

**PENGEMBANGAN APLIKASI BERGERAK PEMESANAN
AIR GALON DENGAN METODE WATERFALL (STUDI
KASUS DEPOT SUEGER WATER NGEMPLAK
SLEMAN)**



Disusun Oleh:

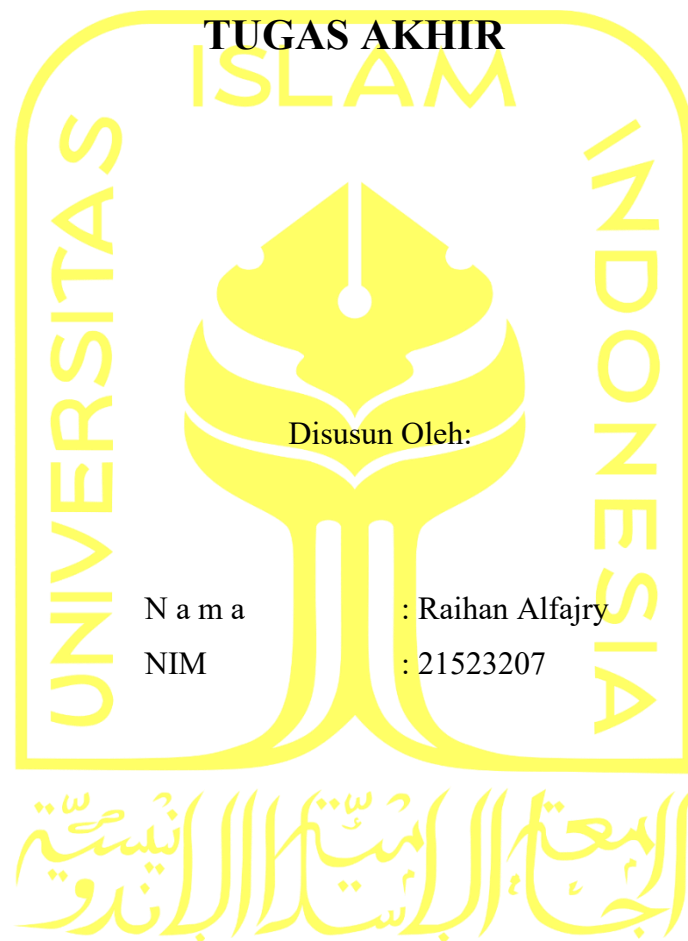
Nama : Raihan Alfajry
NIM : 21523207

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2026

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**PENGEMBANGAN APLIKASI BERGERAK PEMESANAN
AIR GALON DENGAN METODE WATERFALL (STUDI
KASUS DEPOT SUEGER WATER NGEMPLAK
SLEMAN)**



Yogyakarta, 24 April 2026

Pembimbing,

(Kurniawan Dwi Irianto, S.T., M.Sc.)

HALAMAN PENGESAHAN DOSEN PENGUJI

PENGEMBANGAN APLIKASI BERGERAK PEMESANAN AIR GALON DENGAN METODE WATERFALL (STUDI KASUS DEPOT SUEGER WATER NGEMPLAK SLEMAN)

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 24 April 2026

Tim Penguji

Kurniawan Dwi Irianto, S.T., M.Sc.

Anggota 1

Sheila Nurul Huda, S.Kom., M.Cs.

Anggota 2

Moh. Idris, S.Kom., M.Kom.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Raihan Alfajry

NIM : 21523207

Tugas akhir dengan judul:

PENGEMBANGAN APLIKASI BERGERAK PEMESANAN AIR GALON DENGAN METODE WATERFALL (STUDI KASUS DEPOT SUEGER WATER NGEMPLAK SLEMAN)

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 24 April 2026


Raihan Alfajry

HALAMAN PERSEMBAHAN

Dengan rasa Syukur yang mendalam kepada Allah SWT atas segala limpahan Rahmat, kasih sayang, dan karunia-Nya, skripsi ini penulis persembahkan kepada:

1. Kedua orang tua saya, yang selalu memberikan kasih sayang, doa, dukungan moral, serta menjadi sumber inspirasi dalam menjalani setiap Langkah kehidupan. Tanpa kasih sayang, dan pengorbanan orang tua penulis, penulis tidak akan mampu berada pada titik ini.
2. Kakak tercinta, yang selalu memberikan semangat, doa, dan dorongan dalam setiap proses perjuangan.
3. Teman-teman seperjuangan, atas kebersamaan, dukungan, tawa, dan perjuangan Bersama yang tidak akan pernah terlupakan.
4. Untuk diri sendiri, sebagai pengingat bahwa proses, usaha, dan ketekunan akan membawa hasil terbaik.

HALAMAN MOTO

"Kesabaran adalah unsur paling menyehatkan dalam hidup kita."

"Jagalah salatmu. Karena saat kamu kehilangan salat, maka kamu akan kehilangan segalanya."

"Ummar Bin Khattab"

"Dan tolong-menolonglah kamu dalam kebaikan dan takwa."

(QS. Al-Maidah: 2)

KATA PENGANTAR

Puji Syukur kehadiran Allah SWT atas segala limpahan Rahmat, dan hidayah-Nya sehingga penulis dapat menyelesaikan penyusunan laporan skripsi yang berjudul “Pengembangan Aplikasi Bergerak Pemesanan Air Galon dengan Metode Waterfall” dengan baik dan tepat waktu.

Penyusunan Penyusunan skripsi ini bertujuan untuk mengembangkan aplikasi berbasis Android yang dapat membantu proses pemesanan galon secara digital serta mendukung pengelolaan pesanan oleh penjual, sehingga mampu memberikan kemudahan bagi pengguna dalam melakukan pemesanan air galon secara lebih praktis, cepat, dan terstruktur.

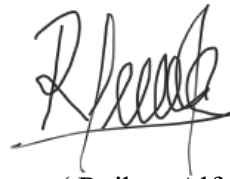
Dalam proses penyusunan skripsi ini, penulis menghadapi beberapa kendala, terutama terkait manajemen waktu. Namun berkat dukungan dan motivasi dari berbagai pihak, kendala tersebut dapat dilalui dengan baik hingga laporan ini dapat terselesaikan.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT atas segala berkah dan Rahmat yang diberikan.
2. Kedua orang tua tercinta, atas doa, dukungan, dan motivasi yang tiada henti diberikan.
3. Kakak saya tercinta yang telah memberikan doa dan dukungan.
4. Bapak Kurniawan Dwi Irianto, S.T., M.Sc., selaku dosen pembimbing, yang telah menyediakan waktu, tenaga, serta memberikan arahan serta bimbingan dalam proses penyusunan skripsi ini.
5. Bapak dan Ibu dosen pengajar program studi informatika, atas ilmu yang telah diajarkan dan diberikan selama masa perkuliahan.
6. Teman-teman seperjuangan, yaitu Reiky, Arga, Alya, Emil, Aliah, Akhtar, Sabina, yang selalu memberikan dukungan, dan semangat dalam menyelesaikan tugas akhir ini.
7. Semua pihak lain yang tidak dapat disebutkan satu per satu namun telah membantu dalam proses penelitian maupun penyusunan laporan ini.

Penulis menyadari bahwa skripsi ini masih banyak kurangnya, baik dari isi maupun penyajiannya. Penelitian ini diharapkan dapat memberikan manfaat bagi pembaca, menjadi referensi bagi pengembang penelitian selanjutnya, serta menjadi kontribusi untuk perkembangan ilmu pengetahuan dan teknologi, khususnya pada bidang pengembangan aplikasi *mobile*.

Yogyakarta, 24 April 2026

A handwritten signature in black ink, appearing to read 'Raihan', with a long horizontal stroke extending from the bottom of the letters.

(Raihan Alfajry)

SARI

Perkembangan teknologi mobile mendorong kebutuhan akan sistem yang mampu mendukung proses pemesanan air galon secara *digital*, praktis, dan efisien. Penelitian ini bertujuan untuk mengembangkan aplikasi pemesanan galon berbasis Android yang dapat membantu pembeli dalam melakukan pemesanan air galon serta membantu penjual dalam mengelola data pesanan, riwayat transaksi, dan informasi jenis galon secara lebih terstruktur. Aplikasi dikembangkan menggunakan metode Waterfall yang meliputi tahapan analisis kebutuhan, desain sistem, implementasi, dan pengujian. Dalam implementasinya, aplikasi memanfaatkan Firebase Authentication untuk autentikasi pengguna, Firebase Realtime Database untuk penyimpanan data jenis galon, serta Cloud Firestore untuk penyimpanan data pengguna dan data pesanan. Pengujian fungsional dilakukan menggunakan Black box *Testing* dan menunjukkan bahwa seluruh fitur utama aplikasi berjalan sesuai dengan kebutuhan dan skenario yang dirancang. Selain itu, pengujian kegunaan menggunakan System Usability Scale (SUS) memperoleh skor rata-rata 75,2 yang termasuk dalam kategori *Good*, sehingga menunjukkan bahwa aplikasi dapat diterima dan cukup mudah digunakan oleh pengguna.

Kata Kunci: Pemesanan Galon, Android, Waterfall, Firebase, Black box *Testing*, System Usability Scale (SUS).

GLOSARIUM

Activity Diagram	Diagram alur aktivitas yang menggambarkan proses jalannya sistem.
Android Studio	Lingkungan pengembang terpadu (IDE) resmi untuk pengembangan aplikasi berbasis android.
Authentication	Proses verifikasi identitas pengguna sebelum mendapatkan akses ke sistem.
Black Box Testing	Metode pengujian perangkat lunak yang fokusnya pada pengujian fungsionalitas tanpa melihat kode program di dalamnya.
COD (Cash on Delivery)	Metode pembayaran yang dilakukan saat pesanan diterima oleh pembeli di lokasi pengantaran.
Firebase	Platform layanan backend milik Google yang menyediakan <i>hosting database</i> , autentikasi pengguna, dan sinkronasi data real-time.
Firebase Firestore	Layanan basis data milik Firebase yang digunakan untuk menyimpan dan mengelola data aplikasi secara terstruktur dalam bentuk koleksi dan dokumen.
Firebase Realtime Database	Layanan basis data dari Firebase yang menyimpan dan menyinkronkan data secara langsung.
Galon	Wadah air minum berkapasitas besar yang digunakan untuk memenuhi kebutuhan air minum sehari-hari dan menjadi objek utama dalam proses pemesanan pada aplikasi.
Kode OTP	Kode verifikasi sekali pakai yang dikirim atau digunakan untuk memastikan bahwa nomor handphone yang didaftarkan benar-benar milik pengguna.
Kotlin	Bahasa pemrograman utama yang digunakan dalam pengembangan aplikasi android.
Nomor Handphone	Nomor handphone pengguna yang digunakan sebagai data kontak dalam proses registrasi, verifikasi akun, dan pemesanan galon.
QRIS (Quick Response Code Indonesian Standard)	Standar kode QR pembayaran di Indonesia yang digunakan untuk melakukan transaksi digital secara non-tunai melalui aplikasi pembayaran.

Black box(SUS)	metode pengukuran Tingkat kebergunaan aplikasi berdasarkan hasil penilaian pengguna.
Use Case Diagram	Diagram UML yang menggambarkan interaksi antara actor dan system melalui Kumpulan fungsi-fungsi utama sistem.
Waterfall	Metode pengembangan perangkat lunak dengan tahapan berurutan.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI.....	ix
GLOSARIUM.....	x
DAFTAR ISI	xii
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan	4
1.7.1 Pendahuluan	4
1.7.2 Tinjauan Pustaka	5
1.7.3 Metodologi Penelitian	5
1.7.4 Hasil dan Pembahasan	5
1.7.5 Kesimpulan dan Saran	5
BAB II TINJAUAN PUSTAKA.....	6
2.1 Galon.....	6
2.2 Android Studio.....	6
2.3 Firebase	7
2.4 Cloudinary.....	8
2.5 <i>Use case diagram</i>	9
2.6 Black box(SUS)	10
2.7 <i>Activity Diagram</i>	13
2.8 Black box <i>Testing</i>	15
2.9 Kajian Pustaka.....	15
2.10 <i>Waterfall</i>	19
BAB III METODE PENELITIAN	21
3.1 Analisis Kebutuhan	22
3.1.1 Teknik Pengumpulan Data.....	22
3.1.2 Analisis Kebutuhan Fungsional	28
3.1.3 Analisis Kebutuhan Non-Fungsional	28
3.2 Desain Aplikasi	29
3.2.1 <i>Use case diagram</i>	29
3.2.2 <i>Activity Diagram</i>	31
3.2.3 Rancangan Antarmuka	43
3.3 Pemrograman	58
3.4 Pengujian.....	58

BAB IV HASIL DAN PEMBAHASAN	59
4.1 Fitur Aplikasi	59
4.1.1 Fitur Pemesanan Air Galon.....	59
4.1.2 Fitur Riwayat Transaksi dan Status Pesanan	60
4.1.3 Fitur Pengelolaan Pesanan oleh Penjual	60
4.2 Implementasi Sistem	60
4.2.1 Implementasi Halaman Login.....	61
4.2.2 Implementasi Halaman Registrasi	61
4.2.3 Implementasi Halaman Home Pembeli.....	62
4.2.4 Implementasi Halaman Riwayat Pesanan Pembeli	63
4.2.5 Implementasi Halaman Akun Pembeli	67
4.2.6 Implementasi Dialog Pemesanan Galon dan Mekanisme Pembayaran ...	68
4.2.7 Implementasi Halaman Home Penjual.....	74
4.2.8 Implementasi Halaman Riwayat Pesanan Penjual	75
4.2.9 Implementasi Halaman Pengaturan Jenis dan Harga Galon	76
4.2.10 Implementasi Perubahan Status Pesanan	77
4.2.11 Implementasi Sistem Notifikasi dan Batas Waktu Pembayaran	91
4.3 Integrasi Firebase	96
4.3.1 Firebase Authentication	97
4.3.2 Firebase Realtime Database	97
4.3.3 Cloud Firestore.....	98
4.4 Integrasi Layanan Cloudinary	100
4.5 Pengujian Black box	101
4.6 Pengujian System Usability Scale (SUS).....	106
4.7 Analisis Kelebihan dan Kekurangan Aplikasi	109
4.8 Pembahasan.....	112
BAB V KESIMPULAN DAN SARAN.....	116
5.1 Kesimpulan	116
5.2 Saran.....	116
DAFTAR PUSTAKA.....	118
LAMPIRAN	123

DAFTAR TABEL

Tabel 2.1 Simbol <i>Use Case Diagram</i>	9
Tabel 2.2 Tabel Pertanyaan SUS	11
Tabel 2.3 Tabel Skala Likert SUS	11
Tabel 2.4 Simbol <i>Activity Diagram</i>	14
Tabel 2.5 Analisis Penelitian Terdahulu dan Kebaruan Penelitian.....	16
Tabel 3.1 Struktur database akun.....	55
Tabel 3. 2 Struktur Desain Database Galon.....	56
Tabel 3.3 Struktur Desain Database Users	56
Tabel 3. 4 Struktur Desain Database <i>Order</i>	57
Tabel 4.1 Pengujian Black box	101
Tabel 4.2 Pengujian <i>System Usability Scale</i>	107
Tabel 4.3 Rata-rata skor per pertanyaan	108
Tabel 4.4 Rekapitulasi Penilaian.....	108

DAFTAR GAMBAR

Gambar 2.1 Logo Android Studio	6
Gambar 2.2 Logo Firebase.....	8
Gambar 3.1 Metode <i>Waterfall</i>	21
Gambar 3.2 Lokasi Depot Sueger Water	22
Gambar 3.3 <i>Use case diagram</i> Aplikasi	30
Gambar 3.4 Activity Diagram Registrasi Pembeli	32
Gambar 3.5 <i>Activity Diagram</i> Login Pengguna.....	33
Gambar 3.6 <i>Activity Diagram</i> Home Pembeli	34
Gambar 3.7 <i>Activity Diagram</i> Riwayat Pesanan Pembeli	35
Gambar 3.8 <i>Activity Diagram</i> Pengaturan Akun Pembeli dan Ubah Lokasi.....	36
Gambar 3.9 <i>Activity Diagram</i> Pemesanan Galon dengan Metode COD	37
Gambar 3.10 Activity Diagram Pemesanan Galon dengan Metode QRIS	38
Gambar 3.11 <i>Activity Diagram</i> Home Penjual	39
Gambar 3.12 <i>Activity Diagram</i> Riwayat Pesanan Penjual.....	40
Gambar 3.13 <i>Activity Diagram</i> Kelola Status Pesanan COD	41
Gambar 3.14 <i>Activity Diagram</i> Kelola Status Pesanan QRIS	42
Gambar 3.15 <i>Activity Diagram</i> Kelola Jenis dan Harga Galon	43
Gambar 3.16 Sketsa Desain tampilan Halaman Login	44
Gambar 3.17 Sketsa Desain Tampilan Halaman <i>Home</i> Penjual.....	45
Gambar 3.18 Tampilan Dialog Ubah <i>Password</i>	46
Gambar 3.19 Sketsa Desain Tampilan Halaman Riwayat Pesanan	46
Gambar 3.20 Sketsa Tampilan Halaman Pengaturan Jenis & Harga.....	47
Gambar 3.21 Tampilan Dialog Tambah, Edit, dan Hapus Jenis Galon	47
Gambar 3.22 Tampilan Dialog Ubah Status Pesanan COD	48
Gambar 3.23 Tampilan Dialog Verifikasi dan Catatan Status Pesanan QRIS	48
Gambar 3.24 Tampilan Halaman Login	49
Gambar 3.25 Tampilan Halaman Registrasi	49
Gambar 3.26 Sketsa Desain Tampilan Halaman Home.....	50
Gambar 3.27 Sketsa Desain Tampilan Halaman Pesanan Galon	51
Gambar 3.28 Tampilan Dialog Pembayaran QRIS.....	52
Gambar 3.29 Tampilan Dialog Hasil Pesanan QRIS Status Menunggu Konfirmasi Penjual .	52
Gambar 3.30 Tampilan Dialog Hasil Pesanan QRIS Status Menunggu Pembayaran	53

Gambar 3.31 Sketsa Desain Tampilan Halaman Riwayat Pesanan	54
Gambar 3.32 Sketsa Tampilan Halaman Akun.....	54
Gambar 3.33 Sketsa Tampilan Dialog Edit Lokasi	55
Gambar 4.1 Implementasi Halaman Login.....	61
Gambar 4.2 Implementasi Halaman Registrasi	62
Gambar 4.3 Implementasi Halaman Home Pembeli.....	63
Gambar 4.4 Implementasi Halaman Riwayat Pesanan Pembeli pada Tab Diproses	64
Gambar 4.5 Implementasi Halaman Riwayat Pesanan Pembeli pada Tab Selesai.....	65
Gambar 4.6 Tampilan Pratinjau Bukti Pengantaran pada Halaman Riwayat Pembeli.....	66
Gambar 4.7 Dialog Perbesaran Foto Bukti Pengantaran pada Sisi Pembeli.....	67
Gambar 4.8 Implementasi Halaman Akun Pembeli	68
Gambar 4.9 Implementasi Dialog Pemesanan Galon	69
Gambar 4.10 Tampilan Pesanan dengan COD pada Halaman Home Pembeli	70
Gambar 4.11 Implementasi Dialog Pembayaran QRIS	71
Gambar 4.12 Dialog Hasil Pesanan QRIS dengan Status Menunggu Konfirmasi Penjual	72
Gambar 4.13 Dialog Hasil Pesanan QRIS dengan Status Menunggu Pembayaran.....	73
Gambar 4.14 Tampilan Status Pesanan QRIS pada Halaman Home Pembeli	74
Gambar 4.15 Implementasi Halaman Home Penjual.....	75
Gambar 4.16 Implementasi Halaman Riwayat Pesanan Penjual	76
Gambar 4.17 Implementasi Halaman Pengaturan Jenis dan Harga Galon	77
Gambar 4.18 Tampilan Pesanan COD pada Sisi Penjual Sebelum Perubahan Status.....	78
Gambar 4.19 Dialog Kelola Pesanan COD pada Sisi Penjual	78
Gambar 4.20 Tampilan Perubahan Status Pesanan COD Menjadi Dikirim pada Sisi Penjual	79
Gambar 4.21 Dialog Perubahan Status Lanjutan Pesanan COD pada Sisi Penjual.....	80
Gambar 4.22 Tampilan Hasil Akhir Perubahan Status Pesanan COD pada Sisi Penjual.....	81
Gambar 4.23 Tampilan Pesanan QRIS dengan Status Menunggu Konfirmasi Penjual pada Sisi Penjual.....	82
Gambar 4.24 Dialog Kelola Pesanan QRIS pada Sisi Penjual	83
Gambar 4.25 Dialog Pengisian Catatan Pembayaran Kurang pada Sisi Penjual.....	84
Gambar 4.26 Tampilan Pesanan QRIS dengan Status Pembayaran Kurang pada Sisi Penjual	84
Gambar 4.27 Tampilan Pesan dari Penjual terkait Pembayaran Kurang pada Sisi Pembeli ...	85
Gambar 4.28 Dialog Pembayaran QRIS untuk Melanjutkan Pembayaran pada Sisi Pembeli	86

Gambar 4.29 Tampilan Pesanan QRIS dengan Status Menunggu Pembayaran pada Sisi Pembeli	87
Gambar 4.30 Dialog Pemilihan Metode Pelunasan oleh Pembeli	88
Gambar 4.31 Tampilan Pesanan dengan Status Sisa Bayar COD	89
Gambar 4.32 Tampilan Pesanan QRIS dengan Status Pembayaran Diterima pada Sisi Penjual	90
Gambar 4.33 Dialog Perubahan Status Pengiriman Pesanan QRIS pada Sisi Penjual	90
Gambar 4.34 Dialog Penyelesaian Status Akhir Pesanan QRIS pada Sisi Penjual	91
Gambar 4.35 Tampilan Notifikasi Pembuatan Pesanan pada Sisi Pembeli	92
Gambar 4.36 Tampilan Notifikasi Pembatalan Otomatis akibat Batas Waktu Habis	93
Gambar 4.37 Tampilan Notifikasi Verifikasi Pembayaran QRIS pada Sisi Pembeli	94
Gambar 4.38 Tampilan Notifikasi Perubahan Status Pengiriman pada Sisi Pembeli	95
Gambar 4.39 Tampilan Notifikasi Pesanan Baru Masuk pada Sisi Penjual	96
Gambar 4.40 Tampilan Integrasi Firebase dengan Android Studio	96
Gambar 4.41 Tampilan Firebase Authentication	97
Gambar 4.42 Struktur Data Jenis Galon pada Firebase Realtime Database	98
Gambar 4.43 Struktur Data <i>Users</i> pada Cloud Firestore	99
Gambar 4.44 Struktur Data <i>Orders</i> pada Cloud Firestore	100
Gambar 4.45 Konfigurasi Upload Preset pada Cloudinary	101

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada era digital, pemanfaatan teknologi informasi semakin melekat dalam aktivitas masyarakat untuk memenuhi kebutuhan sehari-hari. Berbagai layanan pemesanan berbasis aplikasi maupun website berkembang karena mampu mempercepat proses transaksi, memperluas jangkauan layanan, serta memudahkan pengelolaan data pelanggan dan pesanan. Transformasi digital ini juga relevan diterapkan pada layanan pemenuhan kebutuhan rumah tangga yang bersifat rutin, termasuk pemesanan air minum isi ulang/air galon, agar proses pemesanan menjadi lebih praktis dan terstruktur (Devansyah dkk., 2025; Firdaus dkk., 2025).

Menjaga kesehatan merupakan suatu hal yang penting untuk dilakukan oleh setiap orang. Banyaknya penyakit yang berkembang saat ini menunjukkan pentingnya menjaga kesehatan. Salah satu upaya seseorang untuk menjaga kesehatannya adalah dengan mengonsumsi air. Air merupakan komponen penting yang dibutuhkan dalam tubuh manusia, hal ini dikarenakan 80% dalam tubuh manusia terdiri dari cairan dan membutuhkan 2,5 liter air setiap harinya (Kusumawardani & Larasati, 2020). Dengan mengonsumsi air yang cukup dapat mengurangi risiko terkena penyakit tertentu, seperti batu ginjal, dehidrasi, dan lain-lain. Konsumsi air yang cukup setiap hari juga berperan penting dalam menjaga keseimbangan tubuh serta membantu proses metabolisme yang sehat (Abadi dkk., 2021).

Untuk memenuhi kebutuhan konsumsi air tersebut, masih banyak keluarga yang memenuhi kebutuhan air minum melalui air galon karena relatif mudah diperoleh dan lebih efisien untuk kebutuhan harian. Namun, proses pemesanan air galon di lapangan masih sering dilakukan secara manual, misalnya pelanggan harus datang langsung ke depot/toko atau melakukan pemesanan melalui telepon/WhatsApp. Cara ini berpotensi menimbulkan berbagai kendala, seperti ketidakpastian jadwal pengantaran, serta keterbatasan pencatatan riwayat pesanan pelanggan (Fiki dkk., 2021; Syukron dkk., 2023).

Permasalahan lain yang kerap muncul pada pemesanan manual adalah risiko kesalahan informasi pesanan (jumlah/jenis galon, alamat, dan waktu antar), serta kendala dalam proses pengantaran karena pengantar tidak selalu memiliki informasi lokasi pelanggan yang jelas. Penelitian terkait layanan depot air menunjukkan bahwa pemesanan dan pengantaran yang masih mengandalkan komunikasi sederhana (telepon/percakapan pesan) dan/atau ingatan pengantar dapat menyulitkan pencarian alamat pelanggan dan berdampak pada keterlambatan

layanan ((Fiki dkk., 2021; Mahatmi dkk., 2022)). Selain itu, ketika pemesanan dilakukan melalui pesan singkat/telepon, pelanggan juga dapat mengalami ketidaknyamanan akibat menunggu terlalu lama atau harus melakukan konfirmasi ulang karena pesanan terlewat ((Syukron dkk., 2023)).

Dari sisi pelaku usaha (depot/toko air galon), pengelolaan pesanan secara manual juga menyulitkan pencatatan transaksi, rekap penjualan, serta pengelolaan data pelanggan secara rapi. Kondisi ini dapat memicu *human error*, memperlambat pelayanan, dan menyulitkan pemilik usaha dalam memantau pesanan yang masuk maupun membuat laporan. Sejumlah penelitian pengembangan sistem pemesanan air minum isi ulang menunjukkan bahwa digitalisasi pemesanan dapat membantu mengurangi kesalahan pencatatan, mempercepat proses layanan, dan meningkatkan efisiensi operasional usaha (Devansyah dkk., 2025; Firdaus dkk., 2025).

Berdasarkan permasalahan tersebut, diperlukan sebuah solusi berupa aplikasi pemesanan air galon yang mampu memfasilitasi pelanggan dalam melakukan pemesanan secara lebih mudah dan terdokumentasi, serta membantu pihak depot atau toko dalam mengelola pesanan secara terpusat. Konsep pengembangan aplikasi pemesanan air galon atau isi ulang berbasis Android maupun website telah banyak diteliti dan terbukti dapat mempermudah proses pemesanan serta meningkatkan kualitas layanan bagi pelanggan maupun pelaku usaha (Fiki dkk., 2021; Ratnawati dkk., 2019; Syukron dkk., 2023). Oleh karena itu, penelitian ini difokuskan pada pengembangan aplikasi berbasis *mobile* untuk pemesanan galon sebagai media pemesanan yang praktis dan efektif, serta mampu mendukung pengelolaan transaksi secara lebih tertib dan efisien, sehingga dapat memenuhi kebutuhan pelanggan akan pemesanan air galon yang cepat sekaligus meningkatkan kualitas pelayanan dan efisiensi pengelolaan pesanan pada depot.

1.2 Rumusan Masalah

Bagaimana merancang dan membangun aplikasi pemesanan air galon berbasis Android untuk membantu pelanggan dalam melakukan pemesanan dan membantu depot dalam mengelola pesanan?

1.3 Batasan Masalah

- a. Penelitian hanya membahas proses pemesanan air galon dan pengelolaan pesanan pada depot.

- b. Aplikasi hanya dapat digunakan pada *smartphone* Android.
- c. Sistem ini hanya berjalan secara *online*.

1.4 Tujuan Penelitian

Mengembangkan aplikasi pemesanan air galon berbasis Android untuk mempermudah proses transaksi antara pembeli dan penjual, yang dilengkapi dengan fitur riwayat pesanan, pilihan metode pembayaran (tunai dan non-tunai), serta foto bukti pengantaran dalam pemesanan galon.

1.5 Manfaat Penelitian

1. Bagi Pembeli
 - a. Mempermudah proses pemesanan air galon kapan saja dan di mana saja tanpa harus datang langsung atau menelepon pihak depot.
 - b. Membantu pembeli mengetahui status pesannya dengan jelas (apakah sedang diproses, dikirim, atau selesai) melalui riwayat transaksi.
 - c. Memberikan kemudahan dalam membayar tagihan melalui pilihan metode tunai (COD) maupun non-tunai (QRIS), termasuk jika ada kekurangan bayar.
 - d. Memberikan rasa aman karena adanya fitur foto bukti pengantaran yang memastikan galon sudah benar-benar sampai di lokasi.
2. Bagi Penjual
 - a. Mempermudah penjual dalam mencatat pesanan masuk, mengelola daftar harga, dan memperbarui stok galon secara digital.
 - b. Membantu penjual menghindari kerugian melalui fitur pengecekan pembayaran QRIS sebelum galon dikirim, serta membatalkan otomatis pesanan yang tidak dibayar.
 - c. Meningkatkan kualitas pelayanan dan kepercayaan pelanggan terhadap depot dengan adanya bukti foto setiap kali pesanan selesai diantar.
3. Bagi Penulis
 - a. Mendapatkan pengalaman berharga dan memperluas wawasan dalam pengembangan aplikasi.
 - b. Dapat melatih kemampuan dalam pemrograman dan perancangan antarmuka.

1.6 Metodologi Penelitian

1. Analisis Kebutuhan

Pada tahapan ini bertujuan untuk melakukan analisis kebutuhan pada sistem seperti apa saja yang dibutuhkan oleh sistem. Pada tahapan analisis kebutuhan ini informasi terkait kebutuhan aplikasinya didapatkan melalui wawancara dengan pihak penjual galon dan pembeli di Depot.

2. Desain Sistem

Pada tahapan ini bertujuan untuk memaparkan desain antarmuka pengguna dan struktur dari *database* untuk aplikasi pemesanan galon. Rancangan tersebut dirancang untuk memvisualisasikan interaksi antara pengguna dan antarmuka, serta untuk memastikan agar data diatur dan disimpan secara terstruktur dalam sistem.

3. Pemrograman/Implementasi

Pada tahapan ini dilakukan pemrograman untuk mengembangkan aplikasi pemesanan galon berdasarkan dengan desain yang telah dikerjakan sebelumnya. Pembuatan aplikasi ini menggunakan bahasa pemrograman kotlin.

4. Pengujian

Pada Pada tahapan ini dilakukan pengujian untuk mengetahui apakah aplikasi dapat berjalan sesuai fungsinya serta meminimalkan kesalahan pada sistem. Pengujian dilakukan menggunakan metode *Black box Testing* untuk menguji fungsi aplikasi berdasarkan input dan output, serta metode *System Usability Scale (SUS)* untuk menilai tingkat kegunaan dan penerimaan aplikasi oleh pengguna.

1.7 Sistematika Penulisan

Sistematika penulisan bertujuan untuk memberikan gambaran dari struktur penelitian agar lebih terarah dan sistematis. Adapun sistematika penulisan dalam penelitian ini adalah sebagai berikut:

1.7.1 Pendahuluan

Pada bab ini akan dibahas pendahuluan yang terdiri atas latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan.

1.7.2 Tinjauan Pustaka

Pada bab ini akan membahas berbagai teori dan referensi yang berkaitan dengan topik penelitian ini, tools dan metode yang digunakan dalam pengembangan aplikasi, serta kajian pustaka terdahulu yang relevan dengan penelitian.

1.7.3 Metodologi Penelitian

Pada bab ini akan membahas metode penelitian yang dilakukan dalam pengembangan aplikasi, mulai dari tahapan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian.

1.7.4 Hasil dan Pembahasan

Pada bab ini menyajikan pembahasan mengenai hasil implementasi aplikasi, tampilan antarmuka, serta hasil pengujian fungsional menggunakan black box dan evaluasi *usability* menggunakan metode System Usability Scale (SUS).

1.7.5 Kesimpulan dan Saran

Pada bab ini berisi kesimpulan dari hasil penelitian serta saran untuk pengembangan aplikasi di masa mendatang.

BAB II

TINJAUAN PUSTAKA

2.1 Galon

Galon merupakan kemasan air minum berukuran besar yang digunakan sebagai wadah penyimpanan dan penyaluran air minum untuk kebutuhan sehari-hari. Dalam kajian mengenai air minum dalam kemasan galon, Maulana & Riski Hapsari, (2024) menjelaskan bahwa proses produksi air minum dalam kemasan mencakup tahapan penyaringan, disinfeksi, dan pengisian produk, sehingga kemasan galon tidak hanya berfungsi sebagai wadah, tetapi juga menjadi bagian penting dalam menjaga mutu produk selama proses distribusi hingga dikonsumsi. Dalam penelitian Citra Agustina, (2021) menegaskan bahwa kualitas air minum perlu ditinjau dari keberadaan cemaran mikrobiologi seperti *coliform* dan *Escherichia coli*, karena keberadaan bakteri tersebut menjadi indikator bahwa mutu air minum belum sepenuhnya aman untuk dikonsumsi. Dengan demikian, galon dapat diartikan sebagai kemasan air minum berkapasitas besar yang mendukung kepraktisan penggunaan, namun tetap harus memenuhi aspek kebersihan, keamanan, dan pengendalian mutu agar air yang dikonsumsi tetap layak dan aman.

2.2 Android Studio



Gambar 2.1 Logo Android Studio

Android studio merupakan sebuah *Integrated Development Environment* (IDE) yang resmi untuk pengembangan aplikasi android. Android studio dibuat berbasis kode editor dan alat developer dari IntelliJ IDEA. Android studio juga menawarkan fitur-fitur seperti sistem build berbasis gradle yang fleksibel, kemampuan untuk membangun berbagai versi aplikasi untuk perangkat android, serta editor layout visual yang sangat intuitif. Dengan terintegrasi nya

android studio ke *Android Software Development Kit* (SDK), pengembang aplikasi dapat langsung menguji dan menerbitkan aplikasinya ke perangkat android (Sibuea dkk., 2022).

Android studio mendukung dua bahasa pemrograman yang utama, yaitu Java dan Kotlin. Dalam penelitian ini, bahasa pemrograman kotlin yang dipilih karena keunggulannya dalam efisiensi penulisan kode serta fitur-fitur modern yang mendukung produktivitas, sehingga diharapkan dapat mempermudah proses pengembangan aplikasi ini.

2.3 Firebase

Firebase merupakan sebuah *platform* pengembangan aplikasi yang disediakan oleh Google, yang di dalamnya menawarkan berbagai jenis layanan yang berguna untuk membantu pengguna dalam pengembangan aplikasi yang berkualitas tinggi. Salah satu fitur utama yang ada di dalam Firebase ini adalah *Real-time Database*, yang memungkinkan pengguna untuk melakukan penyimpanan dan pengambilan data secara *real-time*. Sehingga memungkinkan aplikasi untuk memberikan pengalaman pengguna yang responsif dan interaktif (Sugiyatno, 2023).

Firebase *Real-time Database* sangat ideal digunakan dalam penggunaan aplikasi yang membutuhkan penyampaian hasil informasi secara cepat. Data yang telah disimpan di dalam Firebase dapat langsung dilihat dan diperbarui oleh pengguna dalam hitungan milidetik tanpa perlu melakukan refresh halaman secara manual. Firebase juga mendukung berbagai jenis data seperti string, number, long, dan boolean. Semua data yang terdapat di dalam Firebase disusun dalam *format* JSON, sehingga membuat pengelolaan data menjadi lebih efisien (Andrianto & Munandar, 2022).

Pada Selain itu, Firebase juga menyediakan layanan Firebase Authentication yang digunakan untuk mengelola proses autentikasi pengguna. Layanan ini memudahkan pengembang dalam menerapkan sistem login dan registrasi karena telah menyediakan mekanisme autentikasi yang terintegrasi. Firebase Authentication mendukung beberapa metode masuk, seperti email dan kata sandi, nomor telepon, serta akun pihak ketiga seperti Google, Facebook, GitHub, dan Twitter. Dengan adanya layanan ini, proses autentikasi pengguna dapat diterapkan dengan lebih praktis serta membantu meningkatkan kemudahan pengguna dalam mengakses aplikasi (Chougale dkk., 2021).

Firebase juga memiliki layanan basis data lain yaitu Cloud Firestore. Cloud Firestore merupakan basis data NoSQL berbasis dokumen yang menyimpan data dalam bentuk koleksi dan dokumen, sehingga struktur data yang digunakan menjadi lebih fleksibel dan terorganisir.

Data pada Cloud Firestore disimpan sebagai objek JSON dan dapat diakses melalui API yang telah disediakan, sehingga memudahkan pengembang dalam mengelola data sesuai kebutuhan aplikasi. Dengan model penyimpanan tersebut, Cloud Firestore sesuai digunakan untuk aplikasi yang membutuhkan pengelolaan data secara terstruktur, seperti data pengguna dan data transaksi (Chougale dkk., 2021).

Pada penelitian ini, Firebase digunakan untuk mendukung kebutuhan backend pada aplikasi. Firebase Authentication dimanfaatkan untuk proses registrasi dan login pengguna, Firebase Realtime Database digunakan untuk menyimpan data jenis galon, sedangkan Cloud Firestore digunakan untuk menyimpan data pengguna dan data pesanan. Dalam pemanfaatannya, penggunaan layanan Firebase tersebut diharapkan dapat membantu aplikasi berjalan dengan lebih terstruktur, efisien, dan sesuai dengan kebutuhan sistem pemesanan galon yang dikembangkan.



Gambar 2.2 Logo Firebase

2.4 Cloudinary

Cloudinary merupakan layanan penyimpanan media berbasis cloud yang dapat digunakan untuk menyimpan gambar pada aplikasi berbasis web maupun mobile. Dalam pengembangan sistem, Cloudinary berfungsi sebagai tempat penyimpanan gambar yang diunggah oleh pengguna, sehingga file gambar tidak disimpan secara langsung pada server aplikasi.

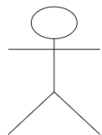
Pada proses penggunaannya, gambar yang diunggah melalui aplikasi akan dikirim dan disimpan ke layanan Cloudinary. Setelah proses unggah berhasil, Cloudinary akan menghasilkan URL gambar. URL tersebut kemudian disimpan ke dalam database sistem, sedangkan file gambar tetap tersimpan pada server Cloudinary. Dengan cara ini, aplikasi hanya perlu menyimpan alamat gambar, bukan file gambar secara langsung.

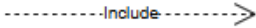
Penggunaan Cloudinary sebagai penyimpanan gambar dapat membantu mengurangi penggunaan ruang penyimpanan pada server utama aplikasi. Selain itu, pengelolaan gambar menjadi lebih efisien karena file gambar tersimpan pada layanan cloud yang memang dirancang untuk menangani penyimpanan dan distribusi media. Penelitian oleh Dey, Jaiswal, Hemasabari (2025) menjelaskan bahwa Cloudinary dapat digunakan sebagai layanan cloud-based media storage and management yang membantu proses unggah, penyimpanan, dan pengelolaan media pada aplikasi. Oleh karena itu, dalam penelitian ini Cloudinary digunakan secara khusus sebagai layanan penyimpanan gambar, di mana file gambar disimpan pada Cloudinary dan URL gambar disimpan pada database aplikasi.

2.5 Use case diagram

Use case diagram (UCD) merupakan salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan hubungan antara pengguna dengan sistem yang dikembangkan. Diagram ini menampilkan fungsi-fungsi utama dari sistem dalam bentuk *use case* dan menunjukkan siapa saja yang bisa berinteraksi dengan fungsi tersebut (Hafsari dkk., 2023). *Use case diagram* berperan penting dalam tahap analisis karena dapat menjelaskan kebutuhan fungsional secara visual dan mudah dipahami oleh pengembang maupun pengguna non-teknis (Ichsandi dkk., 2025). UCD juga membantu menentukan batasan sistem dan ruang lingkup fungsional yang akan dikembangkan, UCD juga berfungsi untuk menggambarkan alur interaksi antara pengguna dan sistem dalam menentukan keputusan, sehingga memudahkan tim dalam mengidentifikasi fitur utama yang dibutuhkan (Sariangsah, 2024). Simbol *Use Case Diagram* dapat dilihat pada tabel 2.1 Simbol *Use Case Diagram*.

Tabel 2.1 Simbol *Use Case Diagram*.

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i> .
	<i>Use Case</i> : Abstraksi dan interaksi antara sistem dan aktor.

	<i>Directed Association Relationship</i> : hubungan asosiasi yang diarahkan hanya kepada satu arah saja.
	Relasi untuk menunjukkan satu <i>use case</i> dasar memerlukan use case tambahan untuk menjalankan fungsinya.

Pada penelitian ini, *Use case diagram* sebagai alat pemodelan awal yang menggambarkan bagaimana pengguna berinteraksi dengan sistem, misalnya “pemesanan galon”, “melihat riwayat pemesanan”. Hal ini memungkinkan pengembang merumuskan kebutuhan fungsional sistem secara lebih sistematis dan mempersiapkan tahap desain antarmuka serta implementasi dengan landasan yang jelas.

2.6 Black box(SUS)

Black box(SUS) merupakan metode evaluasi standar yang digunakan untuk mengukur tingkat kegunaan suatu sistem atau aplikasi berdasarkan persepsi dari pengguna. Sistem pengujian ini pertama kali dikembangkan oleh John Brooke pada tahun 1986 dan dikembangkan hingga saat ini, metode ini menjadi salah satu metode paling populer karena sistem pengujian ini bersifat sederhana, cepat, serta mampu memberikan hasil kuantitatif yang dapat dipercaya. SUS terdiri dari 10 pernyataan dengan skala *likert* 1-5, yang menilai kemudahan penggunaan, efisiensi, konsistensi, dan kenyamanan sistem bagi pengguna. Hasil dari seluruh pernyataan tersebut diubah menjadi skor 0-100, yang dimana skor yang lebih tinggi menunjukkan tingkat usability yang lebih baik (Sukma dkk., 2023).

Metode SUS ini banyak digunakan dalam berbagai penelitian untuk menilai kualitas dari antarmuka pengguna dan pengalaman pengguna pada aplikasi berbasis web ataupun *mobile*. Pada penelitian Susila dan Arsa (2023) menggunakan metode SUS untuk mengevaluasi sistem pemesanan menu secara mandiri di restoran, dan hasilnya membantu mengidentifikasi aspek antarmuka yang perlu disempurnakan agar lebih mudah dalam penggunaannya. Lupita Dyayu dkk (2023) juga menerapkan SUS pada aplikasi PeduliLindungi dan menemukan bahwa metode ini efektif untuk menilai persepsi dari pengguna terhadap kemudahan dalam navigasi dan tampilan aplikasinya. Pada penelitian evaluasi *usability* E-modul basis data, Putra dkk., (2024) juga menggunakan SUS dan menyimpulkan bahwa SUS mampu memberikan gambaran objektif mengenai kepuasan pengguna terhadap sistem pendidikan digital.

Dalam instrumen System Usability Scale (SUS), butir pertanyaan disusun secara selang-seling antara pernyataan bernada positif dan negatif. Pernyataan pada nomor ganjil, yaitu 1, 3, 5, 7, dan 9, bersifat positif karena mencerminkan pandangan pengguna terkait kemudahan serta kenyamanan saat menggunakan sistem. Sementara itu, pernyataan pada nomor genap, yakni 2, 4, 6, 8, dan 10, bersifat negatif dan berfungsi untuk mengungkap kemungkinan adanya kesulitan, tingkat kerumitan, maupun hambatan yang dialami pengguna selama berinteraksi dengan sistem. Pola penyusunan yang bergantian ini dilakukan untuk membantu menjaga konsistensi jawaban responden sekaligus meningkatkan keabsahan hasil pengukuran usability sistem.

Tabel 2.2 Tabel Pertanyaan SUS

No	Pertanyaan
1.	Saya berpikir bahwa saya akan sering menggunakan sistem ini
2.	Saya merasa sistem ini terlalu kompleks
3.	Saya merasa sistem ini mudah digunakan
4.	Saya membutuhkan bantuan teknis untuk dapat menggunakan sistem ini
5.	Saya merasa fitur-fitur dalam sistem ini terintegrasi dengan baik
6.	Saya merasa terdapat terlalu banyak inkonsistensi dalam sistem ini
7.	Saya merasa kebanyakan orang akan dapat mempelajari sistem ini dengan cepat
8.	Saya merasa sistem ini membingungkan untuk digunakan
9.	Saya merasa percaya diri dalam menggunakan sistem ini
10.	Saya perlu membiasakan diri terlebih dahulu sebelum dapat menggunakan sistem ini

Untuk menilai tingkat usability aplikasi, responden diminta memberikan tanggapan terhadap sepuluh pernyataan pada kuesioner SUS menggunakan pilihan jawaban berdasarkan skala Likert, seperti yang disajikan pada Tabel 2.3 berikut.

Tabel 2.3 Tabel Skala Likert SUS

Jawaban	Skor Nilai
Sangat Tidak Setuju	1
Tidak Setuju	2
Netral	3

Setuju	4
Sangat Setuju	5

Untuk mendapatkan kesimpulan mengenai kinerja sistem, metode System Usability Scale (SUS) menggunakan perhitungan skor berdasarkan rumus yang telah ditetapkan. Pada butir pertanyaan bernomor ganjil, skor dihitung dengan cara mengurangi nilai jawaban responden (X) dengan angka 1 (Prayoga & Kristiana, 2024), sebagaimana ditunjukkan pada Persamaan (2.1).

$$\text{Skor Pertanyaan Ganjil} = X - 1 \quad (2.1)$$

Sementara itu, pada pertanyaan bernomor genap, skor dihitung dengan mengurangi nilai maksimum pada skala, yaitu 5, dengan nilai jawaban yang diberikan responden (X) (Rivdyho & Bahri, 2022), sebagaimana dinyatakan pada Persamaan (2.2).

$$\text{Skor Pertanyaan Genap} = 5 - X \quad (2.2)$$

Berdasarkan hasil perhitungan tersebut, tahap berikutnya adalah menentukan nilai subtotal, yaitu dengan menjumlahkan skor dari seluruh pertanyaan, baik yang bernomor ganjil maupun genap. Proses perolehan nilai subtotal ini ditunjukkan pada Persamaan (2.3).

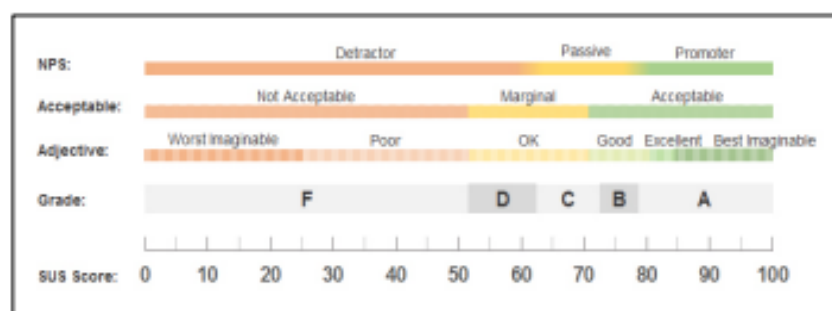
$$\text{Skor Subtotal} = \text{Total Skor Ganjil} + \text{Total Skor Genap} \quad (2.3)$$

Setelah nilai subtotal diperoleh, langkah berikutnya adalah menghitung skor akhir setiap responden dengan mengalikan nilai subtotal tersebut dengan konstanta 2,5 (Rivdyho & Bahri, 2022). Proses penentuan skor akhir responden ini ditunjukkan pada Persamaan (2.4).

$$\text{Skor Akhir Responden} = \text{Subtotal} \times 2.5 \quad (2.4)$$

Setelah seluruh skor akhir dari masing-masing responden diperoleh, tahap berikutnya adalah menghitung skor akhir System Usability Scale (SUS). Perhitungan skor akhir SUS dilakukan dengan cara menjumlahkan seluruh skor akhir responden, kemudian membaginya dengan jumlah responden yang terlibat dalam pengujian. Tullis & Stetson, (2004) menyatakan

bahwa SUS merupakan salah satu instrumen evaluasi usability yang mampu memberikan hasil reliabel meskipun digunakan pada jumlah sampel yang relatif kecil. Dalam penelitiannya, SUS menunjukkan konsistensi hasil yang tinggi dan pada ukuran sampel 12 partisipan telah mampu menghasilkan kesimpulan yang sama dengan dataset penuh setidaknya 90% pada sebagian besar pengujian. Berdasarkan temuan tersebut, ukuran sampel sekitar 12–14 partisipan dinilai cukup untuk memperoleh hasil evaluasi usability yang *reliable*. SUS memiliki rentang penilaian dalam pengukuran kategori penggunaan aplikasi, rentang penilaian ini dapat dilihat melalui penelitian (Maulana dkk., 2025), yang dapat dilihat sebagai berikut:



Gambar 2.3 Interpretasi Skor SUS

Gambar di atas menunjukkan visualisasi interpretasi skor System Usability Scale (SUS) berdasarkan kategori penerimaan (acceptability), tingkat kualitas (grade), serta deskripsi kualitatif (adjective rating). Pada diagram tersebut, skor SUS dibagi menjadi tiga kelompok utama yaitu Not Acceptable, Marginal, dan Acceptable, yang masing-masing merepresentasikan tingkat usability dari suatu sistem. Semakin tinggi skor yang diperoleh, maka semakin baik tingkat kegunaan dan penerimaan sistem oleh pengguna.

Dengan demikian, penggunaan SUS sangat relevan untuk digunakan di penelitian ini karena mampu memberikan data kuantitatif yang mudah dianalisis guna menilai sejauh mana pengguna merasa nyaman, efisien, dan puas dalam menggunakan aplikasi Cek Galon. Hasil dari evaluasi SUS dapat menjadi dasar untuk melakukan perbaikan desain antarmuka dan meningkatkan kualitas pengalaman pengguna.

2.7 Activity Diagram

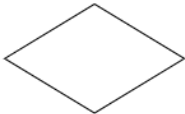
Activity Diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan alur kerja (*workflow*) atau alur aktivitas dari suatu sistem secara visual. Diagram ini menggambarkan urutan langkah aktivitas yang terjadi di

dalam sebuah proses atau sistem mulai dari awal hingga akhir, termasuk keputusan, percabangan, dan jalur paralel yang mungkin terjadi dalam eksekusi proses. Dengan demikian, *Activity Diagram* memudahkan tim pengembang serta pemangku kepentingan non-teknis untuk memahami keseluruhan proses bisnis atau logika sistem yang sedang dirancang (Nasution dkk., 2023).

Dalam konteks perancangan sistem, *Activity Diagram* juga berperan penting dalam mendukung tahap analisis dan desain karena diagram ini dapat menjelaskan tahapan proses, urutan aktivitas, dan keterkaitan antar langkah dengan jelas. Hal ini membantu dalam merinci proses yang telah diidentifikasi dari *Use Case Diagram* menjadi aliran aktivitas yang terstruktur. Misalnya, dalam penelitian yang menggunakan UML, *Activity Diagram* digunakan untuk memvisualisasikan aktivitas seperti “Login”, “Memilih Menu”, “Memproses Data”, dan “Logout” dalam sebuah sistem informasi, sehingga memungkinkan tim pengembang dan stakeholder memahami logika dan alur sistem sebelum melakukan implementasi (Narulita dkk., 2024).

Dengan notasi visual yang mudah dipahami, *Activity Diagram* efektif digunakan untuk memastikan bahwa alur proses sistem telah dipahami secara bersama oleh analis, pengembang, dan pengguna non-teknis, serta dapat membantu dalam mengidentifikasi skenario kegagalan atau jalur alternatif yang perlu ditangani dalam sistem (Syahrizal, 2025). Simbol *Activity Diagram* dapat dilihat pada Tabel 2.4.

Tabel 2.4 Simbol *Activity Diagram*

Simbol	Keterangan
	<i>Initial State</i> : Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	<i>Activity</i> : Aktivitas yang dilakukakn sistem, dimana aktivitas biasanya diawali dengan kata kerja.
	<i>Decesion</i> : Asosiasi percabangan dimana jika ada pilihan aktivitas.
	<i>Transition</i> : Menghubungkan satu komponen dengan komponen lainnya.

	<i>Join</i> : Penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	<i>Final State</i> : Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Dalam penelitian ini, *Activity Diagram* digunakan untuk memodelkan alur proses aplikasi Cek Galon, mulai dari *login* pengguna, pemesanan galon, verifikasi data, hingga pencatatan riwayat pemesanan. Dengan adanya diagram ini, alur interaksi sistem dapat divisualisasikan secara jelas sehingga mempermudah penentuan kebutuhan fungsional, memastikan tidak ada proses yang terlewat, dan meningkatkan ketepatan desain sebelum implementasi dilakukan.

2.8 Black box *Testing*

Pengujian *Blackbox* merupakan sebuah metode pengujian perangkat lunak yang berfokus pada pengujian fungsionalitas pada sistem tanpa perlu mengetahui struktur internal atau kode programnya. Metode pengujian ini bertujuan untuk menemukan kecacatan yang ada pada fungsionalitas sistem, termasuk pada kesalahan akses basis data, atau antarmuka pada sistem. Dalam pengujian ini, penguji hanya berfokus pada input dan output pada sistem tanpa melihat bagaimana proses internalnya bekerja (Ijudin & Saifudin, 2020).

Pada penelitian ini, pengujian *Blackbox* digunakan sebagai metode untuk melakukan pengujian sistem, yang bertujuan untuk memastikan bahwa sistem yang diuji dapat berfungsi secara optimal dan sesuai dengan kebutuhan pengguna.

2.9 Kajian Pustaka

Pada subbab ini akan dilakukan analisis dari penelitian-penelitian terdahulu yang relevan dengan topik penelitian ini, yang bertujuan untuk melihat perbandingan antara masing-masing metode yang telah dilakukan oleh peneliti terdahulu. Berikut adalah tabel rincian dari masing-masing metode yang telah digunakan oleh peneliti terdahulu, yang dapat dilihat pada Tabel 2.5 dibawah ini.

Tabel 2.5 Analisis Penelitian Terdahulu dan Kebaruan Penelitian

No	Nama Peneliti & Tahun	Judul Penelitian	Fitur Utama	Keterbatasan Penelitian Terdahulu	Gap dan Kebaruan Penelitian
1	(Fiki dkk., 2021)	Aplikasi Pemesanan Air Galon Online Dengan Fitur Tracking Posisi Pengantar Galon Berbasis Android.	Menyediakan aplikasi pemesanan galon berbasis Android dengan fitur pelacakan posisi pengantar galon.	Penelitian berfokus pada pelacakan pengantar, tetapi belum menyediakan mekanisme pembayaran online, verifikasi pembayaran, pelunasan kekurangan pembayaran, dan bukti foto saat pesanan selesai.	Penelitian ini menambahkan logika transaksi yang lebih kompleks melalui pembayaran QRIS, status pembayaran kurang, opsi pelunasan ulang, serta pengelolaan status pesanan secara real-time.
2	(Ratnawati dkk., 2022)	Aplikasi Pembelian Air Minum Isi Ulang Berbasis Android.	Mengembangkan aplikasi Android untuk pembelian air minum isi ulang, pemilihan jenis air, serta membantu kurir dalam proses pengantaran melalui lokasi konsumen.	Sistem masih menggunakan pendekatan transaksi dasar dan belum membahas mekanisme pembayaran digital yang fleksibel, verifikasi pembayaran, pembatalan otomatis (auto-cancel), maupun bukti pengantaran berbasis foto.	Penelitian ini mengembangkan aplikasi Android yang tidak hanya mendukung pemesanan galon, tetapi juga menyediakan metode pembayaran COD dan QRIS, validasi status pembayaran, serta bukti penyelesaian pesanan melalui foto pengantaran.

No	Nama Peneliti & Tahun	Judul Penelitian	Fitur Utama	Keterbatasan Penelitian Terdahulu	Gap dan Kebaruan Penelitian
3	(Tampubolon & Maslan, 2022)	Perancangan Aplikasi E-Galon Berbasis Android	Merancang aplikasi E-Galon berbasis Android untuk membantu transaksi pemesanan air galon secara online.	Pembayaran dan transaksi masih bersifat umum atau linier, yaitu pesanan diproses tanpa mekanisme khusus untuk kasus pembayaran kurang atau pembayaran yang belum dikonfirmasi.	Penelitian ini menawarkan mekanisme pembayaran yang lebih adaptif, yaitu pembeli dapat tetap melanjutkan pesanan ketika nominal pembayaran QRIS kurang melalui opsi pelunasan ulang atau konversi sisa pembayaran menjadi COD.
4	(Devansyah dkk., 2025)	Rancang Bangun Sistem Informasi Pemesanan Air Minum Isi Ulang Berbasis Framework Laravel.	Mengembangkan sistem berbasis web menggunakan Laravel dengan fitur pemesanan, pencatatan pelanggan, notifikasi status pesanan, serta dukungan pembayaran digital seperti QRIS dan e-wallet.	Sistem telah mendukung pembayaran digital, tetapi belum menjelaskan mekanisme penanganan pembayaran kurang, auto-cancel berbasis validasi status pembayaran, serta bukti pengantaran sebagai syarat penyelesaian pesanan.	Penelitian ini memperluas konsep pembayaran digital dengan fitur split payment pada QRIS, smart auto-cancel selama 15 menit, serta validasi status transaksi sebelum pembatalan otomatis dilakukan.
5	(Firdaus dkk., 2025)	Rancang Bangun Aplikasi Pemesanan	Mengembangkan aplikasi Android dengan integrasi Firebase dan	Sistem telah memanfaatkan Firebase dan layanan lokasi,	Penelitian ini menggabungkan Firebase, QRIS, COD, notifikasi,

No	Nama Peneliti & Tahun	Judul Penelitian	Fitur Utama	Keterbatasan Penelitian Terdahulu	Gap dan Kebaruan Penelitian
		Air Isi Ulang Berbasis Android dengan Integrasi Firebase dan Location Based Service di Depot Bumi Qta.	<i>Location Based Service</i> (LBS) untuk mendukung pemesanan air isi ulang serta pelacakan lokasi.	tetapi belum menonjolkan pengelolaan kasus pembayaran QRIS kurang, pilihan sisa bayar COD, dan bukti foto pengantaran yang dapat dilihat pembeli setelah pesanan selesai.	auto-cancel, dan bukti foto pengantaran sehingga proses pemesanan tidak hanya real-time tetapi juga lebih aman, fleksibel, dan terdokumentasi.

Berdasarkan Tabel 2.5, dapat diketahui bahwa penelitian mengenai sistem pemesanan air galon atau air minum isi ulang telah banyak dilakukan oleh peneliti sebelumnya, baik dalam bentuk aplikasi berbasis Android maupun berbasis web. Penelitian yang dilakukan oleh Ratnawati dkk. (2019), Fiki dkk. (2021), Tampubolon & Maslan (2022), Devansyah dkk. (2025), serta Firdaus dkk. (2025) menunjukkan bahwa digitalisasi layanan depot air minum dapat membantu meningkatkan efektivitas proses pemesanan, pengelolaan data pelanggan, pencatatan transaksi, pelacakan lokasi pengantaran, hingga penerapan pembayaran digital. Berbagai penelitian tersebut memberikan kontribusi yang signifikan dalam mendukung transformasi layanan pemesanan air minum menjadi lebih cepat, praktis, dan terorganisir.

Meskipun demikian, berdasarkan hasil analisis terhadap penelitian terdahulu, masih terdapat beberapa keterbatasan yang belum banyak dibahas. Sebagian besar penelitian berfokus pada proses pemesanan, pelacakan pengantaran, dan pembayaran digital secara umum, namun belum menyediakan mekanisme penanganan transaksi yang lebih adaptif. Beberapa penelitian telah mendukung pembayaran digital, tetapi belum menjelaskan proses yang dilakukan apabila terjadi kekurangan pembayaran, keterlambatan konfirmasi pesanan dari penjual, maupun pengelolaan status transaksi yang memerlukan tindak lanjut sebelum pesanan dapat diselesaikan. Selain itu, sebagian penelitian juga belum menyediakan mekanisme validasi penyelesaian pengantaran yang dapat dijadikan bukti bahwa pesanan benar-benar telah diterima oleh pelanggan.

Berdasarkan gap tersebut, penelitian ini mengembangkan aplikasi pemesanan air galon berbasis Android yang dilengkapi dengan fitur-fitur yang berfokus pada penguatan proses transaksi dan validasi layanan. Aplikasi menyediakan metode pembayaran menggunakan QRIS dan Cash on Delivery (COD). Selain itu, ketika pembayaran QRIS yang dilakukan pelanggan memiliki nominal yang kurang dari total tagihan, sistem memberikan fleksibilitas kepada pelanggan untuk melakukan pelunasan kembali melalui QRIS atau melanjutkan sisa pembayaran menggunakan metode COD. Mekanisme ini dirancang untuk mengakomodasi kondisi transaksi yang tidak sepenuhnya dapat ditangani oleh sistem pembayaran digital pada penelitian terdahulu.

Selanjutnya, aplikasi menerapkan fitur smart auto-cancel pada pesanan yang menggunakan metode pembayaran QRIS. Sistem akan memberikan batas waktu pembayaran selama 15 menit, namun proses pembatalan otomatis dilakukan dengan mempertimbangkan status transaksi yang sedang berlangsung. Dengan demikian, pesanan yang telah diverifikasi atau berada pada tahap pembayaran kurang tidak langsung dibatalkan oleh sistem. Selain itu, penelitian ini juga menyediakan fitur bukti pengantaran berupa foto yang diunggah oleh penjual sebagai dokumentasi bahwa pesanan telah berhasil dikirim kepada pelanggan. Bukti pengantaran tersebut dapat digunakan sebagai bentuk validasi penyelesaian layanan sehingga status pesanan selesai tidak hanya bergantung pada perubahan status secara manual.

Dengan demikian, kebaruan penelitian ini tidak terletak pada implementasi pemesanan air galon secara online, karena konsep tersebut telah banyak diterapkan pada penelitian sebelumnya. Kebaruan penelitian ini terletak pada penerapan mekanisme transaksi yang lebih adaptif melalui penanganan pembayaran QRIS dengan nominal kurang, opsi pelunasan menggunakan QRIS maupun COD, penerapan smart auto-cancel berbasis status transaksi, serta penyediaan bukti foto pengantaran sebagai validasi penyelesaian layanan. Fitur-fitur tersebut diharapkan dapat meningkatkan fleksibilitas pembayaran, keamanan transaksi, serta keandalan proses penyelesaian pesanan dalam aplikasi pemesanan air galon berbasis Android.

2.10 Waterfall

Metode *Waterfall* merupakan salah satu model pengembangan perangkat lunak yang bersifat berurutan, satu per satu, dimana setiap langkah harus diselesaikan sebelum beralih ke langkah berikutnya. Model ini disebut *waterfall* karena alurnya mengalir ke bawah seperti air terjun, dimana setiap tahapan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahapan berikutnya. Tahapan utama dalam metode *waterfall* ini meliputi analisis kebutuhan,

desain sistem, implementasi atau pemrograman, pengujian sistem, dan pemeliharaan (Widodo dkk., 2024). Metode ini banyak digunakan karena prosesnya yang terstruktur, mudah dipahami, dan cocok untuk proyek yang memiliki kebutuhan sistem yang jelas sejak awal.

Metode *waterfall* banyak digunakan dalam penelitian dan pengembangan aplikasi berbasis *website* maupun *mobile*. Pada penelitian (Pricillia & Zulfachmi, 2021) menjelaskan bahwa metode *waterfall* sangat cocok digunakan pada proyek dengan kebutuhan yang jelas dan stabil karena setiap tahapan memiliki dokumentasi yang lengkap sehingga memudahkan pengembang maupun pengguna dalam memahami proses pengembangannya. Pada penelitian (Hamdi & Fadli, 2025) juga menggunakan metode *waterfall*, dan menunjukkan bahwa metode *waterfall* mampu mempercepat proses pengembangan aplikasi UMKM karena struktur dari tahapannya dapat membantu tim fokus dalam menyelesaikan satu proses sebelum lanjut ke proses berikutnya.

Pada penelitian ini metode *waterfall* relevan digunakan karena mampu memberikan alur pengembangan yang sistematis dan terkontrol, sehingga proses perancangan hingga implementasi aplikasi Cek Galon dapat dilakukan dengan lebih rapi, terarah, dan terdokumentasi dengan baik.

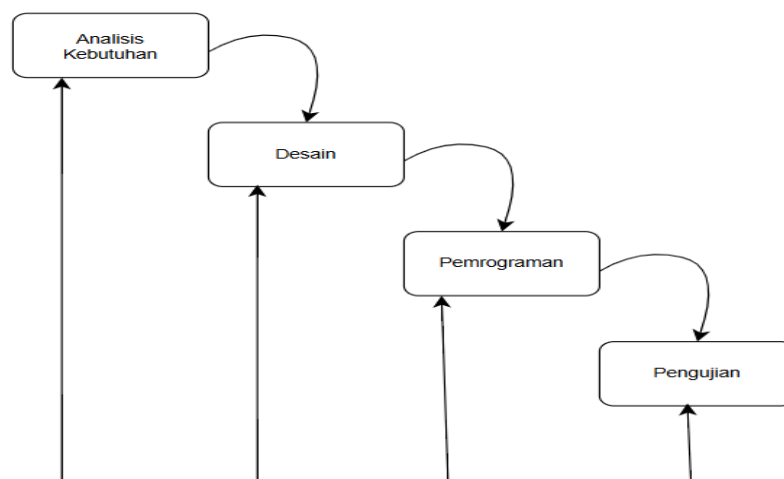
BAB III

METODE PENELITIAN

Aplikasi pemesanan galon merupakan sebuah sistem aplikasi yang bertujuan untuk membantu proses pemesanan dan pengelolaan layanan air galon secara lebih praktis dan terintegrasi. Aplikasi ini dirancang untuk mengatasi kendala pada proses pemesanan manual yang masih sering terjadi di lapangan, seperti pelanggan harus datang langsung ke depot/toko atau melakukan pemesanan melalui telepon/WhatsApp yang berpotensi menimbulkan kesalahan informasi, keterlambatan layanan, serta pencatatan pesanan yang tidak terdokumentasi dengan baik.

Melalui aplikasi ini, pelanggan dapat melakukan pemesanan air galon secara digital, sementara pihak depot dapat mengelola data pesanan dan transaksi secara lebih tertib. Informasi pemesanan yang masuk akan tersimpan di dalam sistem sehingga memudahkan proses pelayanan serta membantu pengelolaan riwayat pesanan.

Aplikasi dikembangkan dengan menggunakan metode penelitian Waterfall yang bersifat sistematis dan berurutan. Metode ini terdiri dari beberapa tahap, yaitu analisis kebutuhan, desain, pemrograman, dan pengujian. Berikut adalah gambar fase yang ada pada metode penelitian Waterfall.



Gambar 3.1 Metode *Waterfall*

(Pressman, 2015)

3.1 Analisis Kebutuhan

Analisis kebutuhan dalam pengembangan aplikasi dilakukan untuk menentukan kebutuhan fungsional dan non-fungsional yang perlu diimplementasikan ke dalam sistem. Data kebutuhan diperoleh melalui wawancara langsung dengan pemilik depot dan pembeli air galon di Depot Sueger Water yang beralamat di Jalan Pamungkas, Umbulmartani, Ngemplak, Sleman. Wawancara ini bertujuan untuk menggali kebutuhan pengguna secara langsung serta memperoleh gambaran mengenai permasalahan yang terjadi pada proses pemesanan air galon yang masih dilakukan secara manual. Melalui proses tersebut, peneliti memperoleh informasi mengenai kebutuhan utama sistem, seperti kemudahan dalam melakukan pemesanan galon, pengelolaan data pesanan, penyediaan informasi status pesanan, penyimpanan riwayat transaksi, serta pengaturan data pendukung layanan seperti lokasi pengantaran dan data jenis galon.



Gambar 3.2 Lokasi Depot Sueger Water

3.1.1 Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian ini adalah metode wawancara. Wawancara merupakan salah satu metode pengumpulan data yang dilakukan melalui interaksi langsung antara peneliti dan responden dalam bentuk tanya jawab. Teknik ini memungkinkan peneliti memperoleh informasi secara lebih mendalam mengenai kondisi nyata di lapangan, termasuk kebiasaan pengguna, kendala yang dihadapi, serta kebutuhan yang

diharapkan dari sistem yang akan dikembangkan. Melalui wawancara, peneliti dapat memahami permasalahan yang terjadi pada proses pemesanan air galon dan menjadikannya sebagai dasar dalam menentukan kebutuhan sistem yang sesuai. Proses ini membantu peneliti memperoleh data yang lebih rinci karena adanya komunikasi langsung antara kedua belah pihak (Trivaika & Senubekti, 2022).

A. Data Hasil Wawancara kepada Pemilik Usaha Galon Depot

Wawancara dilakukan kepada pemilik usaha Depot Sueger Water sebagai pihak yang terlibat langsung dalam proses penerimaan, pengelolaan, pembayaran, dan pengantaran pesanan air galon. Pertanyaan yang diajukan difokuskan pada alur pemesanan yang berjalan saat ini, kendala operasional yang sering dihadapi, serta kebutuhan sistem yang diharapkan dapat mendukung kegiatan pelayanan depot secara lebih terstruktur dan efisien.

1. Bagaimana alur pemesanan air galon yang berjalan saat ini (mulai dari pelanggan memesan sampai pesanan selesai diantar)?

Pemilik depot menjelaskan bahwa proses pemesanan yang berjalan saat ini masih dilakukan secara manual, baik melalui WhatsApp maupun pelanggan yang datang langsung ke depot. Untuk pesanan antar, penjual terlebih dahulu menanyakan nama pelanggan, alamat atau lokasi pengantaran, jenis galon, dan jumlah pesanan. Setelah informasi tersebut diperoleh, pesanan diproses dan selanjutnya diantarkan ke alamat pelanggan sampai selesai.

2. Melalui media apa saja pelanggan biasanya memesan (datang langsung/telepon/WhatsApp), dan mana yang paling sering digunakan?

Media pemesanan yang digunakan oleh pelanggan meliputi WhatsApp, telepon, dan datang langsung ke depot. Namun, media yang paling sering digunakan adalah WhatsApp karena dinilai lebih praktis, cepat, dan memudahkan pelanggan dalam menyampaikan detail pesanan.

3. Informasi apa saja yang wajib dicatat saat ada pesanan masuk?

Informasi yang dicatat oleh penjual saat menerima pesanan meliputi nama pemesan, alamat atau lokasi pengantaran, jenis galon yang dipesan, serta jumlah galon yang dibutuhkan.

4. Kendala apa yang paling sering terjadi dalam proses pemesanan dan pengantaran, serta apa penyebab utamanya?

Kendala yang paling sering terjadi adalah ketidakjelasan alamat atau lokasi pelanggan, sehingga menyulitkan penjual dalam menemukan tempat pengantaran. Hal ini umumnya disebabkan oleh kurangnya informasi atau patokan lokasi yang diberikan oleh pelanggan.

5. Bagaimana cara penjual mengelola pesanan yang masuk dan memberi tahu status pesanan kepada pelanggan?

Penjual mengelola pesanan berdasarkan urutan pesanan yang masuk, baik melalui WhatsApp maupun pelanggan yang datang langsung. Informasi pelanggan biasanya tersimpan dalam riwayat percakapan WhatsApp, seperti nama, nomor, dan alamat. Konfirmasi status pesanan dilakukan melalui komunikasi langsung dengan pelanggan melalui WhatsApp. Namun, belum terdapat sistem khusus untuk mengelola dan menampilkan status pesanan secara terstruktur karena masih dilakukan secara manual.

6. Jika dibuat aplikasi pemesanan air galon, apakah hal tersebut akan membantu dalam kegiatan operasional depot?

Pemilik depot menyatakan bahwa aplikasi pemesanan air galon akan sangat membantu dalam kegiatan operasional, terutama jika dilengkapi dengan fitur pengelolaan dan pencatatan pesanan secara terstruktur. Hal ini dapat meminimalkan risiko pesanan terlewat atau tertumpuk.

7. Apakah pernah ada pesanan yang terlewat atau tertumpuk? Apa penyebabnya?

Pemilik depot menjelaskan bahwa pesanan yang terlewat atau tertumpuk jarang terjadi. Namun, kondisi tersebut dapat terjadi ketika jumlah pesanan meningkat secara bersamaan. Penyebab utamanya adalah banyaknya pesanan yang masuk dalam waktu yang hampir bersamaan sehingga menyulitkan pengelolaan secara manual.

B. Data Hasil Wawancara kepada Pembeli Galon

Wawancara dilakukan kepada empat orang pembeli galon dengan latar belakang pengguna rumah tangga. Adapun pertanyaan yang diajukan kepada setiap responden bersifat sama untuk memperoleh sudut pandang yang konsisten terkait kebutuhan dan permasalahan dalam pemesanan galon air.

a. Responden 1: Fauzan

1. Biasanya Anda memesan air galon lewat apa?

Responden biasanya memesan air galon melalui WhatsApp karena dianggap mudah dan tidak ribet.

2. Bagaimana alur pemesanan yang dilakukan?

Responden melakukan pemesanan dengan mengirim pesan melalui WhatsApp, menentukan jenis galon (seperti Aqua atau Le Minerale), kemudian membagikan lokasi. Setelah itu, galon akan diantar ke rumah dan dapat ditunggu atau ditinggalkan di depan rumah.

3. Informasi apa yang biasanya disampaikan ke penjual?

Informasi yang diberikan berupa lokasi melalui fitur share location, serta jenis merek galon yang diinginkan.

4. Kendala yang sering dialami?

Kendala utama yang sering dialami adalah lambatnya respon dari penjual.

5. Seberapa penting mengetahui status pesanan?

Mengetahui status pesanan dinilai sangat penting agar pengguna dapat memperkirakan waktu kedatangan dan merasa lebih tenang karena mendapatkan informasi yang jelas.

b. Responden 2: Rafiq

1. Biasanya Anda memesan air galon lewat apa?

Responden memesan melalui WhatsApp karena tidak perlu datang langsung ke depot.

2. Bagaimana alur pemesanan yang dilakukan?

Responden memesan melalui WhatsApp, kemudian meninggalkan uang di dekat galon. Setelah pulang kerja, galon biasanya sudah diganti oleh penjual.

3. Informasi apa yang biasanya disampaikan ke penjual?

Informasi yang diberikan meliputi nama, alamat lengkap, serta patokan lokasi tempat tinggal.

4. Kendala yang sering dialami?

Kendala yang sering terjadi adalah lambatnya respon dari penjual, sehingga responden tidak mengetahui apakah pesanan sudah diproses atau belum.

5. Seberapa penting mengetahui status pesanan?

Mengetahui status pesanan sangat penting agar pengguna mengetahui apakah pesanan sudah diproses atau belum, mengingat sering terjadi keterlambatan respon dari penjual.

c. Responden 3: Syahputra

1. Biasanya Anda memesan air galon lewat apa?

Responden biasanya memesan dengan datang langsung ke depot air isi ulang karena sudah menjadi pelanggan tetap.

2. Bagaimana alur pemesanan yang dilakukan?

Responden membawa galon kosong ke depot, memilih jenis air, kemudian menunggu pengisian dan membawa kembali galon tersebut ke rumah.

3. Informasi apa yang biasanya disampaikan ke penjual?

Informasi yang disampaikan hanya berupa jenis air yang diinginkan karena pemesanan dilakukan secara langsung di lokasi.

4. Kendala yang sering dialami?

Kendala utama adalah kurangnya informasi terkait jam operasional depot (buka atau tutup), sehingga responden sering harus datang langsung untuk memastikan.

5. Seberapa penting mengetahui status pesanan?

Mengetahui status pesanan sangat penting, terutama untuk mengetahui kondisi depot, harga, serta kemudahan dalam melakukan pemesanan tanpa harus datang langsung.

d. Responden 4: Arya

1. Biasanya Anda memesan air galon lewat apa?

Responden memesan melalui WhatsApp karena tidak memiliki waktu untuk datang langsung ke depot.

2. Bagaimana alur pemesanan yang dilakukan?

Responden menanyakan ketersediaan galon melalui WhatsApp, kemudian melakukan pembayaran melalui transfer bank atau QRIS setelah mendapatkan konfirmasi. Selanjutnya, galon akan diantar ke alamat yang telah diberikan.

3. Informasi apa yang biasanya disampaikan ke penjual?

Informasi yang diberikan meliputi alamat tempat tinggal, jenis galon, jumlah pesanan, serta konfirmasi harga.

4. Kendala yang sering dialami?

Kendala utama yang sering dialami adalah tidak adanya informasi yang jelas mengenai estimasi waktu pengantaran, sehingga pengguna sering mengalami ketidakpastian.

5. Seberapa penting mengetahui status pesanan?

Mengetahui status pesanan dinilai penting agar pengguna dapat memantau proses pemesanan dan mengetahui jika terjadi keterlambatan atau perubahan kondisi layanan.

Berdasarkan Berdasarkan hasil wawancara dengan pemilik depot dan empat orang pembeli, dapat disimpulkan bahwa proses pemesanan air galon yang berjalan saat ini masih didominasi oleh komunikasi manual melalui WhatsApp, telepon, atau pemesanan langsung di depot. Meskipun cara tersebut dinilai cukup praktis oleh pengguna, dalam pelaksanaannya masih ditemukan beberapa kendala, seperti lambatnya respon penjual, ketidakjelasan lokasi pengantaran, belum tersedianya informasi status pesanan secara terstruktur, serta belum adanya pencatatan pesanan dan data pelanggan yang tersimpan secara rapi.

Dari sisi pembeli, kebutuhan utama yang muncul meliputi kemudahan dalam melakukan pemesanan air galon, ketersediaan informasi lokasi pengantaran yang jelas, pilihan metode pembayaran yang sesuai, serta adanya riwayat transaksi dan status pesanan yang dapat dipantau secara langsung. Sementara itu, dari sisi penjual, kebutuhan utama yang ditemukan adalah tersedianya sistem yang dapat membantu menerima dan mengelola pesanan, menyimpan data pelanggan, mencatat transaksi, serta mengatur data pendukung layanan seperti jenis galon, harga, dan stok secara lebih terstruktur.

Berdasarkan temuan tersebut, aplikasi perlu dikembangkan untuk mendukung proses pemesanan air galon secara digital, pencatatan dan riwayat transaksi/pesanan, serta pengelolaan pesanan oleh pihak depot. Dengan adanya fitur-fitur tersebut, sistem diharapkan mampu membantu pengguna dalam melakukan pemesanan secara lebih mudah, sekaligus membantu pihak depot dalam meningkatkan ketertiban pengelolaan layanan.

3.1.2 Analisis Kebutuhan Fungsional

Kebutuhan Kebutuhan fungsional merupakan kebutuhan utama yang harus dimiliki oleh aplikasi agar dapat mendukung proses pemesanan air galon secara digital sesuai dengan kebutuhan pengguna. Berdasarkan hasil wawancara dengan pemilik depot dan pembeli, kebutuhan fungsional aplikasi adalah sebagai berikut.

1. Pemesanan air galon oleh pelanggan

Aplikasi harus menyediakan fasilitas bagi pelanggan untuk melakukan pemesanan air galon secara *digital*. Pemesanan mencakup pemilihan jenis galon, jumlah pesanan, alamat pengantaran, catatan lokasi, serta metode pembayaran agar proses pemesanan lebih mudah, tersimpan dengan baik, dan dapat digunakan kembali pada pemesanan berikutnya. Selain itu, aplikasi juga perlu menampilkan informasi akun pengguna yang berkaitan dengan proses pemesanan, seperti email, nomor handphone, dan data lokasi pengantaran.

2. Pencatatan dan riwayat transaksi/pesanan

Aplikasi harus menyimpan data pesanan dan transaksi secara otomatis agar pelanggan dapat melihat riwayat pemesanan, memantau status pesanan, serta mengetahui perkembangan proses layanan. Dari sisi depot, penyimpanan data ini juga membantu pemantauan aktivitas pesanan secara lebih rapi dan terstruktur.

3. Pengelolaan pesanan oleh pihak depot

Sistem harus mampu membantu pihak depot dalam menerima dan mengelola pesanan yang masuk, termasuk melihat detail pesanan, melakukan konfirmasi, memverifikasi pembayaran, memperbarui status pesanan, serta mengelola data pendukung layanan seperti jenis galon, harga, dan stok yang tersedia. Dengan adanya fitur ini, alur pelayanan menjadi lebih teratur, terkontrol, dan memudahkan operasional depot.

3.1.3 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional mengacu pada kualitas aplikasi secara keseluruhan, seperti kinerja aplikasi, keamanan aplikasi. Kebutuhan ini berperan penting dalam memastikan pengalaman pengguna yang memuaskan, dengan fokus pada bagaimana aplikasi bekerja secara efisien dan dapat diandalkan, serta memenuhi standar yang diinginkan oleh pengguna. Berikut adalah kebutuhan non-fungsional.

1. Antarmuka pengguna yang intuitif

Antarmuka dari aplikasi harus dirancang sederhana dan mudah untuk digunakan oleh pengguna, dengan informasi yang jelas dan mudah diakses. Tampilan antarmuka yang ramah pengguna akan mempermudah penggunaan aplikasi dan mengurangi risiko kesalahan saat menggunakan aplikasi.

2. Tingkat penerimaan pengguna (*User Acceptance*)

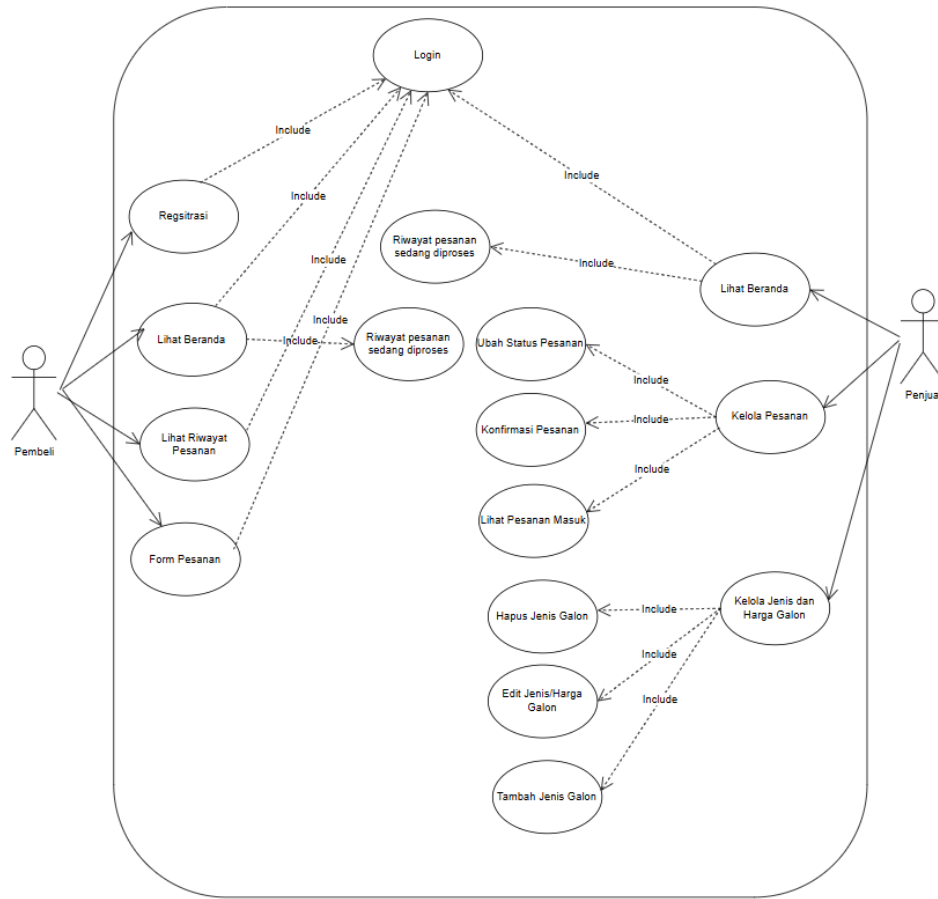
Aplikasi harus dapat diterima dengan baik oleh pengguna, yang ditunjukkan melalui tingkat kepuasan dan kesediaan pengguna untuk menggunakan aplikasi secara berkelanjutan.

3.2 Desain Aplikasi

Bagian ini menjelaskan rancangan antarmuka pengguna dan struktur *database* untuk aplikasi pemesanan galon. Pada tahapan ini berperan sebagai langkah awal yang penting dalam proses pengembangan aplikasi. Rancangan ini bertujuan untuk memvisualisasikan interaksi pengguna dengan antarmuka serta bagaimana data akan diorganisir dan disimpan di dalam sistem. Penjelasan berikut menguraikan desain antarmuka pengguna yang dirancang untuk memastikan kemudahan navigasi, serta struktur *database* yang dirancang untuk mendukung pengelolaan data yang efisien dan terstruktur.

3.2.1 *Use case diagram*

Pada tahap ini dilakukan pembuatan diagram *use case* yang bertujuan untuk menggambarkan masing-masing hak akses kepada setiap pengguna. Berikut adalah diagram *use case* yang digunakan dalam penelitian ini.



Gambar 3.3 *Use case diagram* Aplikasi

Berdasarkan *Use Case Diagram* pada gambar di atas, sistem Aplikasi Pemesanan Air Galon melibatkan dua aktor, yaitu Pembeli dan Penjual. Pembeli dapat menggunakan sistem untuk melakukan proses pemesanan air galon secara digital. Sebelum menggunakan fitur utama, Pembeli dapat melakukan Registrasi (bagi pengguna baru) dan Login agar dapat mengakses layanan dalam aplikasi. Setelah berhasil masuk, Pembeli akan diarahkan pada halaman Beranda, yang menampilkan informasi riwayat pesanan yang sedang diproses sebagai ringkasan pesanan yang masih berjalan. Selain itu, Pembeli juga dapat mengakses menu Lihat Riwayat Pesanan untuk melihat daftar pesanan yang pernah dilakukan dan untuk melihat detail pesanan untuk mengetahui informasi pesanan secara lebih lengkap. Pembeli juga dapat melakukan pemesanan melalui Form Pesanan sebagai proses utama dalam melakukan order air galon.

Di sisi lain, aktor Penjual juga melakukan Login untuk mengakses sistem. Penjual akan diarahkan pada halaman Beranda yang menampilkan ringkasan riwayat pesanan yang sedang diproses, sehingga penjual dapat memantau pesanan aktif yang perlu ditangani. Penjual

memiliki fungsi utama Kelola Pesanan, yang mencakup beberapa aktivitas (ditunjukkan dengan relasi include) yaitu Lihat Pesanan Masuk, Konfirmasi Pesanan, serta Ubah Status Pesanan (misalnya pesanan diterima, diproses, diantar, hingga selesai) agar alur layanan lebih tertib dan terkontrol. Selain pengelolaan pesanan, Penjual juga dapat mengatur data produk melalui use case Kelola Jenis dan Harga Galon, yang mencakup proses Tambah Jenis Galon, Edit Jenis/Harga Galon, dan Hapus Jenis Galon (ditunjukkan dengan relasi include). Hal ini bertujuan agar data jenis galon dan harga yang tersedia pada aplikasi selalu sesuai dengan kondisi di depot.

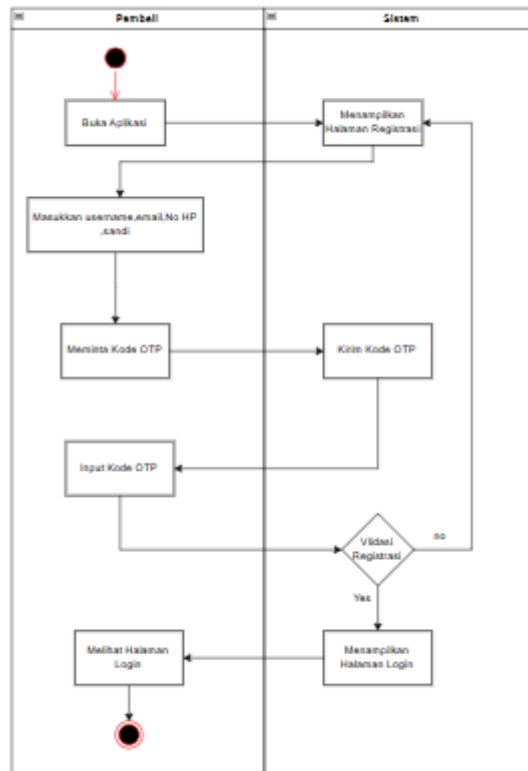
Dengan demikian, *Use Case Diagram* tersebut menggambarkan interaksi Pembeli dan Penjual dalam sistem pemesanan air galon, mulai dari proses akses sistem (registrasi/login), pemantauan pesanan melalui beranda, pembuatan pesanan oleh pembeli, hingga pengelolaan pesanan serta pengelolaan jenis dan harga galon oleh penjual.

3.2.2 *Activity Diagram*

Activity diagram merupakan salah satu diagram dalam *unified modeling language* (UML) yang digunakan untuk menggambarkan alur aktivitas atau proses kerja dalam suatu sistem secara berurutan. Diagram ini menunjukkan langkah-langkah yang dilakukan oleh pengguna dan sistem, mulai dari kondisi awal hingga kondisi akhir, termasuk percabangan keputusan yang mungkin terjadi selama proses berlangsung. *Activity diagram* digunakan untuk membantu memahami bagaimana suatu fitur atau fungsi dalam sistem berjalan secara logis dan sistematis. Dengan adanya diagram ini, pengembang dan pihak terkait dapat melihat alur proses secara jelas, sehingga memudahkan dalam tahap perancangan, implementasi, maupun pengujian sistem.

A. *Activity Diagram* Registrasi Pembeli

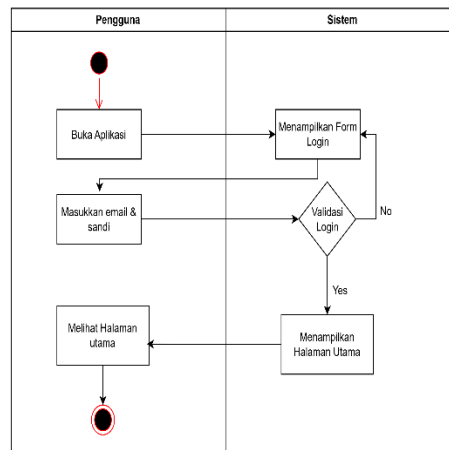
Activity diagram registrasi pembeli menggambarkan alur ketika pengguna baru membuat akun pada aplikasi. Proses dimulai saat pengguna membuka halaman registrasi, kemudian mengisi data yang dibutuhkan seperti username, email, nomor handphone, lokasi, dan password. Setelah itu, sistem melakukan validasi terhadap data yang dimasukkan. Jika data valid, sistem akan menyimpan data pengguna ke dalam database dan mengarahkan pengguna ke halaman login. Namun, apabila terdapat data yang tidak sesuai atau belum lengkap, sistem akan menampilkan pesan kesalahan dan pengguna diminta memperbaiki data terlebih dahulu. Tampilan gambar *Activity diagram* registrasi pembeli dapat dilihat pada Gambar 3.4.



Gambar 3.4 Activity Diagram Registrasi Pembeli

B. Activity Diagram Login Pengguna

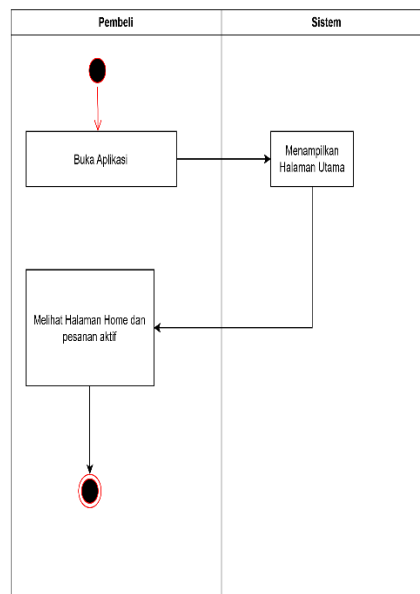
Activity diagram login pengguna menggambarkan alur autentikasi saat pengguna masuk ke dalam aplikasi. Proses dimulai ketika pengguna membuka aplikasi dan sistem menampilkan halaman login. Selanjutnya, pengguna memasukkan email dan password yang telah terdaftar, kemudian sistem memverifikasi kecocokan data tersebut. Apabila data benar, sistem akan mengarahkan pengguna ke halaman utama sesuai dengan perannya, yaitu sebagai pembeli atau penjual. Sebaliknya, jika data login tidak sesuai, sistem akan menampilkan informasi kesalahan dan pengguna diminta memasukkan ulang data dengan benar. Tampilan gambar *Activity diagram* login pengguna dapat dilihat pada Gambar 3.5.



Gambar 3.5 Activity Diagram Login Pengguna

C. Activity Diagram Home Pembeli

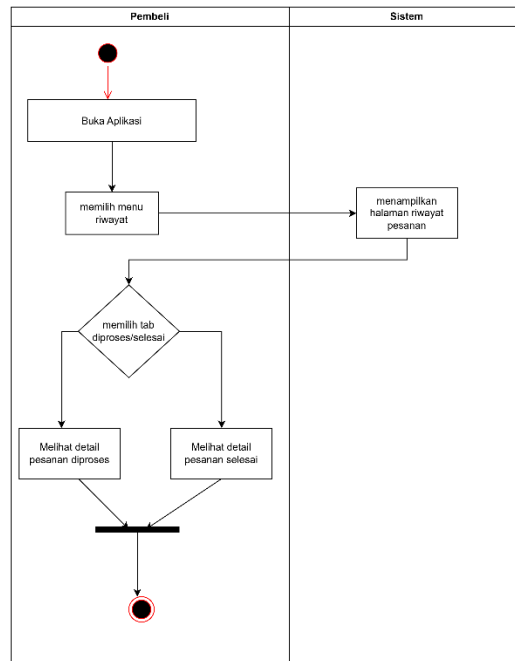
Activity diagram home pembeli menggambarkan alur ketika pembeli berhasil masuk ke aplikasi dan mengakses halaman utama. Proses dimulai setelah sistem memverifikasi akun pembeli, lalu sistem menampilkan halaman home yang berisi sapaan pengguna dan daftar pesanan aktif yang masih berlangsung. Pada halaman ini, pembeli dapat melihat ringkasan informasi pesanan yang sedang diproses, serta melanjutkan ke fitur lain seperti membuat pesanan baru, melihat riwayat pesanan, atau membuka halaman akun. Diagram ini menunjukkan bahwa halaman home berfungsi sebagai pusat aktivitas utama bagi pembeli dalam menggunakan aplikasi. Tampilan gambar *Activity diagram home* pembeli dapat dilihat pada Gambar 3.6



Gambar 3.6 *Activity Diagram* Home Pembeli

D. *Activity Diagram* Riwayat Pesanan Pembeli

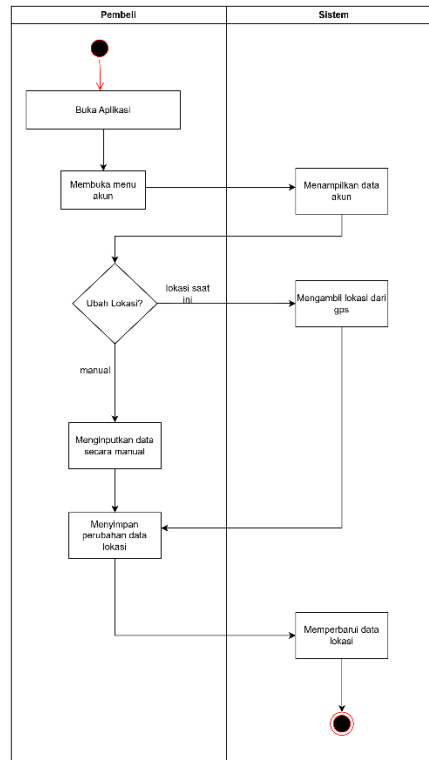
Activity diagram riwayat pesanan pembeli menggambarkan alur saat pembeli membuka halaman riwayat untuk melihat pesanan yang pernah dilakukan. Proses dimulai ketika pembeli memilih menu riwayat, kemudian sistem menampilkan daftar pesanan yang dikelompokkan berdasarkan status, yaitu diproses dan selesai. Pembeli dapat melihat informasi detail dari setiap pesanan, seperti jenis galon, jumlah, metode pembayaran, total harga, alamat pengantaran, dan status pesanan. Melalui alur ini, sistem membantu pembeli memantau perkembangan pesanan secara lebih jelas dan terstruktur. Tampilan gambar *Activity diagram* riwayat pesanan pembeli dapat dilihat pada Gambar 3.7.



Gambar 3.7 Activity Diagram Riwayat Pesanan Pembeli

E. Activity Diagram Pengaturan Akun Pembeli dan Ubah Lokasi

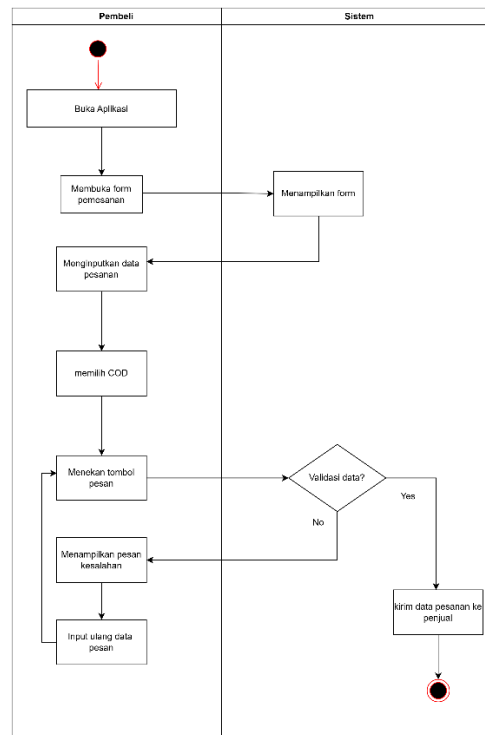
Activity diagram akun pembeli dan ubah lokasi menggambarkan alur ketika pembeli membuka halaman akun untuk melihat informasi profil dan memperbarui lokasi pengantaran. Proses dimulai saat pengguna memilih menu akun, kemudian sistem menampilkan data seperti username, email, nomor handphone, lokasi, dan peran pengguna. Selanjutnya, pembeli dapat memilih untuk mengubah lokasi secara manual atau menggunakan lokasi saat ini. Setelah perubahan disimpan, sistem akan memperbarui data lokasi pada akun pengguna sehingga dapat digunakan kembali pada proses pemesanan berikutnya. Tampilan gambar *Activity diagram* pengaturan akun pembeli dan ubah lokasi dapat dilihat pada Gambar 3.8.



Gambar 3.8 *Activity Diagram* Pengaturan Akun Pembeli dan Ubah Lokasi

F. *Activity Diagram* Pemesanan Galon dengan Metode COD

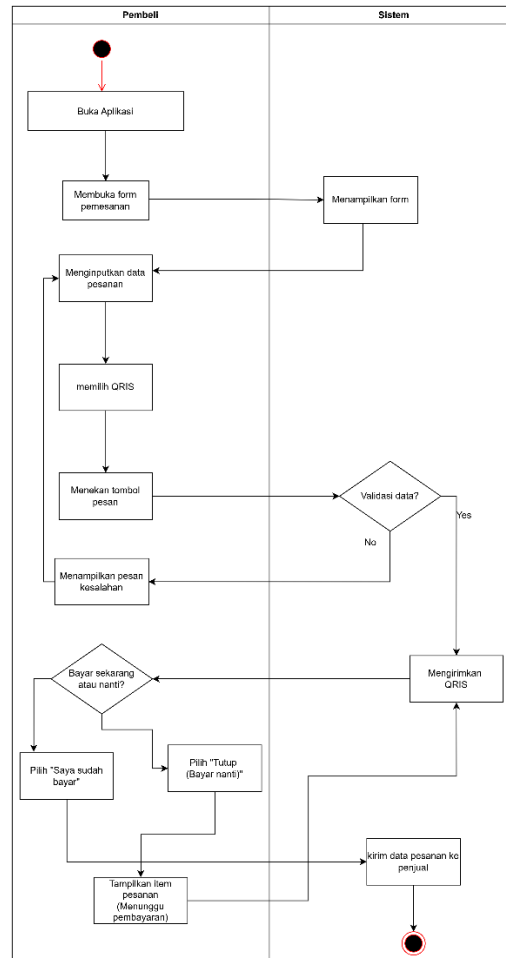
Activity diagram pemesanan galon dengan metode COD menggambarkan alur saat pembeli melakukan pemesanan dan memilih pembayaran di tempat. Proses dimulai ketika pembeli membuka form pemesanan, lalu mengisi data yang diperlukan seperti jenis galon, jumlah pesanan, lokasi pengantaran, catatan, dan metode pembayaran COD. Sistem kemudian melakukan validasi data yang dimasukkan. Jika data valid, pesanan akan disimpan ke dalam sistem dan diteruskan ke sisi penjual untuk diproses lebih lanjut. Diagram ini menunjukkan bahwa pada metode COD, pesanan dapat langsung masuk ke alur layanan karena pembayaran dilakukan saat barang diantar. Tampilan gambar *Activity diagram* pemesanan galon dengan metode COD dapat dilihat pada Gambar 3.9.



Gambar 3.9 *Activity Diagram* Pemesanan Galon dengan Metode COD

G. *Activity Diagram* Pemesanan Galon dengan Metode QRIS

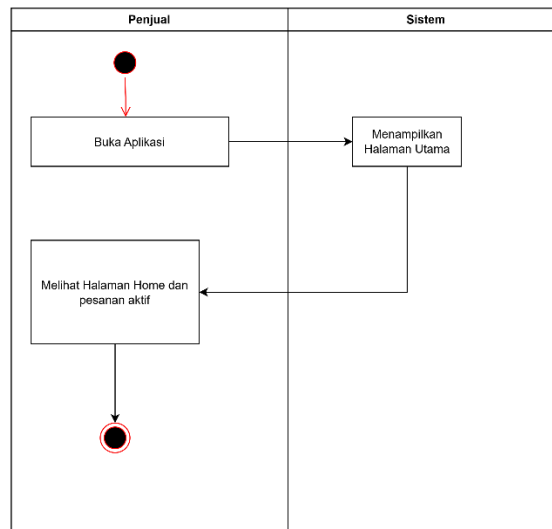
Activity diagram pemesanan galon dengan metode QRIS menggambarkan alur saat pembeli melakukan pemesanan dengan pembayaran digital. Proses dimulai ketika pembeli mengisi form pemesanan dan memilih metode pembayaran QRIS, kemudian sistem menampilkan dialog pembayaran berupa kode QR dan nominal yang harus dibayarkan. Pada tahap ini, pembeli dapat memilih menandai bahwa pembayaran sudah dilakukan atau menutup dialog untuk membayar nanti. Jika pembayaran ditandai sudah dilakukan, sistem menyimpan pesanan dengan status menunggu konfirmasi penjual. Jika pembayaran belum diselesaikan, sistem tetap menyimpan pesanan dengan status menunggu pembayaran. Tampilan gambar *Activity diagram* pemesanan galon dengan metode QRIS dapat dilihat pada Gambar 3.10.



Gambar 3.10 Activity Diagram Pemesanan Galon dengan Metode QRIS

H. Activity Diagram Home Penjual

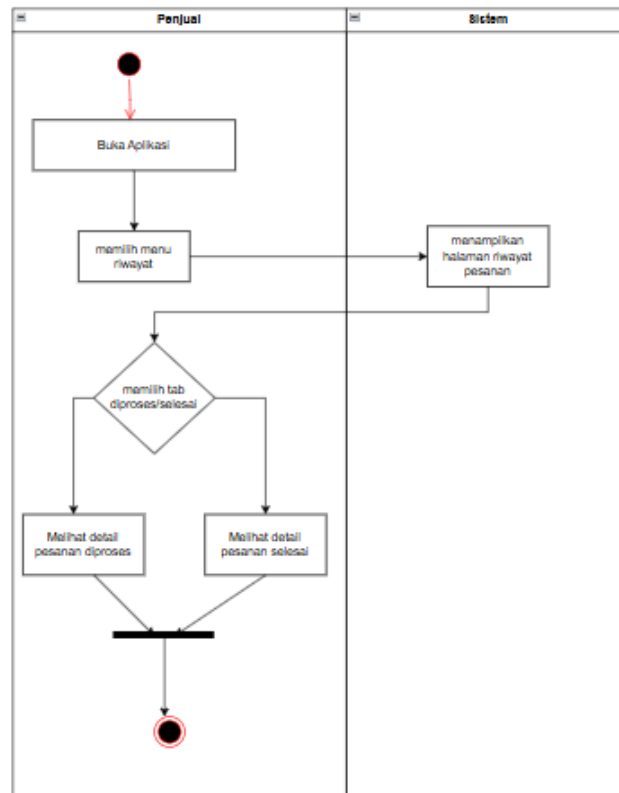
Activity diagram home penjual menggambarkan alur ketika penjual berhasil login dan mengakses halaman utama aplikasi. Setelah proses login berhasil, sistem menampilkan halaman home penjual yang berisi daftar pesanan aktif yang perlu ditindaklanjuti. Informasi yang ditampilkan meliputi nama pemesan, jenis galon, jumlah pesanan, metode pembayaran, alamat, total harga, dan status pesanan. Dari halaman ini, penjual dapat melanjutkan ke proses pengelolaan pesanan atau melihat riwayat transaksi yang telah tercatat. Dengan demikian, halaman home penjual menjadi pusat pemantauan pesanan yang sedang berjalan. Tampilan gambar *Activity diagram* home penjual dapat dilihat pada Gambar 3.11.



Gambar 3.11 *Activity Diagram* Home Penjual

I. *Activity Diagram* Riwayat Pesanan Penjual

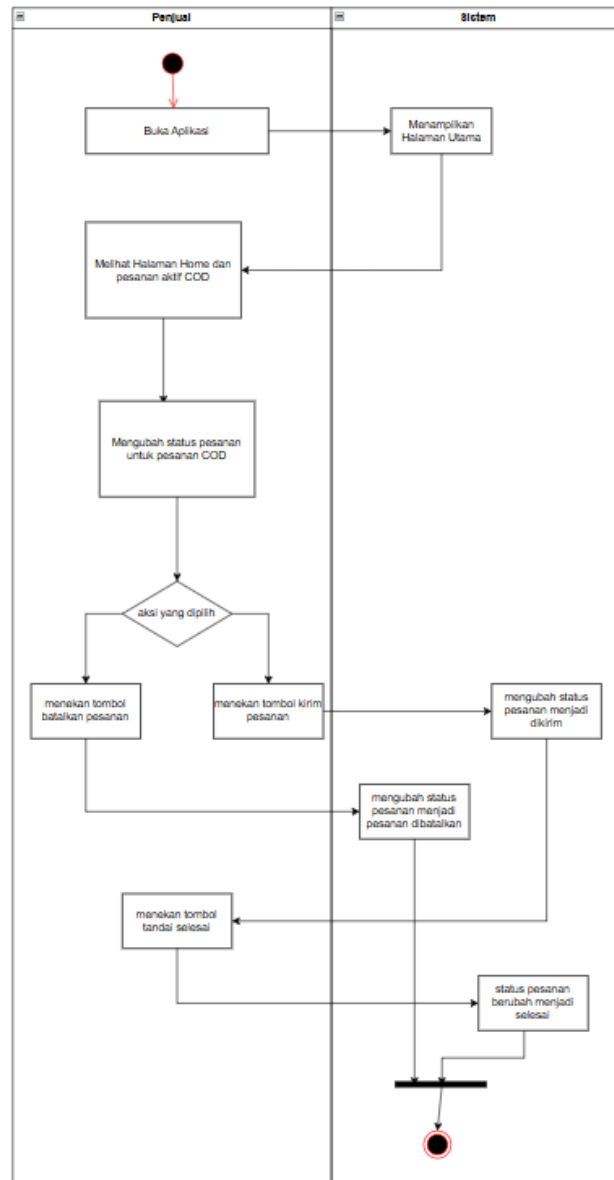
Activity diagram riwayat pesanan penjual menggambarkan alur saat penjual membuka halaman riwayat untuk melihat seluruh data pesanan yang pernah masuk ke dalam sistem. Proses dimulai ketika penjual memilih menu riwayat, lalu sistem menampilkan daftar pesanan berdasarkan status diproses dan selesai. Pada halaman ini, penjual dapat melihat detail pesanan seperti nama pembeli, jenis galon, jumlah, alamat, metode pembayaran, total harga, dan waktu pemesanan. Diagram ini menunjukkan bahwa halaman riwayat memudahkan penjual dalam memantau transaksi yang sedang berjalan maupun yang telah berakhir. Tampilan gambar *Activity diagram* riwayat pesanan penjual dapat dilihat pada Gambar 3.12.



Gambar 3.12 *Activity Diagram* Riwayat Pesanan Penjual

J. Activity Diagram Kelola Status Pesanan COD

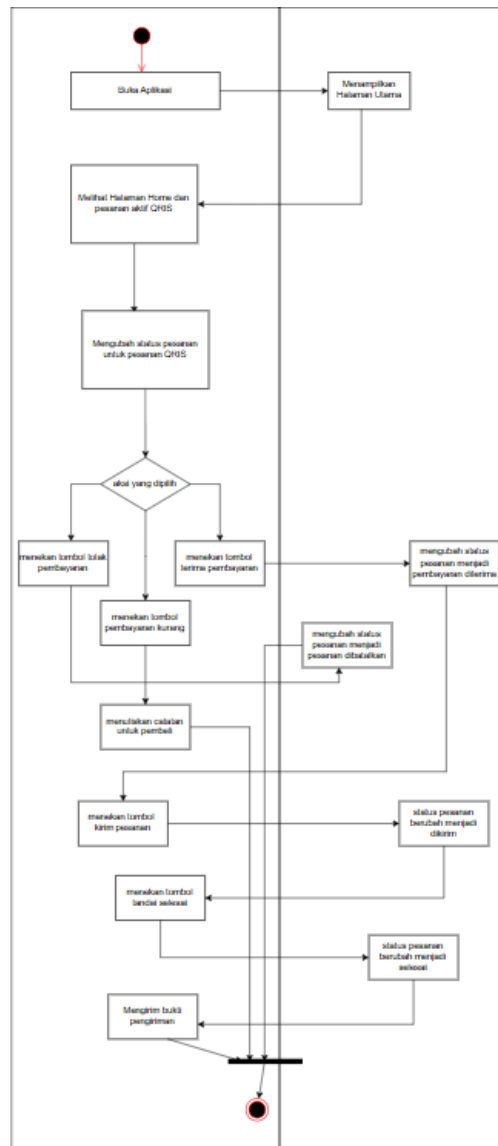
Activity diagram kelola status pesanan COD menggambarkan alur perubahan status pesanan oleh penjual pada transaksi dengan metode pembayaran COD. Proses dimulai ketika penjual memilih salah satu pesanan COD yang masuk, kemudian sistem menampilkan opsi pengelolaan status pesanan. Penjual dapat mengubah status pesanan menjadi dikirim, selesai, atau batal sesuai kondisi layanan di lapangan. Setelah status diperbarui, sistem menyimpan perubahan tersebut dan menampilkan informasi terbaru pada sisi penjual maupun pembeli. Dengan alur ini, perkembangan pesanan COD dapat dipantau secara lebih jelas oleh kedua pihak. Tampilan gambar *Activity diagram* kelola status pesanan COD dapat dilihat pada Gambar 3.13.



Gambar 3.13 *Activity Diagram* Kelola Status Pesanan COD

K. *Activity Diagram* Kelola Status Pesanan QRIS

Activity diagram kelola status pesanan QRIS menggambarkan alur verifikasi pembayaran dan perubahan status pesanan oleh penjual. Proses dimulai ketika pesanan QRIS masuk ke sistem dengan status menunggu konfirmasi penjual, lalu penjual membuka menu pengelolaan status pesanan. Pada tahap ini, penjual dapat memilih menerima pembayaran, menandai pembayaran kurang, atau menolak pembayaran. Jika pembayaran diterima, pesanan dapat dilanjutkan ke tahap dikirim dan selesai. Namun, jika pembayaran belum sesuai, sistem akan menyimpan catatan yang dapat dilihat pembeli sebagai informasi tindak lanjut pembayaran. Tampilan gambar *Activity diagram* kelola status pesanan QRIS dapat dilihat pada Gambar 3.14.

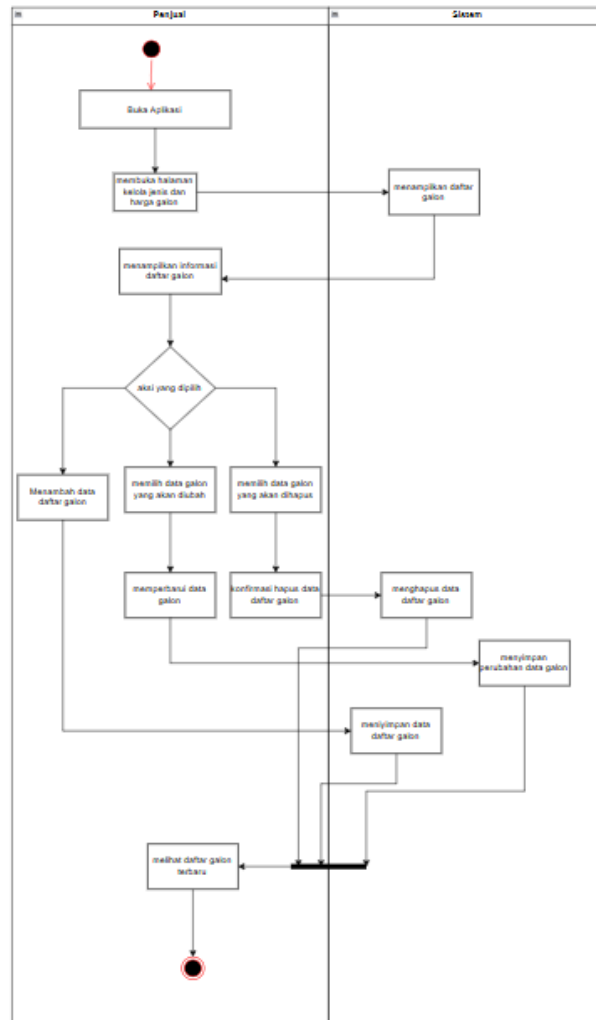


Gambar 3.14 *Activity Diagram* Kelola Status Pesanan QRIS

L. *Activity Diagram* Kelola Jenis dan Harga Galon

Activity diagram kelola jenis dan harga galon menggambarkan alur aktivitas penjual dalam mengatur data galon yang tersedia pada aplikasi. Proses dimulai ketika penjual membuka halaman pengaturan jenis dan harga galon, kemudian sistem menampilkan daftar data galon yang tersimpan. Pada halaman ini, penjual dapat memilih menambah data galon baru, mengedit data galon yang sudah ada, atau menghapus data galon tertentu. Jika penjual menambah atau mengedit data, sistem akan menampilkan form input dan menyimpan perubahan ke database setelah data dikonfirmasi. Jika penjual memilih hapus, sistem terlebih

dahulu menampilkan konfirmasi sebelum data dihapus dari sistem. Tampilan gambar *Activity diagram* kelola jenis dan harga galon dapat dilihat pada Gambar 3.15.



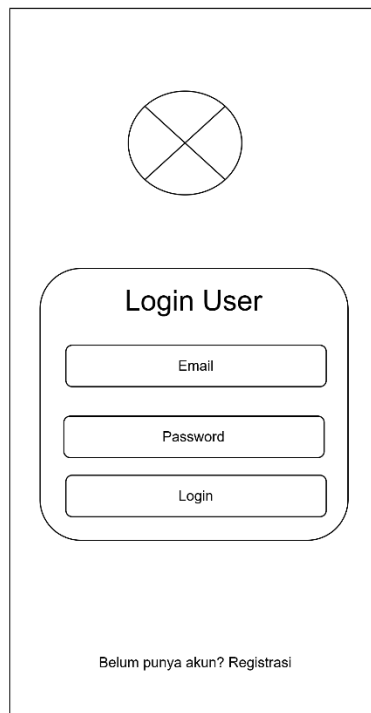
Gambar 3.15 *Activity Diagram* Kelola Jenis dan Harga Galon

3.2.3 Rancangan Antarmuka

Rancangan antarmuka merupakan sebuah proses perancangan elemen visual dan interaktif dalam sebuah aplikasi atau sistem untuk menciptakan pengalaman pengguna yang intuitif, efisien, dan mudah digunakan. Rancangan antarmuka yang baik memperhatikan kebutuhan pengguna, kemudahan dalam navigasi, serta konsistensi tampilan. Pendekatan yang berfokus pada pengguna, terbukti mampu meningkatkan kenyamanan dan efektivitas dalam penggunaan aplikasi secara keseluruhan (Ansori dkk., 2023). Berikut adalah rincian dari sketsa desain aplikasi.

A. Antarmuka untuk Penjual

a. Halaman Login Penjual



The sketch shows a vertical rectangular container. At the top center is a circle with an 'X' inside. Below this is a rounded rectangular box containing the text 'Login User'. Underneath the box are three stacked rectangular input fields labeled 'Email', 'Password', and 'Login'. At the bottom of the container, centered, is the text 'Belum punya akun? Registrasi'.

Gambar 3.16 Sketsa Desain tampilan Halaman Login

b. Halaman Home Penjual



Gambar 3.17 Sketsa Desain Tampilan Halaman *Home* Penjual

c. Dialog Ubah Password Penjual

CekGalon

Selamat datang,

Ubah Password x

Password Lama

Password Baru

Minimal 6 Karakter

Konfirmasi Password Baru

Batal Simpan

Home Riwayat Settings

Gambar 3.18 Tampilan Dialog Ubah *Password*

d. Halaman Riwayat Pesanan Penjual

CekGalon

Riwayat Pesanan

Diproses Selesai

07 Nov 2024 13.38 wib

Sedang Diproses

Nama Pemesan

Galon air RO

Metode : QRIS

Catatan: Pembayaran Diterima

JL. Candirejo

081111111

Qty 1 x Rp14.000,00 Total Rp14.000,00

Ubah Status Pesanan

Home Riwayat Akun

CekGalon

Riwayat Pesanan

Diproses Selesai

06 Nov 2024 17.10 wib

Selesai

Nama Pemesan

Galon air RO

Metode : QRIS

Catatan: Pembayaran Diterima

JL. Candirejo

081111111

Qty 1 x Rp14.000,00 Total Rp14.000,00

Lihat Bukti Pengiriman

Home Riwayat Akun

CekGalon

Riwayat Pesanan

Diproses Selesai

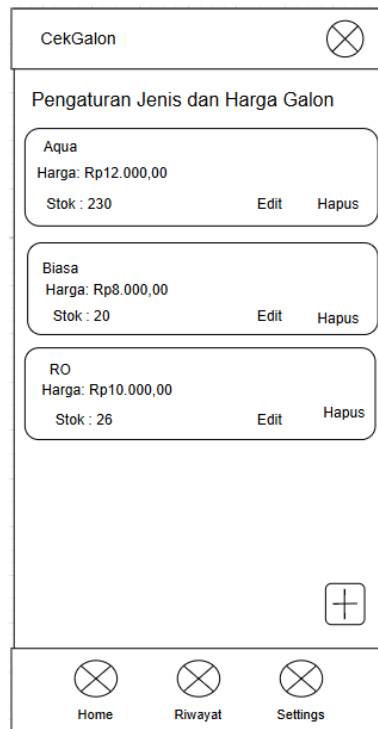
06 Nov 2024

Bukti Pengantaran

Home Riwayat Akun

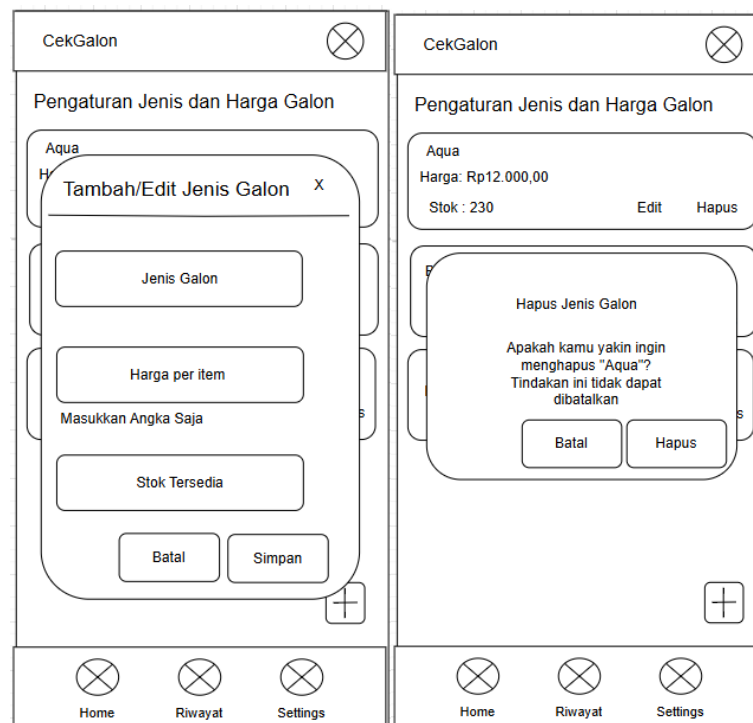
Gambar 3.19 Sketsa Desain Tampilan Halaman Riwayat Pesanan

e. Halaman Kelola Jenis dan Harga Galon



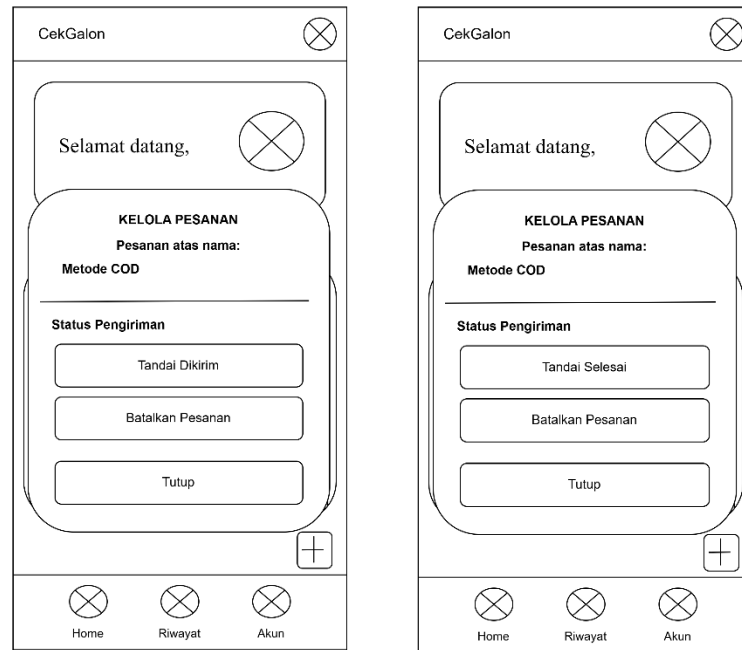
Gambar 3.20 Sketsa Tampilan Halaman Pengaturan Jenis & Harga

f. Dialog Tambah, Edit, dan Hapus Jenis Galon



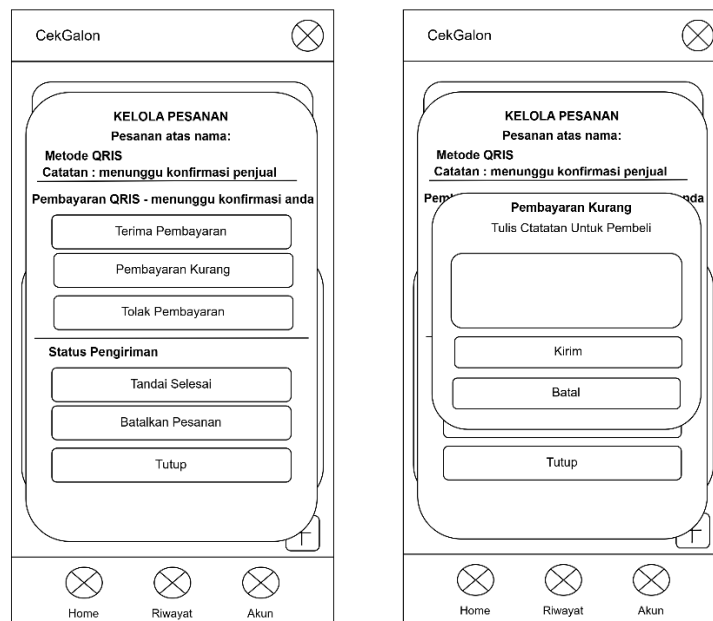
Gambar 3.21 Tampilan Dialog Tambah, Edit, dan Hapus Jenis Galon

g. Dialog Ubah Status Pesanan COD



Gambar 3.22 Tampilan Dialog Ubah Status Pesanan COD

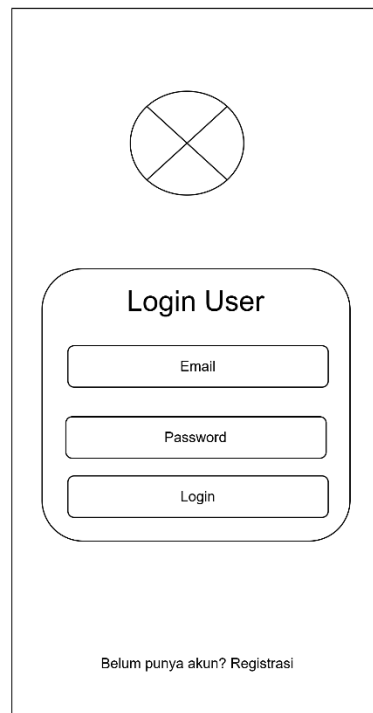
h. Dialog Verifikasi dan Catatan Status Pesanan QRIS



Gambar 3.23 Tampilan Dialog Verifikasi dan Catatan Status Pesanan QRIS

B. Halaman Antarmuka Untuk Pembeli

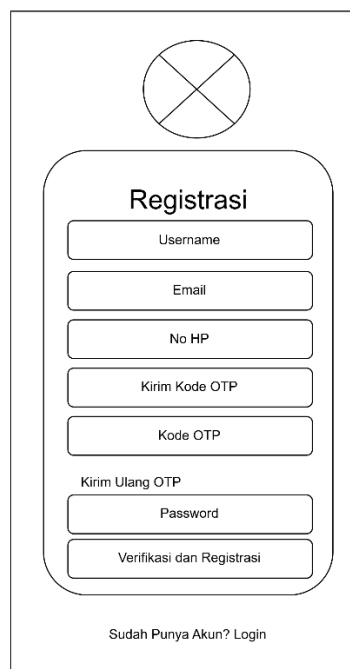
a. Halaman Login



A mobile app login screen. At the top is a circular logo with an 'X' inside. Below it is a rounded rectangle containing the title 'Login User'. Under the title are three input fields: 'Email', 'Password', and 'Login'. At the bottom of the screen is a link that says 'Belum punya akun? Registrasi'.

Gambar 3.24 Tampilan Halaman Login

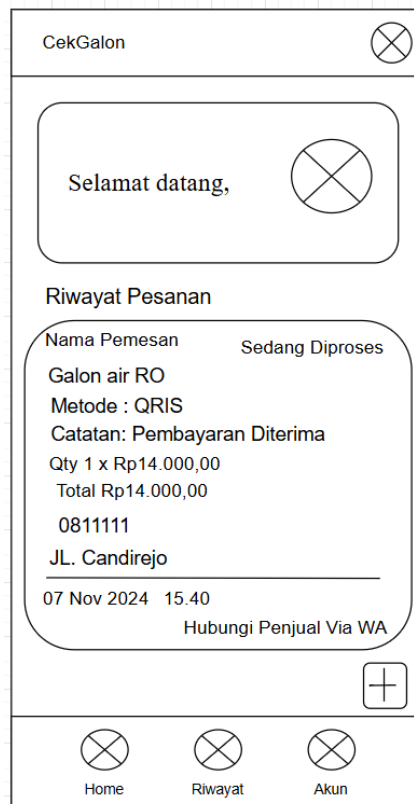
b. Halaman Registrasi



A mobile app registration screen. At the top is a circular logo with an 'X' inside. Below it is a rounded rectangle containing the title 'Registrasi'. Under the title are seven input fields: 'Username', 'Email', 'No HP', 'Kirim Kode OTP', 'Kode OTP', 'Kirim Ulang OTP', and 'Password'. At the bottom of the rounded rectangle is a button labeled 'Verifikasi dan Registrasi'. At the bottom of the screen is a link that says 'Sudah Punya Akun? Login'.

Gambar 3.25 Tampilan Halaman Registrasi

c. Halaman *Home*



Gambar 3.26 Sketsa Desain Tampilan Halaman Home

d. Dialog Pemesanan Galon

CekGalon

Selamat datang.

Pemesanan Galon

Pesan Galon X

Nama

Jenis Galon

Stok tersedia: 24
Harga per item: Rp14.000,00

Kuantitas

Total Harga

Lokasi

No HP

Metode Pembayaran

QRIS: Scan QRIS, masukkan nominal sesuai total

Pesan

Jika ada kendala silakan tekan ini

COD bayar saat galon diantar

Pesan

Jika ada kendala silakan tekan ini

Gambar 3.27 Sketsa Desain Tampilan Halaman Pesanan Galon

e. Dialog Pembayaran QRIS

CekGalon

Pembayaran QRIS

QR CODE

Cara Bayar:

1. Scan QRIS diatas
2. Masukkan nominal sesuai informasi pesanan/catatan penjual
3. Selesaikan Pembayaran
4. klik "Saya Sudah Bayar" agar penjual bisa cek

Saya Sudah Bayar

Tutup

Home Riwayat Akun

Gambar 3.28 Tampilan Dialog Pembayaran QRIS

f. Dialog Hasil Pesanan QRIS Status Menunggu Konfirmasi Penjual

CekGalon

Selamat datang,

Pemesanan Galon

Menunggu Konfirmasi Penjual

Pembayaranmu Sudah ditandai. Penjual akan memverifikasi pembayaran terlebih dahulu.

Atas nama :
Galon Aqua (Qty 1)
Total Rp14.000,00
Metode QRIS

Lokasi:
Kos Putra Candirejo

Catatan Lokasi: Kos Putra Candirejo

Status Pemabayaran:
Menunggu konfirmasi penjual(QRIS). Nominal
Rp14.000,00

Oke

QRIS: Scan QRIS, masukkan nominal sesuai total

Pesan

Jika ada kendala silakan tekan ini

Gambar 3.29 Tampilan Dialog Hasil Pesanan QRIS Status Menunggu Konfirmasi Penjual

g. Dialog Hasil Pesanan QRIS Status Menunggu Pembayaran

CekGalon

Selamat datang,

Pemesanan Galon

Menunggu Pembayaran

Pesanan sudah dibuat, tetapi belum dibayar.
Pesanan akan diproses setelah kamu menyelesaikan pembayaran QRIS.

Atas nama :
Galon Aqua (Qty 1)
Total Rp14.000,00
Metode QRIS

Lokasi:
Kos Putra Candirejo

Catatan Lokasi: Kos Putra Candirejo

Status Pemabayaran:
Menunggu konfirmasi penjual(QRIS). Nominal
Rp14.000,00

Oke

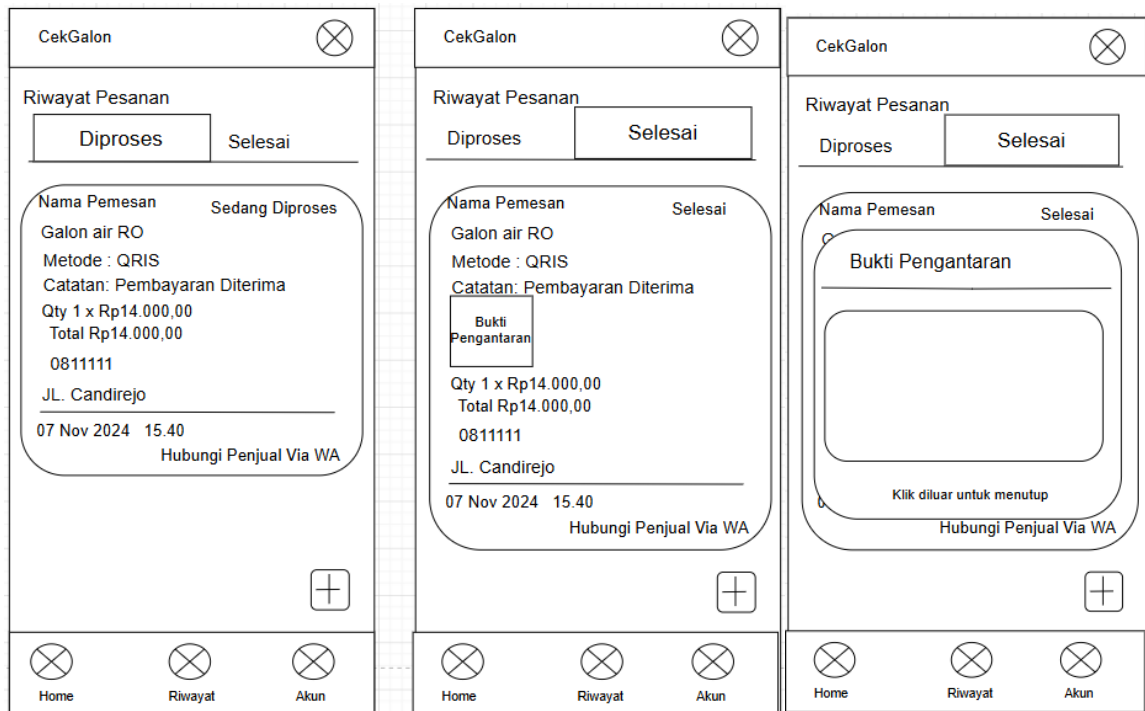
QRIS: Scan QRIS, masukkan nominal sesuai total

Pesan

Jika ada kendala silakan tekan ini

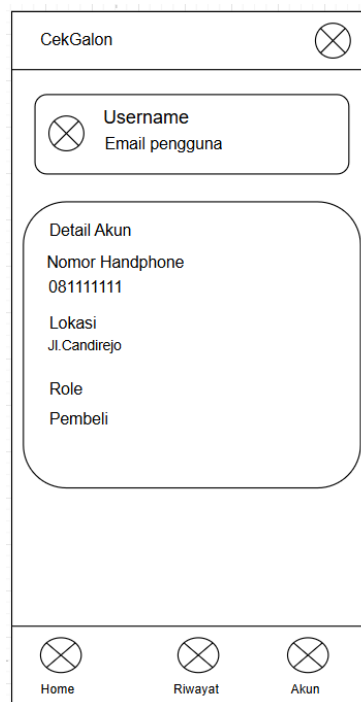
Gambar 3.30 Tampilan Dialog Hasil Pesanan QRIS Status Menunggu Pembayaran

h. Halaman Riwayat Pesanan



Gambar 3.31 Sketsa Desain Tampilan Halaman Riwayat Pesanan

i. Halaman Akun dan Pengaturan Lokasi



Gambar 3.32 Sketsa Tampilan Halaman Akun

j. Dialog Edit Lokasi



Gambar 3.33 Sketsa Tampilan Dialog Edit Lokasi

C. Desain Struktur *Database*

Desain struktur database merupakan perancangan susunan data yang digunakan untuk mendukung kebutuhan aplikasi agar data dapat tersimpan secara terorganisir, mudah diakses, dan sesuai dengan fungsi sistem. Pada aplikasi pemesanan galon, struktur data dirancang untuk mendukung proses autentikasi pengguna, pengelolaan data jenis galon, penyimpanan data pengguna, serta pencatatan transaksi pesanan. Dalam implementasinya, aplikasi ini memanfaatkan layanan Firebase, yaitu Firebase Authentication, Firebase Realtime Database, dan Cloud Firestore. Berikut ini adalah struktur database yang digunakan dalam aplikasi.

a. Database Akun

Database akun digunakan untuk mendukung proses autentikasi pengguna pada aplikasi. Pada bagian ini, data utama yang digunakan adalah email, password, dan nomor handphone sebagai identitas unik pengguna. Struktur database akun dapat dilihat pada Tabel 3.1.

Tabel 3.1 Struktur database akun

No	Nama Kolom	Tipe Data	Keterangan
1	email	String	Alamat email pengguna

2	password	String	Kata sandi pengguna
3	phone	String	Nomor Hp pengguna

b. Database Galon

Database galon digunakan untuk menyimpan data jenis galon yang tersedia pada aplikasi. Data ini meliputi nama jenis galon, harga, dan stok galon yang masih tersedia. Struktur database galon dapat dilihat pada Tabel 3. 2.

Tabel 3. 2 Struktur Desain Database Galon

```
{
  "galon_types": {
    "id_galon_types": {
      "nama": "String",
      "harga": "Int",
      "stok": "Int"
    }
  }
}
```

c. Database Users

Database users digunakan untuk menyimpan data pendukung pengguna yang melengkapi data akun. Data ini mencakup informasi seperti username, email, nomor handphone, lokasi, role, dan waktu pembuatan akun. Struktur database users dapat dilihat pada Tabel 3.3.

Tabel 3.3 Struktur Desain Database Users

```
{
  "users": {
    "id_users": {
      "catatanLokasi": "String",
      "createdAt": "Timestamp",
      "email": "String",
      "lokasi": "String",
      "noHp": "String",
      "phoneVerified": "Boolean",

```

```

    "role": "String",
    "uid": "String",
    "updatedAt": "Timestamp",
    "username": "String"
  }
}
}

```

d. Database *Order*

Database *order* digunakan untuk menyimpan seluruh data pesanan yang dilakukan oleh pembeli. Data yang disimpan mencakup identitas pesanan, data pembeli, jenis galon yang dipesan, jumlah pesanan, total harga, lokasi pengantaran, metode pembayaran, status pesanan, catatan tambahan, bukti pengantaran, dan waktu pembuatan pesanan. Struktur database order dapat dilihat pada Tabel 3. 4.

Tabel 3. 4 Struktur Desain Database *Order*

```

{
  "orders": {
    "id_orders": {
      "catatan": "String",
      "catatanLokasi": "String",
      "createdAt": "Timestamp",
      "id": "String",
      "jenis_galon": "String",
      "kuantitas": "Int",
      "lokasi": "String",
      "bukti_pengantaran": "null",
      "metode_pembayaran": "String",
      "nama": "String",
      "phone": "String",
      "status": "String",
      "total_harga": "Int",
      "uid": "String"
    }
  }
}

```

```

}
}

```

3.3 Pemrograman

Setelah tahap desain aplikasi selesai dilakukan, langkah berikutnya adalah pemrograman. Pada tahap ini, rancangan sistem yang telah dibuat kemudian diimplementasikan menjadi aplikasi pemesanan galon berbasis Android agar dapat digunakan sesuai dengan kebutuhan pengguna. Proses pemrograman dilakukan berdasarkan hasil analisis kebutuhan, rancangan antarmuka, activity diagram, use case diagram, serta struktur database yang telah disusun pada tahap sebelumnya. Bahasa pemrograman yang digunakan dalam pengembangan aplikasi ini adalah Kotlin karena memiliki sintaks yang ringkas, mendukung pengembangan aplikasi Android secara modern, dan sesuai digunakan dalam implementasi sistem yang terintegrasi dengan layanan Firebase (Nurlaila dkk., 2022).

3.4 Pengujian

Pengujian pada aplikasi dalam penelitian ini merupakan tahap penting sebelum aplikasi digunakan oleh pengguna. Pengujian dilakukan untuk mengurangi potensi kesalahan sistem. Pada penelitian ini digunakan dua metode pengujian, yaitu pengujian black box dan System Usability Scale (SUS). Pengujian black box berfokus pada antarmuka serta kesesuaian antara *input* dan *output* aplikasi yang bertujuan untuk memastikan setiap fungsi sistem berjalan sesuai dengan kebutuhan tanpa melihat proses program yang digunakan (Pratama & Dadaprawira, 2023).

Selain itu, dilakukan pula pengujian menggunakan System Usability Scale (SUS) untuk mengevaluasi tingkat kemudahan penggunaan aplikasi dari sudut pandang pengguna. Metode SUS digunakan untuk mengukur aspek *usability* seperti kemudahan memahami antarmuka, kenyamanan penggunaan, serta kepuasan pengguna secara keseluruhan berdasarkan kuesioner penilaian yang terdiri dari sepuluh pernyataan standar.

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini disajikan hasil pengembangan aplikasi pemesanan galon berdasarkan rancangan sistem yang telah dijelaskan pada Bab III. Tahap pengembangan dilakukan dengan mengacu pada metode Waterfall yang meliputi analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Hasil pengembangan mencakup implementasi aplikasi berbasis Android menggunakan bahasa pemrograman Kotlin serta integrasi Firebase sebagai layanan autentikasi dan penyimpanan data. Aplikasi yang dikembangkan dirancang untuk mendukung proses pemesanan air galon secara digital, pengelolaan pesanan oleh penjual, penyimpanan riwayat transaksi, serta pengelolaan data pendukung layanan seperti akun pengguna, lokasi pengantaran, jenis galon, harga, dan stok. Seluruh fitur yang diimplementasikan disesuaikan dengan kebutuhan pengguna yang diperoleh dari hasil wawancara dan analisis kebutuhan sistem.

4.1 Fitur Aplikasi

Bagian ini menjelaskan fitur-fitur utama yang berhasil dikembangkan pada aplikasi berdasarkan hasil analisis kebutuhan fungsional pada Bab III. Fitur-fitur tersebut dirancang untuk mendukung proses pemesanan air galon oleh pelanggan, pencatatan dan riwayat transaksi, serta pengelolaan pesanan oleh pihak depot. Secara umum, aplikasi pemesanan galon dikembangkan dengan tiga fitur utama, yaitu fitur pemesanan galon, fitur riwayat transaksi dan status pesanan, serta fitur pengelolaan pesanan oleh penjual.

4.1.1 Fitur Pemesanan Air Galon

Fitur pemesanan galon dikembangkan untuk memudahkan pelanggan dalam melakukan pemesanan air galon secara digital. Melalui fitur ini, pengguna dapat mengisi form pemesanan yang memuat data nama, jenis galon, harga, kuantitas, alamat pengantaran, catatan lokasi, serta metode pembayaran. Selain itu, sistem juga menampilkan informasi akun yang berkaitan dengan proses pemesanan, seperti email, nomor handphone, dan data lokasi pengantaran yang dapat digunakan kembali pada pemesanan berikutnya. Metode pembayaran yang disediakan dalam aplikasi meliputi COD dan QRIS.

4.1.2 Fitur Riwayat Transaksi dan Status Pesanan

Fitur riwayat transaksi dan status pesanan dikembangkan agar pengguna dapat melihat daftar pesanan yang pernah dilakukan beserta perkembangan proses layanannya. Pada sisi pembeli, riwayat pesanan ditampilkan ke dalam dua kelompok, yaitu tab diproses dan tab selesai. Tab diproses memuat pesanan dengan status menunggu pembayaran, diproses, dan dikirim, sedangkan tab selesai memuat pesanan dengan status selesai dan dibatalkan. Melalui fitur ini, pengguna dapat memantau perkembangan pesanan secara lebih jelas tanpa harus melakukan konfirmasi manual kepada penjual. Selain itu, untuk pesanan yang telah selesai, fitur ini juga menyediakan transparansi berupa pratinjau foto bukti pengantaran yang diunggah oleh penjual.

4.1.3 Fitur Pengelolaan Pesanan oleh Penjual

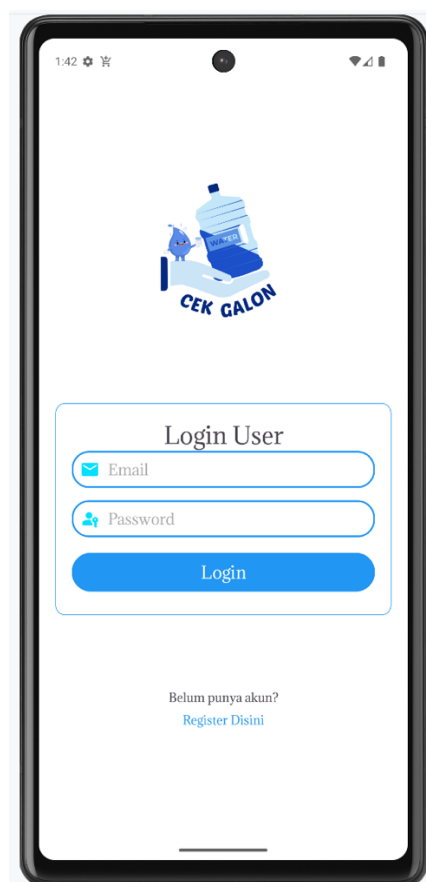
Fitur pengelolaan pesanan oleh penjual dikembangkan untuk membantu pihak depot dalam menerima, memverifikasi, dan mengelola pesanan yang masuk dari pembeli. Melalui fitur ini, penjual dapat melihat detail pesanan, melakukan konfirmasi pesanan, memverifikasi pembayaran, serta memperbarui status pesanan sesuai tahapan layanan. Penjual juga dapat mengambil dan mengunggah foto bukti pengantaran sebagai bentuk pertanggungjawaban penyelesaian pesanan. Di samping itu, fitur ini mengakomodasi fleksibilitas pembayaran melalui mekanisme pelunasan ganda (*split payment*) apabila terjadi kekurangan pada pembayaran non-tunai. Selain itu, penjual juga dapat mengelola data pendukung layanan berupa jenis galon, harga, dan stok yang tersedia, sehingga proses operasional depot dapat berjalan dengan lebih tertib dan terstruktur.

4.2 Implementasi Sistem

Berdasarkan kebutuhan sistem yang diperoleh dari hasil analisis dan wawancara, peneliti kemudian mengembangkan rancangan antarmuka dalam bentuk sketsa dan melanjutkannya ke tahap implementasi menggunakan Android Studio dengan bahasa pemrograman Kotlin. Tahap implementasi dilakukan dengan merealisasikan setiap fitur utama yang telah dirancang pada bab sebelumnya ke dalam bentuk tampilan antarmuka dan fungsi sistem yang dapat digunakan oleh pembeli maupun penjual.

4.2.1 Implementasi Halaman Login

Halaman login digunakan sebagai gerbang awal bagi pengguna untuk masuk ke dalam aplikasi. Pada halaman ini, pengguna diminta memasukkan email dan password yang telah terdaftar, kemudian menekan tombol login untuk melakukan proses autentikasi. Setelah data berhasil diverifikasi, sistem akan mengarahkan pengguna ke halaman utama sesuai dengan perannya, yaitu sebagai pembeli atau penjual. Selain itu, pada bagian bawah halaman juga disediakan tautan menuju halaman registrasi bagi pengguna yang belum memiliki akun. Implementasi halaman login dapat dilihat pada Gambar 4.1.

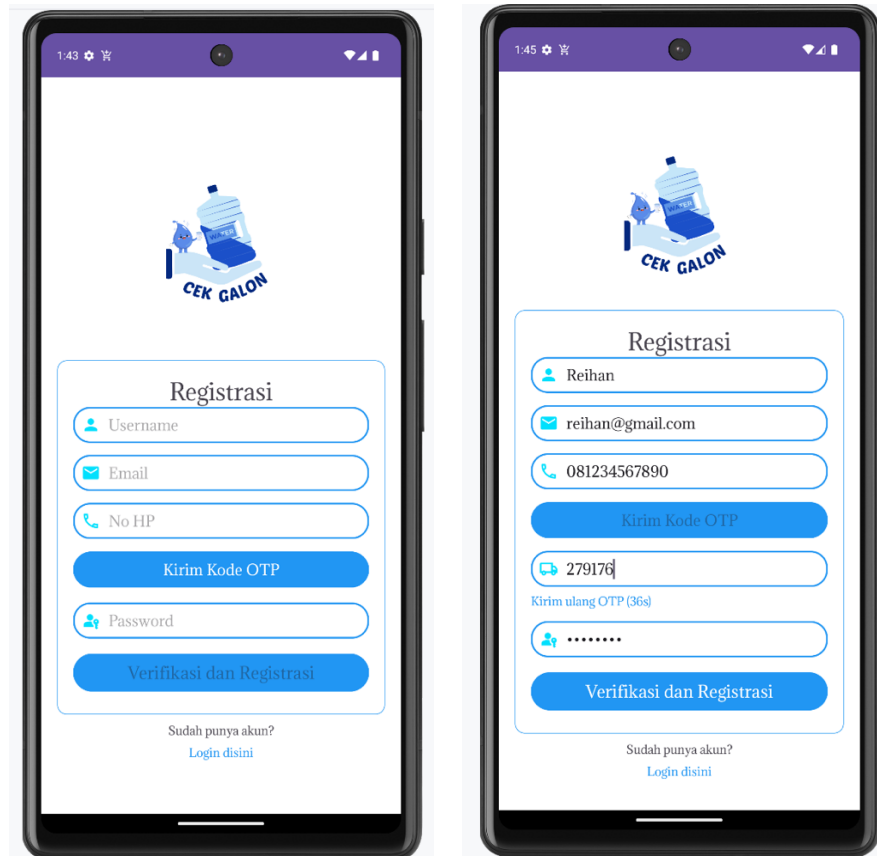


Gambar 4.1 Implementasi Halaman Login

4.2.2 Implementasi Halaman Registrasi

Halaman registrasi digunakan untuk pembuatan akun baru pada sisi pembeli. Pada halaman ini, pengguna diminta mengisi data berupa nama, email, nomor handphone, kode OTP, dan password. Proses registrasi diawali dengan pengisian nomor handphone dan pengiriman kode OTP sebagai langkah verifikasi, kemudian pengguna memasukkan kode yang diterima sebelum menyelesaikan pendaftaran akun. Setelah seluruh data dinyatakan valid,

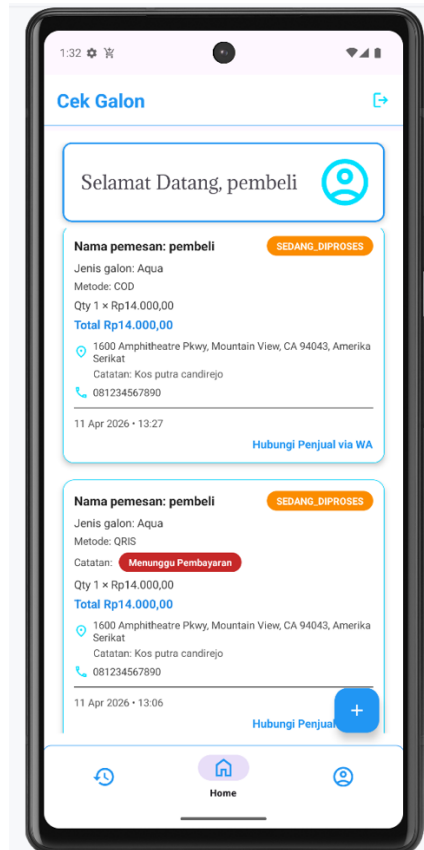
sistem akan menyimpan data pengguna dan mengarahkan pengguna untuk masuk melalui halaman login. Implementasi halaman registrasi dapat dilihat pada Gambar 4.2.



Gambar 4.2 Implementasi Halaman Registrasi

4.2.3 Implementasi Halaman Home Pembeli

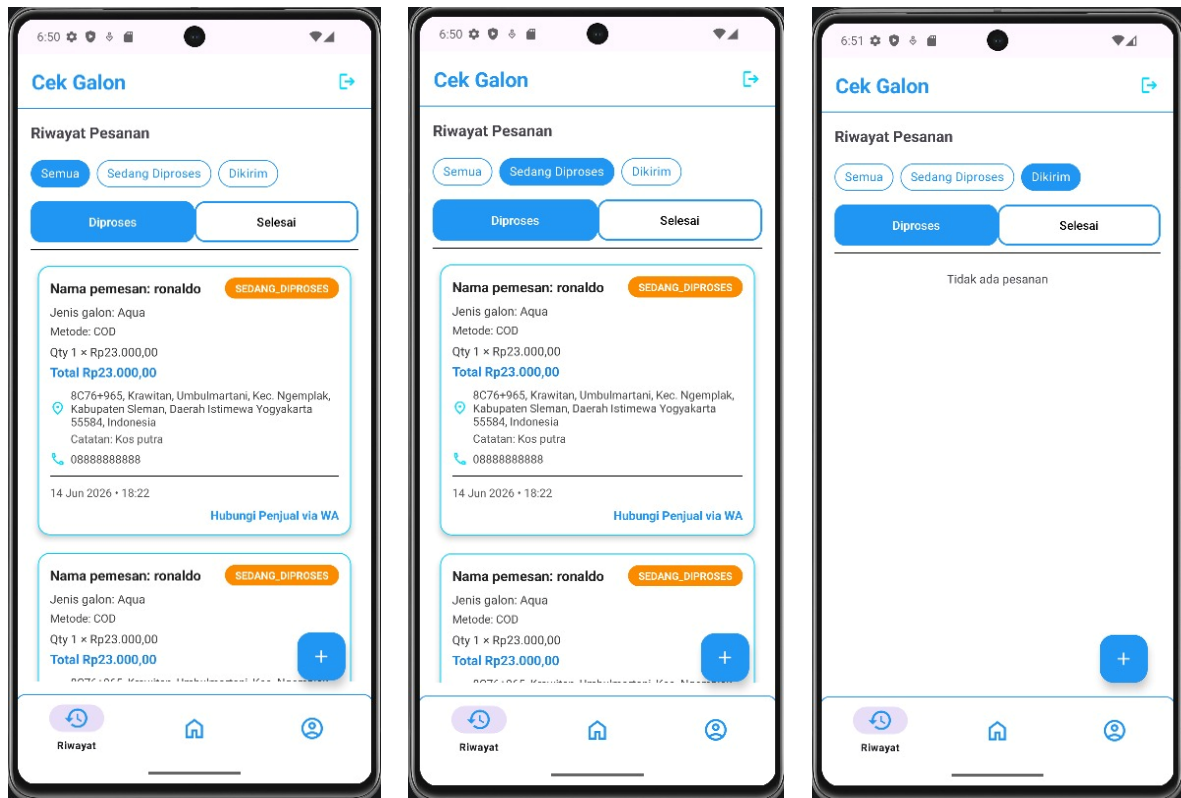
Halaman home pembeli berfungsi sebagai tampilan utama yang tampil setelah pengguna berhasil masuk ke dalam aplikasi. Pada halaman ini, sistem menampilkan sapaan selamat datang sesuai nama pengguna yang tersimpan pada akun, serta daftar pesanan aktif yang masih berada dalam proses layanan. Informasi yang ditampilkan pada setiap item pesanan meliputi nama pemesan, jenis galon, metode pembayaran, jumlah pesanan, total harga, alamat pengantaran, catatan lokasi, nomor handphone, serta waktu pemesanan. Selain itu, halaman ini juga menyediakan tombol tambah untuk membuka dialog pemesanan galon baru. Implementasi halaman home pembeli dapat dilihat pada Gambar 4.3.



Gambar 4.3 Implementasi Halaman Home Pembeli

4.2.4 Implementasi Halaman Riwayat Pesanan Pembeli

Halaman riwayat pesanan pembeli digunakan untuk menampilkan daftar pesanan yang pernah dilakukan oleh pengguna secara lebih terstruktur. Pada halaman ini, sistem membagi riwayat pesanan ke dalam dua tab, yaitu Diproses dan Selesai. Tab Diproses menampilkan pesanan yang masih berada pada tahap layanan, seperti status sedang diproses dan dikirim, sedangkan tab Selesai menampilkan pesanan yang telah selesai atau dibatalkan. Setiap kartu pesanan memuat informasi nama pemesan, jenis galon, metode pembayaran, jumlah pesanan, total harga, alamat pengantaran, catatan lokasi, nomor handphone, waktu pemesanan, serta tombol untuk menghubungi penjual melalui WhatsApp. Dengan adanya pengelompokan ini, pengguna dapat memantau riwayat transaksi dengan lebih mudah sesuai kondisi pesannya. Implementasi halaman riwayat pesanan pembeli pada tab Diproses dapat dilihat pada Gambar 4.4, sedangkan tampilan tab Selesai dapat dilihat pada Gambar 4.5.

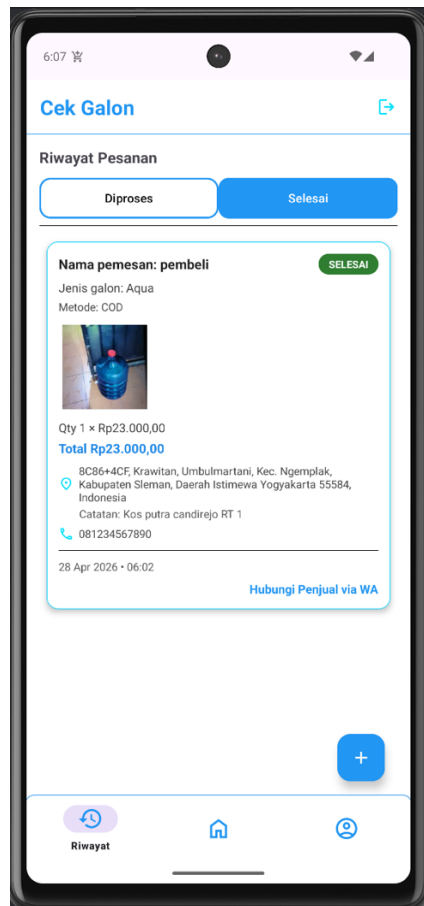


Gambar 4.4 Implementasi Halaman Riwayat Pesanan Pembeli pada Tab Diproses

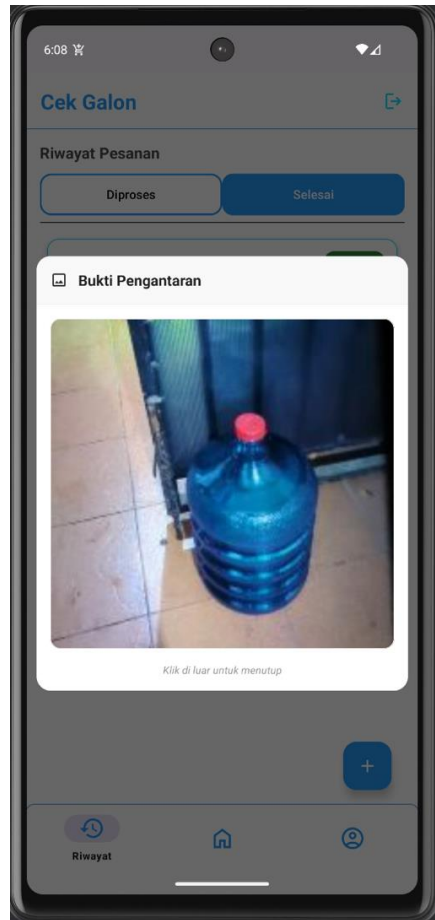


Gambar 4.5 Implementasi Halaman Riwayat Pesanan Pembeli pada Tab Selesai

Sebagai bentuk transparansi penyelesaian pesanan, aplikasi menampilkan pratinjau (thumbnail) foto bukti pengantaran pada item pesanan yang telah berstatus Selesai. Apabila pembeli menekan pratinjau tersebut, sistem akan menampilkan dialog yang memperbesar foto bukti pengantaran, sehingga pembeli dapat memverifikasi bahwa galon telah benar-benar sampai di lokasi dengan aman. Tampilan pratinjau foto pada item pesanan dapat dilihat pada Gambar 4.6, sedangkan dialog perbesaran foto dapat dilihat pada Gambar 4.7.



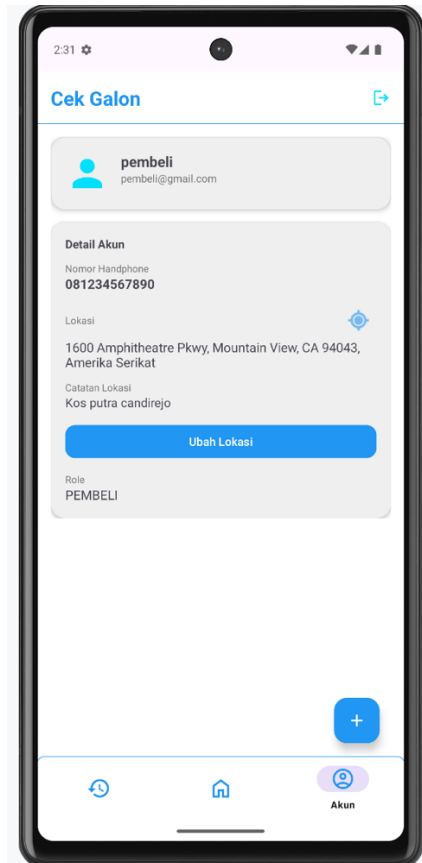
Gambar 4.6 Tampilan Pratinjau Bukti Pengantaran pada Halaman Riwayat Pembeli



Gambar 4.7 Dialog Perbesaran Foto Bukti Pengantaran pada Sisi Pembeli

4.2.5 Implementasi Halaman Akun Pembeli

Halaman akun pembeli digunakan untuk menampilkan informasi identitas pengguna yang tersimpan pada sistem, seperti nama pengguna, email, nomor handphone, lokasi, catatan lokasi, dan peran pengguna sebagai pembeli. Pada halaman ini, pengguna dapat melihat data akun yang digunakan saat registrasi, serta memperbarui informasi lokasi pengantaran melalui tombol Ubah Lokasi atau ikon lokasi untuk mengambil posisi saat ini secara otomatis. Dengan adanya halaman ini, data akun dan lokasi pengantaran dapat dikelola dengan lebih mudah sehingga mendukung proses pemesanan galon pada transaksi berikutnya. Implementasi halaman akun pembeli dapat dilihat pada Gambar 4.8.



Gambar 4.8 Implementasi Halaman Akun Pembeli

4.2.6 Implementasi Dialog Pemesanan Galon dan Mekanisme Pembayaran

Dialog pemesanan galon digunakan sebagai sarana utama bagi pembeli untuk melakukan pemesanan air galon secara langsung melalui aplikasi. Dialog ini ditampilkan ketika pengguna menekan tombol tambah pada halaman utama pembeli. Pada dialog pemesanan, sistem menampilkan beberapa komponen input yang meliputi nama pemesan, jenis galon, informasi stok tersedia, harga per item, kuantitas, total harga, lokasi pengantaran, catatan lokasi, nomor handphone, serta metode pembayaran. Data nama dan nomor handphone ditampilkan berdasarkan data akun pengguna, sedangkan lokasi pengantaran dapat menggunakan lokasi yang telah tersimpan maupun diperbarui kembali melalui fitur ubah lokasi atau pengambilan lokasi saat ini. Implementasi dialog pemesanan galon dapat dilihat pada Gambar 4.9.

Pesan Galon

Informasi Pelanggan

Nama: **ronaldo**

No HP: **0888888888**

Jenis Galon: **RO - Rp8.000,00**

Stok tersedia: 164

Harga per item: Rp8.000,00

Kuantitas: **2**

Jumlah Harga: **Rp16.000,00**

Lokasi Pengantaran

8C76-965, Krawitan, Umbulmartani, Kec. Ngemplak, Kabupaten Sleman, Daerah Istimewa Yogyakarta 55584, Indonesia

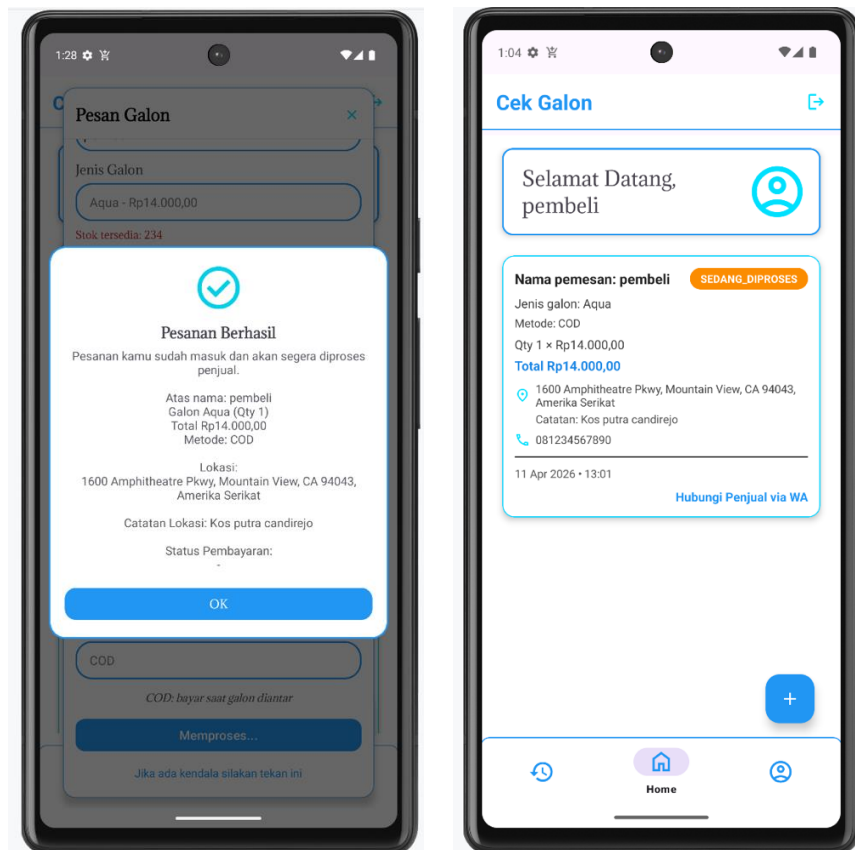
Catatan lokasi: Kos putra

[Ubah lokasi](#)

Metode Pembayaran

Gambar 4.9 Implementasi Dialog Pemesanan Galon

Setelah pengguna melengkapi data pesanan, sistem akan memproses pesanan berdasarkan metode pembayaran yang dipilih, yaitu Cash on Delivery (COD) atau QRIS. Pada metode COD, pesanan yang telah diisi lengkap akan langsung dikirim ke sistem dan diteruskan ke sisi penjual untuk diproses lebih lanjut. Pada kondisi ini, stok galon akan langsung dikurangi sesuai jumlah pesanan, kemudian pesanan ditampilkan pada daftar pesanan aktif dengan status sedang diproses. Mekanisme ini digunakan karena pembayaran dilakukan saat pesanan diantar ke lokasi pengguna. Tampilan pesanan dengan metode COD pada halaman utama pembeli dapat dilihat pada Gambar 4.10.



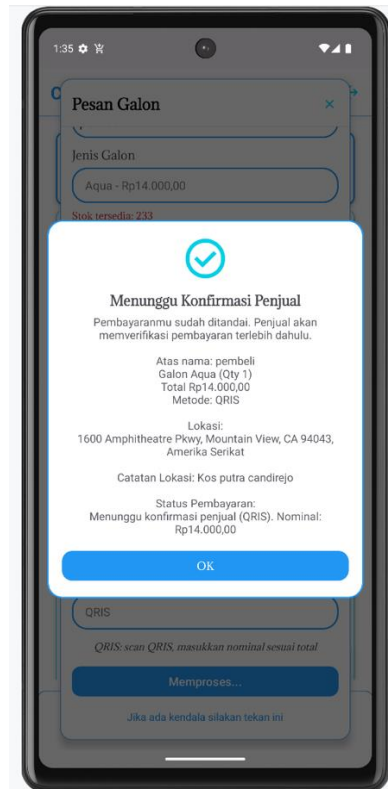
Gambar 4.10 Tampilan Pesanan dengan COD pada Halaman Home Pembeli

Berbeda dengan metode COD, pada metode QRIS sistem akan menampilkan dialog pembayaran QRIS setelah data pesanan berhasil dibuat. Dialog tersebut menampilkan kode QR beserta nominal pembayaran yang harus dibayarkan oleh pengguna. Pada tahap ini, pengguna diberikan batas waktu pembayaran selama 15 menit sejak pesanan dibuat. Selain itu, pengguna juga diberikan dua pilihan, yaitu tombol Saya sudah bayar dan tombol Tutup. Apabila pengguna menekan tombol Saya sudah bayar, sistem akan menandai pesanan sebagai menunggu konfirmasi penjual, sehingga penjual dapat melakukan verifikasi pembayaran sebelum pesanan diproses lebih lanjut. Implementasi dialog pembayaran QRIS dapat dilihat pada Gambar 4.11.



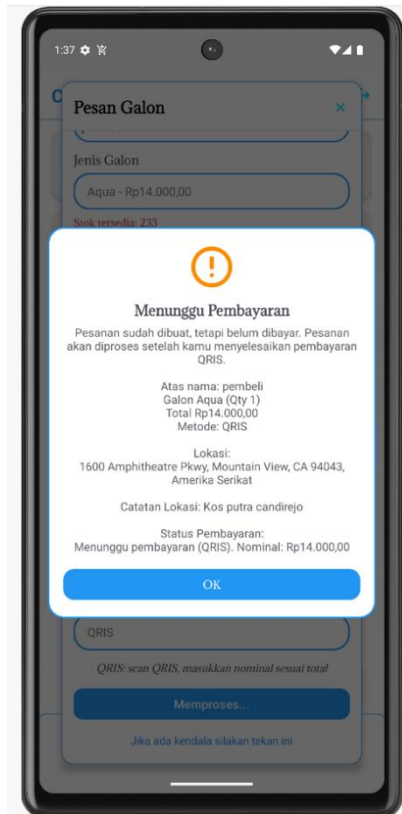
Gambar 4.11 Implementasi Dialog Pembayaran QRIS

Setelah pengguna menekan tombol Saya sudah bayar, sistem menampilkan dialog hasil pesanan sebagai bentuk umpan balik bahwa pembayaran telah ditandai dan pesanan berada pada status menunggu konfirmasi penjual. Pada kondisi ini, pesanan belum sepenuhnya diproses hingga penjual melakukan verifikasi terhadap pembayaran yang masuk. Dialog hasil pesanan dengan status menunggu konfirmasi penjual dapat dilihat pada Gambar 4.12.



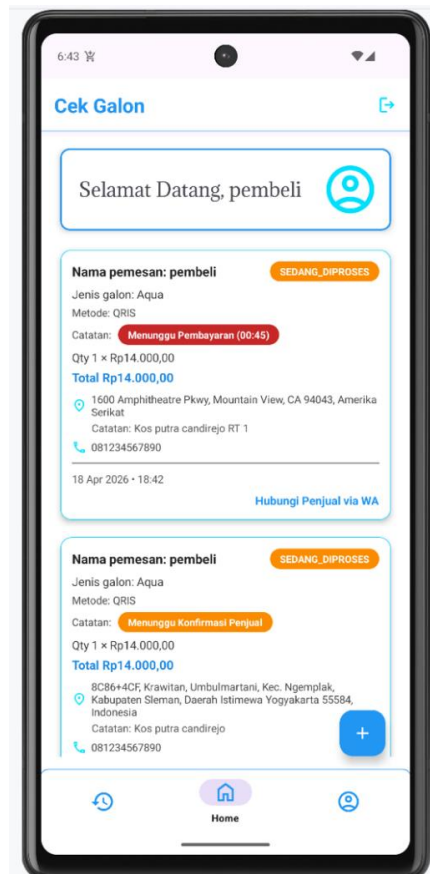
Gambar 4.12 Dialog Hasil Pesanan QRIS dengan Status Menunggu Konfirmasi Penjual

Namun, apabila pengguna menekan tombol Tutup pada dialog pembayaran QRIS, pesanan tetap tersimpan di sistem dengan status menunggu pembayaran. Hal ini menunjukkan bahwa pesanan telah dibuat, tetapi pembayaran belum diselesaikan, sehingga pesanan belum dapat diproses lebih lanjut oleh penjual. Pengguna masih dapat melanjutkan pembayaran tanpa perlu mengisi ulang data pesanan dari awal, selama pembayaran dilakukan sebelum batas waktu 15 menit berakhir. Jika sampai melewati 15 menit pengguna belum juga menyelesaikan pembayaran, maka pesanan akan secara otomatis dibatalkan oleh sistem. Dialog hasil pesanan dengan status menunggu pembayaran dapat dilihat pada Gambar 4.13.



Gambar 4.13 Dialog Hasil Pesanan QRIS dengan Status Menunggu Pembayaran

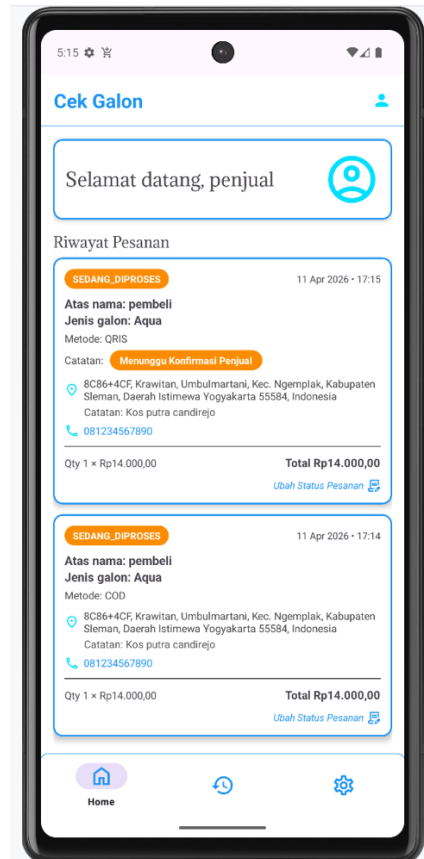
Setelah proses pemesanan selesai, informasi pesanan akan ditampilkan pada halaman utama pembeli sesuai dengan kondisi pembayaran dan status prosesnya. Pada pesanan dengan metode QRIS, sistem menampilkan keterangan status pembayaran, seperti Menunggu Konfirmasi Penjual atau Menunggu Pembayaran, sehingga pengguna dapat mengetahui kondisi pesannya secara langsung. Dengan mekanisme ini, proses pemesanan dan pembayaran menjadi lebih terstruktur karena pengguna dapat memantau perkembangan pesanan tanpa harus melakukan konfirmasi manual kepada penjual. Tampilan status pesanan QRIS pada halaman utama pembeli dapat dilihat pada Gambar 4.14.



Gambar 4.14 Tampilan Status Pesanan QRIS pada Halaman Home Pembeli

4.2.7 Implementasi Halaman Home Penjual

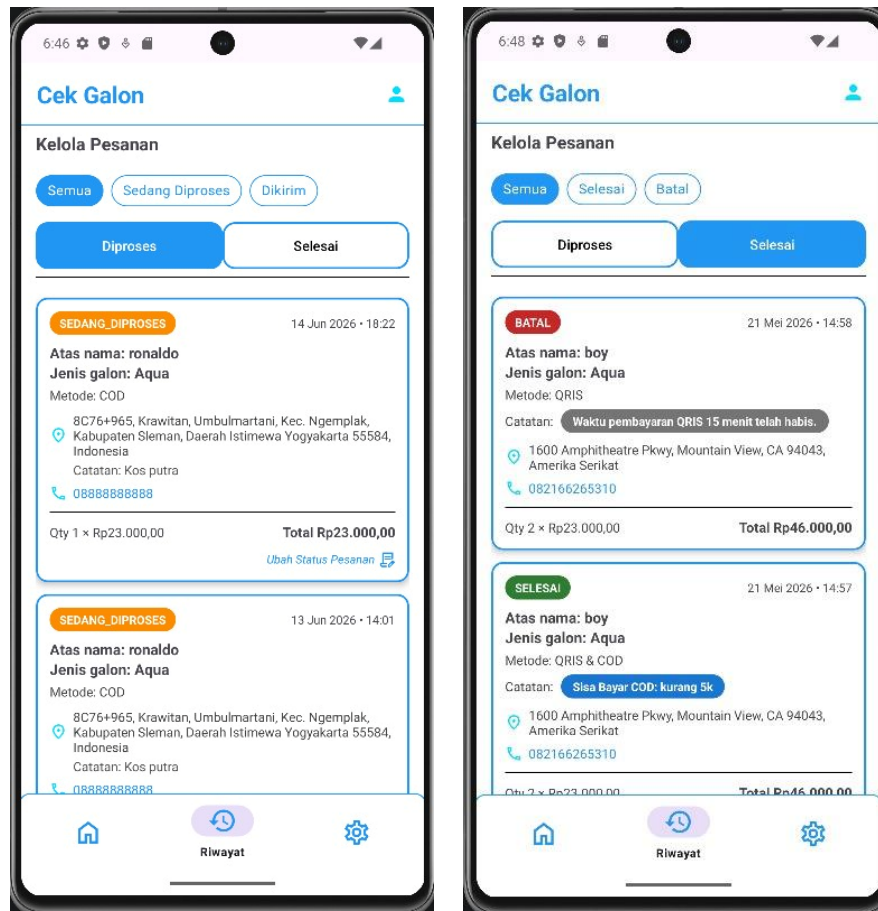
Halaman home penjual digunakan sebagai tampilan utama setelah penjual berhasil masuk ke dalam aplikasi. Pada halaman ini, sistem menampilkan sapaan sesuai nama pengguna serta daftar pesanan aktif yang perlu segera ditangani oleh penjual. Setiap item pesanan memuat informasi penting, seperti nama pemesan, jenis galon, metode pembayaran, catatan pembayaran, alamat pengantaran, nomor handphone, jumlah pesanan, total harga, dan waktu pemesanan. Selain itu, pada setiap pesanan juga tersedia tombol Ubah Status Pesanan yang digunakan untuk menindaklanjuti proses pesanan sesuai kondisi layanan. Implementasi halaman home penjual dapat dilihat pada Gambar 4.15.



Gambar 4.15 Implementasi Halaman Home Penjual

4.2.8 Implementasi Halaman Riwayat Pesanan Penjual

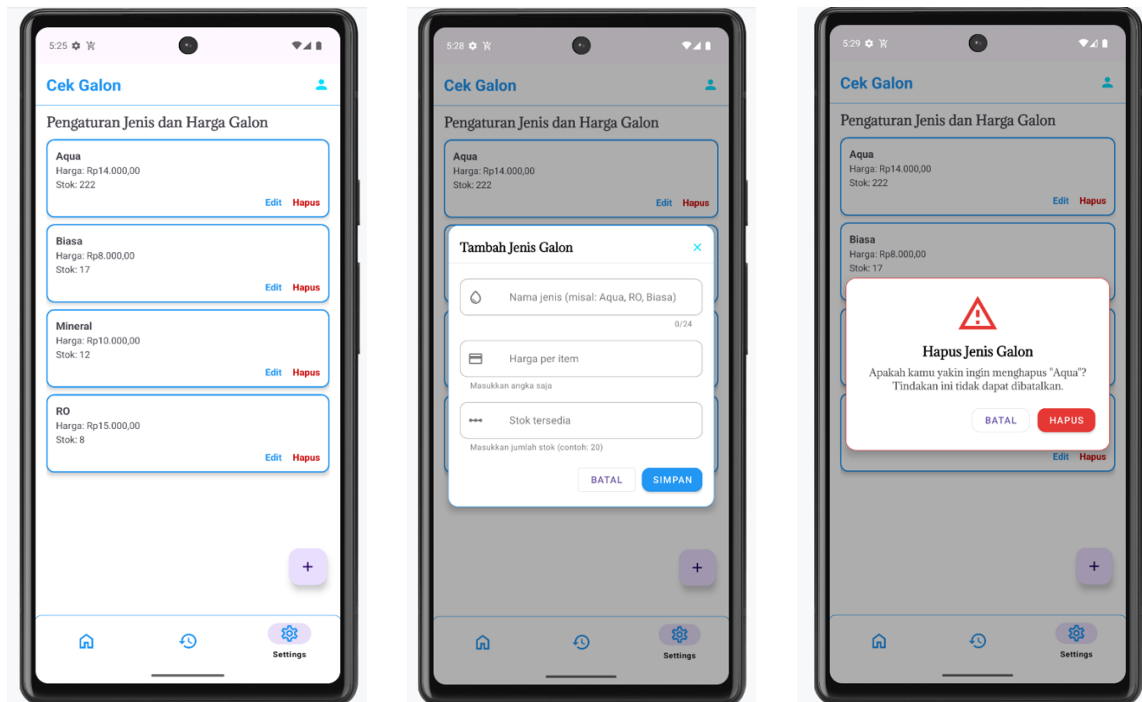
Halaman riwayat pesanan penjual digunakan untuk menampilkan dan mengelola seluruh pesanan yang masuk ke dalam sistem secara lebih terstruktur. Pada halaman ini, pesanan dibagi ke dalam dua tab, yaitu Diproses dan Selesai. Tab Diproses menampilkan pesanan yang masih berada dalam tahap penanganan, seperti status sedang diproses dan dikirim, sedangkan tab Selesai menampilkan pesanan yang telah selesai atau dibatalkan. Setiap kartu pesanan memuat informasi nama pemesan, jenis galon, metode pembayaran, catatan pembayaran, alamat pengantaran, nomor handphone, jumlah pesanan, total harga, serta waktu pemesanan. Selain itu, penjual juga dapat melakukan tindak lanjut terhadap pesanan melalui tombol Ubah Status Pesanan yang tersedia pada setiap data. Dengan adanya pembagian ini, penjual dapat memantau perkembangan pesanan dengan lebih mudah sesuai kondisi layanan. Implementasi halaman riwayat pesanan penjual pada tab Diproses dapat dilihat pada Gambar 4.16.



Gambar 4.16 Implementasi Halaman Riwayat Pesanan Penjual

4.2.9 Implementasi Halaman Pengaturan Jenis dan Harga Galon

Halaman pengaturan jenis dan harga galon digunakan oleh penjual untuk mengelola data galon yang tersedia di dalam aplikasi. Pada halaman ini, sistem menampilkan daftar jenis galon beserta informasi harga dan stok yang masih tersedia. Setiap data galon dilengkapi dengan tombol Edit dan Hapus untuk memudahkan penjual dalam memperbarui maupun menghapus data yang sudah ada. Selain itu, tersedia tombol tambah yang digunakan untuk menampilkan dialog penambahan jenis galon baru dengan isian nama jenis galon, harga per item, dan stok tersedia. Dengan adanya halaman ini, penjual dapat mengelola data pendukung layanan secara lebih mudah dan terstruktur sesuai kondisi operasional depot. Implementasi halaman pengaturan jenis dan harga galon dapat dilihat pada Gambar 4.17.



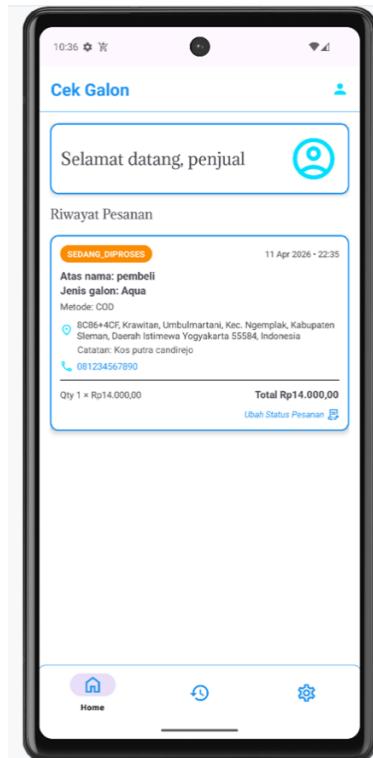
Gambar 4.17 Implementasi Halaman Pengaturan Jenis dan Harga Galon

4.2.10 Implementasi Perubahan Status Pesanan

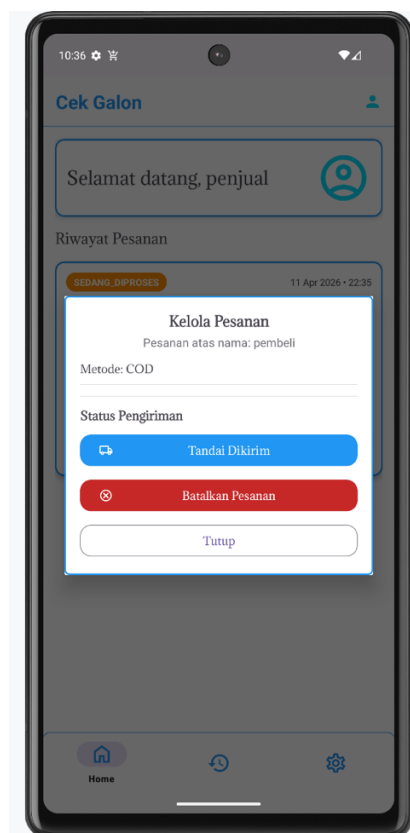
Pada sistem aplikasi, perubahan status pesanan dilakukan oleh penjual melalui fitur kelola pesanan. Mekanisme ini digunakan untuk memperbarui kondisi pesanan sesuai tahap layanan yang sedang berjalan, sehingga informasi yang ditampilkan pada sisi penjual dan pembeli dapat tetap sinkron. Perubahan status pesanan pada aplikasi ini dibedakan berdasarkan metode pembayaran yang digunakan, yaitu *Cash on Delivery* (COD) dan QRIS.

A. Perubahan Status Pesanan dengan Metode COD

Pada pesanan dengan metode COD, penjual dapat langsung menindaklanjuti pesanan yang masuk karena pembayaran dilakukan saat galon diantar ke lokasi pembeli. Pesanan yang baru masuk akan ditampilkan pada sisi penjual dengan status sedang diproses, kemudian penjual dapat membuka dialog kelola pesanan melalui tombol Ubah Status Pesanan. Tampilan pesanan COD sebelum dilakukan perubahan status dapat dilihat pada Gambar 4.18, sedangkan dialog kelola pesanan COD dapat dilihat pada Gambar 4.19.

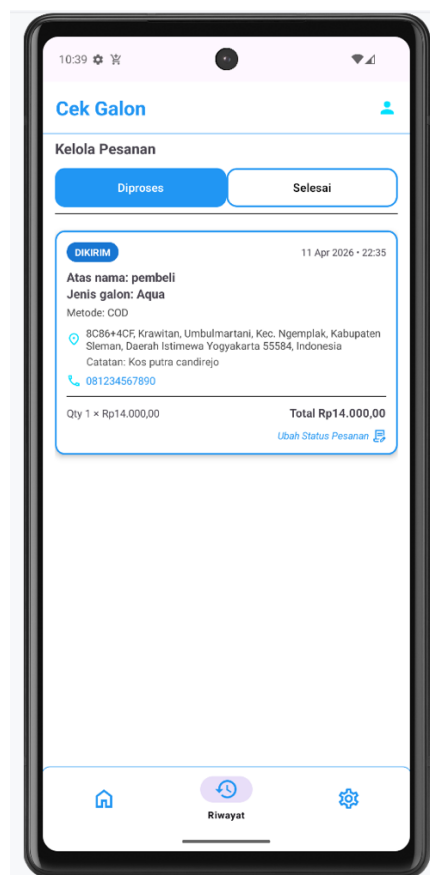


Gambar 4.18 Tampilan Pesanan COD pada Sisi Penjual Sebelum Perubahan Status



Gambar 4.19 Dialog Kelola Pesanan COD pada Sisi Penjual

Setelah dialog kelola pesanan dibuka, penjual dapat memilih tindakan Tandai Dikirim untuk mengubah status pesanan dari sedang diproses menjadi dikirim. Perubahan tersebut akan langsung tampil pada daftar pesanan penjual sebagai penanda bahwa pesanan sedang berada dalam proses pengantaran. Setelah galon berhasil diantar, penjual dapat kembali membuka dialog kelola pesanan untuk menandai pesanan sebagai selesai atau batal sesuai kondisi layanan di lapangan. Tampilan perubahan status pesanan COD menjadi dikirim dapat dilihat pada Gambar 4.20, sedangkan dialog lanjutan untuk menandai pesanan sebagai selesai dapat dilihat pada Gambar 4.21.

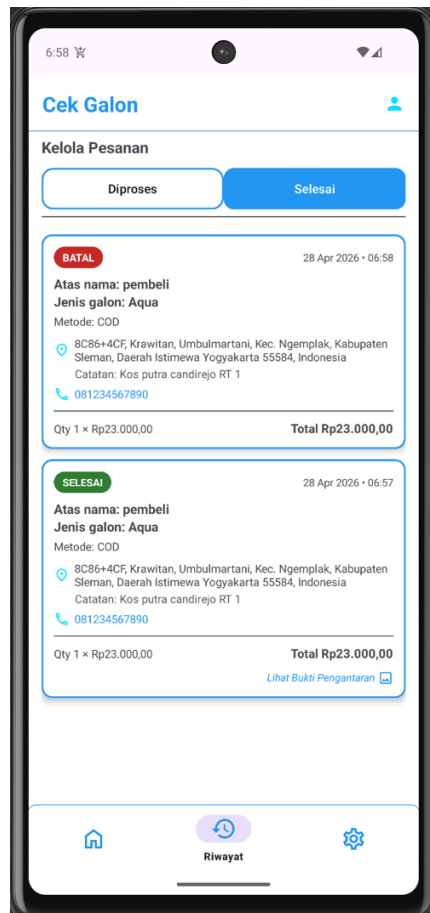


Gambar 4.20 Tampilan Perubahan Status Pesanan COD Menjadi Dikirim pada Sisi Penjual



Gambar 4.21 Dialog Perubahan Status Lanjutan Pesanan COD pada Sisi Penjual

Jika penjual memilih Tandai Selesai, maka pesanan akan berpindah ke tab Selesai dengan status selesai. Sebaliknya, jika penjual memilih Batalkan Pesanan, maka pesanan akan ditampilkan pada tab yang sama dengan status batal. Dengan mekanisme ini, sistem dapat mencatat hasil akhir pesanan secara lebih jelas dan terstruktur. Tampilan hasil akhir perubahan status pesanan COD pada sisi penjual dapat dilihat pada Gambar 4.22.

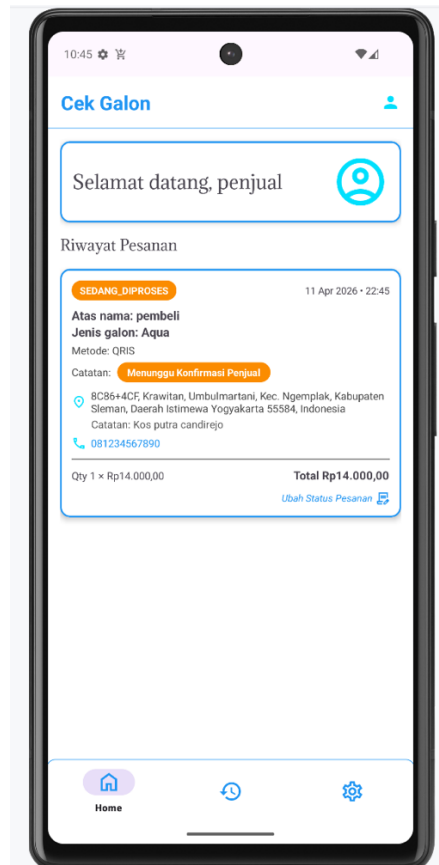


Gambar 4.22 Tampilan Hasil Akhir Perubahan Status Pesanan COD pada Sisi Penjual

Perubahan status yang dilakukan penjual pada pesanan COD juga akan memengaruhi tampilan pada sisi pembeli. Dengan demikian, pembeli dapat mengetahui apakah pesannya masih diproses, sedang dikirim, telah selesai, atau dibatalkan tanpa perlu melakukan konfirmasi manual kepada penjual.

B. Perubahan Status Pesanan dengan Metode QRIS

Pada pesanan dengan metode QRIS, mekanisme perubahan status pesanan dilakukan dengan alur yang lebih bertahap karena penjual perlu memverifikasi pembayaran terlebih dahulu. Setelah pembeli menandai bahwa pembayaran telah dilakukan, pesanan akan tampil pada sisi penjual dengan status sedang diproses dan catatan Menunggu Konfirmasi Penjual. Kondisi ini menunjukkan bahwa pesanan belum dapat dikirim sebelum penjual melakukan verifikasi pembayaran. Tampilan pesanan QRIS dengan status menunggu konfirmasi penjual dapat dilihat pada Gambar 4.23.



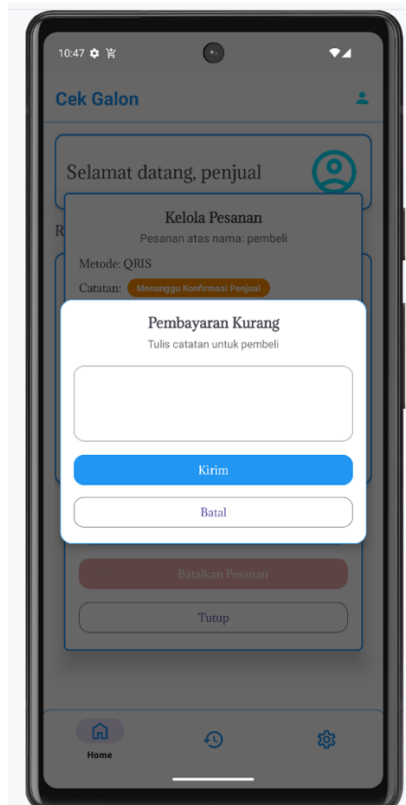
Gambar 4.23 Tampilan Pesanan QRIS dengan Status Menunggu Konfirmasi Penjual pada Sisi Penjual

Ketika tombol Ubah Status Pesanan dipilih, sistem akan menampilkan dialog kelola pesanan yang memuat tiga pilihan verifikasi pembayaran, yaitu Terima Pembayaran, Pembayaran Kurang, dan Tolak Pembayaran. Pada tahap ini, tombol untuk status pengiriman masih belum dapat digunakan sebelum keputusan pembayaran ditetapkan. Dialog kelola pesanan QRIS dapat dilihat pada Gambar 4.24.

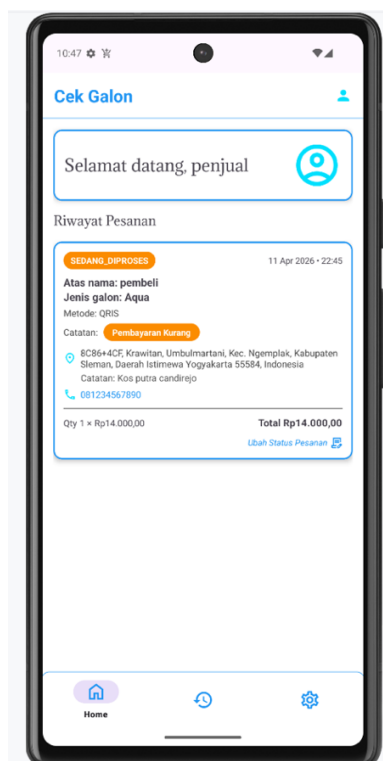


Gambar 4.24 Dialog Kelola Pesanan QRIS pada Sisi Penjual

Apabila penjual memilih Pembayaran Kurang, sistem akan menampilkan dialog tambahan untuk menuliskan catatan kepada pembeli mengenai kekurangan pembayaran yang harus dilunasi. Setelah catatan dikirim, informasi tersebut akan tersimpan pada pesanan dan ditampilkan kembali pada sisi penjual sebagai penanda bahwa pembayaran belum sepenuhnya diterima. Dialog pengisian catatan pembayaran kurang dapat dilihat pada Gambar 4.25, sedangkan tampilan pesanan QRIS dengan status Pembayaran Kurang pada sisi penjual dapat dilihat pada Gambar 4.26.

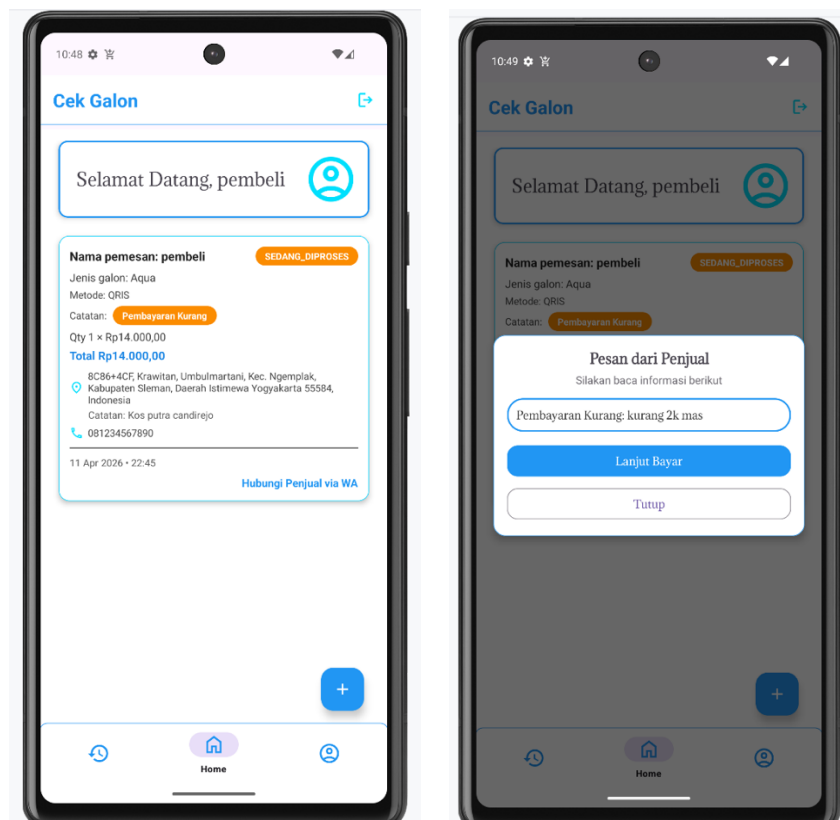


Gambar 4.25 Dialog Pengisian Catatan Pembayaran Kurang pada Sisi Penjual



Gambar 4.26 Tampilan Pesanan QRIS dengan Status Pembayaran Kurang pada Sisi Penjual

Setelah penjual memberikan catatan bahwa pembayaran kurang, sistem juga menampilkan informasi tersebut pada sisi pembeli. Pembeli akan menerima pesan dari penjual yang menjelaskan kondisi pembayaran dan diberikan opsi untuk melanjutkan pembayaran melalui dialog QRIS. Dengan mekanisme ini, pembeli tidak perlu membuat pesanan baru, melainkan cukup menyelesaikan pembayaran pada pesanan yang sama. Tampilan pesan dari penjual pada sisi pembeli dapat dilihat pada Gambar 4.27, sedangkan dialog pembayaran QRIS untuk melanjutkan pembayaran dapat dilihat pada Gambar 4.28.

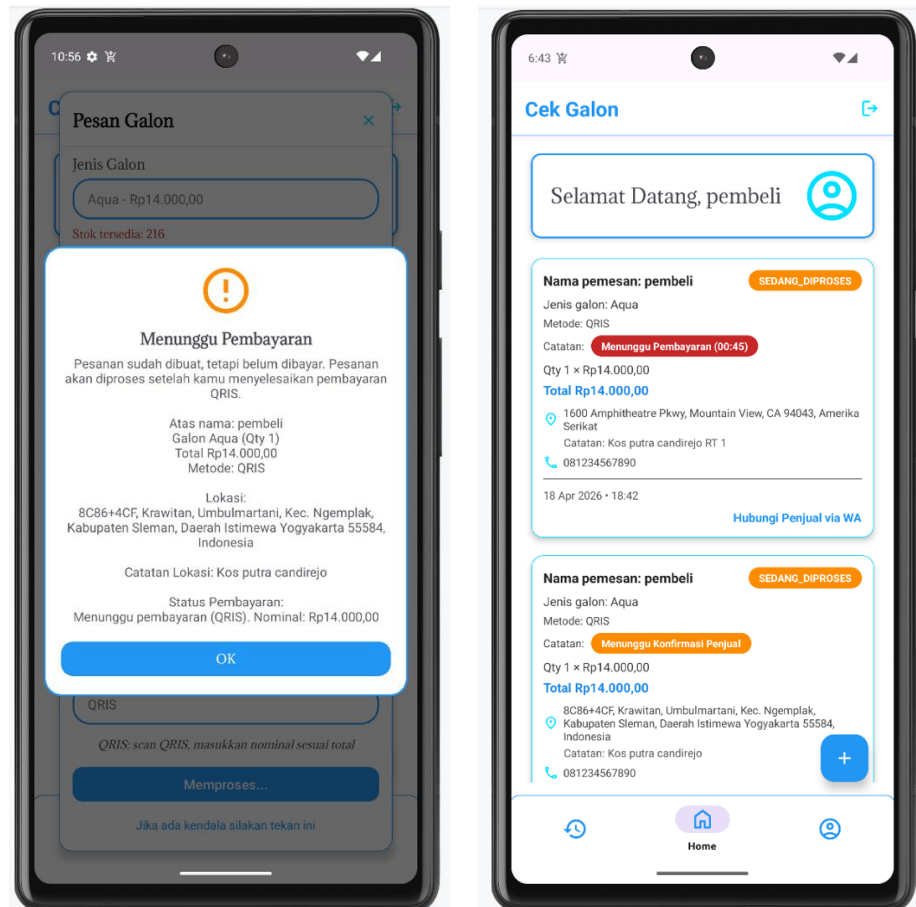


Gambar 4.27 Tampilan Pesan dari Penjual terkait Pembayaran Kurang pada Sisi Pembeli



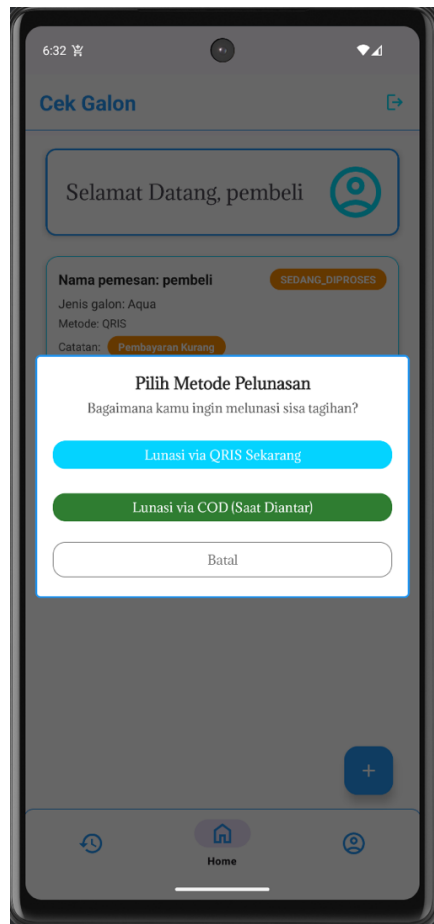
Gambar 4.28 Dialog Pembayaran QRIS untuk Melanjutkan Pembayaran pada Sisi Pembeli

Apabila pembeli belum langsung menyelesaikan pembayaran QRIS, maka pesanan akan tetap ditampilkan pada sisi pembeli dengan catatan Menunggu Pembayaran. Pada kondisi ini, sistem memberikan batas waktu pembayaran selama 15 menit. Apabila hingga batas waktu tersebut pembayaran belum dilakukan, maka pesanan akan dibatalkan secara otomatis oleh sistem. Kondisi ini menunjukkan bahwa pesanan telah dibuat, namun belum dapat diproses lebih lanjut oleh penjual sebelum pembayaran dilengkapi. Tampilan pesanan QRIS dengan status menunggu pembayaran pada sisi pembeli dapat dilihat pada Gambar 4.29.



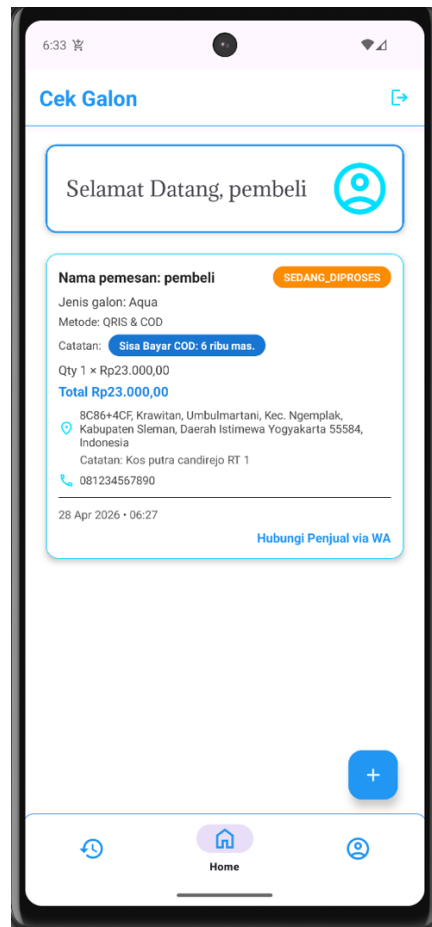
Gambar 4.29 Tampilan Pesanan QRIS dengan Status Menunggu Pembayaran pada Sisi Pembeli

Setelah penjual memberikan catatan bahwa pembayaran kurang, sistem juga menampilkan informasi tersebut pada sisi pembeli. Pembeli diberikan opsi untuk melanjutkan pembayaran melalui dialog pemilihan metode pelunasan. Pada tahap ini, pembeli dapat memilih untuk melunasi sisa tagihan melalui transfer QRIS kembali, atau mengubah sisa tagihan menjadi metode COD (pembayaran saat galon diantar). Tampilan dialog pemilihan metode pelunasan pada sisi pembeli dapat dilihat pada Gambar 4.30.



Gambar 4.30 Dialog Pemilihan Metode Pelunasan oleh Pembeli

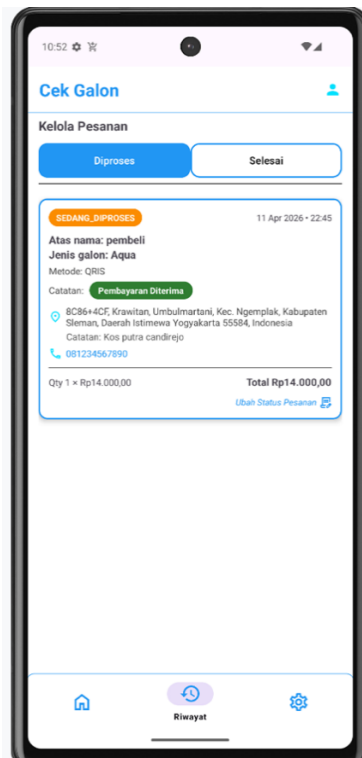
Apabila pembeli memilih opsi pelunasan via COD, sistem akan secara otomatis mengubah metode pembayaran menjadi "QRIS & COD" dan memperbarui catatan pesanan menjadi "Sisa Bayar COD" beserta rincian nominal kekurangannya. Pembaruan ini langsung ditampilkan pada antarmuka penjual dengan label penanda berwarna biru, yang memberikan sinyal bahwa penjual dapat langsung memproses pengiriman dan menagih sisa kekurangan secara tunai di lokasi. Pendekatan ini memberikan fleksibilitas transaksi tanpa mengharuskan pembeli membatalkan pesanan. Tampilan pesanan dengan status pelunasan COD dapat dilihat pada Gambar 4.31.



Gambar 4.31 Tampilan Pesanan dengan Status Sisa Bayar COD

Sebaliknya, apabila pembeli memilih opsi pelunasan via QRIS, mekanisme pembayaran akan berjalan seperti biasa. Sistem akan menampilkan kembali dialog pembayaran QRIS, dan pembeli melakukan transfer sesuai sisa tagihan. Setelah dikonfirmasi oleh pembeli, pesanan akan kembali berstatus "Menunggu Konfirmasi Penjual" agar penjual dapat melakukan verifikasi ulang terhadap dana pelunasan yang baru masuk.

Setelah pembayaran dinyatakan lengkap dan penjual memilih Terima Pembayaran, maka catatan pesanan akan berubah menjadi Pembayaran Diterima. Pada tahap ini, pesanan (baik yang telah lunas via QRIS maupun yang menggunakan pelunasan COD) dapat dilanjutkan ke proses pengiriman melalui tombol Tandai Dikirim. Tampilan pesanan QRIS dengan status Pembayaran Diterima dapat dilihat pada Gambar 4.32, sedangkan dialog perubahan status lanjutan menjadi dikirim dapat dilihat pada Gambar 4.33.

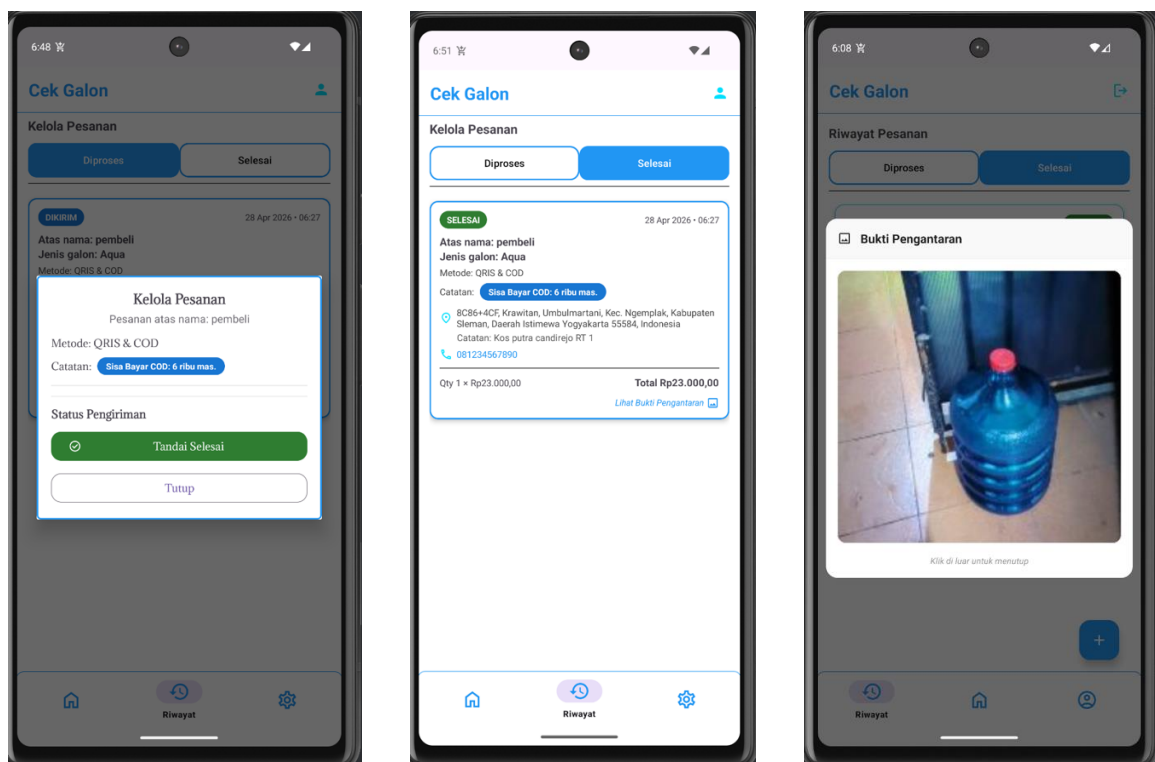


Gambar 4.32 Tampilan Pesanan QRIS dengan Status Pembayaran Diterima pada Sisi Penjual



Gambar 4.33 Dialog Perubahan Status Pengiriman Pesanan QRIS pada Sisi Penjual

Tahap akhir dari siklus pesanan adalah penyerahan pesanan kepada pembeli. Saat penjual hendak mengubah status pesanan dari "Dikirim" menjadi "Selesai", sistem mewajibkan penjual untuk mengambil foto bukti pengantaran menggunakan fitur kamera perangkat. Setelah foto diambil, sistem akan mengunggah gambar tersebut secara otomatis ke penyimpanan cloud dan menyimpan tautan (URL) gambar pada dokumen pesanan di Firestore. Mekanisme ini memastikan adanya bukti pengantaran untuk setiap transaksi yang telah diselesaikan. Dialog penyelesaian pesanan dapat dilihat pada Gambar 4.34.



Gambar 4.34 Dialog Penyelesaian Status Akhir Pesanan QRIS pada Sisi Penjual

Dengan demikian, perubahan status pesanan pada metode QRIS tidak hanya berfungsi untuk menunjukkan tahap layanan, tetapi juga menjadi media verifikasi pembayaran dan komunikasi antara penjual dan pembeli. Mekanisme ini membuat proses transaksi menjadi lebih terstruktur karena penjual dapat memastikan pembayaran terlebih dahulu, sedangkan pembeli tetap memperoleh informasi yang jelas mengenai kondisi pesannya.

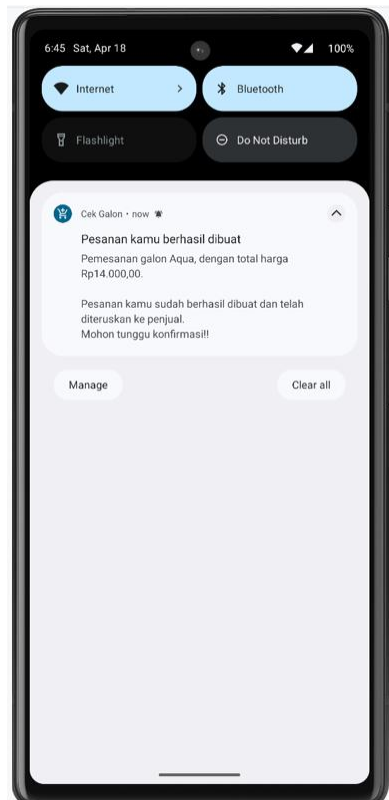
4.2.11 Implementasi Sistem Notifikasi dan Batas Waktu Pembayaran

Sistem Implementasi sistem notifikasi dan batas waktu pembayaran digunakan untuk memberikan informasi terkini kepada pembeli maupun penjual terkait perkembangan status

pesanan. Melalui fitur ini, pengguna dapat mengetahui setiap tahapan transaksi secara langsung tanpa harus terus-menerus membuka dan mengecek aplikasi. Selain itu, sistem ini juga mengelola mekanisme batas waktu pembayaran khusus untuk pesanan dengan metode QRIS. Pada mekanisme ini, sistem akan membatalkan pesanan secara otomatis apabila pembeli tidak segera mengonfirmasi pembayaran dalam rentang waktu yang telah ditentukan. Implementasi dari fitur notifikasi dan batas waktu pembayaran ini dibedakan berdasarkan alur interaksi pada sisi pembeli dan sisi penjual.

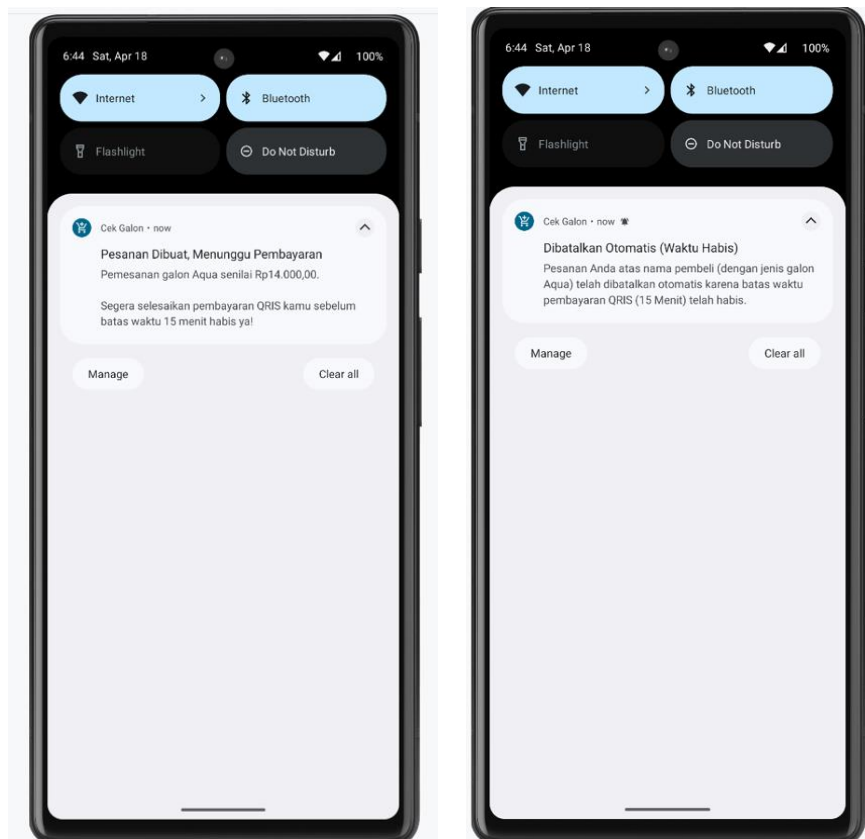
A. Notifikasi dan Batas Waktu pada Sisi Pembeli

Pada sisi pembeli, sistem akan menampilkan notifikasi (*pop-up banner*) pada beberapa tahapan pesanan. Saat pesanan baru berhasil dibuat, sistem memberikan notifikasi yang berbeda berdasarkan metode pembayaran. Pada metode COD, notifikasi berisi informasi bahwa pesanan telah berhasil dibuat dan diteruskan ke penjual. Sedangkan pada metode QRIS, notifikasi berisi pengingat agar pembeli segera menyelesaikan tahap pembayaran. Tampilan notifikasi pembuatan pesanan pada sisi pembeli dapat dilihat pada Gambar 4.35.



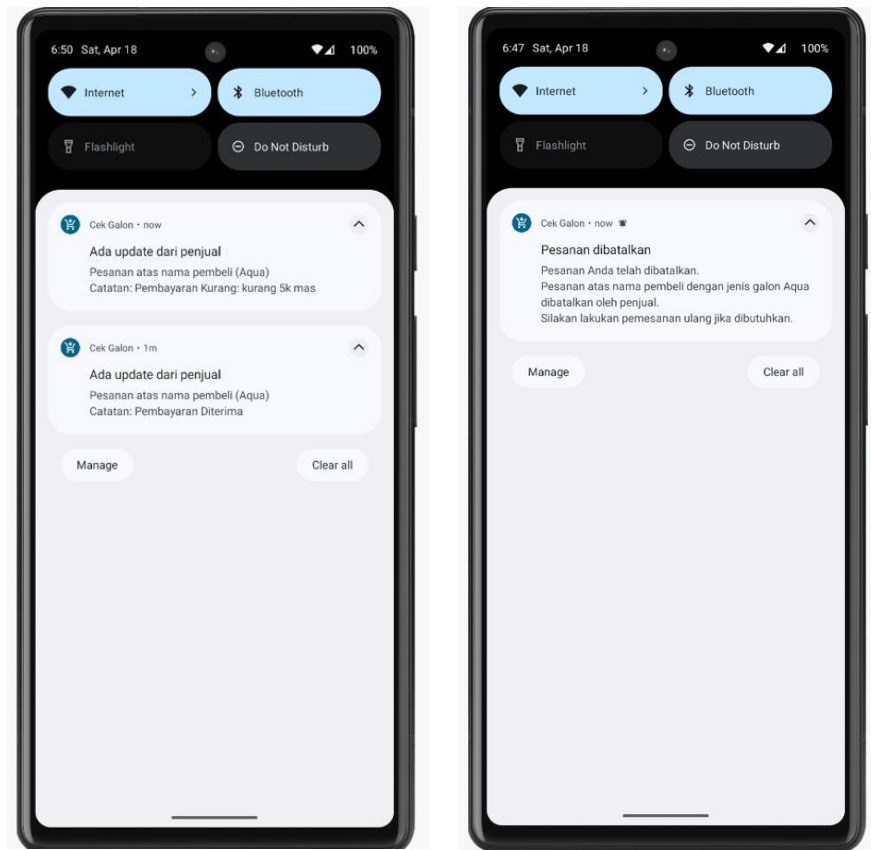
Gambar 4.35 Tampilan Notifikasi Pembuatan Pesanan pada Sisi Pembeli

Khusus untuk pesanan dengan metode QRIS, sistem menerapkan batas waktu pembayaran selama 15 menit. Waktu hitung mundur ini ditampilkan langsung pada antarmuka aplikasi. Apabila pembeli tidak menyelesaikan pembayaran dan mengonfirmasinya dalam tenggat waktu tersebut, pesanan akan dibatalkan secara otomatis oleh sistem, dan pembeli akan menerima notifikasi pembatalan akibat batas waktu habis. Tampilan notifikasi waktu pembayaran habis dapat dilihat pada Gambar 4.36.



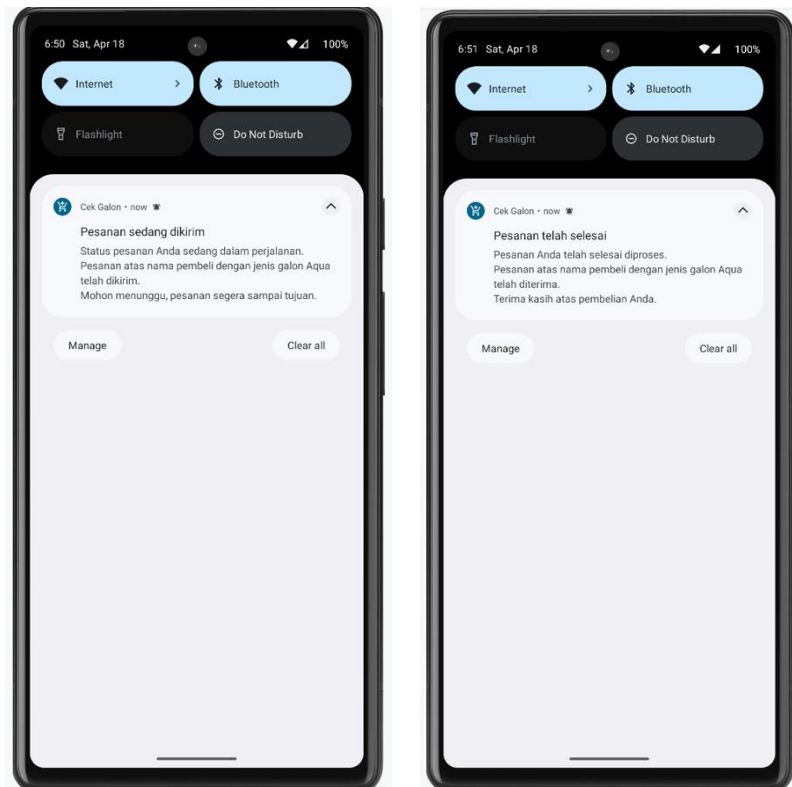
Gambar 4.36 Tampilan Notifikasi Pembatalan Otomatis akibat Batas Waktu Habis

Selanjutnya, pada metode QRIS, pembeli juga akan menerima notifikasi khusus yang memberikan pembaruan (update) ketika penjual melakukan verifikasi terhadap pembayaran yang dikirimkan. Notifikasi ini meliputi status "Pembayaran Diterima" jika nominal pembayaran sudah sesuai, "Pembayaran Kurang" jika nominal belum mencukupi dan membutuhkan pelunasan, serta "Pembayaran Ditolak" jika terjadi kendala pada transaksi. Tampilan notifikasi verifikasi pembayaran QRIS ini dapat dilihat pada Gambar 4.37.



Gambar 4.37 Tampilan Notifikasi Verifikasi Pembayaran QRIS pada Sisi Pembeli

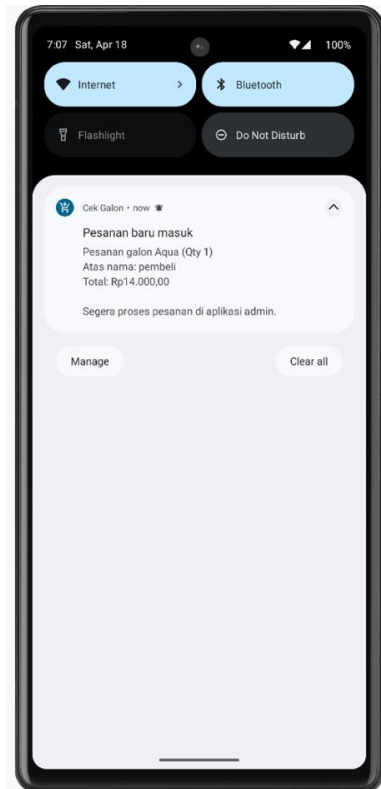
Setelah tahap pembayaran selesai dan statusnya diterima oleh penjual, alur pesanan QRIS akan kembali seperti pada metode COD. Pada tahap ini, pembeli akan menerima notifikasi lanjutan ketika penjual memperbarui status pengiriman pesanan, yaitu saat pesanan diubah menjadi dikirim, atau jika pesanan telah selesai diantar oleh penjual. Tampilan notifikasi perubahan status pengiriman dapat dilihat pada Gambar 4.38.



Gambar 4.38 Tampilan Notifikasi Perubahan Status Pengiriman pada Sisi Pembeli

B. Notifikasi pada Sisi Penjual

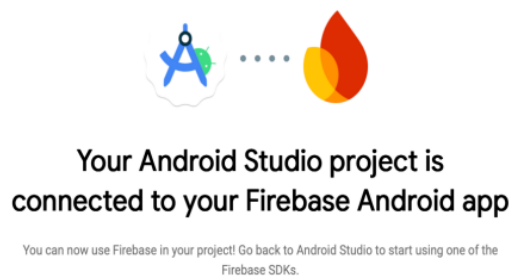
Pada sisi penjual, sistem memfokuskan pemberitahuan pada pesanan baru yang masuk agar penjual dapat segera melakukan tindakan layanan. Pada pesanan dengan metode COD, notifikasi pesanan masuk akan langsung ditampilkan sesaat setelah pembeli membuat pesanan. Sementara itu, pada pesanan dengan metode QRIS, notifikasi pesanan baru hanya akan dikirimkan kepada penjual setelah pembeli mengonfirmasi pembayaran melalui tombol "Saya sudah bayar". Hal ini dirancang agar penjual tidak terganggu oleh pemberitahuan pesanan yang masih berada dalam proses pembayaran awal oleh pembeli. Tampilan notifikasi pesanan masuk pada sisi penjual dapat dilihat pada Gambar 4.39.



Gambar 4.39 Tampilan Notifikasi Pesanan Baru Masuk pada Sisi Penjual

4.3 Integrasi Firebase

Pada tahapan ini, pengembangan aplikasi menggunakan Firebase sebagai layanan *database* untuk menangani kebutuhan autentikasi dan pengelolaan data aplikasi. Firebase digunakan sebagai tempat penyimpanan data terpusat yang memungkinkan pertukaran data secara *real-time* antara aplikasi dan server. Proses penghubungan aplikasi dengan firebase dilakukan melalui penyesuaian konfigurasi pada *firebase console* serta integrasi proyek di android studio, sehingga aplikasi dapat beroperasi dengan dukungan layanan firebase secara optimal. Keberhasilan proses tersebut dapat dilihat pada tampilan status koneksi proyek pada android studio, sebagaimana ditunjukkan pada Gambar 4.40.

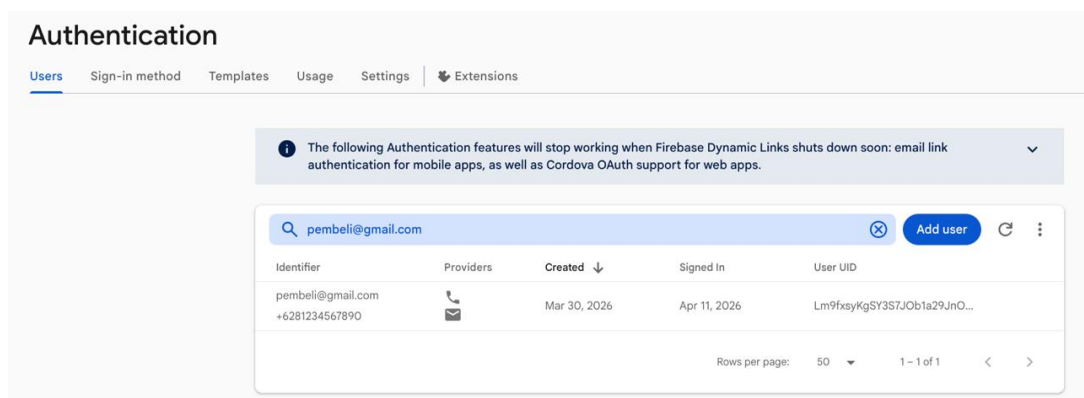


Gambar 4.40 Tampilan Integrasi Firebase dengan Android Studio

Setelah proses integrasi berhasil dilakukan, tahap berikutnya adalah penerapan layanan Firebase sesuai fungsi masing-masing pada aplikasi. Dalam penelitian ini, layanan yang digunakan meliputi Firebase Authentication, Firebase Realtime Database, dan Cloud Firestore. Firebase Authentication digunakan untuk proses login dan registrasi akun, Firebase Realtime Database digunakan untuk menyimpan data jenis galon, sedangkan Cloud Firestore digunakan untuk menyimpan data pengguna dan data pesanan. Pembagian ini dilakukan agar pengelolaan data pada aplikasi dapat berjalan lebih terstruktur sesuai kebutuhan fitur yang dikembangkan.

4.3.1 Firebase Authentication

Firebase Authentication digunakan untuk mengelola proses autentikasi pengguna pada aplikasi. Melalui layanan ini, pengguna dapat melakukan registrasi dan login menggunakan email, nomor *handphone* dan kata sandi. Setiap akun yang berhasil dibuat akan tercatat pada menu Authentication di Firebase Console dan memiliki identitas unik yang digunakan sistem pada proses autentikasi. Tampilan data akun pada Firebase Authentication dapat dilihat pada Gambar 4.41.

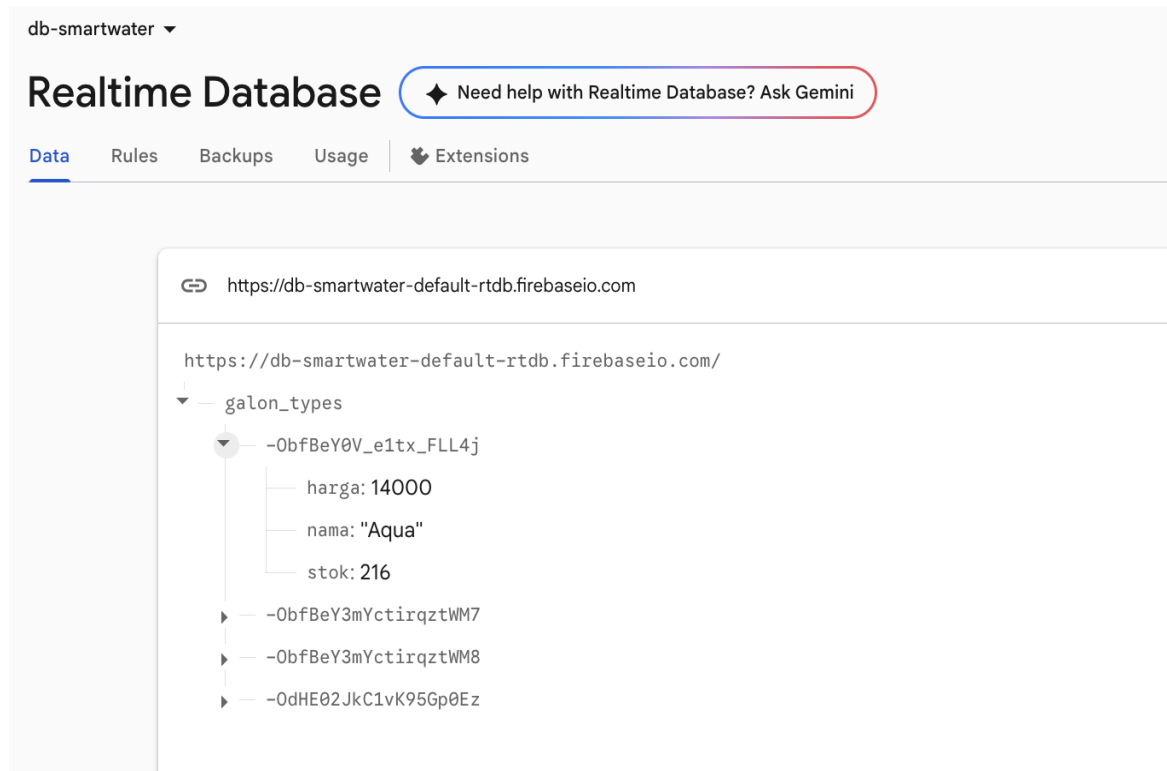


Gambar 4.41 Tampilan Firebase Authentication

4.3.2 Firebase Realtime Database

Firebase Realtime Database digunakan untuk menyimpan data jenis galon yang tersedia pada aplikasi. Data yang disimpan meliputi nama galon, harga, dan stok yang tersedia. Informasi tersebut digunakan sebagai acuan utama pada proses pemesanan, karena sistem akan menampilkan pilihan galon berdasarkan data yang tersimpan pada database. Penggunaan Realtime Database pada data galon memudahkan sistem dalam menampilkan perubahan data secara langsung, terutama ketika penjual menambah, mengubah, atau menghapus data jenis

galon. Struktur data jenis galon pada Firebase Realtime Database dapat dilihat pada Gambar 4.42.



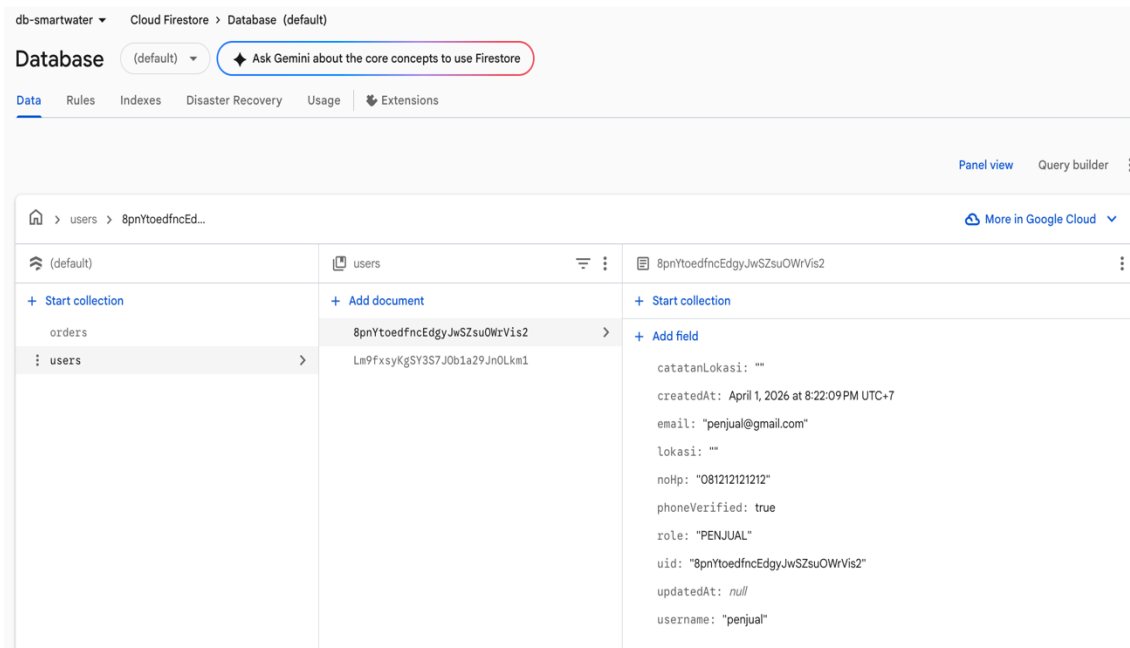
Gambar 4.42 Struktur Data Jenis Galon pada Firebase Realtime Database

4.3.3 Cloud Firestore

Cloud Firestore digunakan untuk menyimpan data pengguna dan data pesanan pada aplikasi. Layanan ini dipilih karena mendukung penyimpanan data dalam bentuk koleksi dan dokumen, sehingga pengelolaan data menjadi lebih terstruktur dan mudah disesuaikan dengan kebutuhan sistem. Pada penelitian ini, Cloud Firestore digunakan untuk menyimpan dua koleksi utama, yaitu *users* dan *orders*.

A. Struktur Data *Users*

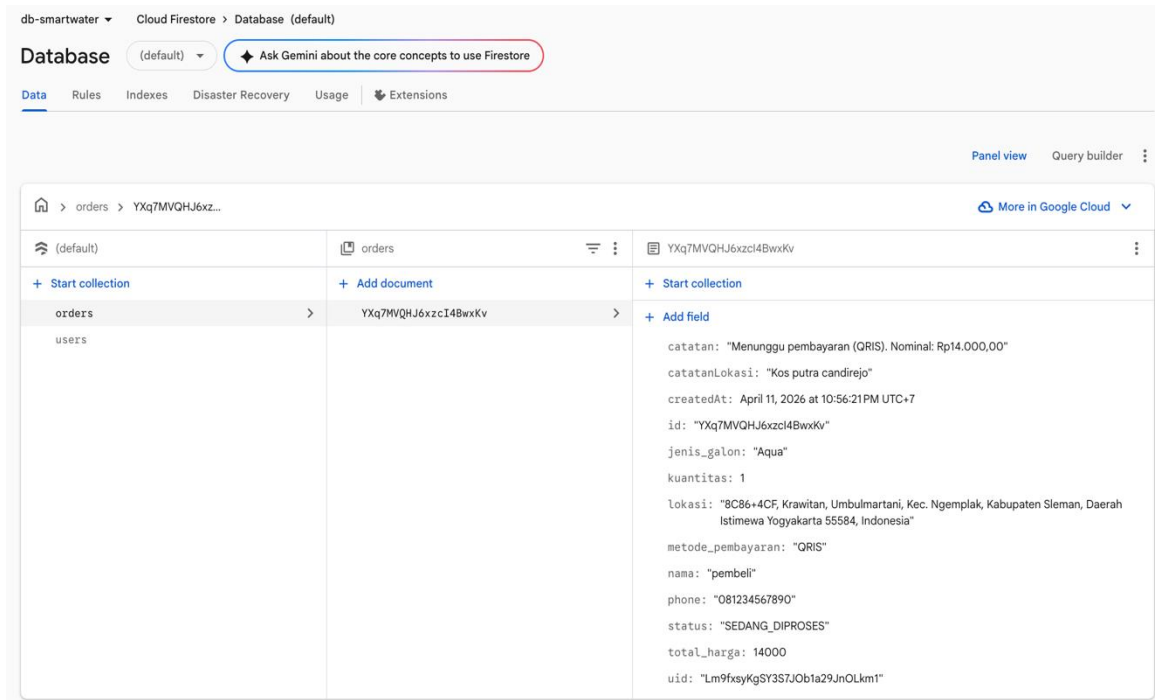
Koleksi *users* digunakan untuk menyimpan data pengguna yang telah terdaftar pada aplikasi. Data yang disimpan meliputi username, email, nomor handphone, lokasi, peran pengguna, dan waktu pembuatan akun. Setiap dokumen pengguna disimpan berdasarkan identitas unik tertentu, sehingga sistem dapat membedakan data masing-masing akun dengan jelas. Data pada koleksi ini digunakan untuk menampilkan informasi akun pengguna serta mendukung proses pemesanan galon. Struktur data pengguna pada Cloud Firestore dapat dilihat pada Gambar 4.43.



Gambar 4.43 Struktur Data *Users* pada Cloud Firestore

B. Struktur Data *Orders*

Koleksi *orders* digunakan untuk menyimpan seluruh data transaksi pemesanan galon yang dilakukan oleh pembeli. Data yang tersimpan meliputi identitas pesanan, nama pemesan, nomor handphone, jenis galon, jumlah pesanan, lokasi pengantaran, metode pembayaran, total harga, status pesanan, catatan tambahan, dan waktu pembuatan pesanan. Data ini digunakan untuk menampilkan riwayat pesanan pada sisi pembeli serta membantu penjual dalam memantau dan memperbarui status pesanan sesuai proses layanan yang berjalan. Struktur data pesanan pada Cloud Firestore dapat dilihat pada Gambar 4.44.



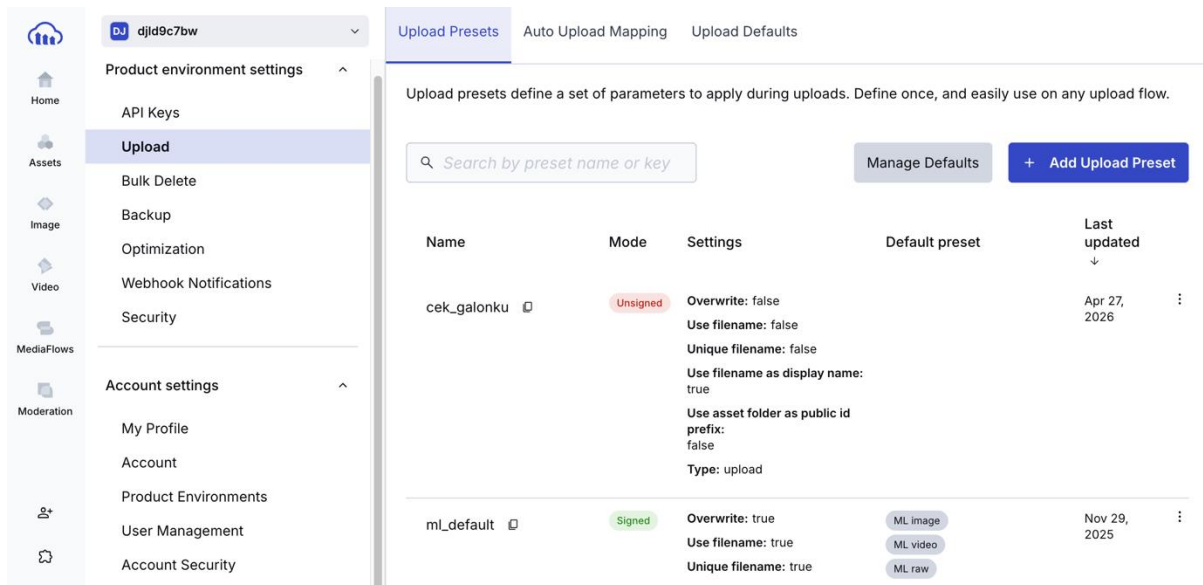
Gambar 4.44 Struktur Data Orders pada Cloud Firestore

4.4 Integrasi Layanan Cloudinary

Selain menggunakan layanan Firebase, aplikasi ini juga mengintegrasikan Cloudinary sebagai layanan manajemen media berbasis cloud. Cloudinary dimanfaatkan untuk menangani proses pengunggahan, penyimpanan, dan pengelolaan file gambar, khususnya foto bukti pengantaran yang diunggah oleh penjual setelah pesanan selesai diproses.

Pada prosesnya, ketika penjual menyelesaikan pesanan dan mengambil gambar melalui kamera perangkat, file gambar tersebut akan diunggah secara asinkron ke server Cloudinary melalui API yang telah dikonfigurasi sebelumnya. Setelah proses unggah berhasil, Cloudinary akan menghasilkan tautan URL gambar (secure URL) yang dapat diakses secara aman. URL tersebut kemudian disimpan ke dalam dokumen pesanan pada Cloud Firestore sebagai referensi data gambar.

Selanjutnya, pada sisi pembeli, sistem akan mengambil URL tersebut dan menampilkan gambar menggunakan library Glide, sehingga proses pemuatan gambar dapat dilakukan secara efisien tanpa harus menyimpan file secara langsung di dalam basis data aplikasi. Penggunaan Cloudinary pada penelitian ini bertujuan untuk mengoptimalkan pengelolaan media, mengurangi beban penyimpanan pada Firebase, serta meningkatkan performa aplikasi dalam menampilkan gambar.



Gambar 4.45 Konfigurasi Upload Preset pada Cloudinary

4.5 Pengujian Black box

Pengujian pada aplikasi dilakukan menggunakan metode black box testing, yaitu metode pengujian yang berfokus pada fungsi sistem berdasarkan masukan dan keluaran tanpa memperhatikan struktur kode program. Pengujian ini bertujuan untuk memastikan bahwa setiap fitur utama aplikasi dapat berjalan sesuai dengan kebutuhan yang telah dirancang. Skenario pengujian disusun untuk mencakup proses registrasi, login, pemesanan galon, pengelolaan riwayat pesanan, pengaturan akun dan lokasi, pengelolaan data galon, pengelolaan pesanan oleh penjual, perubahan kata sandi akun penjual, serta pengujian sistem notifikasi dan batas waktu pembayaran QRIS. Hasil pengujian black box disajikan padapada Tabel 4.1.

Tabel 4.1 Pengujian Black box

No	Fitur	Skenario Uji	Data Uji	Hasil yang Diharapkan	Hasil Pengujian
1	Registrasi	Registrasi akun baru dengan data valid	Username, email, <i>password</i> , no hp, Lokasi.	Data tersimpan dan pengguna dialihkan ke halaman login	Berhasil
		Registrasi dengan data tidak lengkap	Salah satu field wajib dikosongkan	Sistem menolak registrasi dan menampilkan pesan kesalahan	Berhasil

No	Fitur	Skenario Uji	Data Uji	Hasil yang Diharapkan	Hasil Pengujian
		Registrasi dengan konfirmasi password tidak sesuai	Password dan konfirmasi password berbeda	Sistem menolak registrasi dan menampilkan pesan kesalahan	Berhasil
2	Login	Login dengan <i>email</i> dan <i>password</i> benar	<i>Email</i> dan <i>password</i> sesuai data registrasi	Pengguna berhasil masuk dan diarahkan ke halaman home	Berhasil
		Login dengan email atau password salah	Email dan password tidak sesuai data registrasi	Sistem menolak login dan menampilkan pesan kesalahan	Berhasil
3	Home (Pembeli)	Menampilkan informasi sapaan dan riwayat pesanan yang sedang berjalan	Pengguna berhasil diarahkan ke halaman <i>home</i>	Halaman menampilkan sapaan dan Riwayat pesanan yang sedang berjalan	Berhasil
4	Pemesanan Galon	Pemesanan galon dengan data valid dan metode COD	Jenis galon, kuantitas, lokasi, catatan lokasi, metode pembayaran COD	Pesanan tersimpan dan ditampilkan pada riwayat pesanan dengan status awal sesuai proses sistem	Berhasil
		Pemesanan galon dengan data valid dan metode QRIS	Jenis galon, kuantitas, lokasi, catatan lokasi, metode pembayaran QRIS	Sistem menampilkan dialog pembayaran QRIS dan menyimpan pesanan sesuai status pembayaran	Berhasil
		Pemesanan dengan field wajib kosong	Kuantitas atau metode pembayaran tidak diisi	Sistem menolak pemesanan dan menampilkan pesan kesalahan	Berhasil
		Perhitungan total harga pesanan	Harga galon dan kuantitas diisi	Sistem menampilkan total harga sesuai hasil perhitungan	Berhasil
5	Riwayat Pesanan Pembeli	Menampilkan daftar pesanan	Pengguna membuka halaman Riwayat	Sistem menampilkan pesanan sesuai kelompok status	Berhasil

No	Fitur	Skenario Uji	Data Uji	Hasil yang Diharapkan	Hasil Pengujian
		diproses dan selesai			
		fitur hubungi penjual via WA	pengguna menggunakan fitur hubungi penjual via WA	aplikasi mengarahkan ke WA dengan nomor penjual yang sesuai	Berhasil
		Memilih metode pelunasan saat status Pembayaran Kurang	Pembeli menekan opsi "Lunasi via COD" pada dialog pilihan	Metode pembayaran otomatis berubah menjadi "QRIS & COD" dan catatan menjadi "Sisa Bayar COD"	Berhasil
		Melihat foto bukti pengantaran	Pembeli menekan <i>thumbnail</i> foto pada pesanan yang sudah Selesai	Sistem memunculkan dialog pop-up yang memperbesar foto bukti pengantaran	Berhasil
6	Akun Pembeli	Menampilkan data akun pembeli	Username, email, no hp, Lokasi, role	Username dan email tampil sesuai registrasi, role otomatis tampil sebagai pembeli	Berhasil
7	Pengaturan Lokasi Pembeli	Mengubah Lokasi secara manual	Menginputkan Lokasi baru	Sistem menyimpan Lokasi baru dan menampilkan data Lokasi terbaru,	Berhasil
		Menggunakan fitur "Pakai Lokasi Saya"	Pengguna menekan fitur "Pakai Lokasi Saya"	Alamat otomatis terisi berdasarkan koordinat Lokasi pengguna	Berhasil
8	Home (Penjual)	Menampilkan halaman utama penjual	Pengguna berhasil diarahkan ke halaman <i>home</i>	Halaman <i>home</i> penjual menampilkan sapaan dan daftar pesanan aktif	Berhasil
9	Pengaturan Jenis & Harga Galon (Penjual)	Menambah jenis galon baru	Penjual menekan ikon +, mengisi nama, harga, stok lalu simpan	Jenis galon baru muncul pada daftar dengan data nama, harga, dan stok yang sesuai	Berhasil
		Mengedit data jenis galon	Penjual mengedit harga atau stok kemudian simpan	Data jenis galon pada daftar terbaru sesuai perubahan	Berhasil
		Menghapus jenis galon	Penjual menghapus pada salah satu data	Data jenis galon tersebut hilang dari daftar	Berhasil

No	Fitur	Skenario Uji	Data Uji	Hasil yang Diharapkan	Hasil Pengujian
		Input harga dan stok dengan data tidak valid	Input huruf pada kolom harga atau stok	Sistem menolak input dan hanya menerima angka	Berhasil
10	Kelola Pesanan	Menampilkan daftar pesanan diproses dan selesai	Penjual membuka halaman Kelola pesanan	Seluruh pesanan tampil sesuai status masing-masing, dilengkapi nama pemesan, jenis galon, alamat, jumlah, dan total harga	Berhasil
		Mengubah status pesanan COD	Status diubah dari diproses menjadi dikirim/selesai/batal	Sistem memperbarui status pesanan pada sisi penjual dan pembeli	Berhasil
		Verifikasi pesanan QRIS	Penjual memilih terima pembayaran, pembayaran kurang, atau tolak pembayaran	Sistem memperbarui status dan catatan pesanan sesuai tindakan penjual	Berhasil
		Memasukkan jumlah pesanan melebihi stok galon yang tersedia	Jenis galon dipilih, kuantitas pesanan lebih besar dari stok tersedia	Sistem menolak pemesanan dan menampilkan pesan bahwa stok galon tidak mencukupi	Berhasil
		Mengunggah foto bukti pengantaran	Penjual mengunggah foto dari kamera saat mengubah status menjadi Selesai	Sistem berhasil mengunggah foto, menyimpannya di database, dan mengubah status pesanan menjadi Selesai	Berhasil
11	Akun penjual	Penjual mengubah <i>password</i> dengan data valid	<i>Password</i> lama benar, password baru, konfirmasi password baru sesuai	Sistem menyimpan password baru dan penjual dapat login menggunakan password baru	Berhasil

No	Fitur	Skenario Uji	Data Uji	Hasil yang Diharapkan	Hasil Pengujian
		Penjual mengubah password dengan password lama salah	Password lama salah, password baru, konfirmasi password baru sesuai	System menolak perubahan password dan menampilkan pesan kesalahan	Berhasil
		Penjual mengubah password dengan konfirmasi password baru tidak sama	Password lama benar, password baru, konfirmasi password baru tidak sama	System menolak perubahan password dan menampilkan pesan kesalahan	Berhasil
		Penjual mengubah password dengan field kosong	Salah satu kolom password lama, password baru, atau konfirmasi password dikosongkan	Sistem menolak perubahan password dan menampilkan pesan kesalahan	Berhasil
12	Notifikasi (Pembeli)	Menerima notifikasi pesanan COD dibuat	Pembeli menekan tombol Pesan pada metode COD	Sistem memunculkan notifikasi bahwa pesanan berhasil dibuat dan menunggu konfirmasi	Berhasil
		Menerima notifikasi untuk pembayaran QRIS	Pembeli menekan tombol "Tutup" pada dialog QRIS	Sistem memunculkan notifikasi untuk segera melakukan pembayaran	Berhasil
		Menerima notifikasi perubahan status/catatan dari penjual	Penjual mengubah status menjadi Dikirim, Selesai, atau memperbarui catatan pembayaran	Sistem secara <i>real-time</i> memunculkan notifikasi pop-up sesuai pembaruan yang dilakukan penjual	Berhasil
13	Notifikasi (Penjual)	Menerima notifikasi pesanan baru masuk (COD)	Pembeli menekan tombol Pesan dengan metode COD	Sistem memunculkan notifikasi pesanan baru masuk di sisi penjual	Berhasil

No	Fitur	Skenario Uji	Data Uji	Hasil yang Diharapkan	Hasil Pengujian
		Menerima notifikasi pesanan baru masuk (QRIS)	Pembeli menekan tombol "Saya sudah bayar" pada dialog QRIS	Sistem memunculkan notifikasi pesanan baru masuk di sisi penjual	Berhasil
14	Batas waktu pembayaran (QRIS)	Hitung mundur berjalan pada dialog dan item riwayat pesanan	Pembeli memilih metode QRIS sebagai metode pembayaran	Timer 15 menit berjalan secara <i>real-time</i> di sisi pembeli.	Berhasil
		Pembatalan pesanan secara otomatis saat batas waktu habis	Pembeli membiarkan waktu 15 menit habis tanpa menekan tombol "Saya sudah bayar"	Sistem otomatis membatalkan pesanan, dan memunculkan notifikasi pembatalan otomatis	Berhasil

4.6 Pengujian System Usability Scale (SUS)

Setelah pengujian menggunakan black box, aplikasi dinyatakan berjalan dengan baik, tahap selanjutnya adalah melakukan pengujian System Usability Scale (SUS) untuk mengukur tingkat kegunaan (*usability*) dari aplikasi yang telah dikembangkan. Pengujian usability ini dilakukan untuk mengetahui sejauh mana aplikasi pemesanan galon dapat digunakan dengan mudah, nyaman, dan sesuai dengan kebutuhan pengguna, khususnya dalam mendukung aktivitas konsumsi air sehari-hari.

System Usability Scale (SUS) merupakan metode pengujian usability yang menggunakan kuesioner dengan skala Likert satu sampai lima, yaitu 1 (sangat tidak setuju), 2 (tidak setuju), 3 (netral), 4 (setuju), dan 5 (sangat setuju). Skor yang diperoleh dari kuesioner kemudian diolah menggunakan ketentuan sebagai berikut:

- Setiap pernyataan bernomor ganjil, skor diperoleh dari nilai jawaban dikurangi satu.
- Setiap pernyataan bernomor genap, skor diperoleh dari lima dikurangi nilai jawaban.
- Seluruh skor dari pernyataan ganjil dan genap dijumlahkan, kemudian hasilnya dikalikan dengan 2,5.

- d. Skor akhir SUS diperoleh dari nilai rata-rata dengan membagi total skor seluruh responden dengan jumlah responden.

Pengujian ini melibatkan 15 responden yang terdiri dari 14 pengguna galon dan 1 penjual galon. Setiap responden diminta untuk menggunakan aplikasi, kemudian mengisi kuesioner SUS berdasarkan pengalaman penggunaan yang dirasakan.

Berdasarkan hasil pengujian yang dilakukan, diperoleh nilai rata-rata SUS yang menunjukkan bahwa aplikasi memiliki tingkat *usability* yang baik dan dapat diterima oleh pengguna. Hasil pengujian menggunakan SUS kemudian diolah dalam bentuk tabel, dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian *System Usability Scale*

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
R1	3	2	4	2	3	3	4	1	4	3
R2	4	2	5	1	5	2	5	2	4	1
R3	2	4	1	3	2	3	2	4	2	3
R4	5	3	4	2	5	3	5	1	4	3
R5	3	3	4	2	5	1	3	3	4	2
R6	4	2	4	2	4	3	3	2	4	5
R7	4	2	4	1	5	2	4	2	5	2
R8	4	2	5	4	5	2	5	2	4	4
R9	5	1	5	1	5	1	5	1	5	1
R10	4	2	5	2	4	2	5	1	4	2
R11	4	2	4	2	4	2	5	1	5	1
R12	4	1	4	2	4	2	4	1	5	2
R13	5	3	4	3	5	4	5	4	5	4
R14	5	2	4	3	4	1	5	3	4	2
R15	5	2	4	2	5	1	5	1	5	2

Hasil dari penilaian seluruh responden didapatkan dari perhitungan dengan aturan SUS yaitu pertanyaan dengan nomor ganjil nilai yang di dapat dikurangi 1, dan untuk pertanyaan nomor genap adalah 5 dikurangi nilai yang di dapat. Untuk hasil rata rata setiap pertanyaan dapat dilihat pada Tabel 4.3.

Tabel 4.3 Rata-rata skor per pertanyaan

Pertanyaan	Total Skor	Rata-rata
P1	61	4,07
P2	33	2,20
P3	61	4,07
P4	32	2,13
P5	65	4,33
P6	32	2,13
P7	65	4,33
P8	29	1,93
P9	64	4,27
P10	37	2,47

Berdasarkan Tabel 4.3, pertanyaan P5 dan P7 memperoleh nilai rata-rata tertinggi yaitu sebesar 4,33. Hal ini menunjukkan bahwa responden merasa fitur-fitur yang tersedia pada aplikasi dapat digunakan dengan baik dan mendukung kebutuhan pengguna. Sementara itu, pertanyaan P8 memperoleh nilai rata-rata terendah sebesar 1,93. Karena P8 merupakan pernyataan negatif dalam SUS, nilai yang rendah menunjukkan bahwa sebagian besar responden tidak mengalami kesulitan dalam menggunakan aplikasi. Secara keseluruhan, nilai rata-rata pada pertanyaan ganjil cenderung tinggi dan pada pertanyaan genap cenderung rendah, yang mengindikasikan tingkat usability aplikasi yang baik. Untuk hasil rekapitulasi akhirnya dapat dilihat pada Tabel 4.4.

Tabel 4.4 Rekapitulasi Penilaian

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Total Skor	Total Nilai SUS
R1	2	3	3	3	2	2	3	4	3	2	27	67,5
R2	3	3	4	4	4	3	4	3	3	4	35	87,5
R3	1	1	0	2	1	2	1	1	1	2	12	30
R4	4	2	3	3	4	2	4	4	3	2	31	77,5
R5	2	2	3	3	4	4	2	2	3	3	28	70
R6	3	3	3	3	3	2	2	3	3	0	25	62,5
R7	3	3	3	4	4	3	3	3	4	3	33	82,5
R8	3	3	3	1	4	3	4	3	3	1	28	70

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Total Skor	Total Nilai SUS
R9	4	4	4	4	4	4	4	4	4	4	40	100
R10	3	3	4	3	3	3	4	4	3	3	33	82,5
R11	3	3	3	3	3	3	4	4	4	4	34	85
R12	3	4	3	3	3	3	3	4	4	3	33	82,5
R13	4	2	3	2	4	1	4	1	4	1	26	65
R14	4	3	3	2	3	4	4	2	3	3	31	77,5
R15	4	3	3	3	4	4	4	4	4	3	36	87,5
Skor Rata-rata												75,2

Berdasarkan hasil pengujian Black box(SUS), aplikasi memperoleh nilai rata-rata sebesar 75,2. Nilai tersebut berada pada kategori “*Good*” dan termasuk dalam *Acceptable Range*, yang menunjukkan bahwa aplikasi memiliki tingkat kegunaan (*usability*) yang baik dan dapat diterima oleh pengguna. Secara umum, skor SUS diatas 68 dianggap berada di atas nilai rata-rata. Dengan demikian, nilai 75,2 menandakan bahwa aplikasi pemesanan galon mudah digunakan, fitur-fitur utama dapat dipahami dan dioperasikan dengan baik oleh pengguna.

4.7 Analisis Kelebihan dan Kekurangan Aplikasi

Berdasarkan hasil perancangan, implementasi sistem, serta pengujian fungsional dan usability yang telah dilakukan, aplikasi pemesanan galon memiliki beberapa kelebihan dan kekurangan. Analisis ini dilakukan untuk melihat sejauh mana aplikasi telah memenuhi kebutuhan pengguna, sekaligus mengidentifikasi bagian yang masih dapat dikembangkan pada penelitian selanjutnya.

1. Kelebihan Aplikasi

A. Proses pemesanan galon menjadi lebih terstruktur

Aplikasi pemesanan galon mempermudah pembeli dalam melakukan pemesanan galon tanpa harus menghubungi penjual secara manual melalui pesan singkat atau telepon. Seluruh data penting seperti jenis galon, jumlah pesanan, lokasi pengantaran, catatan lokasi, dan metode pembayaran dapat diinput langsung melalui aplikasi sehingga proses pemesanan menjadi lebih tertata dan terdokumentasi.

B. Riwayat pesanan dan status transaksi dapat dipantau dengan lebih jelas

Aplikasi menyediakan fitur riwayat pesanan yang dikelompokkan berdasarkan status diproses dan selesai. Melalui fitur ini, pembeli dapat mengetahui perkembangan

pesanan secara langsung, sedangkan penjual dapat memantau dan mengelola pesanan dengan lebih mudah. Hal ini membantu mengurangi ketidakpastian informasi yang sebelumnya sering terjadi pada proses pemesanan manual.

C. Mendukung dua metode pembayaran sesuai kebutuhan layanan

Aplikasi telah menyediakan metode pembayaran COD dan QRIS. Kehadiran dua pilihan pembayaran ini memberi fleksibilitas kepada pembeli dalam menyesuaikan cara transaksi dengan kebiasaan masing-masing. Pada sisi penjual, sistem juga mendukung proses verifikasi pembayaran QRIS sebelum pesanan dilanjutkan ke tahap pengiriman.

D. Memiliki fitur batas waktu pembayaran otomatis (QRIS)

Pada metode pembayaran QRIS, aplikasi dilengkapi dengan fitur hitung mundur otomatis selama 15 menit sebagai batas waktu pembayaran. Fitur ini bertujuan untuk memastikan bahwa pesanan yang belum dibayar tidak tersimpan terlalu lama di dalam sistem. Apabila hingga batas waktu tersebut pembeli masih belum menyelesaikan pembayaran, maka pesanan akan dibatalkan secara otomatis.

E. Pengelolaan data galon lebih mudah dilakukan oleh penjual

Penjual dapat menambah, mengubah, dan menghapus data jenis galon beserta harga dan stok melalui aplikasi. Fitur ini membantu penjual dalam menjaga kesesuaian informasi produk yang ditampilkan kepada pembeli, sehingga proses operasional menjadi lebih rapi dan terkontrol.

F. Aplikasi mudah digunakan oleh pengguna

Berdasarkan hasil pengujian System Usability Scale (SUS), aplikasi memperoleh nilai rata-rata sebesar 75,2 yang termasuk dalam kategori *Good* dan berada pada *acceptable range*. Hasil ini menunjukkan bahwa aplikasi cukup mudah dipahami, digunakan, dan dapat diterima oleh pengguna dalam mendukung aktivitas pemesanan galon.

G. Fleksibilitas Pelunasan (*Split Payment*)

Sistem mampu menangani kasus pembayaran yang tidak penuh (Pembayaran Kurang) dengan cerdas. Pembeli diberikan fleksibilitas untuk melunasi sisa tagihan melalui QRIS kembali atau mengubah sisa tagihan menjadi metode COD saat galon diantar. Hal ini meminimalisir pembatalan transaksi akibat kesalahan transfer.

H. Transparansi Bukti Pengantaran

Aplikasi mendukung fitur dokumentasi berbasis foto. Penjual dapat melampirkan foto bukti pengantaran yang nantinya dapat dilihat langsung oleh pembeli pada riwayat

pesanan yang telah selesai. Fitur ini meningkatkan kepercayaan dan transparansi antara depot dan pelanggan.

2. Kekurangan Aplikasi

A. Notifikasi belum berjalan secara optimal

Fitur notifikasi pada aplikasi saat ini belum sepenuhnya dapat menampilkan pemberitahuan secara langsung ketika terjadi perubahan data penting di dalam sistem. Hal ini disebabkan karena mekanisme notifikasi belum menggunakan Firebase Cloud Messaging (FCM), melainkan masih mengandalkan kombinasi WorkManager dan AlarmManager pada Android. Pendekatan ini membuat aplikasi perlu melakukan pengecekan data secara berkala terlebih dahulu sebelum menampilkan notifikasi. Meskipun notifikasi dapat muncul secara langsung (*real-time*) ketika aplikasi sedang dibuka di layar (*foreground*) karena memanfaatkan SnapshotListener pada Firestore, performa notifikasi mengalami penurunan ketika aplikasi berada dalam keadaan tertutup atau berjalan di latar belakang (*background*). Akibatnya, pemberitahuan kepada pembeli maupun penjual tidak selalu muncul secara cepat dan pada beberapa kondisi memerlukan jeda waktu tertentu.

B. Nomor *handphone* pengguna tidak dapat diubah setelah registrasi

Pada aplikasi saat ini, data nomor *handphone* yang telah didaftarkan belum dapat diperbarui kembali melalui halaman akun. Keterbatasan ini dapat menjadi kendala ketika pembeli ingin melakukan pemesanan untuk orang lain atau ketika nomor yang digunakan perlu diganti.

C. Verifikasi OTP masih menggunakan nomor uji dari Firebase Authentication

Fitur OTP pada aplikasi masih menggunakan *Phone numbers for testing* yang disediakan oleh Firebase Authentication, sehingga kode OTP yang digunakan belum bersifat dinamis seperti pada sistem produksi. Kondisi ini dipilih karena implementasi OTP dinamis memerlukan aktivasi skema penagihan *billing plan* pada Firebase dengan model “*pay as you go*”. Pada tahap penelitian, pendekatan tersebut belum digunakan karena dikhawatirkan menimbulkan biaya tambahan yang sulit diprediksi.

D. Pembayaran QRIS belum sepenuhnya dinamis

Metode pembayaran QRIS yang digunakan dalam aplikasi saat ini masih memanfaatkan kode QR statis milik depot. Dengan demikian, nominal pembayaran belum dapat terbentuk secara otomatis berdasarkan total harga pesanan seperti pada

sistem pembayaran yang sudah terintegrasi penuh. Agar QRIS dapat berjalan secara dinamis, diperlukan layanan pihak ketiga atau penyedia payment gateway yang umumnya berbayar. Karena pertimbangan biaya implementasi, penelitian ini masih menggunakan QRIS statis sebagai bentuk solusi yang lebih sederhana.

4.8 Pembahasan

Berdasarkan hasil pengembangan dan pengujian yang telah dilakukan, aplikasi menunjukkan bahwa kebutuhan utama pengguna yang diperoleh pada tahap analisis dapat direalisasikan ke dalam bentuk sistem yang berjalan secara fungsional. Permasalahan yang sebelumnya muncul pada proses pemesanan manual, seperti pencatatan pesanan yang tidak terstruktur, keterlambatan informasi status pesanan, serta kesulitan penjual dalam mengelola data pesanan, dapat dikurangi melalui penggunaan aplikasi ini. Pembeli tidak lagi harus bergantung sepenuhnya pada komunikasi melalui WhatsApp atau telepon, karena proses pemesanan, pemantauan riwayat, dan pengecekan status pesanan telah tersedia dalam satu sistem yang terpusat.

Dari sisi implementasi, aplikasi telah berhasil menyediakan alur layanan, mulai dari registrasi dan login pengguna, pengelolaan data akun, pemesanan galon, pemilihan metode pembayaran, hingga pengelolaan pesanan oleh penjual. Pada metode COD, sistem memungkinkan pesanan langsung diteruskan ke penjual untuk diproses lebih lanjut karena pembayaran dilakukan saat pengantaran. Sementara itu, pada metode QRIS, sistem menerapkan alur yang lebih adaptif melalui fitur batas waktu 15 menit serta proses verifikasi bertahap oleh penjual. Perbedaan alur ini menunjukkan bahwa aplikasi tidak hanya berfungsi sebagai media pencatatan pesanan, tetapi juga telah mengakomodasi logika bisnis dan mitigasi risiko transaksi (seperti pembatalan otomatis pesanan yang tidak dibayar) sesuai kondisi operasional di lapangan.

Lebih dari itu, aplikasi juga telah mampu mengakomodasi dinamika operasional lapangan yang lebih kompleks. Salah satunya adalah penanganan masalah pembayaran non-tunai (QRIS) yang tidak mencukupi melalui mekanisme pelunasan ganda (split payment). Sistem memungkinkan pembeli yang mengalami kekurangan transfer untuk tidak perlu membatalkan pesanan dari awal, melainkan mengonversi sisa tagihannya menjadi metode

pelunasan di tempat (COD). Fleksibilitas ini secara signifikan mengurangi potensi transaksi gagal (churn rate) dan menyederhanakan alur komunikasi antara depot dan pelanggan.

Selain pengelolaan transaksi, aplikasi ini juga memanfaatkan integrasi Cloudinary untuk menyimpan dokumentasi berupa foto bukti pengantaran. Pada tahap akhir layanan, penjual diminta untuk melampirkan foto sesaat sebelum menyelesaikan pesanan. Langkah ini ditujukan untuk memberikan kejelasan informasi kepada pembeli bahwa pesanan telah diantar, sekaligus meminimalkan potensi kesalahpahaman antara kedua belah pihak. Dari segi teknis, penyimpanan foto dikelola dengan menyimpan tautan (URL) gambar ke dalam Firestore. Pendekatan ini dipilih agar beban penyimpanan pada database utama tetap ringan sehingga performa aplikasi dapat terjaga dengan baik. Dengan adanya penambahan fitur-fitur tersebut, aplikasi ini diharapkan tidak hanya sebatas menjadi media pencatat pesanan, tetapi juga dapat membantu kelancaran operasional depot sehari-hari.

Dari sisi fungsional aplikasi, hasil pengujian black box menunjukkan bahwa fitur-fitur utama aplikasi telah berjalan sesuai dengan skenario yang dirancang. Proses registrasi, login, pemesanan galon, pengelolaan akun dan lokasi, pengelolaan data galon, perubahan status pesanan, operasional timer QRIS, mekanisme split payment, unggah foto bukti pengantaran, hingga sistem notifikasi dapat dijalankan dengan baik. Validasi terhadap kondisi tertentu, seperti data login yang salah, *field* yang kosong, input harga dan stok yang tidak valid, serta jumlah pesanan yang melebihi stok tersedia, juga menunjukkan hasil sesuai harapan. Dengan demikian, aplikasi telah memenuhi kebutuhan fungsional dasar yang diidentifikasi pada tahap analisis kebutuhan.

Dari sisi *usability*, nilai SUS sebesar 75,2 menunjukkan bahwa aplikasi berada pada kategori *Good* dan dapat diterima oleh pengguna. Hasil ini menunjukkan bahwa antarmuka dan alur penggunaan aplikasi sudah cukup mudah dipahami oleh responden. Dengan kata lain, aplikasi telah mampu memberikan pengalaman penggunaan yang relatif baik bagi pembeli maupun penjual dalam menjalankan aktivitas pemesanan dan pengelolaan pesanan.

Meskipun demikian, hasil implementasi juga menunjukkan bahwa aplikasi masih memiliki beberapa keterbatasan. Salah satu keterbatasan utama terletak pada mekanisme notifikasi aplikasi yang belum dapat berjalan secara *real-time* pada semua kondisi. Pada penelitian ini, sistem notifikasi tidak menggunakan Firebase Cloud Messaging (FCM), tetapi memanfaatkan kombinasi WorkManager dan AlarmManager. Pendekatan ini dipilih karena lebih memungkinkan untuk diterapkan pada tahap pengembangan. Namun, konsekuensinya adalah sistem harus menunggu proses pengecekan data secara berkala terlebih dahulu sebelum

notifikasi dapat ditampilkan. Ketika aplikasi sedang aktif dibuka oleh pengguna (*foreground*), notifikasi dapat tampil secara langsung karena memanfaatkan integrasi Firestore SnapshotListener. Akan tetapi, ketika aplikasi berada dalam keadaan tertutup atau berjalan di latar belakang (*background*), sistem mengandalkan WorkManager yang mengikuti interval minimum penjadwalan pada sistem operasi Android, yaitu sekitar 15 menit. Akibatnya, notifikasi tidak selalu muncul secara langsung ketika terjadi perubahan data. Kondisi inilah yang menyebabkan notifikasi pada aplikasi belum sepenuhnya responsif.

Keterbatasan lain juga terlihat pada fitur verifikasi OTP. Pada tahap implementasi, OTP masih menggunakan *Phone numbers for testing* yang disediakan oleh Firebase Authentication, sehingga kode OTP yang digunakan belum benar-benar dikirim secara dinamis ke nomor pengguna seperti pada aplikasi produksi. Penggunaan nomor uji ini dipilih agar proses registrasi dan pengujian tetap dapat dilakukan dalam ruang lingkup penelitian tanpa harus bergantung pada layanan OTP aktual. Hal ini juga berkaitan dengan pertimbangan biaya, karena implementasi OTP dinamis memerlukan aktivasi *billing plan* Firebase dengan skema “*pay as you go*”. Oleh karena itu, penggunaan *Phone numbers for testing* menjadi solusi sementara yang lebih aman dan terkendali selama proses pengembangan, meskipun konsekuensinya fitur OTP pada aplikasi belum sepenuhnya merepresentasikan kondisi implementasi nyata.

Selain itu, pembayaran QRIS pada aplikasi juga masih bersifat statis. Sistem belum dapat menghasilkan nominal QRIS yang otomatis menyesuaikan total harga pesanan, karena implementasi QRIS dinamis umumnya memerlukan integrasi dengan layanan pihak ketiga atau *payment gateway* yang bersifat berbayar. Oleh karena itu, penelitian ini masih menggunakan QRIS statis milik depot sebagai bentuk implementasi yang lebih sederhana dan ekonomis. Meskipun belum seotomatis sistem pembayaran digital penuh, pendekatan ini tetap dapat menggambarkan alur pembayaran non-tunai yang digunakan pada aplikasi.

Pada sisi data pengguna, nomor handphone juga belum dapat diperbarui kembali setelah registrasi. Keterbatasan ini menunjukkan bahwa sistem manajemen akun pada aplikasi masih dapat dikembangkan lebih lanjut, terutama agar lebih fleksibel dalam mendukung kondisi nyata pengguna, misalnya ketika pembeli ingin melakukan pemesanan untuk orang lain atau saat terjadi perubahan nomor *handphone*.

Secara keseluruhan, hasil penelitian ini menunjukkan bahwa aplikasi telah mampu menjawab kebutuhan dasar layanan pemesanan galon berbasis Android, baik dari sisi pembeli maupun penjual. Aplikasi ini berhasil menghadirkan proses pemesanan yang lebih terstruktur,

riwayat transaksi yang lebih jelas, serta pengelolaan pesanan yang lebih tertib dibandingkan proses manual yang digunakan sebelumnya. Adapun beberapa kekurangan yang masih ada, seperti notifikasi yang belum *real-time*, OTP yang masih menggunakan nomor uji, QRIS yang masih statis, dan keterbatasan pembaruan data akun, dapat menjadi dasar pengembangan lebih lanjut pada penelitian berikutnya agar aplikasi menjadi lebih matang dan siap digunakan secara lebih luas.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, dapat disimpulkan bahwa aplikasi pemesanan galon berhasil dikembangkan sebagai aplikasi berbasis Android yang mampu membantu pengguna dalam memantau kondisi galon air serta melakukan pemesanan galon secara lebih praktis. Aplikasi ini menyediakan fitur utama berupa autentikasi pengguna, pemantauan kondisi galon, pemesanan galon, riwayat pesanan, pengaturan lokasi pengantaran, serta notifikasi status pesanan.

Pengembangan aplikasi dilakukan menggunakan metode Waterfall, yang terdiri dari tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Penerapan metode Waterfall membantu proses pengembangan berjalan secara terstruktur dan sistematis, sehingga setiap tahapan dapat diselesaikan dengan baik sebelum melanjutkan ke tahap berikutnya. Metode ini dinilai sesuai untuk pengembangan aplikasi karena kebutuhan sistem telah didefinisikan dengan jelas sejak awal.

Berdasarkan hasil pengujian Black Box, seluruh fitur utama aplikasi dapat berjalan sesuai dengan skenario yang diuji, mulai dari proses login dan registrasi, pemantauan kondisi galon, pemesanan, mekanisme *split payment* dan unggah foto pengantaran, hingga pengelolaan status pesanan beserta notifikasinya. Selain itu, hasil evaluasi menggunakan System Usability Scale (SUS) memperoleh nilai rata-rata 75,2 yang termasuk dalam kategori *Good*. Hasil ini menunjukkan bahwa aplikasi mudah dipahami, nyaman digunakan, dan dapat diterima dengan baik oleh pengguna. Dengan demikian, secara keseluruhan aplikasi ini dinilai layak untuk diimplementasikan sebagai solusi digital pendukung pengelolaan dan pemesanan galon air.

5.2 Saran

Berdasarkan hasil pengembangan dan evaluasi aplikasi pemesanan galon, terdapat beberapa saran yang dapat dijadikan bahan pengembangan lebih lanjut, yaitu sebagai berikut:

- A. Sistem notifikasi dapat dikembangkan agar berjalan secara lebih cepat dan *real-time*, misalnya dengan memanfaatkan layanan Firebase Cloud Messaging (FCM), sehingga pembeli dan penjual dapat memperoleh pemberitahuan pesanan secara langsung tanpa harus menunggu pengecekan berkala.

- B. Fitur akun pengguna dapat dikembangkan dengan menambahkan kemampuan untuk mengubah nomor handphone setelah proses registrasi, sehingga aplikasi menjadi lebih fleksibel ketika pengguna ingin melakukan pemesanan untuk orang lain atau saat nomor aktif yang digunakan telah berubah.
- C. Fitur OTP dapat dikembangkan menggunakan mekanisme verifikasi nomor handphone yang dinamis agar lebih mendekati implementasi pada kondisi penggunaan nyata. Pengembangan ini dapat dilakukan apabila penelitian selanjutnya telah mempertimbangkan kesiapan penggunaan layanan Firebase Authentication dengan dukungan *billing plan* yang sesuai atau dengan menggunakan metode lain yang serupa.
- D. Metode pembayaran QRIS dapat ditingkatkan menjadi lebih dinamis melalui integrasi dengan layanan pembayaran *digital* atau *payment gateway*, sehingga nominal pembayaran dapat terbentuk secara otomatis sesuai total harga pesanan tanpa perlu penyesuaian manual.

DAFTAR PUSTAKA

- Abadi, D. A., Saputra, L. K. P., & Virginia, G. (2021). Smart Water Dispenser Terintegrasi untuk Monitoring Konsumsi Air Minum Harian. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 5(2), 705. <https://doi.org/10.30865/mib.v5i2.2950>
- Agustina, A. C. (2021). Analisis Cemarkan Coliform dan Identifikasi *Escherichia coli* dari Depo Air Minum Isi Ulang di Kota Semarang. *Life Science*, 10(1). <http://journal.unnes.ac.id/sju/index.php/LifeSci>
- Andrianto, R., & Munandar, M. H. (2022). Aplikasi e-commerce penjualan pakaian berbasis Android menggunakan Firebase Realtime Database. *Journal Computer Science and Information Technology(JCoInT) Program Studi Teknologi Informasi*, 3(1), 20–29. <http://jurnal.ulb.ac.id/index.php/JCoInT/index>
- Ansori, S., Hendradi, P., & Nugroho, S. (2023). Penerapan Metode Design Thinking dalam Perancangan UI/UX Aplikasi Mobile SIPROPMAWA. *Journal of Information System Research (JOSH)*, 4(4), 1072–1081. <https://doi.org/10.47065/josh.v4i4.3648>
- Chougale, P., Yadav, V., & Gaikwad, A. Dr. (2021). Firebase-Overview And Usage. *International Research Journal of Modernization in Engineering Technology and Science*, 3(12). www.irjmets.com
- Devansyah, S., Samala, A. D., Zaus, M. A., & Putra, R. (2025). Rancang Bangun Sistem Informasi Pemesanan Air Minum Isi Ulang Berbasis Framework Laravel. *Jurnal Pendidikan Tambusai*, 9(2).
- Dey, S., Jaiswal, S., & Hemasabari, E. (2025). Development of a Full-Stack Social Media Application Using Spring Boot, React.js, and Cloudinary Multiauthor. Dalam *International Journal of Scientific Research & Engineering Trends* (Vol. 11, Nomor 6).
- Dyayu, A. L., Beny, & Yani, H. (2023). Evaluasi Usability Aplikasi PeduliLindungi Menggunakan Metode Usability Testing dan System Usability Scale (SUS). *Jurnal Manajemen Teknologi dan Sistem Informasi (JMS)*, 3(1). <http://ejournal.unama.ac.id/index.php/jms>
- Fiki, S., Rahmayuni, I., & Prayama, D. (2021). Aplikasi Pemesanan Air Galon Online Dengan Fitur Tracking Posisi Pengantar Galon Berbasis Android. *Jurnal Ilmiah Teknologi Sistem Informasi*, 2(1).
- Firdaus, A. A., Munir, M., & Anisyah, A. (2025). Rancang Bangun Aplikasi Pemesanan Air Isi Ulang Berbasis Android dengan Integrasi Firebase dan Location Based Service di

- Depot Bumi Qta. *Digital Transformation Technology*, 5(1), 443–454.
<https://doi.org/10.47709/digitech.v5i1.6567>
- Hafsari, R., Aribi, E., & Maulana, N. (2023). PERANCANGAN SISTEM INFORMASI MANAJEMEN INVENTORI DAN PENJUALAN PADA PERUSAHAAN PT. INHUTANI V. *Jurnal PROSISKO*, 10(2).
- Hamdi, S., & Fadli, S. (2025). Penerapan Metode Waterfall dalam Pengembangan Website Usaha Mikro, Kecil dan Menengah (Studi Kasus: UMKM Pengrajin Pandan Lanji Bersatu). *Jurnal Mahasiswa Teknik Informatika*, 9(3).
- Ichsandi, Yanto, W., Alhaq, H., Syofiana Sari, R., & Juanda, M. (2025). Impression: Jurnal Teknologi dan Informasi Implementasi UML dalam Desain Sistem Informasi Program Studi SI di Universitas Merangin. *Jurnal Teknologi dan Informasi*, 4(2).
- Ijudin, A., & Saifudin, A. (2020). Pengujian Black Box pada Aplikasi Berita Online dengan Menggunakan Metode Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 5(1). <http://openjournal.unpam.ac.id/index.php/informatika>
- Insani, R., Nasrullah, M., Istyanto, N. P., Arivani, T. D., & Tesalonika, T. A. (2022). Digitalisasi Pemesanan Air Isi Ulang pada UD. Depo Bahagia. *JMM (Jurnal Masyarakat Mandiri)*, 6(6), 5077. <https://doi.org/10.31764/jmm.v6i6.11020>
- Kusumawardani, S., & Larasati, A. (2020). Analisis konsumsi air putih terhadap konsentrasi. *HOLISTIKA*, 4(2).
- Mahatmi, M. F., Hasanuddin, T., & Umar, F. (2022). Implementasi Metode Haversine Formula untuk Menentukan Jarak Terdekat pada Pengantaran Air Galon Depot Anantama Berbasis Android. *Buletin Sistem Informasi dan Teknologi Islam*, 3(1), 69–78.
- Maulana, F. S., Wibowo, F., Badharudin, A. Y., & Wicaksono, A. P. (2025). Analisis Usability Website Sistem Informasi Perpustakaan dengan System Usability Scale (SUS) dan Heuristic Evaluation (HE). *Journal of Information System Research (JOSH)*, 7(1), 1–11. <https://doi.org/10.47065/josh.v7i1.8179>
- Maulana, I., & Hapsari, D. R. (2024). Pengawasan Mutu Proses Produksi Air Minum Dalam Kemasan Galon Di PT. XYZ. *Karimah Tauhid*, 3(11).
- Narulita, S., Nugroho, A., & Abdillah, M. Z. (2024). Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS). *Bridge: Jurnal publikasi Sistem Informasi dan Telekomunikasi*, 2(3), 244–256. <https://doi.org/10.62951/bridge.v2i3.174>

- Nasution, M. F. A., Suendri, & Harahap, A. M. (2023). Customer Service Information System Using Dynamic Priority Scheduling Algorithm At PT Sumatra Sistem Integrasi Corresponding Author. Dalam *Journal of Information System and Technology Research journal homepage* (Vol. 2, Nomor 1). <http://creativecommons.org/licenses/by-sa/4.0/>
- Nurlaila, D., Mutamassikin, & Susanti, T. (2022). Comparative analysis of Kotlin and Java programming language in memory usage. *JISCO(Journal of Information System and Computing) ISSN, 1*(1). <http://fstuinjambi.org/jurnalSI/index.php/SI/index>
- Pratama, S. D., & Dadaprawira, M. N. (2023). Pengujian Black Box Testing Pada Aplikasi Edu Digital Berbasis Website Menggunakan Metode Equivalence Dan Boundary Value. *Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD, 6*(2), 560–569. <https://ojs.trigunadharma.ac.id/index.php/jsk/index>
- Prayoga, E. I., & Kristiana, T. (2024). Evaluasi Usability pada Aplikasi HRMWincorp Menggunakan Metode System Usability Scale (SUS). *Jurnal Informatika dan Teknik Elektro Terapan, 12*(2). <https://doi.org/10.23960/jitet.v12i2.4094>
- Pressman, R. S. (2015). *Software engineering: A practitioner's approach*. (8th ed.). McGraw-Hill Education.
- Pricillia, T., & Zulfachmi. (2021). Survey Paper: Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, RAD). *Bangkit Indonesia, 10*(1).
- Putra, D. M. D. U., Kusuma, A. S., Willdahlia, A. G., & Pande, N. K. N. N. (2024). Evaluasi Usability E-Modul Basis Data Menggunakan Metode System Usability Scale (SUS). *Jurnal Ilmiah Global Education, 5*(2), 1800–1809. <https://doi.org/10.55681/jige.v5i2.2764>
- Ratnawati, F., Azren, M., & Tedyyana, A. (2019). Aplikasi Pembelian Air Minum Isi Ulang Berbasis Android. *Jurnal Teknologi Informasi & Komunikasi Digital Zone, 10*(1), 88. <https://doi.org/10.31849/digitalzone.v10i1>
- Rivdyho, M. A., & Bahri, S. (2022). Analisis Usability Fitur Rating pada Aplikasi Ladder Menggunakan Metode System Usability Scale. *JTRISTE, 9*(2), 12–21.
- Sariangshah, husin. (2024). Perancangan Sistem Pendukung Keputusan dalam Menentukan Menu Makanan untuk Penderita Batu Ginjal. *Jurnal Unitek, 17*(1).
- Sibuea, S., Saputro, M. I., Annan, A., & Widodo, Y. B. (2022). Aplikasi mobile collection berbasis Android pada PT. Suzuki Finance Indonesia. *JURNAL INFORMATIKA DAN TEKNOLOGI KOMPUTER (JITEK)), 2*(1).

- Sugiyatno. (2023). Pengiriman Informasi Real Time Menggunakan Teknologi Database Firebase pada Aplikasi Mobile Android Abstrak (Bahasa Indonesia). Dalam *Bisnis dan Manajemen* (Vol. 21, Nomor 2).
- Sukma, A. P., Yusuf, R., & Dai, R. H. (2023). Analisis pengukuran usability Sistem Informasi Manajemen BAZNAS (SIMBA) menggunakan metode System Usability Scale (SUS). *Journal of System and Information Technology*, 3(2).
- Susila, A. A. N. H., & Arsa, D. M. S. (2023). Analisis System Usability Scale (SUS) dan Perancangan Sistem Self Service Pemesanan Menu di Restoran Berbasis Web. *Majalah Ilmiah UNIKOM*, 21(1). <https://doi.org/10.34010/miu.v21i1.10683>
- Syahrizal. (2025). Sistem Informasi Pengarsipan Surat di Kantor Dinas Ketenagakerjaan Kabupaten Asahan. *Uranus : Jurnal Ilmiah Teknik Elektro, Sains dan Informatika*, 3(1), 152–169. <https://doi.org/10.61132/uranus.v3i1.678>
- Syukron, T. A. S., Ariyanto, L., & W, Th. I. (2023). Sistem Informasi Penjualan Air Galon Berbasis Web. Dalam *Jurnal Elektro dan Teknologi Informasi* (Vol. 2, Nomor 2).
- Tampubolon, P., & Maslan, A. (2022). PERANCANGAN APLIKASI E-GALON BERBASIS ANDROID. *JURNAL COMASIE*, 6(5).
- Trivaika, E., & Senubekti, M. A. (2022). Perancangan Aplikasi Pengelola Keuangan Pribadi Berbasis Android. *JURNAL NUANSA INFORMATIKA*, 16(1). <https://journal.uniku.ac.id/index.php/ilkom>
- Tullis, T. S., & Stetson, J. N. (2004). A Comparison of Questionnaires for Assessing Website Usability. *UPA 2004*.
- Widodo, T. S., Prihati, Y., & Gondohanindijo, J. (2024). Implementasi metode waterfall pada sistem informasi inventaris barang berbasis web di Hotel Grand Edge Semarang. *Journal of Information Technology and Computer Science (INTECOMS)*, 7(5).

LAMPIRAN

Lampiran A: Tempat Lokasi Penelitian



Lampiran B: Wawancara dengan Pihak Depot



Lampiran C: Wawancara dengan Pihak Sebagai Customer Galon

