



Aplikasi Kasir Berbasis *React Native* dan *Database Firebase* Studi Kasus Klinik Kecantikan *Ultra Glow* di Kecamatan Adiwerna

Salwa Aqila Zafira^{1*}, Rudi Heryanto²

¹⁻²Politeknik Pancasakti Global, Indonesia

*Penulis korespondensi: salwaaqilazafira@gmail.com

Abstract. *The development of information technology has driven the need for systems that can improve operational efficiency, including in the beauty service sector. Ultra Glow Beauty Clinic in Adiwerna District still faces challenges in managing transactions, recording customer data, and generating reports manually, which can lead to errors and inefficiencies. This study aims to design and implement a mobile-based cashier application using React Native with Firebase as the database to facilitate transaction processing and real-time data management. The system development method was carried out through the stages of requirement analysis, design, implementation, and testing. System testing was conducted using the Black Box Testing method to ensure that each application function operates according to the specified requirements. The results show that the developed cashier application is able to improve transaction speed and accuracy, simplify the management of customer and service data, and generate reports automatically. Based on the Black Box Testing results, all main features of the application function properly without any significant errors. Therefore, this application can serve as an effective solution to support clinic operations in a more efficient and integrated manner.*

Keywords: *Black Box Testing; Cashier Application; Firebase; Information System; React Native*

Abstrak. Perkembangan teknologi informasi mendorong kebutuhan akan sistem yang mampu meningkatkan efisiensi operasional, termasuk pada sektor layanan kecantikan. Klinik Kecantikan Ultra Glow di Kecamatan Adiwerna masih menghadapi kendala dalam pengelolaan transaksi, pencatatan data pelanggan, serta pembuatan laporan yang dilakukan secara manual, sehingga berpotensi menimbulkan kesalahan dan ketidakefisienan. Penelitian ini bertujuan untuk merancang dan mengimplementasikan aplikasi kasir berbasis mobile menggunakan *React Native* dengan dukungan *database Firebase* guna mempermudah proses transaksi dan manajemen data secara real-time. Metode pengembangan sistem dilakukan melalui tahapan analisis kebutuhan, perancangan, implementasi, dan pengujian. Pengujian sistem menggunakan metode Black Box Testing untuk memastikan setiap fungsi aplikasi berjalan sesuai dengan spesifikasi yang telah ditentukan. Hasil penelitian menunjukkan bahwa aplikasi kasir yang dikembangkan mampu meningkatkan kecepatan dan akurasi transaksi, mempermudah pengelolaan data pelanggan dan layanan, serta menghasilkan laporan secara otomatis. Berdasarkan hasil pengujian Black Box, seluruh fitur utama aplikasi berfungsi dengan baik tanpa ditemukan kesalahan signifikan. Dengan demikian, aplikasi ini dapat menjadi solusi efektif dalam mendukung operasional klinik kecantikan secara lebih efisien dan terintegrasi.

Kata kunci: Aplikasi Kasir; *Black Box Testing*; *Firebase*; *React Native*; Sistem Informasi

1. LATAR BELAKANG

Saat ini perkembangan teknologi dibidang seluler berkembang sangat pesat dari mulai model seluler itu sendiri hingga fungsinya. Salah satu contoh fungsinya telepon seluler bisa dijadikan sebagai media pembelajaran. *Mobile application* adalah aplikasi yang dapat berjalan di ponsel dan dapat digunakan sebagai media pembelajaran, mengolah dan mendapatkan informasi yang bersifat praktis (tidak terikat waktu dan bisa dibawa kemana-mana) (Aziz et al., 2020).

React Native adalah *framework open source* besutan Facebook yang dibuat setelah Facebook sebelumnya membuat *react.js*, *React.JS* sendiri merupakan sebuah *library* dari Facebook yang dapat digunakan untuk membangun antarmuka pengguna (UI). Jadi, *React Native* adalah *framework open source* untuk membuat aplikasi multi-platform (android, ios dan

windows platform ”dalam tahap pengembangan”) dengan bahasa JavaScript, sesuai dengan deskripsi di situs resminya “*Learn once, write anywhere*” (Wiguna et al., 2019).

React Native adalah salah satu *framework* untuk membuat aplikasi mobile dengan menggunakan kode JavaScript. *Framework React Native* memiliki seperangkat komponen bagi platform iOS dan Android untuk membangun aplikasi mobile dengan tampilan yang benar-benar seperti native. Dengan menggunakan *Framework React Native*, kita dapat merender User Interface untuk platform iOS dan Android. *React Native* ini adalah *framework open source*, yang bisa kompatibel dengan platform lain seperti Windows atau tvOS dalam waktu dekat (Malahella & Arwani, 2020).

Firestore adalah API yang disediakan google untuk penyimpanan dan penyelarasan data ke dalam aplikasi Android, iOS, atau web. *Realtime database* adalah salah satu fasilitas yang menyimpan data ke *database* dan mengambil data darinya dengan sangat cepat tetapi *Firestore* bukan hanya *realtime database*, jauh lebih dari itu. *Firestore* memiliki banyak fitur seperti authentication, *database*, storage, hosting, pemberitahuan dan lain-lain (Payara & Tanone, 2022).

Berdasarkan peraturan Menteri Kesehatan Republik Indonesia Nomor 028/Menkes/Per/I/2011, pengertian klinik adalah fasilitas pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan yang menyediakan pelayanan medis dasar dan spesialisasi, diselenggarakan oleh lebih dari satu jenis tenaga kesehatan dan dipimpin oleh seorang tenaga medis (Menkes RI, 2001).

Klinik Kecantikan Ultra Glow merupakan salah satu klinik kecantikan dan kesehatan pada kulit wajah. Klinik tersebut menyediakan layanan jasa seperti konsultasi dan perawatan kulit wajah. Klinik Kecantikan Ultra Glow sudah beroperasi sejak tahun 2011. Beberapa pelayanan di klinik tersebut masih menggunakan cara manual, seperti layanan transaksi penjualan produk ultra glow. Sistem kasir merupakan solusi untuk mempersingkat waktu pada proses input data transaksi produk perawatan kulit wajah, khususnya waktu yang digunakan untuk menunggu antrian. Beberapa klinik kecantikan saat ini sudah banyak yang menerapkan sistem kasir modern.

Berdasarkan studi pendahuluan di Klinik Kecantikan Ultra Glow, untuk melakukan pembayaran transaksi di klinik tersebut pelanggan harus datang terlebih dahulu untuk melakukan pemilihan produk kemudian pelanggan tersebut harus menunggu sesuai antrian dan melakukan transaksi dengan kasir. Pelanggan yang datang terlebih dahulu dapat segera dilayani, sedangkan pelanggan yang lain harus menunggu. Staf kasir juga masih harus mencatat data secara manual ke dalam buku transaksi klinik. Hal ini sering kali

menyebabkan beberapa masalah seperti buku transaksi atau daftar barang yang telah dicatat dapat hilang atau rusak dan proses perekapan data pelanggan menjadi lebih sulit. Untuk itu, diperlukan sebuah aplikasi kasir di Klinik Kecantikan Ultra Glow dengan mengintegrasikan antara teknologi modern dengan smartphone berbasis android. Dengan adanya aplikasi smartphone yang berfungsi untuk mendukung pelayanan kasir, diharapkan mampu menjadi solusi yang lebih efektif dan efisien dibanding metode pemesanan layanan lainnya. Berdasarkan penjelasan latar belakang tersebut, maka penulis memutuskan untuk membuat aplikasi smartphone sebagai penerapan aplikasi kasir pada Klinik Kecantikan Ultra Glow berbasis *React Native*.

2. LANDASAN TEORI

Aplikasi Kasir

Menurut Azizah & Putra (2024), aplikasi kasir adalah aplikasi yang telah dirancang khusus untuk platform mobile (misalnya iOS, android, atau windows mobile) untuk membantu pengguna mengelola transaksi. Dalam banyak kasus, aplikasi kasir memiliki *user interface* dengan mekanisme interaksi unik yang disediakan oleh platform mobile, interoperabilitas dengan sumber daya berbasis android yang menyediakan akses ke beragam informasi yang relevan dengan aplikasi, dan kemampuan pemrosesan lokal untuk pengumpulan, analisis, dan format informasi dengan cara yang paling cocok untuk platform mobile. Selain itu aplikasi mobile menyediakan kemampuan penyimpanan persisten dalam platform.

React Native

Menurut Lestari & Masitoh (2022), *React Native* digunakan untuk membuat aplikasi berbasis mobile, baik itu Android maupun iOS. *React Native* adalah sebuah *framework* berbasis JavaScript yang merupakan sekumpulan library berbasis JavaScript yang dikembangkan oleh Facebook. Syntax dari *React Native* merupakan gabungan antara JavaScript dan XML yang dapat disebut JSX atau TSX.

Firestore Database

Menurut Risqi et al. (2023), *Firestore Database* adalah layanan dari *Firestore*, platform pengembangan aplikasi mobile yang disediakan oleh Google. *Firestore Database* membantu pengembang untuk memahami dan menganalisis perilaku pengguna dalam menggunakan aplikasi mereka. Dengan *Firestore Analytics*, pengembang dapat melacak berbagai metrik penting seperti pengguna aktif, retensi pengguna, konversi, dan banyak lagi. Informasi ini sangat berharga dalam mengidentifikasi tren, memahami preferensi pengguna, dan mengoptimalkan pengalaman pengguna dalam aplikasi.

TypeScript

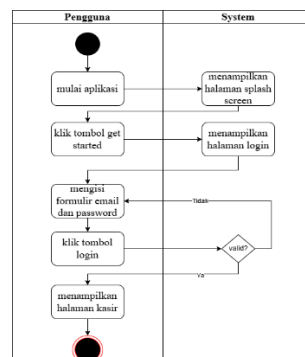
Menurut Bayu et al, (2023), TypeScript adalah bahasa pemrograman yang merupakan pengembangan (*superset*) dari JavaScript dan menambahkan fitur *static typing* (penentuan tipe data secara eksplisit) ke dalam JavaScript. TypeScript dikembangkan oleh Microsoft dan pertama kali dirilis pada tahun 2012. Kode TypeScript tidak bisa langsung dijalankan di browser, melainkan harus dikompilasi (*ditranspile*) terlebih dahulu menjadi JavaScript.

3. METODE PENELITIAN

Metode pengembangan sistem dilakukan melalui tahapan analisis kebutuhan, perancangan, implementasi, dan pengujian. Pengujian dilakukan dengan menggunakan metode black box testing. Menurut Putra (n.d), black box testing adalah metode pengujian perangkat lunak yang dilakukan dengan cara menguji fungsi sistem tanpa mengetahui atau melihat struktur kode program di dalamnya. Pengujian ini hanya berfokus pada input dan output berdasarkan spesifikasi kebutuhan (requirement).

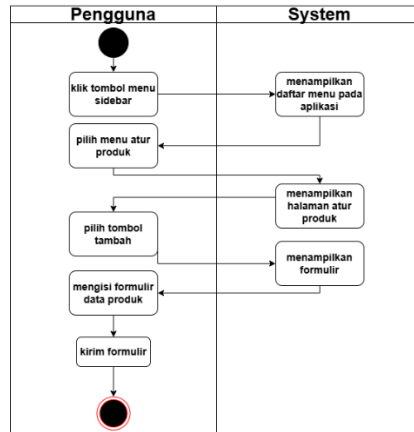
4. HASIL DAN PEMBAHASAN

Activity Diagram Bisnis Yang Berjalan



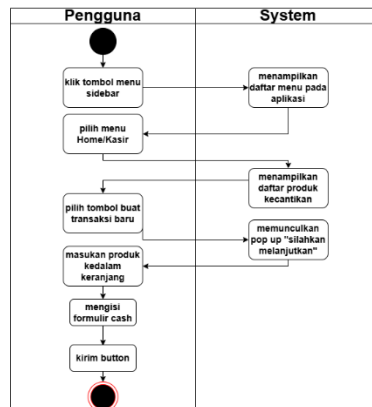
Gambar 1. Activity Login Pengguna

Activity atau aktivitas ini adalah aktivitas yang menunjukkan bagaimana langkah – langkah untuk sebuah proses *login* pada aplikasi kasir. Dimulai dengan pengguna mulai aplikasi, pengguna klik tombol get started pada halaman *splash screen*, pengguna masuk ke dalam halaman *login*, pengguna mengisi email dan *password* form, pengguna klik tombol *login*. Jika email dan *password* benar maka pengguna akan diarahkan ke halaman utama. Jika salah pengguna akan kembali pada halaman *login*.



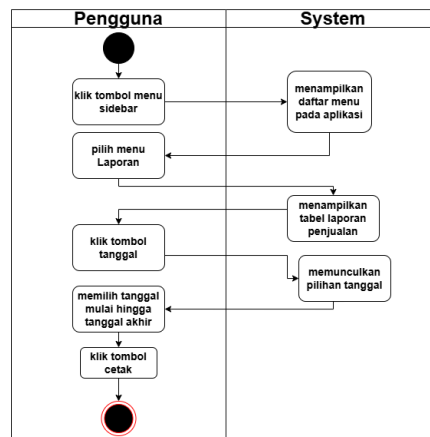
Gambar 2. Activity Tambah Produk

Activity atau aktivitas ini adalah aktivitas yang menunjukkan bagaimana langkah – langkah untuk sebuah proses tambah produk pada aplikasi kasir. Pengguna masuk ke dalam aplikasi, pengguna klik tombol menu atur produk, menampilkan halaman atur produk, pengguna memilih tombol tambah, menampilkan formulir data produk, pengguna harus mengisi lengkap formulir lalu mengirim formulir untuk menyimpan data produk baru.



Gambar 3. Activity Tambah Transaksi

Activity atau aktivitas ini adalah aktivitas yang menunjukkan bagaimana langkah – langkah untuk sebuah proses tambah transaksi atau buat pesanan baru pada aplikasi kasir. Pengguna masuk aplikasi, pengguna klik tombol menu sidebar, aplikasi menampilkan daftar menu sidebar, pengguna memilih menu *home* atau kasir, aplikasi menampilkan daftar produk kecantikan, pengguna memasukkan produk ke dalam keranjang, pengguna mengisi cash formulir lalu klik tombol kirim untuk proses transaksi.

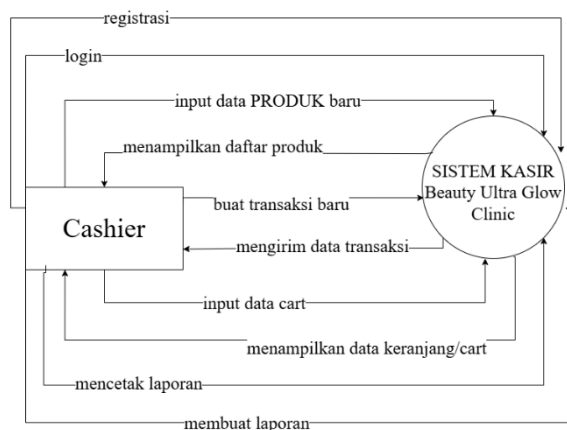


Gambar 4. Activity Cetak Laporan

Activity atau aktivitas ini adalah aktivitas yang menunjukkan bagaimana langkah – langkah untuk sebuah proses cetak laporan penjualan pada aplikasi kasir. Pengguna klik tombol menu *side bar*, aplikasi menampilkan daftar menu pada aplikasi, pengguna memilih menu laporan, aplikasi menampilkan tabel laporan penjualan, pengguna klik tombol tanggal, aplikasi memunculkan pilihan tanggal, pengguna memilih tanggal mulai hingga tanggal akhir, pengguna klik tombol cetak.

Diagram Konteks

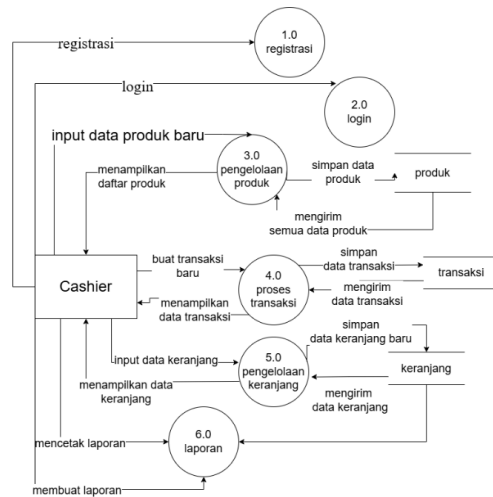
Diagram yang menjelaskan keseluruhan aliran data pada aplikasi sebelum dipecah menjadi bagian yang lebih kecil.



Gambar 5. Diagram Konteks

DFD (Data Flow Diagram)

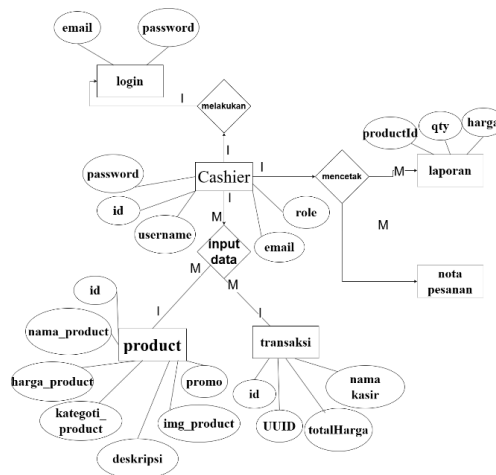
DFD level 0 adalah diagram pecahan awal dari diagram konteks. Pada diagram ini berisi proses-proses yang menjadi pusat aliran data pada aplikasi *cashier ultra glow clinic*



Gambar 6. DFD Level 0

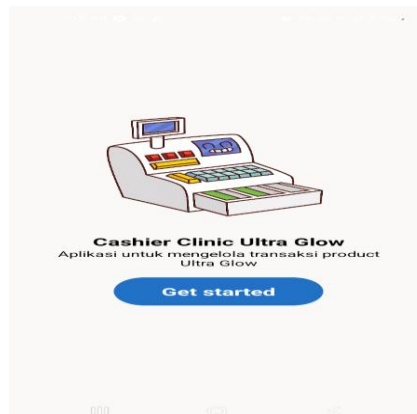
ERD (Entity Relationship Diagram)

Erd adalah diagram yang menggambarkan struktur *database* system, yang akan digunakan dalam pengembangan aplikasi.



Gambar 7. Entity Relationship Diagram

Hasil Implementasi Aplikasi



Gambar 8. Halaman *Splash Screen*

Halaman *splash screen* adalah halaman tampilan pertama pada sebuah aplikasi. Biasanya halaman ini berisi penjelasan singkat mengenai aplikasi.



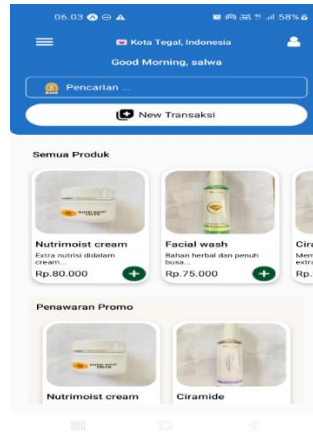
Gambar 9. Halaman *Login*

Pada halaman *login* pengguna harus memasukkan email dan *password* yang telah terdaftar untuk masuk ke dalam halaman utama *cashier*. Jika pengguna memasukkan akun yang belum terdaftar, pengguna tidak bisa melanjutkan untuk masuk ke dalam aplikasi. Pengguna akan diarahkan untuk mendaftar terlebih dahulu.



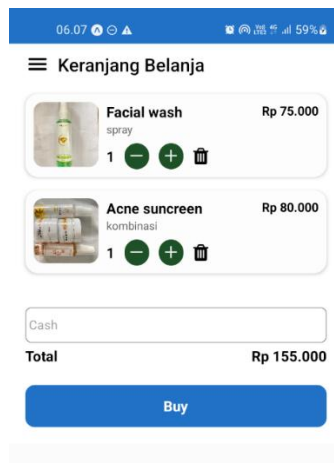
Gambar 10. Halaman Cek Email

Halaman cek email untuk lupa *password*, pada halaman ini pengguna diwajibkan untuk cek email terlebih dahulu untuk memastikan apakah email telah terdaftar di aplikasi. Jika email terdaftar pengguna akan diarahkan ke halaman ganti *password* untuk menyelesaikan fitur lupa *password*.



Gambar 11. Halaman *Home* atau Kasir

Halaman utama dari aplikasi ini adalah halaman yang menampilkan daftar produk kecantikan yang ada di Ultra Glow Clinic Beauty. Pada halaman ini pengguna dapat membuat transaksi baru dan memasukkan pesanan ke dalam keranjang halaman untuk memproses sebuah pesanan baru dari pelanggan.



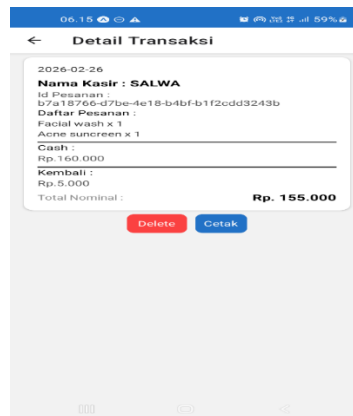
Gambar 12. Halaman Keranjang Belanja

Halaman keranjang belanja merupakan halaman yang menampung daftar pesanan pelanggan yang sebelumnya dimasukkan dari halaman utama, pengguna dapat mengurangi dan menambahkan pesanan pada halaman ini, sekaligus menghapus jika pesanan tidak jadi diproses. Untuk menyelesaikan pesanan pengguna harus mengisi formulir *cash* untuk menampung nominal yang dibayarkan oleh pelanggan, lalu baru klik tombol *buy* untuk menyelesaikan pesanan.



Gambar 13. Halaman Histori Pesanan

History pesanan berisi daftar pesanan pelanggan yang telah berhasil diproses. Pada halaman ini pengguna dapat melihat keseluruhan maupun detail dari pesanan dengan cara klik pesanan yang ingin dilihat, maka aplikasi akan melanjutkan ke halaman detail pesanan.



Gambar 14. Halaman Detail Transaksi

Setelah pengguna klik salah satu pesanan pada halaman histori pesanan, pengguna akan masuk ke dalam halaman detail pesanan. Pada halaman ini pengguna dapat melihat detail pesanan, menghapus pesanan hingga mencetak nota pesanan.



Gambar 15. Halaman Cetak Laporan

Halaman cetak laporan adalah halaman yang berisi tabel dari hasil penjualan *Ultra Glow Clinic Beauty*, pada halaman ini pengguna dapat melihat hasil penjualan dengan menentukan tanggal yang ingin dilihat, dengan cara klik tombol tanggal awal lalu pilih tanggal akhir untuk menentukan laporan yang ingin dilihat. Setelah pengguna melakukan filter tanggal, pengguna dapat langsung melakukan cetak data dengan klik tombol cetak.

Hasil Black Box Testing

Tabel 1. Hasil Pengujian Black Box Testing

No	Fungsi yang diuji	Kondisi	Output yang diharapkan	Status pengujian
1.	<i>Login</i>	<i>Login verifikasi email dan password</i>	Muncul notifikasi <i>login</i> berhasil dan pengguna diarahkan ke halaman utama	Sesuai Harapan
2.	Menambahkan produk baru	Mengisi semua formulir produk dan klik buat	Muncul notifikasi produk berhasil ditambahkan dan pengguna akan diarahkan ke halaman utama	Sesuai Harapan
3.	Mengubah data produk	Memilih produk yang ingin dirubah, merubah data produk di formulir	Muncul notifikasi produk berhasil dirubah, lalu diarahkan ke halaman utama	Sesuai Harapan
4.	Proses transaksi baru	Pengguna pilih menu tanpa klik tombol buat transaksi	Muncul notifikasi dengan pesan “buat pesanan terlebih dahulu”. Pesanan gagal ditambahkan	Sesuai Harapan
5.	Buat transaksi baru keranjang	Pengguna tidak mengisi formulir cash	Sistem akan menampilkan notifikasi “Tolong lengkapi formulir pada keranjang”	Sesuai harapan
6.	Tambah pengguna baru	Pengguna tidak mengisi semua formulir saat ingin menambahkan pengguna baru	Sistem akan menampilkan pemberitahuan formulir harus dilengkapi semua	Sesuai harapan

5. KESIMPULAN DAN SARAN

Berdasarkan pembahasan dan hasil pengujian maka dapat disimpulkan bahwa perancangan aplikasi kasir produk kecantikan dapat dibuat dengan menerapkan teknologi *Firestore realtime database* berbasis *React Native* karena *Firestore* memiliki banyak fitur seperti *authentication, database, storage, hosting*, pemberitahuan sehingga dapat mempermudah pelanggan dalam melakukan proses pengelolaan transaksi pada *Ultra Glow Clinic* tanpa harus menunggu lama. Melihat dari beberapa tahapan yang telah dilaksanakan serta hasil-hasil yang telah dicapai secara umum aplikasi ini telah bekerja cukup baik di mana kasir dapat melakukan pemrosesan transaksi tanpa harus mencatat menggunakan kertas dengan

cara manual sehingga dapat menghemat waktu dalam proses transaksi penjualan. Dengan menggunakan *Firebase* proses transfer data cepat serta aman. Saran pengembangan aplikasi adalah menambahkan fitur pembayaran menggunakan *e-banking*, *e-wallet* untuk mempermudah pelanggan dalam melakukan pembayaran, menambahkan fitur data pelanggan untuk mendata pelanggan-pelanggan tetap untuk diberikan potongan harga untuk pelanggan tetap.

DAFTAR REFERENSI

- Arya, K., Kesuma, B. W., Nyoman, I., Wijaya, Y. A., Gede, I., & Putra, J. E. (2023, October). Implementasi Next.js, Typescript, dan Tailwind CSS untuk pengembangan aplikasi frontend sistem inventory perusahaan apar (Studi kasus: CV Indoka Surya Jaya).
- Aziz, N., Pribadi, G., & Nurcahya, M. S. (2020). Analisa dan perancangan aplikasi pembelajaran bahasa Inggris dasar berbasis Android (Vol. 4, p. 3). November.
- Azizah, H., & Putra, S. D. (2024). Penerapan e-health pada sistem reservasi perawatan kulit wajah di klinik kecantikan dokter Mirda berbasis Android (Vol. 3, p. 2). May.
- Kurniawan, F., & Widiastuti, D. (2020). Pengembangan aplikasi manajemen keuangan berbasis Android menggunakan framework Flutter. *Jurnal Pengembangan Teknologi Informasi*, 3(4), 234–245. <https://doi.org/10.25077/jpti.v3i4.2020.234-245>
- Lestari, P. A., & Masitoh, A. H. (2022). Aplikasi Me-List berbasis Android menggunakan framework React Native. *Jurnal Esensi Infokom: Jurnal Esensi Sistem Informasi dan Sistem Komputer*, 6(2), 26–30. <https://doi.org/10.55886/infokom.v6i2.501>
- Malahella, A. H., & Arwani, I. (2020). Pemanfaatan framework React Native dalam pengembangan aplikasi pemesanan minuman kopi pada kedai Bycoffee. *Jurnal PTIIK UB*.
- Payara, G. R., & Tanone, R. (2022). Penerapan FirebaseRealtimeDatabase pada prototype aplikasi pemesanan makanan berbasis Android (Vol. 4, p. 3). December.
- Pramudya, P., & Saputra, D. (2023). Integrasi aplikasi sistem pembayaran digital dengan blockchain untuk e-commerce. *Jurnal Teknologi Informasi*, 9(1), 76–85. <https://doi.org/10.25077/jti.v9i1.2023.76-85>
- Prasetyo, S., & Irawan, H. (2022). Penerapan teknologi machine learning untuk analisis prediksi cuaca berbasis Android. *Jurnal Ilmu Komputer*, 7(2), 134–144. <https://doi.org/10.55878/jik.v7i2.2034>
- Putra, A. P., Andriyanto, F., Harti, T. D. M., & Puspitasari, W. (n.d.). Pengujian aplikasi point of sale berbasis web menggunakan black box testing.
- Risqi, R., Herlambang, A., & Zukhri, Z. (2023). Penggunaan Firebase Analytics pada pengembangan aplikasi mobile I'm UII dengan framework Flutter. March.
- Santosa, B., & Wijaya, R. F. (2021). Implementasi aplikasi pengelolaan data inventaris menggunakan framework Laravel. *Jurnal Teknologi dan Sistem Informasi*, 5(1), 45–55. <https://doi.org/10.25077/jtsi.v5i1.2021.45-55>

- Setiawan, I., & Ningsih, L. P. (2022). Pengembangan aplikasi sistem informasi pengelolaan tugas dengan menggunakan React JS. *Jurnal Sistem Informasi*, 8(3), 56–67. <https://doi.org/10.25077/jsi.v8i3.2022.56-67>
- Suryanto, M., & Anwar, A. (2021). Rancang bangun aplikasi pengingat jadwal kegiatan berbasis iOS dengan Swift. *Jurnal Aplikasi Teknologi*, 4(2), 101–110. <https://doi.org/10.25077/jat.v4i2.2021.101-110>
- Wiguna, P. D. A., Swastika, I. P. A., & Satwika, I. P. (2019). Rancang bangun aplikasi point of sales distro management system dengan menggunakan framework React Native. *Jurnal Nasional Teknologi dan Sistem Informasi*, 4(3), 149–159. <https://doi.org/10.25077/teknosi.v4i3.2018.149-159>