebase merupakan platform *Backend as a Service* (BaaS) yang menyediakan berbagai layanan untuk mendukung pengembangan aplikasi, seperti Firebase Authentication, Cloud Firestore, Cloud Functions, dan Firebase Cloud Messaging (FCM). Cloud Firestore berfungsi sebagai basis data NoSQL yang mampu melakukan sinkronisasi data secara *real-time*, sedangkan Cloud Functions memungkinkan proses otomatis dijalankan berdasarkan

perubahan data (*event-driven*) tanpa memerlukan pengelolaan server secara mandiri [10], [11]. Firebase Cloud Messaging (FCM) merupakan layanan yang digunakan untuk mengirimkan *push notification* ke perangkat pengguna sehingga informasi dapat diterima meskipun aplikasi tidak sedang aktif dijalankan [5], [12], [13]. Integrasi Cloud Firestore, Cloud Functions, dan FCM memungkinkan implementasi mekanisme pemantauan stok secara proaktif dengan mengirimkan notifikasi otomatis ketika stok produk mencapai batas minimum. Pada penelitian ini, mekanisme tersebut diterapkan untuk mendukung proses pengelolaan stok pada Toko Pestisida X.

2.4 Software Development Life Cycle (SDLC) Waterfall

Software Development Life Cycle (SDLC) merupakan kerangka kerja yang mengatur tahapan pengembangan perangkat lunak secara sistematis mulai dari analisis kebutuhan hingga pemeliharaan sistem. Salah satu model SDLC yang banyak digunakan adalah Waterfall, yang terdiri atas tahapan *requirements analysis*, *system design*, *implementation*, *testing*, dan *maintenance* [15], [16]. Model ini sesuai digunakan pada penelitian yang memiliki kebutuhan sistem yang telah terdefinisi dengan jelas sejak awal sehingga setiap tahapan dapat diselesaikan secara berurutan sebelum melanjutkan ke tahap berikutnya [15], [16].

3. METODE PENELITIAN

Penelitian ini menggunakan metode *Software Development Life Cycle* (SDLC) dengan model Waterfall karena kebutuhan sistem telah didefinisikan dengan jelas sejak awal berdasarkan hasil analisis kebutuhan pada Toko Pestisida X. Model Waterfall dipilih karena memiliki tahapan yang sistematis dan sesuai untuk pengembangan perangkat lunak dengan ruang lingkup yang telah ditentukan [15], [16].

Pengumpulan data dilakukan melalui wawancara dengan pemilik toko untuk mengidentifikasi kebutuhan sistem, observasi terhadap proses transaksi penjualan dan pengelolaan stok yang sedang berjalan, serta studi pustaka mengenai pengembangan aplikasi *Point of Sale*, Flutter, Firebase, dan Firebase Cloud Messaging sebagai landasan teoritis penelitian.

Tahapan penelitian mengikuti lima fase pada model Waterfall sebagaimana ditunjukkan pada Gambar 1, yaitu *requirements analysis*, *system design*, *implementation*, *testing*, dan *maintenance*.

3.1. Analisis Kebutuhan (Requirement Analysis)

Tahap ini bertujuan mengidentifikasi kebutuhan fungsional dan nonfungsional sistem berdasarkan hasil wawancara dan observasi. Analisis difokuskan pada proses transaksi penjualan, pengelolaan produk, pengelolaan stok, laporan penjualan, serta kebutuhan notifikasi stok minimum.

3.2. Perancangan Sistem (System Design)

Berdasarkan hasil analisis kebutuhan, dilakukan perancangan arsitektur sistem, perancangan basis data Cloud Firestore, perancangan antarmuka pengguna, serta pemodelan sistem menggunakan Unified Modeling Language (UML) yang meliputi *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*. Selain itu, dirancang mekanisme pengiriman notifikasi stok menggunakan Cloud Functions dan Firebase Cloud Messaging.

3.3. Implementasi (Implementation)

Tahap implementasi dilakukan menggunakan framework Flutter sebagai pengembang aplikasi Android. Backend sistem memanfaatkan Firebase Authentication untuk autentikasi pengguna, Cloud Firestore sebagai basis data *real-time*, Cloud Functions untuk menjalankan proses otomatis ketika terjadi perubahan data stok, serta Firebase Cloud Messaging untuk mengirimkan *push notification* kepada pemilik toko.

3.4. Pengujian (Testing)

Pengujian sistem dilakukan menggunakan metode Black Box Testing untuk memverifikasi seluruh fungsi aplikasi sesuai kebutuhan sistem [17]. Selanjutnya dilakukan User Acceptance Testing (UAT) dengan melibatkan pengguna akhir untuk mengevaluasi kesesuaian sistem terhadap kebutuhan operasional [18].

3.5. Pemeliharaan (Maintenance)

Tahap pemeliharaan dilakukan sebagai tindak lanjut setelah sistem diimplementasikan,

meliputi perbaikan kesalahan (*bug fixing*) serta penyesuaian sistem apabila terdapat kebutuhan pengembangan lebih lanjut.

Objek penelitian adalah Toko Pestisida X yang berlokasi di Kabupaten Karawang. Sistem yang dikembangkan berupa aplikasi *Point of Sale* berbasis Android dengan mekanisme *login* multi-role untuk kasir dan pemilik toko. Fitur utama aplikasi meliputi pengelolaan produk, transaksi penjualan, laporan penjualan, dashboard pemilik, serta notifikasi otomatis ketika stok produk mencapai batas minimum menggunakan Firebase Cloud Messaging.

4. HASIL DAN PEMBAHASAN

Hasil Pengembangan Sistem

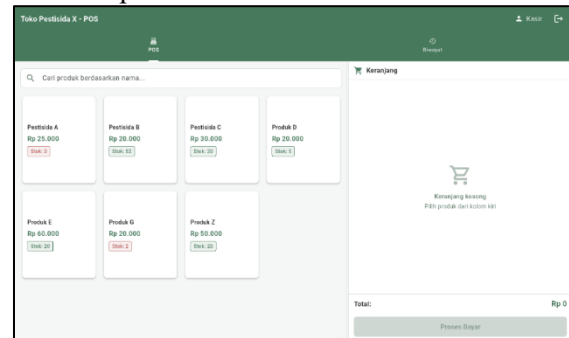
Penelitian ini menghasilkan aplikasi *Point of Sale* (POS) berbasis Android yang dikembangkan menggunakan framework Flutter dengan layanan backend Firebase. Sistem dirancang untuk mendukung proses transaksi penjualan sekaligus menyediakan mekanisme pemantauan stok secara proaktif melalui integrasi Firebase Cloud Functions dan Firebase Cloud Messaging (FCM). Pengembangan sistem dilakukan menggunakan metode SDLC Waterfall yang terdiri atas tahap analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan.

Berdasarkan hasil analisis kebutuhan, sistem dibangun dengan dua jenis hak akses, yaitu Owner dan Kasir. Kasir memiliki akses terhadap proses transaksi penjualan, pengelolaan keranjang belanja, pencetakan struk, serta riwayat transaksi harian. Sementara itu, Owner memperoleh akses terhadap dashboard penjualan, manajemen produk dan stok, laporan penjualan, serta penerimaan notifikasi stok minimum secara otomatis.

Implementasi sistem memanfaatkan Firebase Authentication untuk autentikasi pengguna, Cloud Firestore sebagai basis data real-time, Cloud Functions untuk menjalankan logika bisnis pada sisi server, dan Firebase Cloud Messaging sebagai layanan pengiriman push notification. Arsitektur tersebut memungkinkan sinkronisasi data transaksi dan stok berlangsung secara real-time tanpa memerlukan server yang dikelola secara mandiri.

Gambar 1 memperlihatkan halaman transaksi yang digunakan oleh kasir, sedangkan Gambar 2 menunjukkan dashboard yang

digunakan oleh pemilik toko untuk memantau kondisi operasional secara real-time.



Gambar 1. Antarmuka transaksi pada role Kasir.



Gambar 2. Antarmuka dashboard pada role Owner.

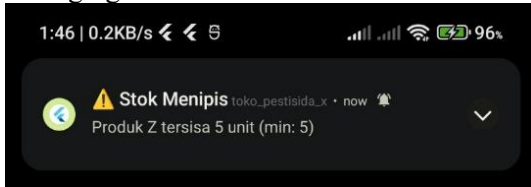
Selain fungsi transaksi, aplikasi menyediakan fitur laporan penjualan dalam format Microsoft Excel, pengelolaan data produk, pengaturan batas minimum stok, serta riwayat transaksi. Implementasi fitur-fitur tersebut memenuhi seluruh kebutuhan fungsional yang diperoleh pada tahap analisis.

Implementasi Notifikasi Stok Minimum

Kontribusi utama penelitian ini terletak pada implementasi mekanisme notifikasi stok minimum yang berjalan secara otomatis setelah transaksi penjualan selesai diproses.

Ketika kasir berhasil menyelesaikan transaksi, sistem memperbarui jumlah stok produk pada Cloud Firestore. Perubahan data tersebut memicu Firebase Cloud Functions untuk melakukan evaluasi terhadap stok setiap

produk yang terjual. Apabila jumlah stok telah mencapai atau berada di bawah nilai minimum yang ditentukan, Cloud Functions mengirimkan push notification kepada perangkat Android milik Owner melalui Firebase Cloud Messaging.



Gambar 3. Tampilan notifikasi stok minimum

Pengujian menunjukkan bahwa notifikasi berhasil diterima pada tiga kondisi penggunaan, yaitu:

- aplikasi berada pada kondisi aktif (*foreground*);
- aplikasi berjalan di latar belakang (*background*);
- perangkat berada pada kondisi layar terkunci (*lock screen*).

Hasil tersebut menunjukkan bahwa mekanisme notifikasi mampu memberikan informasi stok secara proaktif tanpa mengharuskan pemilik toko membuka aplikasi secara manual.

Hasil Pengujian Sistem

Pengujian fungsional dilakukan menggunakan metode Black Box Testing terhadap seluruh fitur utama aplikasi. Sebanyak 24 skenario pengujian disusun berdasarkan kebutuhan fungsional yang telah diidentifikasi pada tahap analisis.

Tabel 1. Hasil Black Box Testing.

Aspek Pengujian	Jumlah Skenario	Berhasil	Persentase
Login & Logout	3	3	100%
Transaksi Penjualan	9	9	100%
Manajemen Produk	6	6	100%
Dashboard & Laporan	3	3	100%
Notifikasi Stok	3	3	100%
Total	24	24	100%

Seluruh skenario pengujian menghasilkan keluaran yang sesuai dengan spesifikasi kebutuhan sehingga tidak ditemukan kegagalan fungsi pada modul utama aplikasi.

Selain pengujian fungsional, penelitian juga melakukan User Acceptance Testing (UAT) untuk mengetahui tingkat penerimaan pengguna terhadap sistem yang dikembangkan. Hasil UAT pada role Kasir memperoleh nilai sebesar 98,10%, sedangkan role Owner memperoleh nilai 100%. Berdasarkan kategori interpretasi skala Likert, kedua nilai tersebut termasuk dalam kategori Sangat Baik, sehingga sistem dinilai telah memenuhi kebutuhan operasional pengguna.

Tabel 2. Hasil User Acceptance Testing (UAT)

Pengguna	Persentase
Kasir	98,10%
Owner	100%

Pembahasan

Hasil penelitian menunjukkan bahwa aplikasi yang dikembangkan berhasil memenuhi tujuan penelitian, yaitu menyediakan sistem Point of Sale berbasis Android yang mampu melakukan pemantauan stok secara proaktif melalui mekanisme push notification otomatis.

Hasil penelitian ini menunjukkan adanya peningkatan fungsional dibandingkan beberapa penelitian sebelumnya yang dilakukan oleh Wisnu *et al.* mengembangkan aplikasi POS berbasis Flutter dengan fitur transaksi dan pengelolaan stok, namun pemantauan stok masih dilakukan secara manual melalui aplikasi [6]. Penelitian oleh Thoriq *et al.* juga telah mengembangkan sistem POS berbasis web dan mobile, tetapi belum menyediakan mekanisme pemberitahuan otomatis ketika stok mencapai batas minimum [7]. Sementara itu, Nurhidayat *et al.* lebih berfokus pada integrasi pembayaran QRIS tanpa mengimplementasikan sistem peringatan stok [14].

Berbeda dengan penelitian-penelitian tersebut, penelitian ini mengintegrasikan Firebase Cloud Functions dan Firebase Cloud Messaging sehingga proses pemantauan stok tidak lagi bergantung pada aktivitas pengguna dalam membuka aplikasi. Mekanisme ini memungkinkan sistem mengirimkan notifikasi secara otomatis segera setelah transaksi menyebabkan stok mencapai batas minimum.

Penggunaan arsitektur serverless Firebase juga memberikan keuntungan dari sisi implementasi karena tidak memerlukan pengelolaan server secara mandiri. Cloud Firestore menyediakan sinkronisasi data secara real-time, sedangkan Cloud Functions memungkinkan proses bisnis dijalankan pada sisi server tanpa membebani perangkat pengguna. Pendekatan tersebut sesuai untuk kebutuhan UMKM yang memerlukan solusi dengan biaya implementasi dan pemeliharaan yang relatif rendah [4], [5].

Hasil Black Box Testing yang mencapai tingkat keberhasilan 100% menunjukkan bahwa seluruh fungsi utama aplikasi telah berjalan sesuai kebutuhan. Selain itu, hasil UAT sebesar 98,10% pada role Kasir dan 100% pada role Owner menunjukkan bahwa sistem diterima dengan sangat baik oleh pengguna. Nilai tersebut mengindikasikan bahwa fitur transaksi, pengelolaan produk, dashboard, laporan penjualan, serta notifikasi stok minimum telah sesuai dengan kebutuhan operasional Toko Pestisida X.

Meskipun demikian, penelitian ini masih memiliki keterbatasan. Pengujian penerimaan pengguna dilakukan pada jumlah responden yang terbatas sehingga hasilnya belum dapat digeneralisasikan pada berbagai jenis UMKM. Selain itu, mekanisme notifikasi yang dikembangkan hanya memanfaatkan ambang batas minimum stok dan belum mempertimbangkan prediksi kebutuhan stok berdasarkan pola penjualan. Pengembangan selanjutnya dapat mengintegrasikan metode peramalan (*forecasting*) sehingga sistem tidak hanya memberikan peringatan ketika stok telah menipis, tetapi juga mampu memprediksi kebutuhan restok berdasarkan histori transaksi.

5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa:

1. Penelitian ini berhasil mengembangkan aplikasi Point of Sale (POS) berbasis Android menggunakan framework Flutter dengan dukungan Firebase Authentication, Cloud Firestore, Cloud Functions, dan Firebase Cloud Messaging (FCM). Sistem yang dikembangkan mampu mendukung proses autentikasi berbasis role, transaksi penjualan, pengelolaan produk dan stok, laporan penjualan, serta notifikasi stok minimum secara otomatis.
2. Implementasi mekanisme Cloud Functions dan Firebase Cloud Messaging (FCM) berhasil mewujudkan sistem pemantauan stok yang bersifat proaktif. Notifikasi stok kritis dapat dikirim secara otomatis kepada pemilik toko ketika jumlah stok mencapai atau berada di bawah batas minimum tanpa memerlukan pemeriksaan aplikasi secara manual.
3. Hasil pengujian menunjukkan bahwa seluruh kebutuhan fungsional sistem telah terpenuhi. Pengujian Black Box Testing terhadap 24 skenario menghasilkan tingkat keberhasilan 100%, sedangkan User Acceptance Testing (UAT) memperoleh tingkat penerimaan pengguna sebesar 98,10% pada role kasir dan 100% pada role owner. Hasil tersebut menunjukkan bahwa aplikasi telah memenuhi kebutuhan operasional Toko Pestisida X.
4. Kelebihan utama sistem terletak pada integrasi layanan Firebase yang memungkinkan sinkronisasi data secara real-time, pengelolaan backend tanpa server mandiri (serverless), serta mekanisme notifikasi stok otomatis yang belum tersedia pada aplikasi POS yang digunakan sebelumnya. Selain itu, penerapan login multi-role memungkinkan pengelolaan hak akses antara owner dan kasir dalam satu aplikasi.
5. Penelitian ini masih memiliki beberapa keterbatasan, yaitu implementasi yang hanya dilakukan pada satu UMKM serta jumlah responden User Acceptance Testing yang masih terbatas. Selain itu, sistem belum menyediakan fitur analisis penjualan, prediksi kebutuhan restok, maupun dukungan pengelolaan multi-cabang.
6. Pengembangan selanjutnya dapat diarahkan pada implementasi analisis penjualan berbasis kecerdasan buatan, prediksi restok menggunakan metode peramalan, integrasi dashboard berbasis web, dukungan multi-store,

serta pengujian pada lebih banyak UMKM untuk mengevaluasi skalabilitas dan efektivitas sistem pada lingkungan operasional yang lebih beragam.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan, bantuan, dan kontribusi dalam pelaksanaan penelitian ini. Semoga segala bentuk bantuan dan kerja sama yang telah diberikan memperoleh balasan yang terbaik serta menjadi amal kebaikan yang mendapat rahmat dan keberkahan dari Allah SWT.

DAFTAR PUSTAKA

- [1] D. A. T. Alwi and H. Ma'sum, "Pengembangan Sistem Inventory Berbasis Flutter UMKM Warung Hj Wiwin," *Jurnal Janitra Informatika dan Sistem Informasi*, vol. 5, no. 1, May 2025, doi: 10.59395/fj0jrm16.
- [2] R. Candwian and M. Septiani, "Aplikasi Pos Kasir Mobile Dengan Fitur Manajemen Stok Terintegrasi Untuk Mendukung Kinerja UMKM," *Bianglala Informatika*, vol. 13, no. 2, pp. 108–114, Sep. 2025, doi: 10.31294/bianglala.v13i2.11421.
- [3] F. I. Afriza, R. K. Pakpahan, G. Gunawan, and N. Nurhayati, "Pengembangan Aplikasi Point of Sales Berbasis Mobile dan Web pada Browenz Coffee," *Jurnal SIFO Mikroskil*, vol. 24, no. 2, pp. 141–156, Oct. 2023, doi: 10.55601/jsm.v24i2.992.
- [4] M. A. Sumarto, "Analisis dan Perancangan Aplikasi Point of Sale (POS) untuk Usaha Mikro, Kecil, dan Menengah (UMKM) dengan Metode Rapid Application Development (RAD)," *Jurnal Studi Komunikasi dan Media*, vol. 27, no. 1, pp. 17–34, Jun. 2023, doi: 10.17933/jskm.2023.5115.
- [5] D. P. Putro, I. Gunawan, and P. E. Suryani, "Software Push Notification Disposisi Persuratan Berbasis Website Menggunakan Firebase Cloud Messaging," 2022. doi: <https://doi.org/10.51519/journalita.volume3.issue3.year2022.page370-381>.
- [6] G. Wisnu, T. Prasetyo, F. Pradana, and B. S. Prakoso, "Pengembangan Aplikasi Point of Sales Warung dan UMKM 'WarunkQu' menggunakan Framework Flutter," 2022. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [7] M. A. Thoriq, I. Adam Hendra Brata, and M. T. Ananta, "Pengembangan Aplikasi Point of Sale Berbasis Website dan Mobile (Studi Kasus: UD. Karya Utama Oxygen)," 2025. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [8] M. Dayfit, A. Rashad, B. Rahayudi, and D. Pramono, "Rancang Bangun Aplikasi Point of Sales Dengan Voice Assistant Berbasis Mobile Menggunakan Framework Flutter (Studi Kasus: Toko 2 Mahkota Plastik)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 9, no. 12, 2025. [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/15763>
- [9] A. Dharmalau, N. Sucahyo, and I. Mukti, "Perancangan Aplikasi Point of Sales (POS) Berbasis Android Menggunakan Framework Flutter di Kafe Elangsta," *JRIS: Jurnal Rekayasa Informasi Swadharma*, vol. 3, no. 2, pp. 6–13, 2023, doi: 10.56486/jris.vol3no2.326.
- [10] V. Kumar and T. Kaur, "Cloud Functions and Serverless Computing," *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, no. 5, pp. 3426–3427, May 2022, doi: 10.22214/ijraset.2022.43163.
- [11] M. D. H. Saputra and T. Dirgahayu, "Pengembangan dan Implementasi Teknologi Serverless Computing pada Sistem Informasi Data Kependudukan Desa dengan Pendekatan Metode Agile," *Journal of Information System Research (JOSH)*, vol. 6, no. 1, pp. 513–523, 2024, doi: 10.47065/josh.v6i1.6075.
- [12] W. Abdusallam, B. Wicaksono, A. Susanto, and Sudaryanto, "Push Notification Menggunakan Firebase Cloud Messaging (FCM) Pada Aplikasi Absensi Karyawan," *Jurnal Sisfotenika*, vol. 11, no. 2, 2021, doi: 10.30700/jst.v11i2.1150.
- [13] M. Meidina, J. Junadhi, U. Rio, and Y. Efendi, "Penerapan Fitur Firebase Cloud Messaging Pada Sistem Kontrol Pembayaran Iuran BPJS Ketenagakerjaan Berbasis Mobile," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 1, 2022, doi: 10.47065/bits.v4i1.1603.
- [14] A. R. Nurhidayat, B. Nugraha, and A. A. Hendriadi, "Perancangan Aplikasi Point of Sales (POS) Berbasis Android dengan QRIS Payment (Studi Kasus: Warung Seblak Tonjong)," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, 2025, doi: 10.23960/jitet.v13i2.6323.
- [15] D. T. Haniva, J. A. Ramadhan, and A. Suharso, "Systematic Literature Review Penggunaan Metodologi Pengembangan Sistem Informasi Waterfall, Agile, dan Hybrid," *Journal of Information Engineering and Educational Technology*, vol. 7, no. 1, pp. 36–42, 2023, doi: 10.26740/jieet.v7n1.p36-42.

- [16] M. Mukhtar, S. Maria, M. Muhammad, V. S. Gunawan, and D. Y. Prasetyo, "Implementasi Metode Waterfall pada Sistem Pengolahan Data Activity Daily Harvested dan Plantation Daily," *Jurnal Teknologi dan Sistem Informasi Bisnis*, vol. 7, no. 3, pp. 467–472, 2025, doi: 10.47233/jteksis.v7i3.2084.
- [17] J. Shadiq, A. Safei, and R. W. R. Loly, "Pengujian Aplikasi Peminjaman Kendaraan Operasional Kantor Menggunakan Black Box Testing," *Information Management for Educators and Professionals*, vol. 5, no. 2, pp. 97–110, 2021, doi: 10.51211/imbi.v5i2.1561.
- [18] N. Aliyah, Nahrin Hartono, and A. A. Muin, "Penggunaan User Acceptance Testing (UAT) Pada Pengujian Sistem Informasi Pengelolaan Keuangan dan Inventaris Barang," *Switch: Jurnal Sains dan Teknologi Informasi*, vol. 3, no. 1, pp. 84–100, 2024, doi: 10.62951/switch.v3i1.330.