

PERANCANGAN DAN PENGEMBANGAN APLIKASI MOBILE SALES ORDER BERBASIS FLUTTER PADA PERUSAHAAN DUA JAYA

Yonatan Widianto, Yosafat Hizkia Pesik*, Yulius Hari

Teknik Informatika, Universitas Widya Kartika
Jl. Sutorejo Prima Utara II No.1 Surabaya, Jawa Timur
yosafatpesik02@gmail.com

ABSTRAK

Perusahaan Dua Jaya, yang bergerak di bidang penjualan peralatan elektronik rumah tangga, menghadapi tantangan dalam proses pemesanan di mana sales masih menggunakan WhatsApp untuk mengirimkan pesanan ke admin, yang kemudian harus memasukkan pesanan tersebut ke dalam sistem berbasis web. Untuk mengatasi masalah ini, dikembangkan aplikasi mobile sales order berbasis Flutter yang dirancang khusus untuk digunakan oleh tim sales. Aplikasi ini memungkinkan sales melakukan pemesanan langsung melalui perangkat mobile tanpa perlu perantara. Pengembangan aplikasi dilakukan menggunakan metode waterfall dengan fokus pada penyederhanaan proses pemesanan bagi sales. Hasil evaluasi menunjukkan bahwa aplikasi ini mendapatkan tingkat kepuasan pengguna sebesar 87,9%, menunjukkan penerimaan positif dari sales.

Kata kunci : Teknologi informasi, sistem pemesanan, aplikasi mobile, Flutter, metode waterfall, Skala likert

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi juga telah mengubah sektor manajemen bisnis. Di era digital ini, teknologi telah menjadi pusat efisiensi dan produktivitas di banyak perusahaan. Perusahaan berlomba-lomba memanfaatkannya untuk memastikan bahwa mereka tetap bersaing secara efektif. Penjualan adalah salah satu sektor yang paling terdampak oleh era aplikasi mobile. Dulu, penjualan melibatkan transaksi manual dan komunikasi yang lambat, tetapi aplikasi mobile kini telah mengubah cara penjualan dilakukan secara mendasar. Aplikasi mobile memungkinkan perusahaan melakukan transaksi real-time, mengakses informasi stok dan logistik, memberikan sales akses ke data penting, dan berinteraksi dengan pelanggan dari mana saja.

Perusahaan Dua Jaya, yang bergerak di bidang penjualan peralatan elektronik rumah tangga, perlu memastikan bahwa mereka memanfaatkan perkembangan teknologi ini untuk memperbaiki dan mengoptimalkan operasional bisnis mereka. Salah satu langkah yang dapat diambil adalah dengan mengembangkan aplikasi mobile. Dengan framework Flutter, yang merupakan teknologi open-source dari Google, perusahaan dapat mengembangkan aplikasi mobile yang efisien dan responsif. Flutter mendukung pengembangan aplikasi yang dapat berjalan di berbagai platform seperti iOS dan Android dengan satu kode sumber.

Aplikasi ini akan memudahkan sales dalam mengakses informasi stok, harga, pelanggan, komisi penjualan, dan status faktur penjualan, sehingga mereka dapat bekerja lebih efektif dan memberikan layanan yang lebih baik kepada pelanggan. Aplikasi mobile ini memungkinkan akses yang lebih cepat, mudah, dan fleksibel bagi sales Dua Jaya tanpa tergantung pada komputer atau laptop. Dengan demikian, pemanfaatan teknologi aplikasi mobile dan penggunaan framework Flutter diharapkan dapat

meningkatkan produktivitas dan daya saing perusahaan Dua Jaya di industri penjualan peralatan elektronik rumah tangga.

2. TINJAUAN PUSTAKA

2.1. Penelitian Sebelumnya

Sebagai dasar pemahaman yang lebih mendalam tentang topik penelitian, pencarian literatur sebelumnya telah dilakukan. Salah satu studi di bidang pengembangan aplikasi mobile menggunakan Flutter adalah penelitian oleh Wang et al., pada 2021 yang berjudul "Designing and Developing an Agricultural Product Sales Application Catalog with a Hybrid Application Development Framework"[1].

Penelitian tersebut bertujuan menggunakan Flutter dalam mengembangkan katalog penjualan produk pertanian untuk aplikasi mobile. Flutter digunakan sebagai framework pengembangan aplikasi mobile dengan fitur pencarian produk berdasarkan kategori dan manajemen data secara real-time melalui sistem backend. Penelitian tersebut menunjukkan bagaimana Flutter dapat digunakan untuk mengembangkan aplikasi penjualan yang terintegrasi dan efisien.

Hasil dari penelitian ini menjadi landasan kuat untuk pengembangan aplikasi mobile sales order di perusahaan Dua Jaya. Fokus utama adalah kemampuan integrasi API dan peningkatan fleksibilitas serta aksesibilitas bagi agen sales dalam mengelola pesanan. Penelitian ini menyoroti pentingnya teknologi mobile untuk meningkatkan efisiensi operasional dan memberikan kemudahan akses informasi bagi pengguna. Dengan menggunakan Flutter, aplikasi yang dikembangkan dapat berjalan di berbagai platform, seperti iOS dan Android, dari satu basis kode yang sama. Hal ini sangat relevan dengan kebutuhan perusahaan Dua Jaya yang menginginkan solusi mobile yang efisien dan mudah diakses oleh sales.

2.2. Framework

Framework adalah kunci utama dalam pengembangan sistem perangkat lunak berorientasi objek berskala besar. Framework memudahkan dan mempercepat pembangunan sistem, meningkatkan produktivitas dengan memungkinkan penggunaan ulang kode yang sama (T. Wijaya, 2023). Framework mencakup seluruh aspek dan mengatur hubungan antar aplikasi. Dengan framework, pengembangan aplikasi menjadi lebih cepat karena menyediakan komponen seperti koneksi basis data, validasi formulir, antarmuka grafis pengguna (GUI), dan keamanan sistem.

Penggunaan framework membuat konsep pengembangan aplikasi menjadi lebih jelas karena adanya aturan tertentu yang harus diikuti. Konsep ini memudahkan kolaborasi dengan orang lain dalam pengembangan sistem dan mempermudah proses debugging. Salah satu framework yang populer dalam pengembangan aplikasi mobile adalah Flutter. Flutter adalah framework open-source yang dikembangkan oleh Google untuk membangun aplikasi mobile, web, dan desktop dari satu basis kode yang sama. Keunggulan Flutter terletak pada kemampuannya untuk menghasilkan aplikasi yang responsif dan berkinerja tinggi dengan antarmuka yang menarik. Dengan Flutter, pengembang dapat membuat aplikasi yang berjalan di berbagai platform seperti iOS dan Android dengan efisiensi tinggi[2].

2.3. MVC

Model-View-Controller (MVC) adalah pola arsitektur perangkat lunak yang membagi aplikasi menjadi tiga komponen utama: Model, View, dan Controller. Model bertanggung jawab mengelola data dan logika bisnis aplikasi. View berfungsi untuk menampilkan data kepada pengguna. Controller bertindak sebagai perantara yang mengatur interaksi antara Model dan View. Penerapan MVC dalam pengembangan aplikasi memungkinkan pemisahan yang jelas antara logika bisnis dan tampilan, memudahkan pengelolaan kode dan kolaborasi antar pengembang. Selain itu, pola MVC memudahkan proses debugging dan pengujian karena setiap komponen dapat diuji secara terpisah[3].

2.4. API

API, atau Application Programming Interface, adalah antarmuka yang memungkinkan aplikasi perangkat lunak berkomunikasi satu sama lain. API menyediakan sekumpulan definisi dan protokol untuk pertukaran data dan layanan antara perangkat lunak. Contoh implementasi API adalah dalam pengembangan aplikasi mobile, seperti API MADRID yang digunakan untuk mengatasi permasalahan remaja dengan menyediakan layanan psikologis berbasis Android[4].

API berfungsi sebagai jembatan antara berbagai sistem perangkat lunak, memungkinkan interoperabilitas dan integrasi yang efisien.

2.5. Waterfall

Waterfall adalah salah satu model SDLC yang paling banyak digunakan. Waterfall adalah model sekuensial yang mengikuti tahapan-tahapan berurutan. Model ini mengharuskan setiap tahap pengembangan perangkat lunak diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya. Nama Waterfall sendiri diambil dari visualisasi aliran air yang bergerak sejalan dari satu tahap ke tahap lainnya. Dalam Waterfall, terdapat lima tahapan utama: analisis, desain, implementasi, testing, dan maintenance[5].

2.6. UML

UML atau Unified Modeling Language adalah bahasa yang digunakan untuk memodelkan dan merancang sistem perangkat lunak. Para pengembang menggunakan UML untuk menggambarkan konsep, struktur, dan hubungan antara berbagai komponen dalam sistem [6].

Contoh diagram UML yang umum digunakan adalah Use Case Diagram dan Class Diagram, yang membantu dalam visualisasi interaksi pengguna dan struktur data sistem.

2.7. Use Case

Use case diagram adalah diagram yang merepresentasikan fungsionalitas sistem dan interaksi antara aktor dengan sistem. Use case diagram menunjukkan "apa" yang dilakukan sistem, bukan "bagaimana" sistem melakukannya. Setiap use case menggambarkan pekerjaan tertentu yang dapat dilakukan oleh sistem. Diagram ini membantu mendefinisikan kebutuhan sistem, berkomunikasi dengan klien, dan merancang test case. Use case dapat menyertakan fungsionalitas use case lain, menghindari duplikasi fungsionalitas, dan memperluas use case lain dengan perilaku tambahan. Hubungan generalisasi menunjukkan spesialisasi antara use case[7].

2.8. Class Diagram

Class diagram adalah diagram yang digunakan untuk menggambarkan struktur dan hubungan antar kelas dalam pemodelan sistem berorientasi objek. Setiap class dalam class diagram mewakili spesifikasi yang dapat diinstansiasi menjadi objek dan memiliki atribut atau properti yang menggambarkan keadaan sistem. Kelas juga menyediakan metode atau fungsi untuk memanipulasi atribut dan menjalankan operasi tertentu. Diagram ini membantu memahami bagaimana data dan fungsi berinteraksi dalam sistem, serta mendefinisikan blueprint untuk objek-objek yang akan dibuat dalam perangkat lunak[8].

2.9. Black Box

Black Box Testing adalah metode pengujian perangkat lunak yang fokus pada bagaimana aplikasi berfungsi dari luar tanpa melihat bagian dalamnya, seperti kode program. Tujuannya adalah untuk memastikan bahwa aplikasi bekerja sesuai dengan yang diharapkan berdasarkan input dan output yang

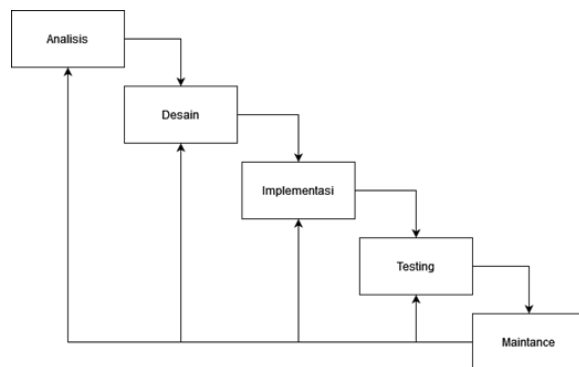
diberikan. Metode ini sangat berguna karena dapat menemukan kesalahan dalam fungsi aplikasi yang mungkin terlewat jika hanya melihat kode. Selain itu, pengujinya tidak perlu memahami detail teknis dari program, sehingga bisa lebih mudah digunakan oleh siapa saja yang ingin memastikan aplikasi berjalan dengan baik[9].

2.10. Skala Likert

Skala Likert adalah metode pengukuran yang sering digunakan dalam penelitian kuantitatif untuk mengukur sikap, opini, dan persepsi seseorang atau sekelompok orang terhadap fenomena tertentu. Skala ini biasanya terdiri dari serangkaian pernyataan dan responden diminta untuk menunjukkan tingkat persetujuan atau ketidaksetujuan mereka terhadap pernyataan tersebut. Skala Likert merupakan skala bipolar yang sering digunakan dalam kuesioner untuk memperoleh data kuantitatif yang kemudian dianalisis secara statistik[10].

3. METODE PENELITIAN

3.1. Tahapan Penelitian



Gambar 1. Metode Waterfall

Penelitian ini menerapkan metode waterfall dalam merancang dan mengembangkan aplikasi mobile sales order berbasis Flutter di perusahaan Dua Jaya. Pemilihan metode ini bertujuan untuk memberikan struktur yang jelas dan langkah-langkah yang terurut, mulai dari analisis kebutuhan, desain, implementasi, pengujian, hingga pemeliharaan. Dengan metode ini, penelitian dapat mencapai hasil yang terarah, memastikan setiap tahap diselesaikan secara menyeluruh sebelum melanjutkan ke tahap berikutnya.

3.2. Analisis

Pada tahap ini, penelitian dimulai dengan analisis masalah dan kebutuhan aplikasi mobile sales order. Ini melibatkan pengumpulan informasi dari perusahaan Dua Jaya mengenai proses penjualan dan manajemen pesanan yang ada. Data yang diperoleh akan digunakan untuk mengidentifikasi kebutuhan bisnis dan teknis yang harus dipenuhi oleh aplikasi.

3.3. Desain

Tahap desain berfokus pada perancangan komponen penting dari aplikasi mobile sales order, termasuk merancang antarmuka pengguna yang intuitif dan responsif menggunakan Flutter. Desain struktur database juga dilakukan untuk memastikan data penjualan dan pesanan dapat disimpan dan diakses dengan efisien. Seluruh logika bisnis dan fungsionalitas aplikasi direncanakan dan didokumentasikan dengan jelas pada tahap ini.

3.4. Implementasi

Pada tahap implementasi, desain yang telah dibuat diwujudkan menjadi kode program yang berfungsi. Pengembang menggunakan Flutter untuk mengimplementasikan desain antarmuka pengguna, membangun sistem manajemen database, dan mengimplementasikan logika bisnis aplikasi. Pengujian unit juga dilakukan untuk memastikan setiap komponen bekerja dengan baik sebelum pengujian integrasi.

3.5. Testing

Tahap pengujian adalah bagian penting dalam pendekatan waterfall. Aplikasi diuji secara menyeluruh untuk memastikan kinerja, keandalan, dan kepatuhan fungsionalitas sesuai dengan kebutuhan yang telah ditentukan. Pengujian ini mencakup pengujian fungsionalitas, integrasi, dan kinerja. Setiap bug yang terdeteksi dicatat dan diperbaiki sebelum melanjutkan ke tahap berikutnya.

3.6. Maintenance

Setelah aplikasi berhasil diuji dan diimplementasikan, tahap pemeliharaan dilakukan. Ini melibatkan penerapan aplikasi di lingkungan produksi perusahaan Dua Jaya dan pemantauan kinerja aplikasi secara berkala.

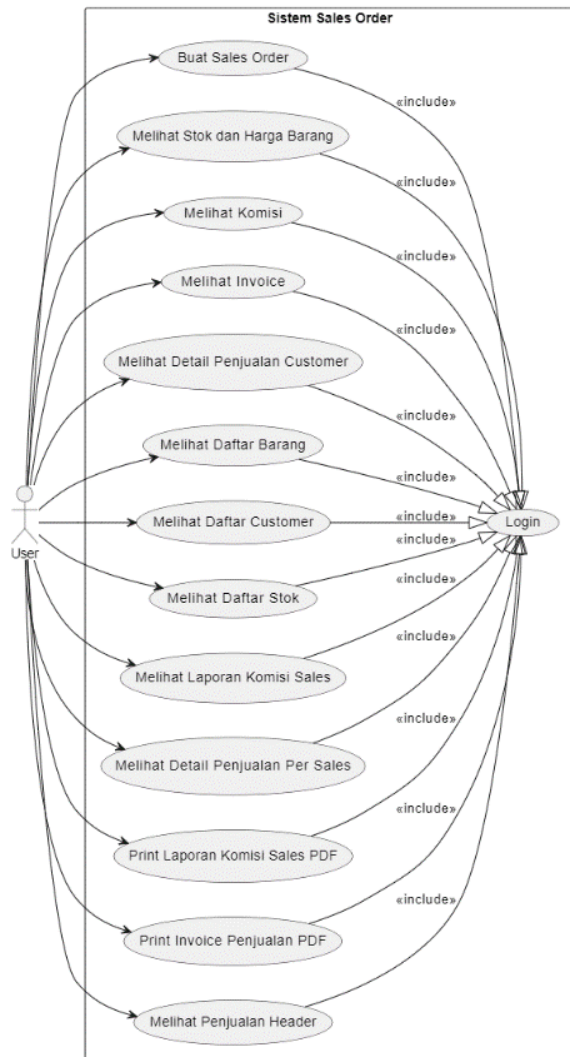
3.7. UML

Tahap berikutnya dari metode *waterfall* adalah pemodelan desain aplikasi. Di penelitian ini penulis akan menggunakan UML untuk pemodelan desain aplikasi. Berikut ini adalah UML yang digunakan.

3.8. Use Case Diagram

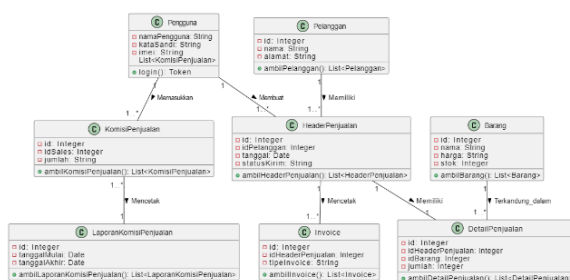
Dalam use case diagram, terdapat satu aktor yaitu User, yang merupakan pengguna utama sistem Sales Order. User dapat melakukan beberapa kegiatan penting dalam sistem setelah login ke dalam sistem untuk mengakses fitur-fitur yang disediakan. Setelah login, User dapat membuat pesanan penjualan baru, melihat stok barang yang tersedia, memantau komisi yang didapatkan dari penjualan, melihat daftar invoice yang terkait dengan pesanan penjualan, dan melihat detail penjualan pelanggan. Selain itu, User juga dapat melihat daftar barang, daftar customer, dan daftar stok. User juga dapat melihat laporan komisi sales, detail penjualan per sales, mencetak laporan komisi sales dalam format PDF, dan mencetak invoice penjualan.

dalam format PDF. Semua aktivitas ini memerlukan login terlebih dahulu untuk memastikan keamanan dan otorisasi pengguna.



Gambar 2. Use case diagram

3.9. Class Diagram



Gambar 3. Class diagram

Class diagram di atas menggambarkan struktur sistem penjualan yang melibatkan beberapa entitas utama, yaitu Pengguna, Barang, Pelanggan, Header Penjualan, Detail Penjualan, Komisi Penjualan, Invoice, dan Laporan Komisi Penjualan. Diagram ini menunjukkan hubungan antar kelas serta atribut dan

metode utama yang dimiliki oleh masing-masing kelas.

Kelas Pengguna merepresentasikan pengguna aplikasi dengan atribut namaPengguna, kataSandi, dan imei. Pengguna dapat melakukan operasi login untuk masuk ke dalam sistem. Selain itu, pengguna juga dapat memasukkan banyak data komisi penjualan.

Kelas Barang menggambarkan barang yang tersedia dalam sistem dengan atribut id, nama, harga, dan stok. Metode ambilBarang digunakan untuk mengambil daftar barang yang tersedia.

Kelas Pelanggan merepresentasikan pelanggan yang melakukan transaksi dengan sistem. Atribut yang dimiliki oleh kelas ini meliputi id, nama, dan alamat. Metode ambilPelanggan digunakan untuk mengambil daftar pelanggan.

Kelas Header Penjualan adalah representasi dari header sebuah transaksi penjualan. Kelas ini mencakup atribut id, idPelanggan, tanggal, dan statusKirim. Metode ambilHeaderPenjualan digunakan untuk mengambil daftar header penjualan. Setiap pengguna dapat membuat banyak header penjualan, dan setiap pelanggan dapat memiliki banyak header penjualan. Setiap header penjualan memiliki satu detail penjualan dan dapat mencetak banyak invoice.

Kelas Detail Penjualan menggambarkan detail dari transaksi penjualan dengan atribut id, idHeaderPenjualan, idBarang, dan jumlah. Metode ambilDetailPenjualan digunakan untuk mengambil daftar detail penjualan. Setiap detail penjualan berisi banyak barang.

Kelas Komisi Penjualan menyimpan informasi tentang komisi yang diterima oleh sales dari transaksi tertentu. Kelas ini memiliki atribut id, idSales, dan jumlah. Metode ambilKomisiPenjualan digunakan untuk mengambil daftar komisi penjualan. Banyak data komisi penjualan mencetak satu laporan komisi penjualan.

Kelas Invoice merepresentasikan faktur yang dihasilkan dari penjualan. Kelas ini memiliki atribut id, idHeaderPenjualan, dan tipeInvoice. Metode ambilInvoice digunakan untuk mengambil daftar invoice. Setiap header penjualan dapat mencetak banyak invoice.

Kelas Laporan Komisi Penjualan melacak komisi penjualan yang diterima selama periode tertentu dengan atribut id, tanggalMulai, dan tanggalAkhir. Metode ambilLaporanKomisiPenjualan digunakan untuk mengambil daftar laporan komisi penjualan.

Hubungan antar kelas dijelaskan dengan asosiasi yang sesuai. Pengguna dapat membuat banyak header penjualan, pelanggan memiliki banyak header penjualan, dan header penjualan mencakup satu detail penjualan. Barang terkandung dalam banyak detail penjualan, header penjualan mencetak banyak invoice, pengguna memasukkan banyak komisi penjualan, dan banyak komisi penjualan mencetak satu laporan komisi penjualan.

4. HASIL DAN PEMBAHASAN

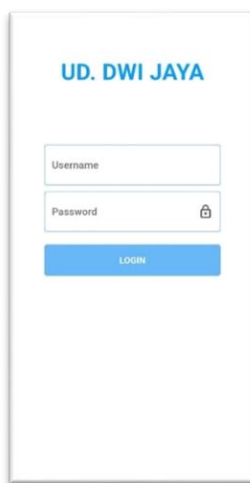
Setelah tahap pemodelan aplikasi, langkah selanjutnya dari metode *waterfall* adalah pengkodean/implementasi. Berikut ini adalah hasil implementasi sistem yang terdiri dari implementasi basis data dan implementasi tampilan sistem.

4.1. Implementasi Sistem

Pada langkah ini, akan dijelaskan tampilan yang sudah dibuat dalam aplikasi ini.

4.2. Halaman Login

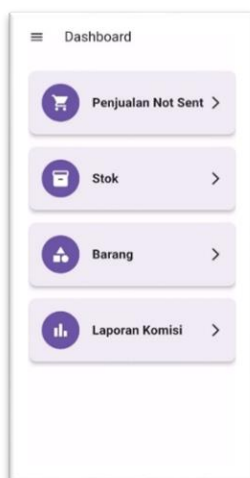
Pada halam ini, user bisa login sesuai dengan email dan password yang sudah terdaftar pada system dan juga pada halaman ini aka ada fungsi untuk mengambil imei perangkat user. Fungsi dari mengambil imei tersebut adalah untuk tujuan keamanan.



Gambar 4. Tampilan login

4.3. Dashboard

Pada halaman Dashboard menampilkan shortcut untuk mengakses halaman yang penting seperti penjualan not sent, stok, barang, dan laporan komisi.



Gambar 5. Tampilan dashboard

4.4. Barang

Pada Halaman barang menampilkan seluruh barang yang dapat kita ambil menggunakan API. Dan pada halaman tersebut kita dapat mencari barang berdasarkan nama item barang serta kita dapat melihat harga barang yang ada mulai dari HP (Harga Pokok) harga terendah dan hingga H3 harga tertinggi untuk menjual barang tersebut.



Gambar 6. Tampilan barang

4.5. Customer

Di Halaman customer menampilkan seluruh data customer mulai dari nama, alamat, no.telp dan sebagainya.



Gambar 7. Halaman customer

4.6. Stok

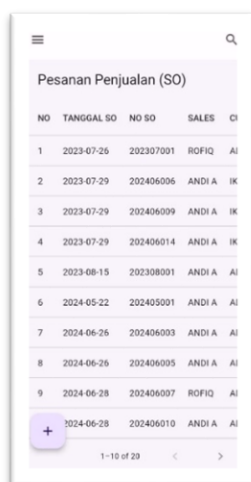
Halaman Stok menampilkan jumlah stok barang yang tersedia dan kita dapat mencari informasi stok barang menggunakan fitur search.



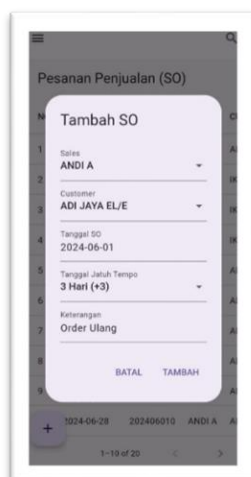
Gambar 8. Halaman stok

4.7. Penjualan Not Sent

Pada halaman ini menampilkan seluruh penjualan dengan status belum dikirim. Pada halaman ini juga kita dapat menambahkan Sales Order baru dan juga kita dapat edit, delete maupun print invoice dari SO tersebut.



Gambar 9. Penjualan not sent

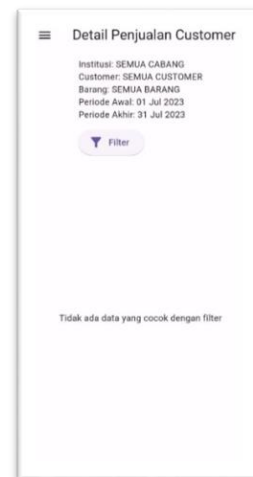


Gambar 10. Dialog tambah so

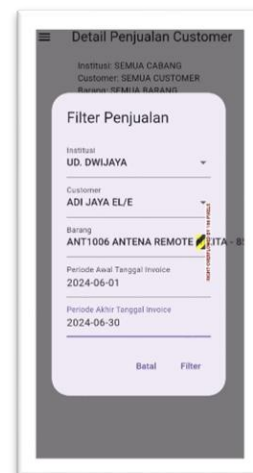
Kita dapat menambahkan SO baru dengan mengklik button tambah dan akan muncul form yang dapat diisi.

4.8. Penjualan Detail Penjualan Customer

Pada Halaman detail penjualan customer user dapat melihat penjualan yang sudah terjadi berdasarkan filter yang ia ditetapkan.



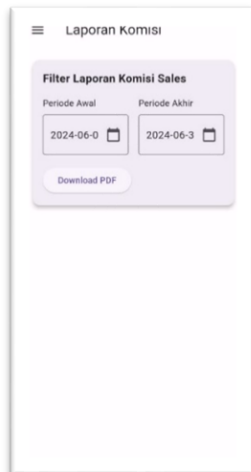
Gambar 11 Halaman detail penjualan customer



Gambar 12. Dialog filter detail penjualan customer

4.9. Komisi Sales

Di halaman kita menetapkan waktu periode untuk mendownload pdf laporan komisi yang ingin didapatkan oleh sales.



Gambar 13. Halaman komisi sales

4.10. Black Box Testing

Tabel 1. Tabel Black Box Testing

No.	Fitur	Hasil yang Diharapkan	Hasil
1	Login	User dapat melakukan login dengan akun yang sudah didaftarkan	Berhasil
2	Navigasi	User bisa berpindah pindah halaman dengan leluasa	Berhasil
3	Logout	User berhasil keluar dari akun yang sedang digunakan	Berhasil
4	Dashboar	Menampilkan Shortcut untuk page penting.	Berhasil
5	Tambah SO	User dapat menambah SO baru pada penjualan not sent.	Berhasil
6	Print PDF	User dapat mencetak PDF komisi sales dan invoice	Berhasil
7	Menampilkan Data	User dapat menerima informasi yang ingin ia cari seperti informasi barang, customer, stok, penjualan, maupun komisi yang ia dapatkan.	Berhasil

4.11. Skala Likert

Setelah aplikasi selesai dibuat maka selanjutnya aplikasi diujicobakan ke user yang kemudian dimintai pendapatnya. Indeks penerimaan user didalam penelitian ini diukur dengan skala likert. Berikut ini adalah rumus skor skala likert.

$$\text{Total Skor} = T \times P_n$$

T = Total jumlah responden yang memilih

P_n = Pilihan angka skor Likert

Kemudian untuk indeks persentase indeks pengukuran ditentukan dengan rumus berikut. Rentangan skor likert yang digunakan dalam penelitian ini adalah 1-5 (jumlah skor likert adalah 5), dan berdasarkan rumus kategori hasil interpretasi

berikut ini maka didapatkan perhitungan $I = 100 / 5 = 20$. Didapatkan rentangan 20 dengan deskripsi sebagai berikut.

$$\text{Indeks \%} = (\text{Total skor} / Y) \times 100\%$$

Y = Skor tertinggi likert x jumlah responden

$$I = 100 / \text{Jumlah Skor (Likert)}$$

Tabel 2. Tabel Skala Likert

No.	Pertanyaan	TxPn	Indeks
1	Apakah aplikasi ini mudah digunakan untuk pengguna baru?	17	85%
2	Apakah saya merasa nyaman menggunakan aplikasi ini untuk membuat SO?	17	85%
3	Apakah aplikasi ini responsif saat digunakan?	18	90%
4	Apakah waktu loading dalam aplikasi ini cepat?	19	95%
5	Apakah aplikasi ini berfungsi dengan lancar tanpa banyak bug?	16.	80%
6	Apakah fitur login berfungsi dengan baik?	20	100%
7	Apakah informasi produk yang ditampilkan dapat memenuhi kebutuhan informasi Anda?	18	90%
8	Apakah informasi pelanggan yang ditampilkan sudah memenuhi kebutuhan Anda?	17	85%
9	Apakah informasi stok yang ditampilkan dapat membantu?	17	85%
10	Apakah fitur laporan komisi sales memberikan informasi yang Anda butuhkan?	18	90%
11	Apakah fitur tambah SO (Sales Order) mudah digunakan dan memudahkan proses penjualan?	16	80%
12	Secara keseluruhan, apakah Anda puas dengan aplikasi ini?	17	85%
13	Apakah Anda merasa aplikasi ini membantu pekerja Anda?	18	90
14	Apakah Anda akan merekomendasikan aplikasi ini kepada rekan kerja Anda	18	90%
Rata – rata presentase			87,9%

5. KESIMPULAN DAN SARAN

Pengembangan aplikasi mobile sales order berbasis Flutter untuk perusahaan Dua Jaya dapat mempermudah pengguna dalam membuat Sales Order dibandingkan dengan cara yang lama. Pengujian menunjukkan bahwa aplikasi berfungsi dengan baik dan diterima dengan baik oleh pengguna, terbukti dari hasil feedback yang menunjukkan tingkat kepuasan yang cukup tinggi. Untuk pengembangan lebih lanjut, beberapa saran yang dapat dipertimbangkan adalah peningkatan fitur keamanan dengan mengganti

otentikasi IMEI dengan autentikasi lainnya seperti menggunakan Android Device ID, karena IMEI sudah mulai dibatasi karena terkait dengan masalah privasi sehingga mulai dari android 10 sudah tidak bisa digunakan lagi IMEI sebagai alat autentikasi[11]. Pengembangan versi iOS juga penting, meskipun Flutter mendukung pengembangan lintas platform, pengujian dan optimisasi khusus untuk perangkat iOS akan memastikan pengalaman pengguna yang konsisten di semua platform.

DAFTAR PUSTAKA

- [1] Y. Wang, A. Wibowo, R. Delima, and A. Rachmat, "Designing and Developing an Agricultural Product Sales Application Catalog with a Hybrid Application Development Framework," *IOP Conf Ser Mater Sci Eng*, vol. 1077, no. 1, p. 012050, Feb. 2021, doi: 10.1088/1757-899X/1077/1/012050.
- [2] T. Wijaya, "Implementasi Aplikasi Hybrid untuk Meningkatkan Produktivitas Pengembang Perangkat Lunak Hybrid App Implementation to Increase Software Development Productivity," 2023. [Online]. Available: <http://ejournal.potensi-utama.ac.id/ojs/index.php/ITJournal/index>
- [3] S. Munir, A. R. Adriansyah, R. Dhelika, and R. Wahyudi, "Implementasi Arsitektur Aplikasi Mvc Pada Perancangan Aplikasi Bank Sampah Berbasis Web," *Jurnal Teknologi Terpadu*, vol. 4, no. 2, p. 318362, Dec. 2018, Accessed: Jun. 30, 2024. [Online]. Available: <https://www.neliti.com/publications/318362/>
- [4] M. S. Putra, Y. Rahma Wati, D. Stkip, and A. A. Dompui, "Pengembangan Aplikasi Psikologi Remaja Berbasis Android (API MADRID) Sebagai Solusi dalam Mengatasi Permasalahan Pada Usia Remaja (Studi pada SMA N 2 Dompui)," *Jurnal Pendidikan*, vol. 9, no. 1, 2021.
- [5] S. Wijaya, A. Andhika, and M. Ilyas, "Development of Sales Information System for SME with the Waterfall Method: A Grocery Store BSR Case," *Jurnal Teknik Informatika (JUTIF)*, vol. 3, no. 4, pp. 1043–1049, 2022.
- [6] L. Z. Sholekhah and T. Rahayu, "PERANCANGAN SISTEM INFORMASI MONITORING KINERJA SALES BERBASIS WEB MENGGUNAKAN KONSEP GAMIFICATION (STUDI KASUS SALES TEAM UNIT BGES PT TELKOM BEKASI)," in *Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya*, 2022, pp. 237–251.
- [7] Suendri, "Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan)," *ALGORITMA: Jurnal Ilmu Komputer dan Informatika*, p. 1, 2018, [Online]. Available: <http://www.omg.org>
- [8] M. Sumiati, R. Abdillah, and A. Cahyo, "Pemodelan Uml Untuk Sistem Informasi Persewaan Alat Pesta," *Jurnal Fasikom*, vol. 11, no. 2, pp. 79–86, 2021.
- [9] E. Novalia and A. Voutama, "Black Box Testing dengan Teknik Equivalence Partitions Pada Aplikasi Android M-Magazine Mading Sekolah," 2022.
- [10] F. Zayid and E. Ferdiana, "Penerapan Algoritma Spatial Map Matching dengan API Menggunakan GPS untuk Posisi Tumpangan Kendaraan," vol. 10, no. 1, pp. 45–56, 2020, doi: 10.36350/jbs.v10i1.
- [11] "dart - get device IMEI in flutter - Stack Overflow." Accessed: Aug. 06, 2024. [Online]. Available: <https://stackoverflow.com/questions/53030779/get-device-imei-in-flutter>