

Implementasi Aplikasi Android dalam Sistem *Restock* UMKM Maju Jaya Accessories

Nurrohmi Zaki ^{1*}, RR. Hajar Puji Sejati ²

^{1,2}Program Studi Informatika, Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta, Kabupaten Sleman, Provinsi Daerah Istimewa Yogyakarta, Indonesia.

Email: nurrohmi.zaki@gmail.com ^{1*}, hajarsejati@staff.uty.ac.id ²

Histori Artikel:

Dikirim 23 Oktober 2024; *Diterima dalam bentuk revisi* 14 November 2024; *Diterima* 20 Desember 2024; *Diterbitkan* 10 Januari 2025. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Indonesia Banda Aceh.

Abstrak

Penelitian ini bertujuan untuk meningkatkan efisiensi dan efektivitas proses restock di UMKM Maju Jaya Accessories melalui pengembangan aplikasi berbasis Android. Sebelumnya, UMKM ini menggunakan metode restock manual yang sering menyebabkan keterlambatan, kesalahan pengelolaan stok, dan komunikasi yang kurang efisien. Aplikasi yang dirancang memungkinkan toko mengajukan permintaan restock secara mandiri, yang kemudian dapat dipantau secara real-time oleh pemilik usaha. Metodologi penelitian menggunakan model Software Development Life Cycle (SDLC) Waterfall. Hasil penelitian menunjukkan peningkatan signifikan dalam ketepatan waktu pengelolaan stok, efisiensi proses restock, dan penurunan biaya operasional yang sebelumnya tinggi akibat kesalahan manual yang sebelumnya sering terjadi seperti kesalahan waktu kepada toko yang belum siap restock. Aplikasi dirancang dengan antarmuka yang sederhana dan mudah digunakan, sehingga dapat diakses oleh pengguna dari berbagai latar belakang. Namun, aplikasi ini masih terbatas untuk pelaku usaha dan mitra saja, tidak dapat digunakan secara umum. Kesimpulannya, penerapan aplikasi ini mampu meningkatkan efektivitas operasional dan memberikan nilai tambah bagi pemilik usaha serta mitra toko melalui pengelolaan stok yang lebih efisien dan terstruktur.

Kata Kunci: Android; Efisiensi; Stok; SDLC; UMKM.

Abstract

This research aims to improve the efficiency and effectiveness of the restocking process at UMKM Maju Jaya Accessories through the development of an Android-based application. Previously, the UMKM used a manual restocking method, which often caused delays, stock management errors, and inefficient communication. The designed application allows stores to independently submit restock requests, which can then be monitored in real-time by the business owner. The research methodology uses the Software Development Life Cycle (SDLC) Waterfall model. The results show a significant improvement in stock management timeliness, restocking process efficiency, and a reduction in operational costs that were previously high due to frequent manual errors, such as timing mistakes for stores that were not ready to restock. The application is designed with a simple and user-friendly interface, making it accessible to users from various backgrounds. However, the application is still limited to business owners and partners only and cannot be used publicly. In conclusion, the implementation of this application improves operational effectiveness and provides added value for business owners and partner stores through more efficient and structured stock management.

Keyword: Android; Efficiency; Restock; SDLC; UMKM.

1. Pendahuluan

Maju Jaya *Accessories* merupakan Usaha Mikro Kecil dan Menengah (UMKM) yang didirikan di Kabupaten Pemalang, Jawa Tengah, lebih dari satu dekade lalu. UMKM adalah entitas usaha produktif yang dimiliki oleh individu atau badan usaha sesuai dengan kriteria yang diatur dalam Undang-Undang No. 20 Tahun 2008 (Hidayat *et al.*, 2022). Usaha ini berfokus pada penyediaan aksesoris dengan harga terjangkau untuk memenuhi kebutuhan pasar, terutama melalui distribusi ke toko-toko kelontong. Sistem operasional yang masih menggunakan metode manual, seperti kunjungan langsung ke toko berdasarkan catatan restock, kerap menghadirkan kendala. Variasi waktu penjualan barang di setiap toko sering kali menyebabkan restock dilakukan terlalu dini atau terlambat. Dalam era digital, UMKM perlu mengadopsi teknologi untuk meningkatkan efisiensi dan daya saing (Alno Rifani *et al.*, n.d.). Untuk menjawab kebutuhan ini, pengembangan sistem berbasis teknologi diperlukan. Salah satu bidang yang terus berkembang adalah teknologi mobile, khususnya perangkat berbasis Android. Platform Android merupakan salah satu sistem yang paling banyak digunakan dan memiliki dampak signifikan dalam mempermudah akses informasi (Frinando & Artaye, 2021; Nurkamid & Budiman, 2021). Aplikasi berbasis Android yang memungkinkan toko mengajukan restock secara mandiri diharapkan dapat mempermudah pengusaha dalam mengelola pengiriman barang, sehingga proses menjadi lebih efisien. Penelitian ini membahas perancangan sistem laporan berbasis Android yang dirancang untuk memudahkan pengusaha dan pemilik toko dalam kegiatan restock. Pengembangan aplikasi dilakukan menggunakan Android Studio dan bahasa pemrograman Kotlin. Android Studio, sebagai *Integrated Development Environment* (IDE) yang dirancang khusus untuk Android, menawarkan fitur seperti sistem build berbasis Gradle, emulator dengan performa tinggi, dan integrasi langsung untuk pengembangan aplikasi (Nasution *et al.*, 2019). Kotlin, sebagai bahasa pemrograman yang berjalan pada *Java Virtual Machine* (JVM), memungkinkan pengembangan aplikasi yang terintegrasi dengan bahasa Java dan mendukung konsep pemrograman berorientasi objek maupun fungsional (Toyib Hidayat & Gede Olka Santoso, 2023).

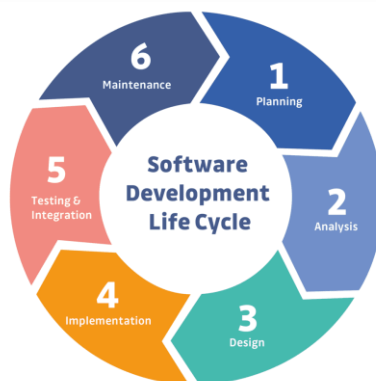
Sistem serupa telah digunakan dalam beberapa penelitian sebelumnya. Contohnya adalah penelitian oleh Mustakim dan Setiawati (2022), yang menggunakan metode prototipe untuk merancang aplikasi berbasis web guna memantau stok barang. Selain itu, penerapan aplikasi akuntansi berbasis Android seperti Si Apik di UMKM menunjukkan kemampuan teknologi ini dalam mempermudah pencatatan keuangan dan pengelolaan data (Hidayah *et al.*, n.d.). Penelitian ini juga berfokus pada solusi berbasis Android untuk meningkatkan efisiensi proses restock, khususnya dalam meminimalkan kesalahan operasional dan meningkatkan keakuratan pengelolaan stok. Tujuan utama penelitian ini adalah untuk membantu pengusaha mengidentifikasi kebutuhan restock secara akurat dan efisien, serta menentukan jenis barang yang perlu direstock dengan tepat. Diharapkan pengelolaan stok yang lebih terstruktur dapat mengurangi risiko kerugian akibat kekurangan atau kelebihan stok. Selain itu, optimalisasi pengelolaan stok juga diharapkan dapat meningkatkan efektivitas operasional dan kepuasan pelanggan melalui ketersediaan barang yang lebih terjamin.

2. Metode Penelitian

Peneliti menggunakan metode SDLC karena tahapan-tahapannya yang runtut dan teratur, sehingga lebih mudah dalam merencanakan setiap langkah pengembangan aplikasi. Selain itu, karena kebutuhan aplikasi sudah diketahui sejak awal, setiap tahap bisa dijabarkan dengan lebih detail.

2.1 Tahapan Penelitian

Penelitian ini menggunakan metode *Software Development Life Cycle* (SDLC) sebagai metode untuk pembuatan aplikasi pada sistem dengan langkah – langkah seperti pada gambar berikut:



Gambar 1. Tahapan Metode SDLC

Metode SDLC *Waterfall* adalah pendekatan yang memiliki ciri khas di mana setiap fase harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase berikutnya. Dengan cara ini, hasil dari setiap fase dapat lebih terfokus dan maksimal, karena tidak ada pengerjaan yang dilakukan secara bersamaan (Nugraha *et al.*, 2018). Tahapan metode SDLC sebagai berikut:

1) *Planning*

Tahap *planning* dalam SDLC adalah langkah awal yang sangat penting, di mana tim pengembang menetapkan tujuan, cakupan, dan strategi untuk pengembangan sistem. Pada fase ini, kebutuhan pengguna dan sasaran bisnis diidentifikasi dan dianalisis untuk memastikan bahwa sistem yang akan dikembangkan sesuai dengan harapan pengguna. Tahap ini juga berfokus pada pengaturan sumber daya yang diperlukan, penyusunan anggaran, dan pembentukan tim proyek yang akan terlibat dalam seluruh proses pengembangan (Pangestu *et al.*, 2012).

2) *Analysis*

Tahap ini bertujuan untuk mengenali dan menganalisis kebutuhan pengguna serta bisnis yang akan dipenuhi oleh sistem. Dalam proses ini, dilakukan pengumpulan informasi melalui wawancara atau observasi dengan pemangku kepentingan untuk mendapatkan data yang diperlukan. Analisis kebutuhan ini menjadi dasar utama dalam pengembangan sistem informasi, dengan fokus pada pengumpulan kebutuhan secara rinci untuk mengarahkan tahapan pengembangan berikutnya (Bariah & Pradina, n.d.).

3) *Design*

Tahap desain dalam SDLC adalah proses kunci yang berfokus pada pengembangan spesifikasi teknis dan arsitektur sistem berdasarkan kebutuhan yang telah diidentifikasi pada fase analisis. Dalam tahap ini, perancang sistem mengembangkan struktur dan komponen utama, seperti antarmuka pengguna dan model data, untuk memastikan bahwa sistem akan berfungsi sesuai dengan tujuan bisnis. Desain ini nantinya akan menjadi panduan untuk proses pengkodean, di mana elemen-elemen yang dirancang diterjemahkan ke dalam perangkat lunak atau aplikasi (Nugraha *et al.*, 2018).

4) *Implementation*

Tahap implementasi sistem adalah fase di mana rancangan yang telah disusun pada tahap-tahap sebelumnya mulai diterapkan atau diwujudkan. Pada fase ini, sistem yang direncanakan mulai dikembangkan dan diintegrasikan ke dalam lingkungan operasional, dengan tujuan memastikan bahwa semua komponen berfungsi dengan baik dan sesuai dengan spesifikasi yang telah ditentukan. Implementasi mencakup pengkodean, pengujian, dan pengaturan infrastruktur yang diperlukan untuk mendukung sistem yang baru (Prayudi, 2018).

5) *Testing*

Pengujian adalah proses di mana sebuah sistem atau komponen dijalankan dalam kondisi tertentu, dengan hasil yang diamati atau dicatat untuk kemudian dievaluasi berdasarkan berbagai aspek dari sistem atau komponen tersebut (Masripah & Ramayanti, 2019).

6) *Maintenance*

Setelah perangkat lunak selesai dikembangkan, tahap berikutnya adalah menjalankannya dan melakukan pemeliharaan. Pemeliharaan ini mencakup perbaikan kesalahan atau bug yang mungkin tidak terdeteksi pada tahap pengujian sebelumnya.

2.2 Pengumpulan Data

Pengumpulan data merupakan tahap penting dalam penelitian ini untuk memastikan seluruh kebutuhan sistem dapat teridentifikasi dengan baik. Data yang dikumpulkan mencakup informasi yang relevan dari berbagai sumber, seperti pemilik toko mitra dan pemasok barang. Pendekatan ini bertujuan untuk mendapatkan gambaran menyeluruh mengenai kebutuhan pengguna dan spesifikasi teknis sistem yang akan dikembangkan.

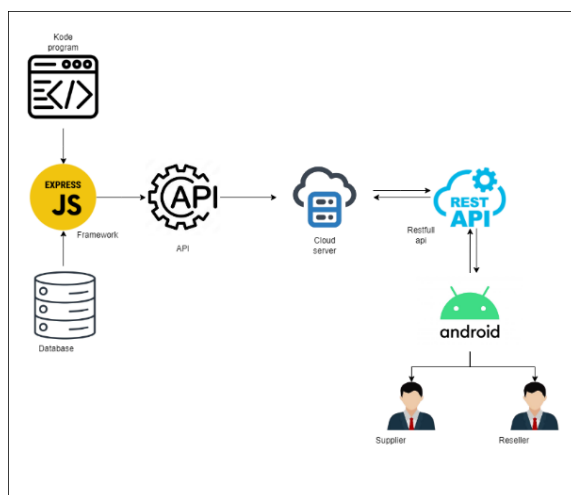
Tabel 1. Sumber Data

No	Data	Keterangan	Sumber
1	Reseller	Informasi lengkap tentang data pemilik toko yang bermitra pada UMKM Maju Jaya <i>Accessories</i>	Reseller
2	Barang	Informasi lengkap tentang jenis dan jumlah barang	Supplier dan Reseller
3	Restock	Riwayat kegiatan <i>restock</i> seperti waktu dan jumlah barang keluar masuk	Supplier dan Reseller

Sumber data diperoleh dari wawancara pada supplier dan beberapa reseller yang sudah menjadi mitra UMKM Maju Jaya *Accessories*, dengan data yang sudah dikumpulkan maka, peneliti dapat melanjutkan pengembangan aplikasi sesuai kebutuhan.

2.3 Arsitektur Model

Aplikasi ini menggunakan sistem aplikasi berbasis android dengan arsitektur model sebagai berikut:



Gambar 2. Arsitektur Model

Arsitektur model menunjukkan interaksi antar komponen dalam sistem perangkat lunak. Kode program mencakup *framework* yang mengelola *database* untuk membuat *endpoint* dengan API, kemudian API di *hosting* ke *cloud server* yang kemudian akan masuk ke arsitektur REST API yang bisa melakukan *request* dan *response* ke *cloud server* dari aplikasi android yang digunakan oleh 2 entitas yaitu *reseller* dan *supplier*.

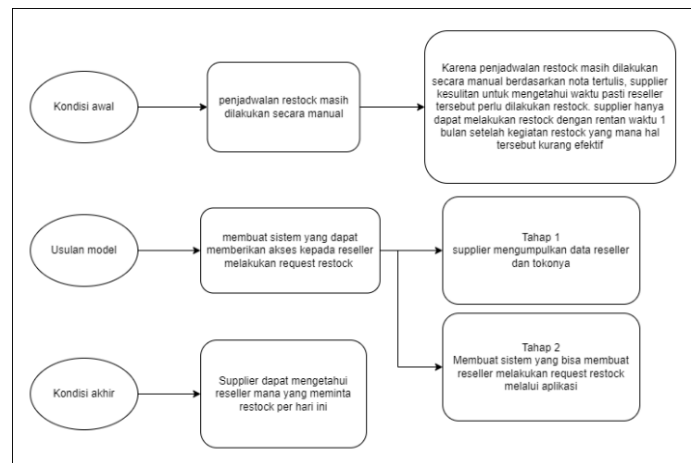
3. Hasil dan Pembahasan

3.1 Hasil

3.1.1 Implementasi Metode

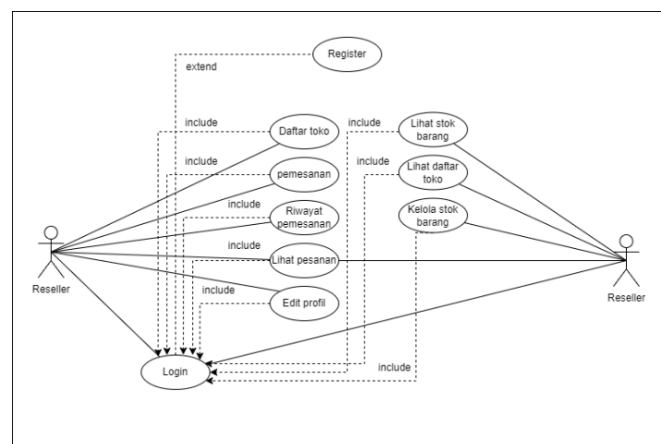
1) *Planning*

Memahami konsep sistem dan masalah yang ingin dipecahkan mencakup penilaian terhadap kelayakan sistem baru dalam hal biaya, waktu, dan perbandingannya dengan sistem yang ada saat ini. Evaluasi ini bertujuan untuk menentukan apakah pengembangan sistem baru tersebut realistis dan sesuai dengan kebutuhan serta keterbatasan yang ada (Dwanoko, n.d.). Pada penelitian ini sistem yang direncanakan bisa menjawab masalah efektifitas pada masalah waktu dan biaya *transport* saat kegiatan *restock* dilakukan. *Planning* atau rencana dari sistem ini dibuat dalam bentuk kerangka penelitian seperti berikut:



Gambar 3. Kerangka Penelitian

Kerangka penelitian pada gambar 3 mendeskripsikan tentang kondisi dari awal sebelum adanya sistem laporan *restock* ini dibuat. Usulan model berisi tahapan ketika proses *restock* berjalan menggunakan sistem. Kondisi akhir merupakan hasil dari penelitian. Kerangka penelitian diperlukan agar alur sistem lebih mudah dipahami. Sedangkan untuk rencana alur jalannya sistem digambarkan dengan diagram *usecase* seperti gambar 4.



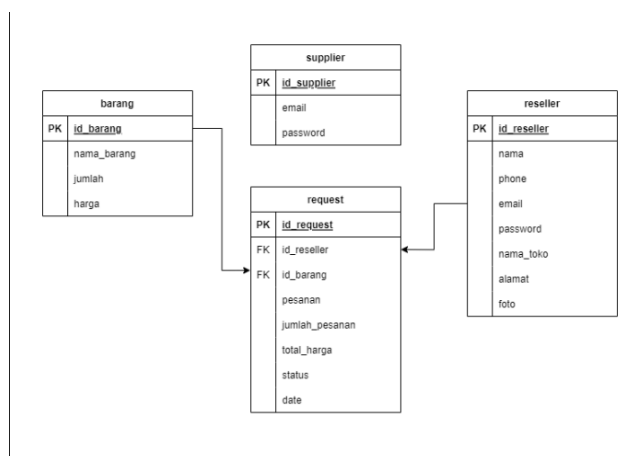
Gambar 4. Diagram *Usecase*

2) *Analysis*

Tahap analisis dan pengumpulan kebutuhan sistem melibatkan pengumpulan informasi yang relevan untuk memahami kebutuhan yang harus dipenuhi oleh sistem. Pada tahap ini, dilakukan penelitian mendalam untuk mengidentifikasi dan mendokumentasikan semua kebutuhan fungsional maupun non-fungsional yang diperlukan dalam pengembangan sistem. Menentukan spesifikasi kebutuhan fungsional dan non-fungsional dari sistem. Pada penelitian ini informasi tentang kebutuhan didapat berdasarkan arsitektur model pada gambar 2, dan di dapat kebutuhan fungsional yaitu ;a) kebutuhan *inputan* seperti *input* data *reseller*, *input* data toko, *input* data barang, *input* pesanan reseller; b) kebutuhan proses seperti registrasi dan login, proses request, proses konfirmasi, proses *update* barang; c) kebutuhan luaran seperti stok barang, riwayat pemesanan, dan daftar toko. Kemudian ada kebutuhan non-fungsional seperti Visual Studio Code, XAMPP, Android Studio, Figma, Draw.io, Postman dan kebutuhan perangkat keras untuk perancangan yaitu laptop MSI GF63 Thin dan ponsel dengan OS android 13.

3) *Design*

Secara garis besar, tahap ini merupakan tahap perancangan sistem, di mana informasi umum tentang rencana pengembangan sistem disusun. Pada tahap ini, dilakukan analisis terhadap kebutuhan-kebutuhan yang diperlukan untuk memastikan sistem dapat dibangun sesuai tujuan dan spesifikasi yang diharapkan (Rahmadani *et al.*, 2019). Merancang arsitektur sistem serta antarmuka aplikasi yang user-friendly dan fungsional. Arsitektur terkait flowchart, database, serta wireframe dan rancangan antarmuka dibuat pada tahap ini. Perancangan database pada sistem ini tergambar pada susunan relasi tabel seperti gambar berikut:



Gambar 5. Relasi Tabel

4) *Implementation*

Pada tahap implementasi, desain sistem yang telah dibuat pada tahap perancangan dikembangkan menjadi kode program. Proses ini melibatkan pengkodean seluruh komponen yang diperlukan hingga sistem siap dijalankan dan diuji fungsionalitasnya (Romindo & Christine, n.d.). pada tahap ini penulis membangun aplikasi berdasarkan desain yang telah dibuat, pengembangan *front-end* menggunakan Android Studio dan Kotlin untuk aplikasi android yang digunakan oleh *reseller* dan *supplier*, pengembangan *back-end* menggunakan Express.js untuk menyediakan API untuk aplikasi dan di integrasi melalui REST API.

3.1.2 Implementasi Antarmuka

Aplikasi ini dirancang dengan konsep multi-user, yang dimana akan terdapat 2 jenis user yang berbeda yaitu *supplier* dari pihak Maju Jaya *Accessories* dan dari pihak *resellernya*. Berikut fitur – fitur yang ada pada aplikasi ini:

1) *On boarding*

Saat aplikasi pertama kali dibuka maka pengguna akan mendapatkan pilihan untuk membuat Akun baru atau langsung masuk ke aplikasi dengan Akun yang sudah ada, tampilannya seperti gambar berikut:

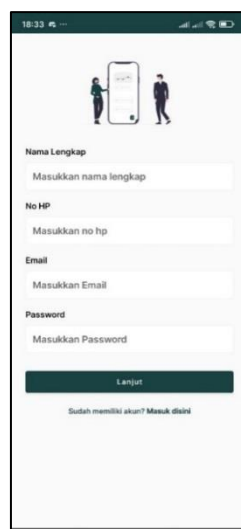


Gambar 6. Halaman *on boarding*

Pada halaman ini, pengguna dapat memilih opsi "Buat Akun Baru" untuk membuat akun yang digunakan untuk mengakses aplikasi. Jika pengguna sudah memiliki akun, mereka bisa langsung menuju halaman *login* dengan memilih menu "Sudah Punya Akun".

2) *Register*

Saat pengguna memilih menu buat Akun baru, maka pengguna akan di arahkan ke halaman pendaftaran yang nantinya data Akun yang diisi akan digunakan untuk masuk ke aplikasi, berikut tampilannya.

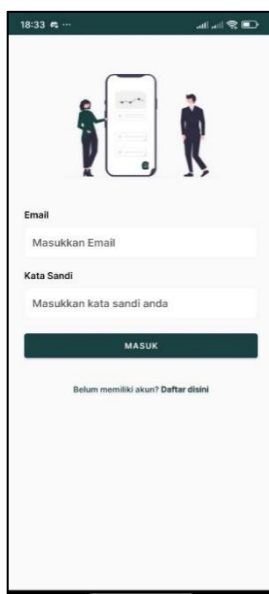


Gambar 7. Halaman *register*

Pada halaman ini, pengguna dapat mendaftarkan diri untuk membuat akun yang akan digunakan untuk mengakses aplikasi. Proses pendaftaran dilakukan dengan cara mengisi formulir yang meminta data diri pengguna, seperti nama lengkap, nomor *handphone*, alamat *email*, serta kata sandi yang akan digunakan untuk *login* ke dalam aplikasi. Semua informasi yang diisi dalam formulir ini akan disimpan dan digunakan untuk keperluan autentikasi pengguna agar dapat mengakses berbagai fitur yang tersedia di aplikasi.

3) *Login*

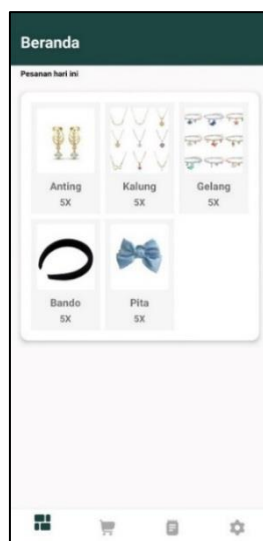
Setelah pengguna mempunyai akun, maka pengguna sudah bisa masuk ke aplikasi menggunakan Akun tersebut, berikut tampilannya.



Gambar 8. Halaman *Login*

4) *Dashboard reseller*

Ketika pengguna berhasil masuk ke aplikasi dengan mengisi data *reseller*, maka akan ditampilkan halaman beranda untuk *reseller* seperti gambar berikut:

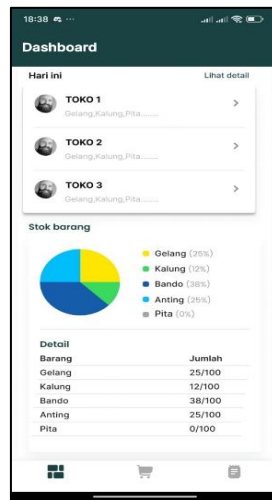


Gambar 9. Halaman beranda *reseller*

Pada halaman ini, pengguna akan melihat informasi mengenai pesanan yang dilakukan pada hari ini. Daftar pesanan yang ditampilkan akan mencakup semua pesanan yang telah dibuat oleh pengguna sesuai dengan permintaan yang diajukan saat melakukan proses pemesanan sebelumnya. Setiap item dalam daftar pesanan akan menunjukkan rincian yang relevan, seperti nama produk, jumlah pesanan, serta status terkini dari setiap pesanan yang telah diminta oleh pengguna. Informasi ini disusun untuk memudahkan pengguna dalam memantau dan memverifikasi pesanan mereka.

5) *Dashboard supplier*

Ketika pengguna berhasil masuk ke aplikasi dengan data *supplier*, maka akan ditampilkan halaman beranda untuk *supplier* seperti gambar berikut:

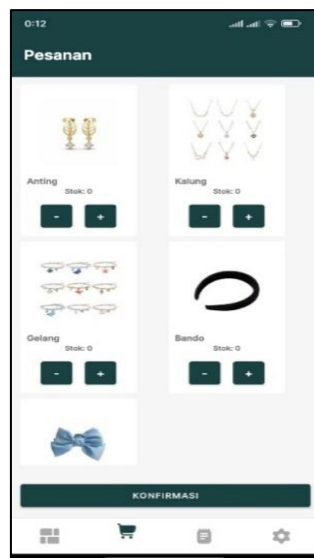


Gambar 10. Halaman beranda *supplier*

Pada halaman ini akan ditampilkan informasi tentang permintaan atau *request* dari *reseller* yang masuk hari ini, dan informasi stok barang yang tersedia dari *supplier*.

6) *Request*

Request adalah fitur utama dari sistem ini, dimana pengguna dapat melakukan *request* atau permintaan untuk *restock* kepada *supplier* untuk tampilan halamannya seperti berikut:

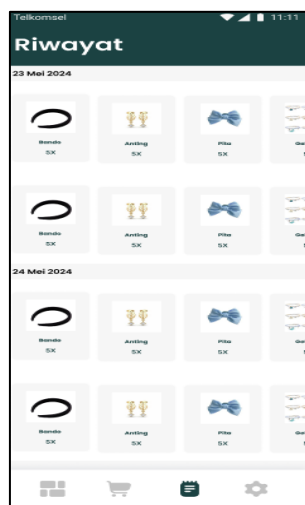


Gambar 11. Halaman *request*

Pada halaman ini, pengguna memiliki kesempatan untuk melakukan permintaan *restock* dengan cara menambah item yang diperlukan sesuai dengan kebutuhan mereka. Pengguna dapat memilih dan menentukan jenis barang yang ingin *diorder* untuk memperbarui stok yang ada. Setelah permintaan tersebut diajukan, data pesanan akan langsung diteruskan ke pihak *supplier* terkait, yang kemudian akan memproses permintaan tersebut dan melakukan *restock* sesuai dengan item yang telah diminta. Dengan cara ini, proses *restocking* dapat berjalan lebih efisien, dan pengguna dapat memastikan ketersediaan barang yang dibutuhkan tepat waktu.

7) *History reseller*

Fitur *history* ini digunakan untuk mencatat kegiatan *restock* yang sudah terjadi, untuk halamannya seperti berikut:

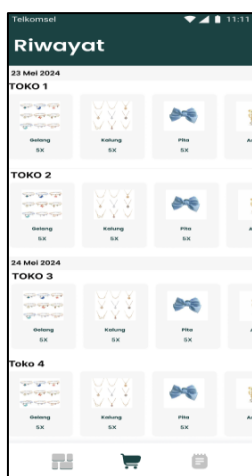


Gambar 12. Halaman *history reseller*

Pada halaman ini pengguna bisa melihat catatan stok barang yang di *restock* sesuai permintaan pada waktu sesuai yang tercatat saat melakukan *request*, catatan ini bisa berguna untuk melihat barang apa saja yang kira – kira tidak terlalu laku pada toko tersebut sehingga bisa mengganti dengan barang yang lain.

8) *History supplier*

Sama seperti *history* pada *reseller*, fitur ini digunakan untuk mencatat kegiatan *restock* yang sudah terjadi, tampilan halamannya seperti berikut:



Gambar 13. Halaman *history supplier*

Pada halaman ini *supplier* bisa melihat catatan *restock* yang sudah terjadi, catatan ini memuat informasi tentang nama toko, waktu *restock* dan barang apa saja yang di *restock* pada waktu itu.

9) *Edit profile*

Edit profile adalah fitur umum yang ada pada aplikasi dengan akun untuk masuk ke dalam aplikasinya, untuk tampilannya seperti berikut:



Gambar 14. Halaman edit profile

Dengan fitur *edit profile* ini, pengguna dapat mengubah data Akun dengan mengisi *form* yang tersedia pada halaman *edit profile*, setelah pengguna mengisi *form* data tersebut, maka informasi tentang data Akun tersebut akan berubah sesuai isi *form* yang diisi.

3.1.3 Pengujian

Pengujian aplikasi dilakukan dengan metode *Blackbox Testing* dengan menguji fungsionalitas aplikasi seperti pada tabel 1 dan *User Acceptance Test* atau *UAT* yaitu dengan cara menguji langsung kepada calon pengguna aplikasi tersebut, pada kasus ini kedua pihak dari sisi *supplier* dan *reseller* mencoba fitur – fitur dari aplikasi ini. Berikut hasil pengujiannya terdapat pada tabel 2.

Tabel 2. Hasil Pengujian *Blackbox Testing*

No	Fungsi yang Diuji	Skenario Pengujian	Input	Output yang Diharapkan	Status
1.	Login Pengguna	Pengguna memasukkan username dan password yang benar	Username dan password valid	Pengguna berhasil masuk ke dalam aplikasi	Berhasil
2.	Login Gagal	Pengguna memasukkan username atau password yang salah	Username / password salah	Muncul pesan kesalahan “Username atau password salah”	Berhasil
3.	Permintaan Restock	Pengguna mengajukan permintaan restock barang	Data barang dan jumlah stok	Permintaan restock berhasil dikirim	Berhasil

4.	Validasi Data Restock	Pengguna mengirim form restock dengan data kosong	Data kosong	Muncul notifikasi "Harap isi semua kolom yang tersedia"	Berhasil
5.	Real-time Monitoring	Pemilik usaha memantau status restock melalui aplikasi	Tidak ada input (akses dashboard)	Data permintaan restock tampil secara real-time	Berhasil
6.	Logout	Pengguna memilih opsi logout dari aplikasi	Tombol logout diklik	Pengguna berhasil keluar dan kembali ke halaman login	Berhasil
7.	Navigasi Menu	Pengguna berpindah dari satu menu ke menu lainnya	Tombol menu ditekan	Menu berpindah sesuai tombol yang diklik	Berhasil
8.	User-Friendly Interface	Pengguna berinteraksi dengan aplikasi	Sentuhan pada tombol atau input	Semua elemen antarmuka berfungsi dengan baik	Berhasil

Tabel 3. Hasil Pengujian *User Acceptance Test*

No	Skenario	Hasil Yang Diharapkan	Hasil Uji Coba
Role User: Reseller			
1.	Halaman On Boarding, klik button Buat Akun Baru	Menampilkan halaman register.	Sesuai
2.	Halaman On Boarding, klik button Sudah Punya Akun	Menampilkan halaman login	Sesuai
3.	Halaman Register, klik button lanjut	Menampilkan halaman login	Sesuai
4.	Halaman Register, klik text button masuk disini	Menampilkan halaman login	Sesuai
5.	Halaman Login, klik button masuk	Berhasil login dan masuk ke halaman beranda	Sesuai
6.	Halaman Login, klik text button Daftar Disini	Menampilkan halaman register	Sesuai
7.	Halaman Dashboard	Menampilkan informasi barang yang sedang di restock hari ini	Sesuai
8.	Halaman Dashboard, klik simbol keranjang (fitur lainnya)	Menampilkan halaman request	Sesuai
9.	Halaman Dashboard, klik simbol nota (fitur lainnya)	Menampilkan halaman history	Sesuai
10.	Halaman Dashboard, klik simbol setting	Menampilkan halaman setting	Sesuai
11.	Halaman request, klik button plus dan minus	Menambah atau mengurangi jumlah item yang akan di restock	Sesuai
12.	Halaman request, klik simbol beranda	Menampilkan halaman dashboard	Sesuai
13.	Halaman request, klik simbol nota (fitur lainnya)	Menampilkan halaman history	Sesuai
14.	Halaman request, klik	Menampilkan halaman setting	Sesuai

	simbol setting		
15.	Halaman history	Menampilkan catatan restock yang sudah terjadi	Sesuai
16.	Halaman history, klik simbol beranda	Menampilkan halaman dashboard	Sesuai
17.	Halaman history, klik simbol keranjang (fitur lainnya)	Menampilkan halaman request	Sesuai
18.	Halaman history, klik simbol setting	Menampilkan halaman setting	Sesuai
19.	Halaman setting, klik button edit profile	Mengubah data pengguna sesuai isi form	Sesuai
20.	Halaman setting, klik simbol beranda	Menampilkan halaman dashboard	Sesuai
21.	Halaman setting, klik simbol keranjang (fitur lainnya)	Menampilkan halaman request	Sesuai
22.	Halaman setting, klik simbol nota (fitur lainnya)	Menampilkan halaman history	Sesuai
Role User: Supplier			
1.	Halaman on boarding, klik button sudah punya akun	Menampilkan halaman login	Sesuai
2.	Halaman login, klik button login	Menampilkan halaman dashboard	Sesuai
3.	Halaman dashboard	Menampilkan informasi toko yang melakukan request hari ini dan stok barang yang tersedia	Sesuai
4.	Halaman dashboard, klik simbol nota (fitur lainnya)	Menampilkan halaman history	Sesuai
5.	Halaman dashboard, klik simbol setting	Menampilkan halaman setting	Sesuai
6.	Halaman history	Menampilkan informasi nama toko, waktu dan barang apa saja yang sudah di restock pada waktu yang tertera	Sesuai
7.	Halaman setting, klik button edit profile	Mengubah data pengguna sesuai isi form	Sesuai
8.	Halaman setting, klik simbol beranda	Menampilkan halaman dashboard	Sesuai
9.	Halaman setting, klik simbol nota (fitur lainnya)	Menampilkan halaman history	Sesuai

Setelah pengujian *Blackbox Testing* dan *User Acceptance Test (UAT)* berhasil dan sistem beroperasi dengan baik, sistem siap diterapkan sepenuhnya dalam lingkungan operasional. Pada titik ini, semua fitur yang telah diuji telah *diverifikasi* oleh calon pengguna aplikasi dan memenuhi kriteria yang ditetapkan, memastikan bahwa sistem dapat memberikan manfaat maksimal bagi pengguna akhir. Pengguna kini dapat mengakses dan memanfaatkan sistem sesuai dengan tujuan yang telah ditetapkan. Pengujian tersebut juga memberikan gambaran bagaimana sistem berfungsi dalam kondisi sebenarnya dan mengidentifikasi area yang mungkin membutuhkan perbaikan lebih lanjut. Tahap selanjutnya adalah implementasi penuh sistem ke dalam lingkungan operasional yang sesungguhnya. Sistem akan diterapkan pada kegiatan *restock* skala kecil dari UMKM Maju Jaya *Accessories*, Pada fase ini, pengembang akan melakukan pengawasan terus-menerus untuk memastikan kinerja sistem tetap stabil dan sesuai dengan yang diharapkan. Ini juga mencakup dukungan berkelanjutan untuk menyelesaikan masalah yang mungkin muncul setelah peluncuran, serta pemeliharaan rutin untuk

mengatasi *bug* atau menambahkan fitur baru berdasarkan masukan pengguna. Pemeliharaan pasca-*implementasi* sangat krusial untuk memastikan sistem tetap relevan dan dapat memenuhi kebutuhan pengguna secara berkelanjutan. Pengembang akan terus memantau dan memperbaiki masalah yang mungkin tidak terdeteksi sebelumnya, serta melakukan pembaruan atau peningkatan sesuai dengan masukan dari pengguna. Untuk pengembangan selanjutnya pengembang akan membuat sistem ini dapat digunakan pada sistem operasi selain android, karena keterbatasan sistem ini hanya bisa digunakan pada sistem operasi android. Selanjutnya, sistem diuji dengan kedua metode pengujian yaitu *blackbox testing* dan *UAT*, setelahnya sistem diuji coba langsung dalam keberlangsungan *restock* pada toko, berikut hasil uji cobanya tertera pada tabel 4.

Tabel 4. Hasil uji coba *restock* took

Minggu	Penjualan sebelum menggunakan sistem (toko)	Penjualan setelah menggunakan sistem (toko)
1	3	5
2	4	6
3	3	6
4	3	5
Total	13	22
Rata – rata	3,25	5,5

3.2 Pembahasan

Penelitian ini memiliki beberapa kesamaan dan perbedaan dengan penelitian sebelumnya dalam hal metode pengembangan, teknologi yang digunakan, dan fokus implementasi. Dari segi metode pengembangan, penelitian ini menggunakan pendekatan *SDLC model Waterfall*, serupa dengan penelitian Mustakim dan Setiawati (2022) yang menggunakan metode prototipe untuk merancang aplikasi manajemen stok berbasis web. Namun, pendekatan *SDLC* memberikan keunggulan melalui proses pengembangan yang lebih terstruktur dan sistematis dibandingkan metode prototipe. Dari sisi teknologi, penelitian yang dilakukan menitikberatkan pada pengembangan aplikasi berbasis *Android* untuk pengelolaan stok, serupa dengan penelitian Frinardo dan Artaye (2021) yang mengembangkan aplikasi marketplace berbasis *Android*. Namun, penelitian ini memiliki fokus yang lebih spesifik pada pengelolaan *restock* dibandingkan transaksi penjualan. Penelitian tersebut berbeda dengan penelitian Hidayah *et al.* (n.d.), yang mengembangkan aplikasi *SI APIK* untuk pencatatan keuangan UMKM. Penelitian ini berkontribusi pada efisiensi operasional melalui pengelolaan *restock* yang lebih akurat dan *real-time*, memberikan solusi yang lebih spesifik untuk kebutuhan inventori UMKM. Dalam hal pengujian, metode *Blackbox Testing* dan *User Acceptance Test (UAT)* yang digunakan dalam penelitian ini juga sejalan dengan penelitian Masripah dan Ramayanti (2019), meskipun fokus penelitian adalah pada penerapan aplikasi berbasis *Android*, bukan sistem berbasis web. Dampak implementasi aplikasi ini terhadap UMKM juga menunjukkan hasil yang signifikan, serupa dengan penelitian Alno Rifani *et al.* (n.d.), yang menekankan peran teknologi informasi dalam meningkatkan efisiensi operasional UMKM. Penelitian ini menawarkan kontribusi baru dengan menghadirkan aplikasi berbasis *Android* yang dirancang khusus untuk mempermudah proses *restock* pada UMKM. Meskipun terdapat kesamaan dengan penelitian sebelumnya, penelitian ini memberikan fokus yang lebih terarah pada solusi pengelolaan stok, menjadikannya sebagai pelengkap penting dalam studi-studi sebelumnya mengenai teknologi informasi untuk UMKM. Penelitian ini berhasil menunjukkan bahwa adopsi teknologi berbasis *Android* tidak hanya meningkatkan efisiensi operasional tetapi juga mengurangi kesalahan dalam pengelolaan stok, yang sebelumnya menjadi kendala utama bagi UMKM.

4. Kesimpulan

Penelitian ini bertujuan untuk memberikan solusi yang dapat meningkatkan kualitas dan efisiensi operasional dalam kegiatan *restock* di Maju Jaya *Accessories*. Berdasarkan hasil penelitian yang dilakukan, penulis menyimpulkan bahwa perancangan sistem permintaan *restock* ini telah berhasil mencapai tujuan yang ditetapkan. Aplikasi yang dikembangkan tidak hanya memiliki antarmuka yang menarik secara *visual*, tetapi juga mudah dipahami oleh berbagai kalangan pengguna, sehingga setiap elemen informasi dapat disampaikan dengan jelas dan efektif. Dengan desain yang intuitif serta ramah pengguna, aplikasi ini memastikan bahwa pengguna dari berbagai latar belakang pendidikan atau teknis dapat dengan mudah mengakses serta memahami fitur-fitur yang tersedia tanpa mengalami kesulitan berarti. Selain itu, sistem ini efektif dalam memenuhi kebutuhan informasi dengan cara yang efisien dan tepat sasaran bagi seluruh pengguna, baik bagi pemilik usaha maupun mitra-mitra toko yang bekerja sama. Implementasi sistem ini membawa peningkatan dalam efektivitas dan efisiensi operasional, terutama dalam hal pengelolaan waktu dan penghematan biaya seperti yang sudah ada pada tabel pengujian. Proses *restock* menjadi lebih cepat, terstruktur, dan terkendali, sehingga waktu yang dibutuhkan untuk mengelola stok barang dan memproses pesanan dapat dikurangi secara signifikan. Efisiensi biaya juga tercapai melalui optimalisasi alur kerja dan pemanfaatan teknologi yang mampu meminimalkan kesalahan dalam proses dan mengurangi kemungkinan keterlambatan. Sebelumnya saat semua proses masih berjalan manual, seringkali terjadi kesalahan waktu saat *restock*, bisa terjadi saat toko hendak dilakukan *restock* ternyata toko tersebut belum siap untuk *restock*, atau bisa juga terjadi toko yang seharusnya sudah siap *restock* justru terlambat dilakukan *restock*. Dengan adanya sistem ini, siklus *restock* tidak hanya menjadi lebih singkat tetapi juga mengurangi beban biaya operasional secara keseluruhan. Hal ini memberikan nilai tambah yang signifikan bagi pengusaha dan pemilik toko dalam menjalankan bisnis mereka, memungkinkan mereka untuk fokus pada peningkatan kualitas pelayanan serta peningkatan keuntungan usaha. Berikutnya sistem ini akan dikembangkan ke sistem operasi selain android, karena sekarang sistem ini masih terbatas pada sistem operasi android saja yang bisa menggunakannya.

5. Daftar Pustaka

- Arisya, N. P., & Miftahuddin, A. PERAN DAN TANTANGAN SISTEM INFORMASI MANAJEMEN DAN MEDIA SOSIAL DALAM BERBAGAI BIDANG.
- Bariah, S. H., & Pradina, D. (2024). Implementasi SDLC Model Prototype Pada Sistem Informasi Company Profile SMP PGRI Bungbulang Berbasis Website. *Petik: Jurnal Pendidikan Teknologi Informasi Dan Komunikasi*, 10(1), 85-97. <https://doi.org/10.31980/jpetik.v10i1.1030>.
- Dwanoko, Y. S. (2016). Implementasi software development life cycle (sdlc) dalam penerapan pembangunan aplikasi perangkat lunak. *Jurnal Teknologi Informasi: Teori, Konsep, dan Implementasi*, 7(2), 143003.
- Frinando, R., & Artaye, K. (2021). Pengembangan aplikasi marketplace untuk usaha mikro kecil menengah (UMKM) berbasis android. <https://doi.org/10.35315/informatika.v13i2.8832>.
- Hidayah, M. R., Probowulan, D., & Aspirandi, R. M. (2021). Pemanfaatan Aplikasi Akuntansi Berbasis Android SI APIK Untuk Menunjang Pelaporan Keuangan UMKM: Studi Kasus Pada UMKM Kerupuk Ikan Ibu Sulastrri Besuki. *Jurnal Ilmiah Akuntansi Kesatuan*, 9(1), 67-78. <https://doi.org/10.37641/jiakes.v9i1.471>.

- Hidayat, A. T., Rio, R., & Santosa, I. G. O. (2023). Membership Application Berbasis Android Dengan Penerapan Kotlin Programming Language Di Wijaya Fitness Center (Wfc). *JUSIM (Jurnal Sistem Informasi Musirawas)*, 8(1), 8-15. <https://doi.org/10.32767/jusim.v8i1.1952>.
- Hidayat, F., Sadali, M., & Sudioanto, A. (2022). Aplikasi Titip Online Produk UMKM Berbasis Android. *Infotek J. Inform. dan Teknol*, 5(1), 128-137. <https://doi.org/10.29408/jit.v5i1.4627>.
- Inggi, R., Prayudi, Y., & Sugiantoro, B. (2018). Penerapan System Development Life Cycle (Sdlc) Dalam Mengembangkan Framework Audio Forensik. *Jurnal SemanTIK*, 4(2), 193-200.
- Masripah, S., & Ramayanti, L. (2019). Pengujian Black Box Pada Sistem Informasi Penerimaan Siswa Baru Berbasis Web. *Information System For Educators And Professionals: Journal of Information System*, 4(1), 1-â.
- Masripah, S., & Ramayanti, L. (2019). Pengujian Black Box Pada Sistem Informasi Penerimaan Siswa Baru Berbasis Web. *Information System For Educators And Professionals: Journal of Information System*, 4(1), 1-â. <https://doi.org/10.36596/jitu.v6i2.820>.
- Nasution, A., Efendi, B., & Siregar, I. K. (2019). Pelatihan membuat aplikasi android dengan android studio pada SMP Negeri 1 Tinggi Raja. *Jurdimas (Jurnal Pengabdian Kepada Masyarakat) Royal*, 2(1), 53-58. <https://doi.org/10.33330/jurdimas.v2i1.321>.
- Nugraha, W., Syarif, M., & Dharmawan, W. S. (2018). Penerapan Metode Sdlc Waterfall Dalam Sistem Informasi Inventori Barang Berbasis Desktop. *JUSIM (Jurnal Sist. Inf. Musirawas)*, 3(1), 22-28.
- Nurhayati, M., & Yanti, S. N. (2024). Peran Teknologi Informasi Dalam Transformasi Bisnis Dan Ekonomi. *Jurnal Review Pendidikan dan Pengajaran (JRPP)*, 7(3), 10008-10012. <https://doi.org/10.31004/jrpp.v7i3.31500>.
- Nurkamid, M., & Budiman, N. A. (2021). Aplikasi Gusjigang Akuntansi Berbasis Android Pada UMKM Kudus Dimasa Pandemi Covid-19. *JUMANJI (Jurnal Masyarakat Informatika Unjani)*, 5(2), 80-88. <https://doi.org/10.26874/jumanji.v5i2.94>.
- Nurlaela, L., Setiawati, D., & Wulandari, F. T. (2024). Penerapan Metode User Centered Design pada Aplikasi UMKM Berbasis Android (Studi Kasus: UMKM Kenyot Susu Boyolali). *JITU: Journal Informatic Technology And Communication*, 8(1), 52-62. <https://doi.org/10.36596/jitu.v8i1.387>.
- Pangestu, H., Alianto, H., & Wijaya, S. F. (2012). Hasil Rancang Bangun Sistem ERP dengan SDLC Model Waterfall: Studi Kasus Sistem Inventori PT Pan Brothers, Tbk. *ComTech: Computer, Mathematics and Engineering Applications*, 3(2), 1036-1042.
- Romindo, R. (2022). Penerapan model sdlc terhadap sistem informasi penjualan dan persediaan bangunan pada cv. nilafa. *Journal Information System Development (ISD)*, 7(1), 62-73.