

APLIKASI INVENTARIS MENGGUNAKAN METODE RAPID APPLICATION DEVELOPMENT (RAD) SEBAGAI INOVASI TEKNOLOGI UNTUK EFISIENSI STOK OBAT

Galih Arya Maulana¹⁾, Joko Aryanto²⁾

^{1,2} Fakultas Sains dan Teknologi, Program Studi Informatika, Universitas Teknologi Yogyakarta, Yogyakarta, Indonesia

email: email: ¹galiharyamaulana135@gmail.com, ²joko.aryanto@uty.ac.id

Abstract

Apotek Dewa Sehat Pati is a company engaged in the pharmaceutical sector and sells medicines. Like other companies, in pharmacies there is also a lot of data that must be managed. Data processing at the pharmacy is still conventional. Pharmacies make records using ledgers. This can cause the service and processing of sales transaction data to be very time consuming. Recording errors often occur and data is also often lost. Sales data is also vulnerable to manipulation by anyone. Based on the existing problems, this research was conducted to create a recording system that is applied to pharmacies. This system aims to create an effective and efficient drug sales system. This pharmacy system was developed using the Rapid Application Development (RAD) method and uses design in the form of designs and diagrams to determine data flow. Kotlin programming language and Firebase database are also used during the coding process of this system. The resulting system was then tested using the Black Box method with 9 scenarios and obtained 100% successful test results. The research produced a mobile-based inventory information system that is able to facilitate the sales process to be more effective and efficient, and make reporting more accurate and timely. Apotek Dewa Sehat Pati can overcome existing problems with the modeling system that has been created.

Keywords: *Pharmacy, Information System, Rapid Application Development (RAD), Kotlin, Firebase*

Abstrak

Apotek Dewa Sehat Pati ialah salah satu perusahaan yang bergerak pada bidang farmasi dan menjual obat-obatan. Seperti perusahaan lainnya, di apotek juga terdapat banyak data yang harus dikelola. Pengolahan data pada apotek tersebut masih bersifat konvensional. Apotek melakukan pencatatan dengan menggunakan buku besar. Hal ini dapat menyebabkan pelayanan dan pengolahan data transaksi penjualan sangat memakan waktu. Kesalahan pencatatan sering terjadi dan data juga sering hilang. Data penjualan juga rentan untuk dimanipulasi oleh siapapun. Berdasarkan permasalahan yang ada, maka penelitian ini dilakukan untuk membuat sebuah sistem pencatatan yang diterapkan pada apotek. Sistem ini bertujuan untuk membuat sistem penjualan obat yang efektif dan efisien. Sistem apotek dikembangkan menggunakan metode *Rapid Application Development* (RAD) dan menggunakan perancangan berupa desain dan diagram untuk mengetahui aliran data. Bahasa pemrograman Kotlin dan database Firebase juga digunakan pada saat proses pengkodean dalam sistem ini. Sistem yang dihasilkan kemudian diuji menggunakan metode Black Box dengan 9 skenario dan didapatkan hasil uji 100% berhasil. Penelitian menghasilkan sistem informasi inventory berbasis mobile yang mampu mempermudah proses penjualan menjadi lebih efektif dan efisien, serta membuat pelaporan menjadi lebih akurat dan tepat waktu. Apotek Dewa Sehat Pati dapat mengatasi permasalahan yang ada dengan adanya sistem pemodelan yang telah dibuat.

Keywords: *Apotek, Sistem Informasi, Rapid Application Development (RAD), Kotlin, Firebase*

1. PENDAHULUAN

Perkembangan kemajuan teknologi informasi berpengaruh pada penggunaan komputer di berbagai sektor kehidupan (Ndaumanu, 2020). Untuk meningkatkan pencapaian derajat kesehatan lebih baik, maka pemerintah mendirikan apotek. Apotek menjadi layanan pertama kesehatan yang ada pada masyarakat. Pada apotek terdapat proses penjualan yang menjadi sumber pendapatan utama. Dalam era globalisasi ini persaingan pasar sangat tajam. Kegiatan penjualan memungkinkan terjadinya transaksi karena kegiatan tersebut melengkapi pembelian barang. Sistem informasi untuk penjualan menjadi alternatif solusi untuk peningkatan pelayanan dalam transaksi penjualan obat melalui internet. Penerapan sistem informasi juga memudahkan pengaksesan data yang diperlukan (Sukma Renanda & Wulandari, 2023).

Pada Apotek Dewa Sehat Pati terdapat permasalahan terkait persediaan. Penggunaan obat di kawasan masyarakat saat ini sangat beragam. Apotek Dewa Sehat Pati harus menyediakan obat yang banyak dan memenuhi kebutuhan masyarakat (Oktaviani et al., 2021). Seluruh transaksi yang terjadi sangat bergantung pada ada atau tidaknya persediaan barang, selain itu persediaan barang juga merupakan hasil usaha yang penting bagi perusahaan dagang dan juga merupakan sumber pendapatan utama. Masyarakat sangat memerlukan sistem yang dapat dengan mudah menampilkan persediaan obat yang tersedia agar penyakit yang diderita bisa segera ditangani. Apotek Dewa Sehat Pati memerlukan sistem yang dapat mengelola data secara cepat dan akurat sehingga dapat meminimalkan dan menghindari penggunaan waktu, biaya, dan tenaga yang signifikan.

Penelitian dilakukan dengan menganalisis beberapa publikasi terdahulu. Beberapa publikasi digunakan sebagai sumber serta pedoman dari penelitian ini. Berdasarkan penelitian yang dilakukan oleh Fikri Hamidy yang membuat sistem informasi bidang akuntansi untuk persediaan obat menggunakan perhitungan harga pokok perpetual FIFO (*First In First Out*). Tahapan pertama yang dilakukan adalah dengan menganalisis sistem yang berjalan. Sistem baru dikembangkan menggunakan metode prototyping dan menghasilkan aplikasi berbasis website yang dapat mengontrol keluar masuknya persediaan obat (Hamidy et al., 2022). Selanjutnya pada penelitian yang dilakukan oleh peneliti sebelumnya (Yuniarthe et al., 2021) yang membuat sistem informasi yang berguna untuk mempermudah proses pengolahan data transaksi dan stok obat dengan membuat aplikasi desktop. Kemudian pada penelitian yang dilakukan oleh peneliti terdahulu (Novita, 2022) juga dijelaskan bahwa pencatatan data pada sebuah buku sangat menyulitkan pekerjaan dan memakan waktu yang banyak. Sehingga sistem informasi dapat menjadi solusi dari masalah yang ada. Pembuatan database menjadi solusi utama dalam penelitian ini. Perancangan sistem informasi pada apotek juga dilakukan oleh peneliti sebelumnya (Amsaras & Dewi, 2022) berupa website. Website tersebut dibuat dengan tujuan untuk memperluas jangkauan sehingga dapat meningkatkan penjualan. Perancangan sistem menggunakan metode Software Development Life Cycle (SDLC) dengan model waterfall. Selanjutnya penelitian yang dilakukan oleh peneliti sebelumnya (Nugraha, 2021) tentang perancangan aplikasi point of sales (POS) yang diterapkan pada Apotek Mitra Sejahtera. Metode pengembangan yang digunakan adalah RUP (Rational Unified Process). Sistem yang dihasilkan berupa aplikasi website yang dapat meningkatkan kinerja dan pelayanan pada apotek sehingga bisa menjadi lebih efektif dan efisien. Beberapa penelitian terdahulu yang dipaparkan diatas dijadikan objek pembandingan dalam penelitian ini.

Penelitian dilakukan untuk mendapatkan solusi dari permasalahan pada Apotek Dewa Sehat Pati. Peneliti membuat sistem penanganan stok obat pada apotek untuk meminimalisir kesulitan pada saat proses pengecekan barang yang dijual maupun yang dibeli. Sistem yang dibuat dapat memberikan akses ke admin terkait pengecekan stok barang secara realtime. Hal itu dapat berdampak pada keakuratan pengambilan keputusan. Fitur – fitur yang diperlukan juga dikembangkan untuk memenuhi kebutuhan penjual dan pembeli dalam bentuk aplikasi berbasis mobile. Aplikasi dikembangkan menggunakan bahasa pemrograman Kotlin. Kotlin adalah bahasa pemrograman yang bersifat pragmatis dan bersifat interoperabilitas sehingga dapat digabungkan dengan bahasa pemrograman java pada suatu proyek yang sama (Toyib Hidayat & Gede Olka Santoso, 2023). Data – data pada sistem juga dikelola menggunakan database firebase yang bersifat realtime. Firebase merupakan penyimpanan basis data Non SQL yang dapat menyimpan beberapa tipe data seperti

<https://doi.org/10.35145/joisie.v8i2.4717>

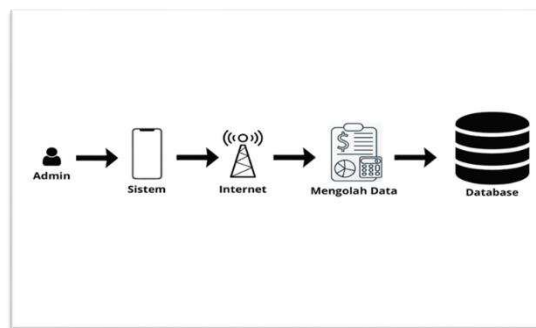
JOISIE licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0)

String, Long, dan Boolean (Andrianto & Haris Munandar, 2022). Dengan adanya database, maka data – data yang tersimpan dapat terjamin keamanannya. Database juga dapat menghubungkan data dari beberapa tabel yang berbeda, sehingga dapat mempermudah proses analisis data (Syahputri et al., 2023). Dengan adanya penelitian ini diharapkan dapat mengatasi masalah yang ada pada Apotek Dewa Sehat Pati.

Sistem penanganan stok obat dapat mempermudah proses transaksi antara penjual dan pembeli dan menjadikannya lebih efektif dan efisien. Terdapat fitur pembayaran yang menjadi keterbaharuan dari sistem yang sudah ada sebelumnya karena dilengkapi oleh fitur pengadaan barang dan fitur checkout obat. Transaksi yang dilakukan juga dapat langsung dilakukan pembayaran pada menu checkout. Hal itu tentu dapat meningkatkan keuntungan apotek karena sistem baru dapat mendukung keputusan pembeli dalam menentukan kebutuhan obatnya lebih cepat.

2. METODE PENELITIAN

Rancangan sistem merupakan tahap awal yang harus dilakukan dalam pengembangan suatu sistem. Rancangan tersebut melibatkan arsitektur serta detail komponen yang digunakan (Aziz et al., 2020). Pada penelitian ini, arsitektur model dibuat untuk menggambarkan rancangan sistem yang dibuat. Berikut adalah gambaran arsitektur model dari sistem.



Gambar 1. Arsitektur Model

2.1. METODE RAPID APPLICATION DESIGN

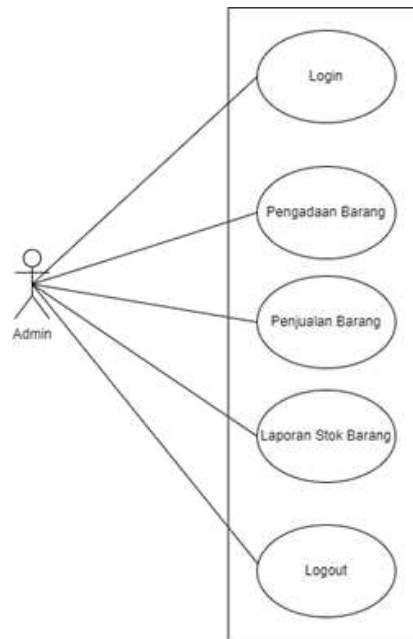
Metode penelitian yang digunakan dalam pengembangan sistem adalah Rapid Application Design (RAD). Metode ini menekankan pembuatan rancangan secara cepat dan umpan balik yang berulang untuk mempercepat pembuatan sistem dengan kualitas yang baik (Singgalen, 2024). Beberapa tahapan Rapid Application Development (RAD) sebagai berikut.

2.1.1. REQUIREMENTS PLANNING

Tahap requirements planning dilakukan dengan mengunjungi Apotek Dewa Sehat Pati pada 8 Juni 2024 untuk melakukan observasi dengan tujuan mengetahui masalah yang ada pada apotek. Masalah yang ada pada Apotek Dewa Sehat Pati diantaranya seperti pengelolaan data obat yang masih memakan waktu. Peneliti juga mendapatkan data berupa stok obat dan macam jenis obat yang tersedia pada apotek. Identifikasi masalah yang ada dilakukan untuk memahami kebutuhan dan solusi dari masalah yang ada pada Apotek Dewa Sehat Pati.

2.1.2. WORKSHOP DESIGN RAD

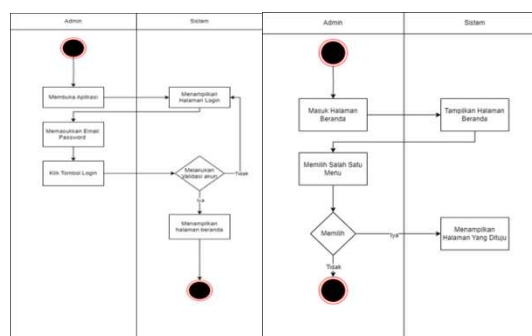
Tahap ini dilakukan pemodelan sebagai solusi dari masalah yang ada. Pemodelan sistem menggunakan UML (Unified Modelling Language) yaitu use case diagram, activity diagram, dan sequence diagram. Use Case Diagram dapat dilihat pada Gambar 2. berikut.



Gambar 2. Use Case Diagram

Pada Use Case Diagram ini admin mengakses login untuk masuk ke dalam sistem, setelah masuk ke dalam sistem admin masuk ke menu beranda, pada menu beranda terdapat beberapa menu yaitu : menu pengadaan barang, menu penjualan barang, menu laporan stok barang, dan menu logout, pada menu pengadaan barang admin dapat menambah stok barang yang sudah habis, lalu pada menu penjualan barang admin dapat mengurangi stok barang yang telah terjual, lalu pada menu laporan stok barang admin menampilkan hasil laporan barang masuk dan barang keluar untuk mengetahui detail stok barang yang telah berhasil transaksi, setelah itu, pada menu logout admin dapat keluar dari sistem.

Selanjutnya adalah activity diagram yang menjadi langkah pertama admin membuka aplikasi. Diagram tersebut dapat dilihat pada Gambar 3 berikut.

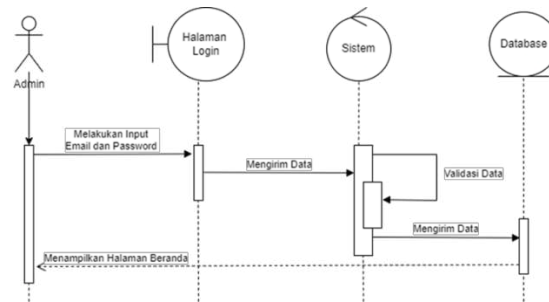


Gambar 3. Activity Diagram

Admin memasukkan email dan password, setelah memasukkan email dan password admin mengklik tombol login lalu sistem memvalidasi akun jika iya maka akan menampilkan halaman beranda jika tidak maka akan menampilkan halaman login lagi. Pada activity menu beranda admin masuk ke halaman beranda lalu melihat tampilan beranda, lalu admin memilih salah satu menu setelah memilih salah satu menu sistem akan menampilkan halaman yang dituju.

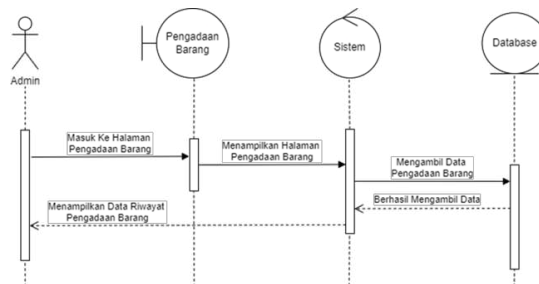
Sequence diagram juga digunakan dalam perancangan sistem ini. Berikut Sequence Diagram Halaman Login yang menjelaskan bahwa admin dapat menginput email dan password, lalu akan

diproses sistem untuk validasi, jika benar akan masuk ke database lalu di kembalikan ke admin. Diagram dapat dilihat pada Gambar 4 dibawah ini



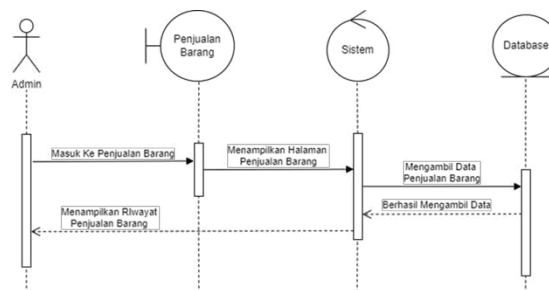
Gambar 4. Sequence Diagram Halaman Login

Sequence Diagram Pengadaan Barang dengan admin masuk ke halaman pengadaan, lalu sistem akan memproses halaman pengadaan barang dan mengambil data pada database, lalu menampilkan data ke admin. Diagram dapat dilihat pada Gambar 5 seperti dibawah ini.



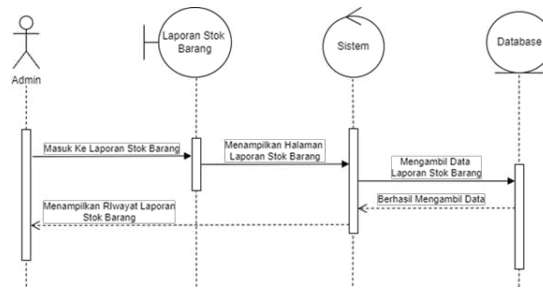
Gambar 5. Sequence Diagram Halaman Pengadaan Barang

Selanjutnya Sequence Diagram Penjualan Barang dengan cara masuk ke menu penjualan barang, lalu sistem memproses halaman penjualan barang dan mengambil data pada database, lalu menampilkan data ke admin. Diagram dapat dilihat pada Gambar 6 dibawah ini.



Gambar 6. Sequence Diagram Halaman Penjualan Barang

Sequence Diagram Laporan Stok Barang dengan cara masuk ke menu laporan barang, lalu sistem memproses halaman laporan barang dan mengambil data pada database, lalu menampilkan data ke admin. Diagram dapat dilihat pada Gambar 7 sebagai berikut.



Gambar 7. Sequence Diagram Halaman Laporan Stok Barang

2.1.3. IMPLEMENTATION

Aplikasi diimplementasikan pada Apotek Dewa Sehat Pati untuk memudahkan pengelolaan data dalam penyetokan obat. Aplikasi dapat diakses dengan mudah menggunakan *smartphone*.

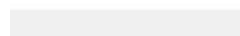
3. HASIL DAN PEMBAHASAN

3.1. ANTARMUKA

Tahap ini melibatkan pengkodean sistem yang menggunakan Kotlin dan Firebase untuk mengelola data yang ada pada sistem. Beberapa tampilan diimplementasikan menjadi halaman mobile diantaranya sebagai berikut.

3.1.1. HALAMAN SPLASH SCREEN

Halaman Splash Screen yang dibuat dengan melakukan pengkodean untuk tampilan terlebih dahulu. Ketika aplikasi dibuka, onCreate dijalankan, menyembunyikan ActionBar dan menampilkan layout activity_splash_screen. Aktivitas ini menunjukkan layar splash selama 2 detik menggunakan Handler yang menjalankan postDelayed dengan penundaan 2000 milidetik (2 detik). Kode dan halaman splash screen dapat dilihat pada Gambar 8 dibawah ini.



Gambar 8. Halaman Splash Screen

Setelah penundaan, aplikasi akan berpindah ke LoginActivity dengan memulai intent baru dan mengakhiri SplashScreenActivity menggunakan finish(). Intinya, kode ini menampilkan layar pembuka singkat sebelum mengarahkan pengguna ke halaman login utama.

<https://doi.org/10.35145/joisie.v8i2.4717>

JOISIE licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0)

3.1.2. HALAMAN LOGIN

Halaman login dikodekan menggunakan fungsi onClickListener untuk tombol login dalam aplikasi Android. Saat tombol login ditekan, kode akan mengambil nilai email dan password dari input pengguna, lalu melakukan validasi email. Kode dan halaman login dapat dilihat pada Gambar 9 dibawah ini



Gambar 9. Halaman Login

Jika email valid, maka proses login akan dilanjutkan dengan memanggil fungsi signIn() dengan email dan password. Jika email tidak valid, akan muncul pesan peringatan meminta pengguna untuk memasukkan email yang valid.

3.1.3. HALAMAN MENU BERANDA

Halaman menu beranda dibuat menggunakan pengkodean. Pada saat aktivitas MainActivity dibuat (onCreate), layout yang terkait dengan MainActivity (R.layout.activity_main) diatur sebagai tata letak tampilan aktif. Selain itu, terdapat inisialisasi beberapa komponen antarmuka pengguna (UI), yaitu ImageButton untuk tombol barang (barangBtn), tombol penjualan (penjualanBtn), tombol laporan stok (laporanStok), dan tombol logout (logoutButton). MainActivity adalah aktivitas utama dalam aplikasi Android ini. Ketika MainActivity dibuat, tata letak tampilan yang digunakan untuk MainActivity diatur untuk ditampilkan kepada pengguna. Kode dan tampilan halaman menu beranda dapat dilihat pada Gambar 10 dibawah.

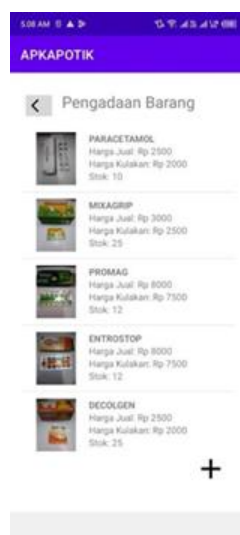


Gambar 10. Halaman Menu Beranda

Selain itu, beberapa komponen antarmuka pengguna (UI) seperti tombol barang, tombol penjualan, tombol laporan stok, dan tombol logout diidentifikasi dan diinisialisasi agar dapat digunakan dalam pengembangan aplikasi lebih lanjut. Dengan demikian, kode ini merupakan langkah awal dalam membangun antarmuka pengguna yang responsif dan interaktif dalam aplikasi Android.

3.1.4. HALAMAN MENU PENGADAAN BARANG

Pada saat aktivitas PengadaanActivity dibuat (onCreate), layout yang terkait diatur untuk ditampilkan kepada pengguna. Kode dan halaman pengadaan barang dapat dilihat pada Gambar 11 dibawah ini.

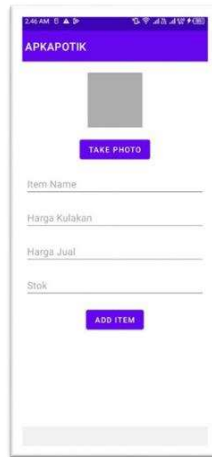


Gambar 11. Halaman Menu Pengadaan Barang

Beberapa komponen antarmuka pengguna (UI) seperti tombol kembali, tombol tambah item, dan ListView diinisialisasi. Kelas ini bertanggung jawab untuk menangani proses pengadaan barang. Ketika tombol kembali ditekan, PengadaanActivity akan berpindah ke MainActivity. Ini memungkinkan pengguna untuk kembali ke halaman utama setelah selesai melakukan proses pengadaan barang.

3.1.5. HALAMAN MENU TAMBAH BARANG

Pada halaman menu tambah barang dilakukan pengkodean. Fungsi `addItemToFirebase()` pada aplikasi Android ini digunakan untuk menambahkan item ke database Firebase. Pertama, kode mengambil nilai dari `EditText` yang berisi informasi tentang nama item, harga kulakan, harga jual, dan stok. Kode dan halaman menu tambah barang dapat dilihat pada Gambar 12 dibawah ini.



Gambar 11. Halaman Menu Tambah Barang

Nilai-nilai ini digunakan untuk membuat objek Item baru dengan UUID sebagai identifikasi gambar. Selanjutnya, jika ada gambar yang terpilih, gambar tersebut diunggah ke Firebase Storage dan URL-nya digunakan untuk mengupdate properti gambar pada objek Item sebelum ditambahkan ke database Firebase. Namun, jika tidak ada gambar terpilih, objek Item langsung ditambahkan ke database tanpa proses unggah gambar. Dengan begitu, fungsi ini memungkinkan pengguna untuk menambahkan item dengan atau tanpa gambar sesuai kebutuhan aplikasi.

3.1.6. HALAMAN MENU PENJUALAN BARANG

Fungsi `loadItemsFromFirebase()` dalam aplikasi Android ini bertujuan untuk memuat daftar item dari database *Firebase*. Kode mengakses child "items" pada database Firebase dan mengambil data yang tersimpan di dalamnya. Data tersebut kemudian diubah menjadi objek Item dan ditambahkan ke dalam *itemList*. Kode dan halaman menu penjualan barang dapat dilihat pada **Gambar 12** dibawah.



Gambar 12. Halaman Menu Penjualan Barang

Selanjutnya, kode menginisialisasi adapter (ItemAdapter) dengan itemList yang sudah terisi dan menampilkan daftar item tersebut ke dalam itemListView. Aksi yang dilakukan ketika item di-klik adalah menambahkan harga jual item ke subtotal dan memperbarui subtotal TextView dengan nilai subtotal yang baru. Namun, kode juga mencakup penanganan kegagalan jika terjadi masalah saat memuat data dari database Firebase. Dengan demikian, fungsi ini memungkinkan aplikasi untuk menampilkan daftar item dari database Firebase dengan responsif dan menangani kegagalan dengan baik.

3.1.7. HALAMAN MENU CHECKOUT BARANG

Kode pada halaman merupakan bagian dari fungsi *onCreate()* dalam Activity pembayaran di aplikasi Android. Saat Activity dibuat, layout ditampilkan dan komponen antarmuka pengguna (UI) diinisialisasi. Kode dan tampilan halaman menu checkout barang dapat dilihat pada **Gambar 13** dibawah.

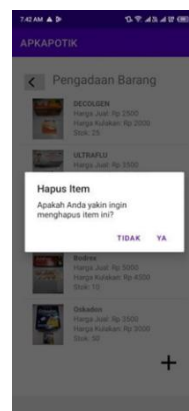


Gambar 13. Halaman Menu Checkout Barang

Nilai subtotal diambil dari Intent sebelumnya dan ditampilkan ke dalam *TextView*. Akses ke database Firebase juga diinisialisasi. Kode menetapkan aksi tombol kembali untuk menyelesaikan Activity. Tombol bayar memeriksa apakah jumlah uang cukup untuk pembayaran. Jika tidak, muncul *AlertDialog*, jika cukup, Activity berpindah ke halaman sukses transaksi. Secara ringkas, fungsi ini mengatur antarmuka pembayaran, mengambil nilai subtotal, menangani interaksi tombol, dan memberikan respons terhadap input pengguna.

3.1.8. HALAMAN MENU HAPUS BARANG

Fungsi *showDeleteConfirmationDialog()* digunakan untuk menampilkan dialog konfirmasi penghapusan item. Kode dan tampilan menu hapus barang dapat dilihat pada **Gambar 14** dibawah.



Gambar 14. Halaman Menu Hapus Barang

Saat dipanggil dengan parameter posisi item, dialog akan ditampilkan dengan pesan konfirmasi "Apakah Anda yakin ingin menghapus item ini?" dan dua pilihan tombol, "Ya" dan "Tidak". Jika pengguna memilih "Ya", item pada posisi yang dipilih akan dihapus dari database Firebase. Pesan Toast akan ditampilkan untuk memberi informasi pengguna tentang keberhasilan atau kegagalan penghapusan item. Dengan demikian, fungsi ini memberikan cara bagi pengguna untuk mengonfirmasi atau membatalkan penghapusan item dari database.

3.1.9. HALAMAN MENU HAPUS BARANG

Pada metode *onCreate()* di dalam *TransaksiSuksesActivity*, layout *activity_transaksi_sukses* diatur sebagai tampilan utama. Kemudian, *titleTextView* dan *selesaiButton* diinisialisasi dengan menghubungkannya ke tampilan yang sesuai di layout. Kode dan halaman transaksi sukses dapat dilihat pada **Gambar 15** dibawah.



Gambar 15. Halaman Transaksi Sukses

Teks pada *titleTextView* diatur menjadi "Transaksi Sukses". Ketika tombol *selesaiButton* ditekan, maka fungsi dari *navigateToHome()* dipanggil untuk mengarahkan pengguna kembali ke halaman utama. Fungsi ini memastikan bahwa pengguna mengetahui bahwa transaksi telah berhasil dan memberikan cara untuk kembali ke beranda aplikasi.

3.2 PENGUJIAN

Tahap selanjutnya ketika sistem telah selesai dibuat adalah pengujian. Pengujian dilakukan untuk mengecek apakah ada bug atau error pada sistem. Apabila terdapat fungsi yang tidak berjalan dengan semestinya maka akan diperbaiki oleh peneliti. Pengujian sistem yang digunakan ialah metode *black box testing*. Test case berupa skenario memiliki fungsi penting dalam tahap ini. Test Case kemudian dibagikan kepada 30 responden untuk diuji (Rafli et al., 2024). Pengujian black box dapat dilihat pada Tabel 1 dibawah ini.

Tabel 1. Pengujian Black Box Testing

No	Skenario Penguji	Expected Result	Result	Ket
1.	Membuka aplikasi	Menampilkan halaman login	Tampil halaman login	[√]
2.	Menuju menu beranda	Menampilkan menu beranda	Tampil halaman menu beranda	[√]

3.	Klik tombol pengadaan barang	Menampilkan beberapa obat masuk	Tampil beberapa obat masuk	[√]
4.	Klik tombol tambah barang	Menampilkan tambah barang	Tampil halaman tambah barang	[√]
5.	Klik item yang akan dihapus	Menampilkan hapus barang	Tampil halaman hapus barang	[√]
6.	Klik halaman penjualan barang	Menampilkan penjualan barang	Tampil halaman penjualan barang	[√]
7.	Klik tombol checkout	Menampilkan checkout barang	Tampil halaman checkout barang	[√]
8.	Klik tombol bayar	Menampilkan transaksi berhasil	Tampil transaksi berhasil	[√]
9.	Klik tombol laporan stok barang	Mengarah pada laporan stok barang	Tertuju pada halaman laporan stok barang	[√]

Berdasarkan hasil dari pengujian diatas yang telah dilakukan, maka dapat disimpulkan bahwa aplikasi dapat digunakan dengan baik. Hal itu diperoleh dari hasil pengujian yang dilakukan oleh 30 respondeng dan didapatkan hasil 100% Berhasil. Halaman splash screen dapat muncul dan pengguna dapat melakukan login. Menu beranda juga dapat menampilkan menu yang tersedia seperti pengadaan barang, penjualan barang, hingga laporan stok barang. Stok barang juga dapat ditambahkan dan dihapus melalui aplikasi yang telah dibuat.

4. SIMPULAN

Penelitian dilakukan menggunakan metode Rapid Application Development (RAD) dengan merancang prototype serta diagram. Sistem juga dikembangkan menggunakan bahasa pemrograman Kotlin dan database firebase. Berdasarkan hasil pengujian yang telah dilakukan dengan melihat dari semua permasalahan yang ada, dapat disimpulkan bahwa aplikasi inventaris berjalan normal. Aplikasi inventaris yang telah dibuat memiliki fitur manajemen penjualan cepat dan mendukung metode pemantauan stok barang secara real-time, pelaporan penjualan, serta sinkronasi data. Tampilan aplikasi memiliki desain yang jelas dan memudahkan admin dalam mengakses aplikasi serta pemilihan warna yang tepat, melakukan usability testing, konsisten dalam elemen desain, dan memberikan feedback yang jelas kepada pengguna. Aplikasi juga telah diuji oleh 30 reponden menggunakan metode black box dan menghasilkan persentase keberhasilan 100%. Aplikasi inventaris mengoptimalkan proses penjualan, mengelola stok secara efisien, menyediakan laporan untuk mendukung bisnis yang lebih baik. Dengan adanya aplikasi ini maka diharapkan Apotek Dewa Sehat Pati dapat dengan mudah mengelola sistem menjadi lebih efektif dan efisien.

5. DAFTAR PUSTAKA

- Singgalen, Y. (2024). Implementation of Rapid Application Development (RAD) for Community-based Ecotourism Monitoring System. *Journal of Information System Research (JOSH)*, 5(2), 520–530. <https://doi.org/10.47065/josh.v5i2.4749>
- Amsaras, P., & Dewi, Y. N. (2022). Analisa Perancangan Sistem Informasi Penjualan Obat Pada Apotek Segar. *Journal of Information System, Applied, Management, Accounting and Research*, 6(4), 2598–8700. <https://doi.org/10.52362/jisamar.v6i4.863>
- Andrianto, R., & Haris Munandar, M. (2022). Aplikasi E-Commerce Penjualan Pakaian Berbasis Android Menggunakan Firebase Raltime Database. *Journal Computer Science and Information Technology(JCoInT) Program Studi Teknologi Informasi*, 3(1). <http://jurnal.ulb.ac.id/index.php/JCoInT/index>
- Aziz, N., Pribadi, G., & Nurcahya, M. (2020). Analisa dan Perancangan Aplikasi Pembelajaran Bahasa Inggris Dasar Berbasis Android. *Jurnal IKRA-ITH Informatika*, 4, 1–5.
- Hamidy, F., Surahman, A., & Hikmah Famelia, R. (2022). Perancangan Sistem Informasi Akuntansi Apotek Menggunakan Metode MPKP (FIFO). *Jurnal Teknokompak*, 16(2), 188–199. <https://doi.org/https://doi.org/10.33365/jtk.v16i2.1479>

<https://doi.org/10.35145/joisie.v8i2.4717>

JOISIE licensed under a Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0)

- Ndaumanu, R. M. (2020). Perancangan Sistem Informasi Persediaan Obat Pada Apotek Rumah Sakit Menggunakan Metode Spiral. *Jurnal Komputer Dan Informatika*, 8(1), 18–27. <https://doi.org/10.35508/jicon.v8i1.2187>
- Novita, N. (2022). Manajemen Proyek Sistem Informasi Pengolahan Data Apotek Berbasis Database. *Jurnal Ilmiah Sistem Informasi*, 2(1), 9–17. <http://ojs.fikom-methodist.net/index.php/methosisfo>
- Nugraha, A. (2021). Perancangan Aplikasi Point Of Sales (POS) Pada Apotek Mitra Sejahtera Berbasis Web. *Jurnal IKRA-ITH Informatika*, 5(1), 74–81.
- Oktaviani, I., Sumarlinda, S., & Widyaningsih, P. (2021). Penerapan Metode PIECES pada Analisis Sistem Informasi Manajemen Apotek. *INFOKES*, 11(1). <https://doi.org/https://doi.org/10.47701/infokes.v11i1.1048>
- Rafli, H., Zen, R., & Nuryasin, I. (2024). Penerapan Whitebox Testing Pada Pengujian Sistem Menggunakan Teknik Basis Path. *Journal Of Information Systems And Informatics Engineering*, 8(1), 101–111. <https://doi.org/10.35145/joisie.v8i1.4229>
- Sukma Renanda, M., & Wulandari, S. (2023). Perancangan Aplikasi Pengelolaan Inventory Stock Amenities Housekeeping Hotel Berbasis Mobile. *JOISIE Journal Of Information System And Informatics Engineering*, 7(2), 214–223.
- Syahputri, K., Irwan, M., & Nasution, P. (2023). Peran Database Dalam Sistem Informasi Manajemen. *Jurnal Akuntansi Keuangan Dan Bisnis*, 1(2), 54–58. <https://jurnal.itc.web.id/index.php/jakbs/index>
- Toyib Hidayat, A., & Gede Olka Santoso, I. (2023). Membership Application Berbasis Android Dengan Penerapan Kotlin Programming Language di Wijaya Fitness Center (WFC). *Jusim : Jurnal Sistem Informasi Musi Rawas*, 8(1).
- Yuniarthe, Yodhi, Fahurian, F., & Nuari, I. (2021). Rancang Bangun Aplikasi Desktop Sistem Persediaan Obat Pada Apotek (Studi Kasus: Apotek Assifa Lampung). *Jurnal Teknologi Dan Informatika (JEDA)*, 2(2).