

PENGEMBANGAN APLIKASI AGEN CV ARJUNA JATI JEPARA BERBASIS ANDROID



Disusun Oleh:

N a m a : Muhammad Saiful Fahmi Hasim
NIM : 20523011

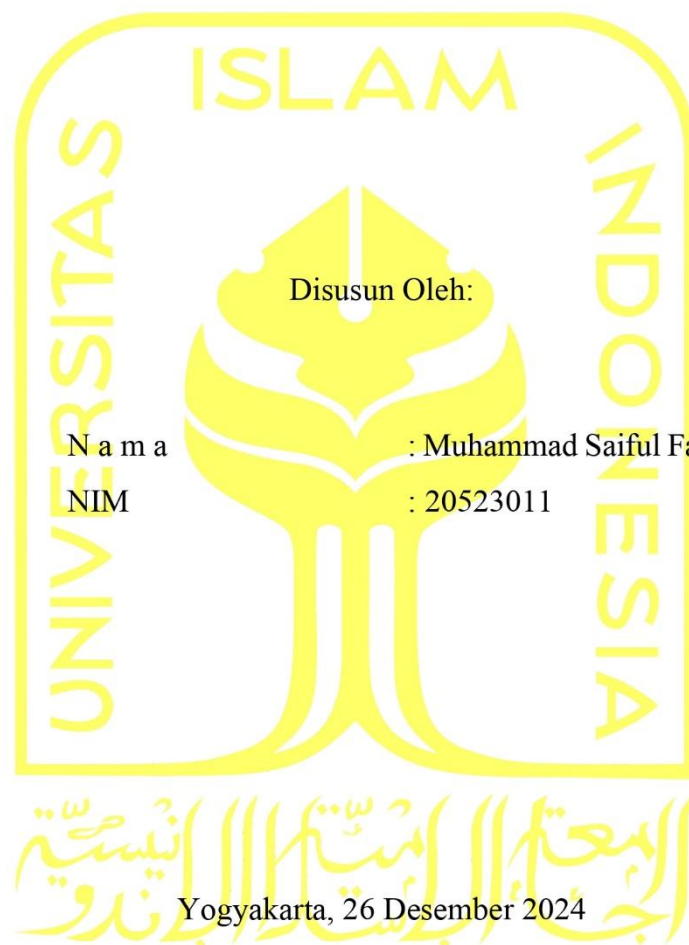
**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2025

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**PENGEMBANGAN APLIKASI AGEN CV ARJUNA JATI
JEPARA BERBASIS ANDROID**

TUGAS AKHIR



Disusun Oleh:

N a m a

: Muhammad Saiful Fahmi Hasim

NIM

: 20523011

Yogyakarta, 26 Desember 2024

Pembimbing,

(Aridhanyati Arifin, S.T., M.Cs)

HALAMAN PENGESAHAN DOSEN PENGUJI

PENGEMBANGAN APLIKASI AGEN CV ARJUNA JATI

JEPARA BERBASIS ANDROID

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, 26 Desember 2024

Tim Penguji

Aridhanyati Arifin, S.T., M.Cs.

Anggota 1

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

Anggota 2

Erika Ramadhani, S.T., M.Eng.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



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

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Muhammad Saiful Fahmi Hasim

NIM : 20523011

Tugas akhir dengan judul:

PENGEMBANGAN APLIKASI AGEN CV ARJUNA JATI JEPARA BERBASIS ANDROID

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, 26 Desember 2024



(Muhammad Saiful Fahmi Hasim)

HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur, saya mengucapkan puji dan terima kasih kepada Allah SWT atas segala nikmat dan karunia-Nya yang telah diberikan, sehingga saya dapat menyelesaikan tugas akhir yang berjudul “PENGEMBANGAN APLIKASI AGEN CV ARJUNA JATI JEPARA BERBASIS ANDROID”. Saya juga mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah berkontribusi dan membantu dalam penelitian ini.

HALAMAN MOTO

"Cara terbaik untuk mendapatkan apa yang Anda inginkan adalah dengan memberikan apa yang orang lain inginkan."

-Warren Buffett

"Kesuksesan adalah guru yang buruk. Ia menggoda orang pintar untuk berpikir bahwa mereka tidak bisa gagal."

-Bill Gates

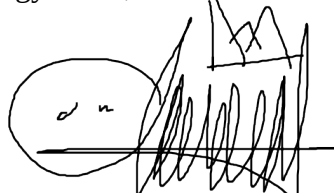
KATA PENGANTAR

Segala puji hanya milik Allah SWT, yang telah memberikan rahmat dan petunjuk-Nya, sehingga saya dapat menyelesaikan laporan tugas akhir dengan judul “PENGEMBANGAN APLIKASI AGEN CV ARJUNA JATI JEPARA BERBASIS ANDROID”. Laporan ini disusun sebagai salah satu syarat untuk menyelesaikan studi di Program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia., saya ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Saya bersyukur kepada Allah SWT yang telah memberikan kesehatan, kelancaran, dan petunjuk-Nya, sehingga saya dapat menyelesaikan laporan tugas akhir ini dengan baik.
2. Terima kasih kepada kedua orang tua dan keluarga saya atas doa, dukungan, dan kasih sayang yang tiada henti.
3. Saya mengucapkan terima kasih kepada Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Jurusan Informatika Universitas Islam Indonesia.
4. Terima kasih juga saya sampaikan kepada Bapak Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Ketua Program Studi Informatika – Program Sarjana.
5. Saya mengucapkan terima kasih kepada Ibu Aridhanyati Arifin, S.T., M.Cs yang dengan sabar memberikan arahan, bimbingan, dan motivasi selama proses penyusunan tugas akhir ini.
6. Terima kasih juga kepada semua pihak yang telah memberikan bantuan yang sangat berarti dalam pelaksanaan tugas akhir ini, meskipun tidak bisa saya sebutkan satu per satu.

Saya menyadari bahwa tugas akhir ini masih jauh dari sempurna, oleh karena itu saya sangat mengharapkan kritik dan saran yang konstruktif untuk perbaikan di masa mendatang. Semoga aplikasi yang dihasilkan dapat memberikan manfaat.

Yogyakarta, 26 Desember 2024



(Muhammad Saiful Fahmi Hasim)

SARI

Penelitian ini dilatarbelakangi oleh permasalahan yang dihadapi CV Arjuna Jati Jepara, yaitu efisiensi rendah, akurasi yang kurang, dan risiko keamanan data dalam sistem kasir manual yang masih digunakan. Sistem manual ini sering kali menyebabkan kesalahan pencatatan, kehilangan data, dan memakan waktu lebih lama dalam proses transaksi. Untuk mengatasi masalah tersebut, penelitian ini bertujuan mengembangkan sebuah aplikasi kasir berbasis *mobile* yang dirancang untuk mengoptimalkan efisiensi, meningkatkan akurasi pencatatan, dan memperkuat keamanan data transaksi.

Penelitian ini menggunakan metode pengembangan perangkat lunak model *Waterfall*. Tahapan yang dilalui meliputi analisis kebutuhan, perancangan sistem, implementasi aplikasi, pengujian, dan pemeliharaan. Setiap tahapan dilakukan secara terstruktur untuk memastikan aplikasi yang dihasilkan memenuhi kebutuhan pengguna. Pengujian aplikasi dilakukan menggunakan metode *Black-box Testing* dan *System Usability Scale* (SUS) untuk mengevaluasi tingkat kepuasan pengguna terhadap fitur dan antarmuka aplikasi.

Hasil penelitian menunjukkan bahwa aplikasi kasir berbasis *mobile* berhasil memenuhi kebutuhan pengguna, meningkatkan efisiensi proses transaksi, mengurangi kesalahan pencatatan, serta menyediakan sistem keamanan data yang lebih baik. Berdasarkan hasil pengujian usabilitas menggunakan metode *System Usability Scale* (SUS), aplikasi ini mendapatkan skor rata-rata 87,5 dari skala 100, yang termasuk dalam kategori "*Excellent*" dan menunjukkan tingkat penerimaan yang sangat positif dari pengguna. Pengujian yang dilakukan terhadap 11 responden, terdiri dari 1 pemilik perusahaan dan 10 karyawan CV Arjuna Jati Jepara, menghasilkan skor SUS individual berkisar antara 80,0 hingga 90,0, dengan skor terendah 80,0 yang masih tergolong baik dan skor tertinggi 90,0 yang menunjukkan kepuasan pengguna yang sangat tinggi. Data ini mengindikasikan bahwa aplikasi tidak hanya berhasil memenuhi kebutuhan fungsional tetapi juga memberikan pengalaman pengguna yang optimal, sehingga dapat mendukung CV Arjuna Jati Jepara dalam meningkatkan efisiensi operasional dan daya saingnya di era digital.

Kata kunci: *mobile*, aplikasi kasir, efisiensi transaksi, *waterfall*, *system usability scale*.

GLOSARIUM

<i>Android</i>	Sistem operasi berbasis <i>Linux</i> yang digunakan pada perangkat <i>mobile</i> , seperti ponsel dan tablet. Digunakan untuk mengembangkan aplikasi <i>mobile</i> dengan berbagai bahasa pemrograman, salah satunya <i>Kotlin</i> .
<i>Black-box Testing</i>	Jenis pengujian perangkat lunak di mana penguji fokus pada fungsionalitas aplikasi, memeriksa apakah aplikasi berfungsi sesuai dengan yang diharapkan berdasarkan input yang diberikan dan output yang dihasilkan, tanpa mengetahui kode sumber internal aplikasi..
<i>Cloud Storage</i>	Layanan penyimpanan data di <i>cloud</i> yang memungkinkan aplikasi untuk menyimpan dan mengakses data dari server jarak jauh, seperti penyimpanan data transaksi atau produk dalam aplikasi kasir.
<i>Firebase</i>	Platform pengembangan aplikasi <i>mobile</i> dan web dari <i>Google</i> yang menyediakan berbagai layanan, seperti database <i>real-time</i> , otentikasi pengguna, penyimpanan data, dan lainnya.
<i>Kotlin</i>	Bahasa pemrograman <i>statically typed</i> yang dirancang untuk menggantikan <i>Java</i> dalam pengembangan aplikasi Android.

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.....	viii
GLOSARIUM.....	ix
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
1.6 Sistematika Penulisan	3
BAB II LANDASAN TEORI DAN KAJIAN PUSTAKA	4
2.1 Landasan Teori.....	4
2.1.1 Android	4
2.1.2 Android Studio	5
2.1.3 Kotlin	6
2.1.4 <i>Firestore Realtime Database</i>	6
2.1.5 <i>Layout XML</i>	7
2.1.6 <i>SharedPreferences</i>	8
2.1.7 <i>Black Box Testing</i>	8
2.1.8 <i>System Usability Scale</i>	9
2.1.9 Integrasi <i>Firestore</i> dengan <i>Android Studio</i>	10
2.2 Kajian Literatur	10
BAB III METODOLOGI PENELITIAN	17
3.1 Pengumpulan Data	18
3.1.1 Wawancara.....	18
3.1.2 Observasi.....	19
3.2 Identifikasi Masalah.....	20
3.3 Perancangan aplikasi.....	20
3.3.1 Analisis kebutuhan.....	20
3.3.2 Perancangan Struktur <i>Database</i>	22
3.3.3 Perancangan Arsitektur Android.....	22
3.3.4 Perancangan <i>UML/Use Case Diagram</i>	22
3.4 <i>Coding</i> Android.....	22
3.4.1 <i>Coding</i> Menggunakan Kotlin dan <i>Android Studio</i>	22
3.4.2 <i>Layout</i> Menggunakan <i>XML</i>	23
3.4.3 <i>Database</i> Menggunakan <i>Firestore</i>	23
3.5 <i>Testing</i>	23
3.5.1 <i>Black-Box Testing</i>	23

3.5.2	<i>Usability Testing</i> dengan <i>System Usability Scale (SUS)</i>	27
	BAB IV HASIL DAN PEMBAHASAN	29
4.1	Hasil Perancangan Aplikasi	29
4.1.1	Hasil Perancangan <i>UML/Use Case Diagram</i>	29
4.1.2	Hasil Perancangan Struktur <i>Database</i>	30
4.1.3	Hasil Perancangan Arsitektur Android	35
4.2	Hasil Pengembangan Aplikasi	51
4.2.1	Halaman <i>Login</i>	52
4.2.2	Halaman <i>Home</i> (Tampilan Perusahaan)	53
4.2.3	Halaman <i>Home</i> (Tampilan Karyawan)	54
4.2.4	Halaman menu karyawan	55
4.2.5	Halaman Menu Produk (Tampilan Perusahaan)	56
4.2.6	Halaman Menu Produk (Tampilan Karyawan)	57
4.2.7	Halaman menu Pembayaran	58
4.2.8	Halaman menu Target (Tampilan Perusahaan)	59
4.2.9	Halaman Menu Target (Tampilan Karyawan)	60
4.2.10	Halaman menu Penjualan	61
4.2.11	Halaman menu Mutasi	62
4.2.12	Halaman menu pengaturan perusahaan	63
4.2.13	Halaman pengaturan akun	64
4.3	<i>Testing</i>	65
4.3.1	<i>Black box testing</i>	65
4.3.2	Pengujian <i>Usabilitas</i>	69
	BAB V KESIMPULAN DAN SARAN	73
5.1	Kesimpulan	73
5.2	Saran	73
	DAFTAR PUSTAKA	75
	LAMPIRAN	79

DAFTAR TABEL

Tabel 2. 1 Tabel Kajian Literatur.....	12
Tabel 3. 1 Analisis Kebutuhan Sistem (Perusahaan).....	21
Tabel 3. 2 Analisis Kebutuhan (Karyawan).....	21
Tabel 3. 3 Tabel skenario pengujian <i>black-box</i> pada akun Perusahaan.....	23
Tabel 3. 4 Tabel skenario pengujian <i>black-box</i> pada akun karyawan	26
Tabel 3. 5 Tabel pernyataan <i>SUS</i>	28
Tabel 4. 1 Tabel pengujian akun perusahaan.....	66
Tabel 4. 2 Tabel pengujian akun karyawan	68
Tabel 4. 3 Tabel pernyataan pengujian <i>sus</i>	70
Tabel 4. 4 Tabel penilaian pengujian <i>sus</i>	70
Tabel 4. 5 Tabel hasil pengujian	71
Tabel 4. 6 Tabel analisis berdasarkan pengelompokan Pernyataan.....	71

DAFTAR GAMBAR

Gambar 3. 1 Alur Penelitian	17
Gambar 4. 1 <i>Use Case Diagram</i>	29
Gambar 4. 2 Struktur <i>Database Relasional</i>	31
Gambar 4. 2. 1 Struktur <i>Database Non-Relasional</i>	33
Gambar 4. 3 Diagram aktivitas <i>login</i>	36
Gambar 4. 4 Diagram aktivitas menu karyawan	37
Gambar 4. 5 Diagram aktivitas menu produk perusahaan	38
Gambar 4. 6 Diagram aktivitas menu pembayaran perusahaan	39
Gambar 4. 7 Diagram aktivitas menu target perusahaan	40
Gambar 4. 8 Diagram aktivitas menu penjualan perusahaan	41
Gambar 4. 9 Diagram aktivitas menu mutasi	42
Gambar 4. 10 Diagram menu pengaturan perusahaan	43
Gambar 4. 11 Diagram pengaturan akun perusahaan	44
Gambar 4. 12 Diagram aktivitas <i>login</i> karyawan	45
Gambar 4. 13 Diagram aktivitas menu produk karyawan	46
Gambar 4. 14 Diagram aktivitas menu pembayaran karyawan	48
Gambar 4. 15 Diagram aktivitas menu target karyawan	48
Gambar 4. 16 Diagram aktivitas menu penjualan karyawan	49
Gambar 4. 17 Diagram aktivitas menu Pengaturan akun karyawan	50
Gambar 4. 18 <i>Wireframe low-fidelity</i>	51
Gambar 4. 19 Halaman <i>login</i>	52
Gambar 4. 20 Halaman <i>Home</i> Perusahaan	54
Gambar 4. 21 Halaman <i>home</i> karyawan	55
Gambar 4. 22 Halaman menu karyawan	56
Gambar 4. 23 Halaman menu produk perusahaan	57
Gambar 4. 24 Halaman menu produk karyawan	58
Gambar 4. 25 Halaman menu pembayaran	59
Gambar 4. 26 Halaman menu target perusahaan	60
Gambar 4. 27 Halaman target karyawan	61
Gambar 4. 28 Halaman menu penjualan	62
Gambar 4. 29 Halaman menu mutasi	63
Gambar 4. 30 Halaman menu pengaturan perusahaan	64

Gambar 4. 31 Halaman pengaturan akun.....	65
---	----

BAB I PENDAHULUAN

1.1 Latar Belakang

CV Arjuna Jati yang berlokasi di Bulungan RT 01 RW 05 Kecamatan Pakis Aji Jepara, merupakan perusahaan yang bergerak di bidang distribusi minuman sejak tahun 2015. Perusahaan yang dipimpin oleh Siti Masruroh ini sebelumnya pernah menjalin kerjasama dengan *Coca Cola*, menunjukkan potensi dan kapabilitas perusahaan dalam industri minuman. Namun dalam operasional hariannya, perusahaan masih menggunakan sistem pencatatan transaksi secara manual yang menimbulkan beberapa permasalahan signifikan.

Permasalahan utama yang dihadapi adalah rendahnya efisiensi dalam proses pencatatan transaksi manual, dimana setiap transaksi harus dicatat dalam buku dan kemudian dipindahkan ke lembar kerja *Excel*. Sistem manual ini mengakibatkan lambatnya proses transaksi dan membutuhkan waktu lebih lama dalam pembuatan laporan (Sasmito, 2017). Selain itu, pencatatan manual juga rentan terhadap kesalahan *human error* seperti kesalahan *input* data, perhitungan yang tidak akurat, dan inkonsistensi format pencatatan.

Risiko yang timbul dari penggunaan sistem manual ini sangat signifikan bagi operasional perusahaan. Kesalahan pencatatan manual dapat mengakibatkan kerugian finansial dan ketidakakuratan dalam pengambilan keputusan bisnis. Data transaksi yang tidak terorganisir dengan baik juga menyulitkan proses pelacakan riwayat transaksi dan analisis performa penjualan. Lebih jauh lagi, penyimpanan data secara fisik berisiko mengalami kerusakan atau kehilangan yang dapat berdampak serius pada stabilitas keuangan Perusahaan (Andriana, 2016).

Sebagai solusi atas permasalahan tersebut, penelitian ini mengusulkan pengembangan aplikasi kasir berbasis *mobile* dengan menggunakan model pengembangan *Waterfall*. Pemilihan model *Waterfall* didasarkan pada karakteristik kebutuhan CV Arjuna Jati yang sudah terdefinisi dengan jelas dan cenderung stabil. Model *Waterfall* sangat efektif untuk proyek dengan kebutuhan yang *well-defined* dan tidak mengalami perubahan signifikan selama proses pengembangan. Model ini menawarkan pendekatan sistematis dan sekuensial yang memungkinkan pengembangan aplikasi yang terstruktur dengan dokumentasi yang lengkap pada setiap tahapnya (Pressman, 2005).

Aplikasi yang dikembangkan akan menggunakan teknologi *Firestore Realtime Database* untuk penyimpanan data, yang menawarkan sinkronisasi data secara *real-time* dan aksesibilitas yang tinggi. Implementasi aplikasi berbasis *mobile* dipilih untuk memberikan fleksibilitas

akses dari berbagai lokasi menggunakan *smartphone*. Aplikasi ini akan dilengkapi dengan fitur pencatatan transaksi otomatis, manajemen stok *real-time*, sistem target penjualan, serta pembedaan hak akses antara pemilik dan karyawan untuk meningkatkan keamanan sistem. Pengembangan aplikasi ini diharapkan dapat mengoptimalkan proses transaksi, meningkatkan akurasi pencatatan, dan memperkuat pengelolaan data operasional CV Arjuna Jati Jepara.

1.2 Rumusan Masalah

- a. Bagaimana mengembangkan aplikasi kasir berbasis mobile yang dapat melakukan pencatatan transaksi secara otomatis dan melakukan pelacakan performa penjualan serta stok barang di CV Arjuna Jati Jepara?
- b. Bagaimana tingkat kebergunaan (*usability*) aplikasi kasir berbasis mobile dari perspektif pengguna di CV Arjuna Jati Jepara?

1.3 Batasan Masalah

- a. Penelitian berfokus pada pengembangan aplikasi kasir berbasis mobile untuk pengelolaan transaksi penjualan di CV Arjuna Jati Jepara.
- b. Lingkup aplikasi meliputi pencatatan transaksi, manajemen stok barang, pelacakan performa penjualan, dan pembuatan laporan transaksi.
- c. Tidak membahas aspek lain dari operasional bisnis di luar sistem kasir dan transaksi.

1.4 Tujuan Penelitian

- a. Menghasilkan aplikasi kasir agen portable berbasis mobile yang dapat mengoptimalkan efisiensi proses transaksi di CV Arjuna Jati Jepara.
- b. Menganalisis pengaruh kebergunaan dari aplikasi kasir agen di CV Arjuna Jati Jepara.

1.5 Manfaat Penelitian

Memberikan kontribusi dalam memahami pentingnya penggunaan teknologi dalam memodernisasi sistem kasir pada bisnis kecil hingga menengah, serta memperkuat daya saing CV Arjuna Jati Jepara di pasar yang semakin kompetitif dan dinamis.

1.6 Sistematika Penulisan

Dalam penyusunan laporan Tugas Akhir, struktur penyajian berperan penting dalam membantu peneliti mengorganisasikan dokumen sesuai dengan standar penelitian, sehingga penelitian menjadi lebih terarah. Berikut ini adalah kerangka laporan Tugas Akhir:

a. **BAB I PENDAHULUAN**

Bab ini berisi pembahasan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, dan struktur laporan.

b. **BAB II KAJIAN PUSTAKA**

Bagian ini memuat berbagai dasar kerangka pemikiran dalam penelitian, yang didapatkan dari studi pustaka untuk memperoleh solusi dari permasalahan yang ada. Bagian ini terdiri Kajian Teori dan Kajian Literatur.

c. **BAB III METODOLOGI**

Bagian metodologi membahas analisis teori yang diterapkan dalam penelitian serta cara mengimplementasikannya ke dalam sistem yang dikembangkan. Materi yang digunakan dalam penelitian ini akan dijelaskan sesuai dengan alur pengembangan yang dilakukan, serta penguraian perancangan sistem.

d. **BAB IV HASIL DAN PEMBAHASAN**

Bagian ini membahas secara rinci hasil penelitian yang dilaksanakan. Hasil penelitian akan disajikan dalam bentuk gambaran alur sistem, gambar kode, tabel, foto ataupun bentuk lainnya yang diuraikan sesuai dengan tahapan penelitian.

e. **BAB V KESIMPULAN DAN SARAN**

Kesimpulan dan saran merupakan bagian terakhir dari penulisan laporan yang membahas tentang kesimpulan hasil penelitian yang telah dilakukan dan berisi saran serta wawasan untuk penelitian selanjutnya yang berkaitan dengan topik bahasan pada penelitian ini.

BAB II

LANDASAN TEORI DAN KAJIAN PUSTAKA

2.1 Landasan Teori

Dalam perkembangan teknologi yang terus melaju pesat, penerapan sistem berbasis digital telah menjadi salah satu upaya strategis untuk meningkatkan kualitas dan efisiensi operasional di berbagai sektor, termasuk bisnis kecil dan menengah. Salah satu inovasi yang mulai banyak diadopsi adalah aplikasi kasir berbasis mobile, yang dirancang untuk menggantikan sistem manual dengan solusi yang lebih modern, efisien, dan aman. Penelitian ini berfokus pada pengembangan aplikasi kasir agen portable di CV Arjuna Jati Jepara, yang tidak hanya bertujuan untuk meningkatkan kecepatan dan akurasi pencatatan transaksi, tetapi juga untuk memperkuat keamanan data dan mendukung modernisasi sistem kasir. Dengan demikian, hasil penelitian ini diharapkan dapat memberikan wawasan baru mengenai pentingnya penerapan teknologi dalam meningkatkan daya saing dan efisiensi operasional usaha kecil dan menengah.

2.1.1 Android

Sistem operasi Android merupakan sebuah platform perangkat mobile berbasis *Linux* yang dirancang untuk menyediakan pengalaman pengguna yang optimal melalui kombinasi antara *middleware*, sistem operasi, dan aplikasi. Sebagai platform terbuka, Android memberikan kebebasan yang luas kepada para pengembang untuk menciptakan berbagai aplikasi inovatif yang dapat berjalan di dalam ekosistemnya. Pendekatan ini tidak hanya mendorong kreativitas dan inovasi, tetapi juga memungkinkan pengembangan solusi yang beragam sesuai dengan kebutuhan pengguna di berbagai bidang (Sholeh dkk., 2022).

Android telah berhasil menjadi salah satu sistem operasi yang paling populer di dunia saat ini, mengungguli berbagai sistem operasi lainnya dalam hal penggunaannya di perangkat mobile. Popularitas ini tidak lepas dari ekosistemnya yang luas, fitur-fitur canggih yang terus berkembang, serta fleksibilitasnya dalam mendukung berbagai jenis perangkat, mulai dari *smartphone* hingga tablet. Keberhasilan Android juga didorong oleh kemampuannya untuk memberikan pengalaman pengguna yang optimal, baik bagi individu maupun pengembang aplikasi (Pratama & Somya, 2021).

Dalam website gs.statcounter.com Android merupakan salah satu sistem operasi yang sangat populer dan banyak digunakan pada berbagai perangkat, seperti *smartphone*,

smartwatch, *smart TV*, dan lainnya. Berbasis *Linux*, Android berhasil menduduki peringkat pertama dalam pangsa pasar sistem operasi mobile di Indonesia per Desember 2024, dengan persentase sebesar 93.41%. Sementara itu, pesaing terbesarnya, iOS, berada di posisi kedua dengan pangsa pasar 6,54%.

2.1.2 Android Studio

Android Studio adalah sebuah *Integrated Development Environment (IDE)* yang dirancang khusus untuk pengembangan aplikasi di platform Android. IDE ini dibangun dengan dasar *IntelliJ IDEA*, yang merupakan *IDE* untuk pemrograman Java. Bahasa utama yang digunakan dalam pengembangan aplikasi adalah Java, sementara untuk mendesain antarmuka atau layout, digunakan bahasa XML. Android Studio juga terhubung dengan *Android Software Development Kit (SDK)*, yang memungkinkan aplikasi diujicobakan langsung ke perangkat Android. Sebagai pengembangan dari *Eclipse*, Android Studio dirancang lebih canggih dan profesional, dengan berbagai fitur kompleks yang telah terintegrasi di dalamnya (Fahnun Putri dkk., 2019).

Android Studio menyediakan lingkungan pengembangan yang lengkap dan profesional, mencakup berbagai alat seperti *Android SDK Tools* yang terintegrasi langsung. Setiap proyek di Android Studio dapat terdiri dari satu atau beberapa modul, yang masing-masing berisi file kode sumber dan sumber daya. Jenis modul yang disediakan mencakup modul aplikasi Android, modul pustaka, dan modul *Google App Engine*. Tampilan proyek secara *default* menggunakan tampilan proyek Android, yang dirancang untuk mempermudah navigasi dan pengelolaan file (Sibuea dkk., 2022).

Lebih lanjut, Android Studio dilengkapi dengan *intelligent code editor* yang mampu menganalisis dan memproses kode secara menyeluruh, membantu pengembang meningkatkan efisiensi dan produktivitas. Fitur *code templates* dan integrasi *GitHub* memungkinkan pengembang untuk memanfaatkan contoh kode yang tersedia atau mengimpor proyek langsung dari *GitHub*. Emulator bawaan mendukung berbagai jenis perangkat dengan ukuran dan bentuk yang beragam, memberikan fleksibilitas untuk menguji aplikasi dalam berbagai skenario (Agustina dkk., 2023).

Meskipun memiliki banyak keunggulan, Android Studio memiliki kendala, terutama ukuran file instalasinya yang besar, mencapai hampir 1 GB untuk Windows. Hal ini dapat menjadi tantangan bagi pengguna dengan koneksi internet lambat. Namun, dengan fitur

canggih dan integrasi lengkap, Android Studio tetap menjadi pilihan utama untuk pengembangan aplikasi Android (Agustina dkk., 2023).

2.1.3 Kotlin

Kotlin adalah bahasa pemrograman yang dirancang secara pragmatis untuk pengembangan aplikasi Android. Bahasa ini menggabungkan konsep pemrograman berorientasi objek (*Object-Oriented Programming* atau *OOP*) dengan paradigma pemrograman fungsional, sehingga memberikan fleksibilitas yang tinggi kepada pengembang dalam membangun aplikasi. Salah satu keunggulan utama Kotlin adalah interoperabilitasnya, yang memungkinkan bahasa ini dapat digunakan bersama dengan Java dalam satu proyek tanpa masalah kompatibilitas. Hal ini membuat Kotlin menjadi pilihan yang populer, terutama bagi pengembang yang ingin memanfaatkan keunggulan dari kedua bahasa secara bersamaan. Tidak hanya terbatas pada aplikasi Android, Kotlin juga mendukung pengembangan aplikasi berbasis desktop, web, hingga aplikasi untuk backend, menjadikannya bahasa pemrograman serbaguna yang dapat digunakan di berbagai platform (Sibarani dkk., 2018).

Menurut Dicoding.com Salah satu fitur penting Kotlin adalah *type inference*, yang memungkinkan pengembang untuk tidak perlu mendeklarasikan tipe data variabel secara eksplisit, karena kompiler Kotlin secara otomatis menentukan tipe data tersebut. Fitur ini memberikan efisiensi dan membuat kode lebih ringkas dibandingkan dengan Java. Selain itu, Kotlin mendukung interoperabilitas penuh dengan Java, memungkinkan kedua bahasa digunakan secara bersamaan dalam satu proyek, sehingga mempermudah transisi bagi pengembang yang sebelumnya menggunakan Java.

2.1.4 *Firestore Realtime Database*

Firestore adalah layanan *cloud* yang menyediakan platform *Backend as a Service (BaaS)*, yang berpusat di *San Francisco, California*. Layanan ini dirancang untuk memudahkan pengembang aplikasi mobile dan web dengan menyediakan infrastruktur yang dibutuhkan untuk pengelolaan backend, sehingga memungkinkan pengembang fokus pada pengembangan logika aplikasi dan fitur utama. *Firestore* didirikan pada tahun 2011 oleh *Andrew Lee* dan *James Tamplin* dengan tujuan menyederhanakan proses pengembangan aplikasi (Andilala dkk., 2021).

Sejak peluncurannya, *Firestore* terus berkembang dengan menawarkan berbagai produk untuk mendukung pengembangan aplikasi secara efisien. Salah satu pencapaian besar *Firestore*

adalah peluncuran layanan berbasis *cloud* pada tahun 2012, yang memberikan fleksibilitas dan skalabilitas bagi para pengembang. Pada Oktober 2014, *Firebase* diakuisisi oleh *Google*, memperluas jangkauan platform ini ke dalam ekosistem *Google Cloud*. Akuisisi ini memungkinkan lebih banyak integrasi dengan layanan *Google* lainnya, memperkuat posisi *Firebase* sebagai solusi andal dalam dunia pengembangan aplikasi (Sonita & Fardianitama, 2018).

Firebase juga dikenal sebagai platform yang mendukung pengembangan aplikasi dengan fitur *real-time*. Ketika data di *Firebase* mengalami perubahan, pembaruan secara otomatis dilakukan di semua perangkat yang terhubung, baik melalui aplikasi berbasis web maupun mobile. Platform ini menyediakan pustaka lengkap untuk berbagai platform web dan mobile, serta kompatibel dengan *framework* seperti *Node.js*, *Java*, dan *JavaScript*. *Firebase* menggunakan *Application Programming Interface (API)* untuk menyimpan dan menyinkronkan data dalam format *JSON* di *cloud*, yang diperbarui secara *real-time* (Sanad dkk., 2018).

Firebase menawarkan berbagai fitur utama, meliputi:

- a. *Analytics*: Memantau perilaku pengguna dalam menggunakan aplikasi dan menyajikan data tersebut dalam satu *dashboard*.
- b. *Develop*: Menyediakan fitur seperti *cloud messaging*, autentikasi, basis data *real-time*, penyimpanan, hosting, laboratorium pengujian, dan pelaporan kerusakan.
- c. *Grow*: Mendukung proses publikasi produk aplikasi, membantu aplikasi menjangkau lebih banyak pengguna.

2.1.5 *Layout XML*

Menurut developer.android.com *Layout XML* adalah struktur dokumen elektronik yang dirancang untuk menyimpan, mengatur, dan mengirim data secara terstruktur dan sistematis. Dalam konteks pengembangan aplikasi mobile, khususnya di platform Android, *XML* sering digunakan untuk mendesain antarmuka pengguna (*UI*). *File XML* memungkinkan pengembang untuk mendefinisikan elemen visual aplikasi secara hierarkis dan terstruktur, mulai dari layout utama hingga elemen interaktif seperti tombol, teks, atau gambar.

Menurut buku *Android Developer Fundamentals Course – Practicals* yang disusun oleh Tim Pelatihan *Developer Google (Developer Training, 2016)*, layout dalam file *XML* digunakan untuk mendefinisikan elemen-elemen *UI* aplikasi Android, seperti komponen *TextView*, *EditText*, *Button*, *ScrollView*, dan *RecyclerView*. File layout ini disusun secara

hierarkis, sehingga memungkinkan pengembang untuk merancang antarmuka pengguna yang kompleks namun tetap terorganisasi. Sebagai bagian dari *Model-View-Presenter (MVP)*, file layout berfungsi untuk memisahkan logika aplikasi (diimplementasikan dalam kode Java atau Kotlin) dari elemen *UI*, sehingga meningkatkan keterbacaan dan maintainability proyek aplikasi.

2.1.6 *SharedPreferences*

Menurut buku *Android Developer Fundamentals Course – Practicals* yang disusun oleh Tim Pelatihan *Developer Google (Developer Training, 2016)*, *SharedPreferences* adalah mekanisme penyimpanan data sederhana pada aplikasi Android yang memungkinkan pengembang menyimpan data dalam bentuk pasangan kunci-nilai secara persisten. Data yang disimpan dalam *SharedPreferences* tetap tersedia bahkan setelah aplikasi ditutup atau perangkat di-restart. *SharedPreferences* sangat cocok digunakan untuk menyimpan informasi seperti preferensi pengguna atau pengaturan aplikasi.

Proses pengelolaan data pada *SharedPreferences* melibatkan metode seperti *put* untuk menulis data dan *get* untuk membaca data berdasarkan kunci tertentu. Untuk penulisan data, pengembang menggunakan objek *SharedPreferences.Editor*, sedangkan pembacaan data dilakukan langsung melalui objek *SharedPreferences*. *SharedPreferences* mendukung penyimpanan berbagai tipe data sederhana seperti *string*, *integer*, *boolean*, dan lain-lain

Secara internal, *SharedPreferences* menggunakan *in-memory storage* di atas *disk storage*. Operasi membaca dan menulis data dilakukan terlebih dahulu pada *in-memory storage*, kemudian disinkronkan ke *disk storage* jika diperlukan. *In-memory storage* ini diimplementasikan sebagai *HashMap*, yang memungkinkan setiap operasi dilakukan dengan efisiensi tinggi, yaitu kompleksitas waktu (Rachman, 2017).

2.1.7 *Black Box Testing*

Black Box Testing adalah salah satu metode pengujian perangkat lunak yang berfokus pada pengujian aspek eksternal atau antarmuka perangkat lunak tanpa memeriksa struktur internal, seperti kode sumber atau arsitektur sistem. Metode ini hanya melihat bagaimana perangkat lunak merespons berbagai jenis input dan memastikan bahwa output yang dihasilkan sesuai dengan spesifikasi atau kebutuhan pengguna. Teknik ini cocok untuk memvalidasi fungsi sistem secara keseluruhan, mendeteksi kesalahan antarmuka, memeriksa batasan data, dan mengidentifikasi inisialisasi atau terminasi yang tidak sesuai (Parlika dkk., 2020).

Pendekatan *Black Box Testing* digunakan untuk menguji fungsi utama aplikasi kasir berbasis Android yang dikembangkan, tanpa memerhatikan struktur *internal* atau kode program sistem yang diuji (Nurudin dkk., 2019). Metode ini dirancang untuk menguji data *input* valid maupun tidak valid, sehingga aplikasi dapat diverifikasi untuk menangani berbagai skenario dengan benar, termasuk validasi data dan respons sistem terhadap input yang diberikan (Nasrullah dkk., 2020).

2.1.8 *System Usability Scale*

System Usability Scale (SUS) merupakan instrumen yang dikembangkan oleh Brooke pada tahun 1996 sebagai alat ukur "*quick and dirty*" untuk mengevaluasi kegunaan (*usability*) suatu sistem. *SUS* terdiri dari 10 pernyataan dengan skala *Likert* 5 poin yang menghasilkan skor akhir dalam rentang 0-100. Kesepuluh pernyataan tersebut meliputi: keinginan menggunakan sistem secara rutin, kompleksitas sistem, kemudahan penggunaan, kebutuhan bantuan teknis, integrasi fungsi, inkonsistensi, kecepatan pembelajaran, kerumitan penggunaan, kepercayaan diri pengguna, dan kebutuhan pembelajaran sebelum menggunakan sistem (Grier dkk., 2013).

Dalam implementasinya, *SUS* dapat digunakan dalam dua skenario utama yaitu evaluasi setelah pengujian langsung (*test-then-measure*) dan evaluasi tanpa tugas langsung. Pada skenario pertama, pengguna menilai sistem setelah melakukan serangkaian tugas tertentu yang memungkinkan perbandingan langsung antar produk ketika pengguna melakukan tugas yang sama atau identik. Sementara pada skenario kedua, penilaian dilakukan berdasarkan pengalaman pengguna dengan sistem tanpa melakukan tugas spesifik, yang berguna untuk mengevaluasi berbagai aspek interface yang mungkin tidak tercakup dalam skenario pengujian terbatas (Nielsen & Levy, 1994).

Penelitian menunjukkan bahwa *SUS* memiliki reliabilitas yang tinggi dan valid sebagai metrik kegunaan sistem komersial. Namun, *SUS* memiliki keterbatasan dalam mengukur aspek ergonomis dari interaksi pengguna dengan sistem. Hasil studi laboratorium dan lapangan menunjukkan bahwa *SUS* tidak dapat diandalkan untuk mengukur ketidaknyamanan atau aktivitas otot yang terkait dengan penggunaan perangkat lunak, yang mengindikasikan bahwa instrumen ini lebih tepat digunakan untuk evaluasi kegunaan umum daripada penilaian aspek ergonomis spesifik (Grier dkk., 2013).

2.1.9 Integrasi *Firebase* dengan *Android Studio*

Berdasarkan jurnal "*Implementation of Firebase Realtime Database to track BRT Trans Semarang*" yang ditulis oleh Wiratno dan Hastuti (2017), implementasi *Firebase Realtime Database* pada aplikasi *Android* memiliki keunggulan utama yaitu kemampuan untuk melakukan *sinkronisasi* data secara otomatis tanpa perlu melakukan *refresh* atau *reload* aplikasi. Seperti yang disebutkan dalam jurnal tersebut, "*The main feature of firebase is able to synchronize the data automatically without make an update request or refresh application*" (Wiratno & Hastuti, 2017). Dalam implementasinya, data disimpan dalam format objek *Java* karena seperti disebutkan dalam jurnal "*Firebase Real time Database use NO-SQL database which means the data is not saved in a SQL table format but a path tree format*" (Wiratno & Hastuti, 2017).

Proses integrasi *Firebase* ke dalam proyek *Android Studio* memerlukan beberapa tahapan konfigurasi. Seperti yang dijelaskan dalam jurnal, "*Android Firebase API needed to get the access to database. This is the most important part to connect our application to the Firebase Real time Database because Android Firebase API is the only way to get the access to database*" (Wiratno & Hastuti, 2017). Tahapan tersebut dimulai dari membuat proyek di *Firebase Console*, menempatkan file *google-services.json* ke direktori *app/*, dan menambahkan *plugin Google Services* serta dependensi *Firebase* yang diperlukan pada *build.gradle*. Setelah konfigurasi selesai dan inisialisasi *Firebase* dilakukan, sistem ini terbukti efektif dimana dalam jurnal disebutkan hasil pengujian menunjukkan "*92% user claims that the application able to provide a usefull information and 84% user claim that our proposed system is worthy to implement*" (Wiratno & Hastuti, 2017).

2.2 Kajian Literatur

Kajian literatur ini bertujuan memberikan landasan teoritis untuk pengembangan aplikasi kasir berbasis *mobile* di CV Arjuna Jati Jepara. Kajian ini merujuk pada jurnal dan publikasi ilmiah yang relevan dan kredibel, yang dikumpulkan melalui platform *Google Scholar* dan alat *Publish or Perish* dengan kata kunci seperti aplikasi kasir, pengembangan aplikasi *Android*, teknologi kasir digital, dan efisiensi transaksi, dalam rentang waktu 2019 hingga 2024.

Proses seleksi jurnal dimulai dengan 200 jurnal yang kemudian disaring menjadi 15 jurnal terpilih berdasarkan relevansi, metodologi, dan kontribusinya. Analisis dilakukan dalam beberapa kategori:

1. Domain Masalah: Jenis aplikasi yang dikembangkan.

2. Metode Pengembangan Aplikasi: Pendekatan yang digunakan dalam pengembangan aplikasi.
3. Fitur yang Ditawarkan: Fitur-fitur yang disediakan oleh aplikasi.
4. Metode Pengujian Aplikasi: Pendekatan pengujian yang diterapkan.
5. Platform: Platform yang digunakan.

Melalui kajian ini, diharapkan dapat ditemukan teori dan praktik terbaik untuk pengembangan aplikasi kasir yang efektif dan efisien.

Tabel 2. 1 Tabel Kajian Literatur

Referensi	Domain Masalah	Metode Penelitian	Metode Pengujian	Fitur aplikasi	Platform Aplikasi
(Kamilla dkk., 2024)	Aplikasi kasir untuk <i>coffee shop</i>	<i>Waterfall</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> • <i>CRUD</i> Produk (tambah, edit, hapus, dan lihat detail produk) • <i>Generate</i> deskripsi produk otomatis menggunakan <i>API Gemini</i> • Tambah pesanan dengan perhitungan total harga Laporan penjualan	Android
(Saputra dkk., 2021)	Aplikasi Kasir di Perusahaan Percetakan	<i>Waterfall</i>	<i>Unit Testing & System Testing</i>	<ul style="list-style-type: none"> • <i>Login</i> • transaksi penjualan • data produk • tambah produk • pengaturan keluar aplikasi. 	Web
(Faizal dkk., 2023)	Aplikasi Kasir Apotik	<i>Waterfall</i>	-	<ul style="list-style-type: none"> • Penghitungan transaksi otomatis. • Penyimpanan data penjualan dalam database. Pencetakan struk pembayaran	Android

Referensi	Domain Masalah	Metode Penelitian	Metode Pengujian	Fitur aplikasi	Platform Aplikasi
				dengan printer <i>Bluetooth</i> .	
(Nuryamin & Risyda, 2024)	Aplikasi Kasir Kedai Kopi	<i>Waterfall</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> Manajemen transaksi laporan penjualan pengelolaan menu 	Web
(Arafat dkk., 2021)	Aplikasi Kasir Nangkringan	<i>Waterfall</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> <i>Input</i> data barang <i>quick order</i> melihat data transaksi tampilan <i>login</i> menu utama 	Android
(Suprianto dkk., 2021)	Aplikasi POS pada Toko Salfa Tarakan	<i>Waterfall</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> Pendataan barang transaksi penjualan laporan pemasukan laporan omzet pengelolaan data pelanggan kas toko 	Web
(Pangestu & Astutik, 2024)	Aplikasi Kasir Toko Kelontong	<i>Waterfall</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> <i>Login</i> data produk transaksi, laporan data pelanggan halaman admin. 	Web
(Riswanto & Fahrudin, 2024)	Aplikasi Kasir Aira Motor	<i>Agile</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> <i>Login</i> 	Android

Referensi	Domain Masalah	Metode Penelitian	Metode Pengujian	Fitur aplikasi	Platform Aplikasi
				<ul style="list-style-type: none"> transaksi penjualan dan servis laporan penjualan dan servis cetak kuitansi data <i>sparepart</i> data pelanggan 	
(Nurwicaksono & Ikamah, 2024)	Aplikasi Kasir <i>Blessing of Shoes</i>	<i>Rapid Application Developers (RAD)</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> Manajemen saldo transaksi stok barang retur laporan cetak struk 	Android
(Sholeh dkk., 2022)	Aplikasi Kasir	<i>Waterfall</i>	<i>Black box Testing</i>	<ul style="list-style-type: none"> <i>Login</i>, pengelolaan produk pencatatan transaksi rekap penjualan cetak struk melalui <i>Bluetooth</i> printer 	Android
(Zahara & Widodo, 2024)	Aplikasi Kasir Toko Cantik	<i>User Centered Design (UCD)</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> Login manajemen stok barang transaksi laporan penjualan pembayaran <i>QRIS</i> tampilan intuitif 	Android

Referensi	Domain Masalah	Metode Penelitian	Metode Pengujian	Fitur aplikasi	Platform Aplikasi
(Rahayu & Fandhika, 2021)	Aplikasi Kasir Digital	Perencanaan, analisis, perancangan, implementasi, dan pengujian (<i>Black Box Testing</i>)	<i>Black Box Testing</i>	<ul style="list-style-type: none"> • Transaksi kasir • input barang • pencatatan penghasilan harian/bulanan • perhitungan pajak penghasilan 	Android
(Gurning dkk., 2022)	Aplikasi Kasir Katering Mama Suci	<i>Rapid Application Development (RAD)</i> , <i>UML</i> , dan <i>Black Box Testing</i>	<i>Black Box Testing</i>	<ul style="list-style-type: none"> • <i>Input</i> pesanan • daftar menu • total tagihan • detail pesanan • laporan histori pesanan 	Andorid
(Rayhan & Fatonah, 2024)	Aplikasi Kasir Kantin	<i>Waterfall</i> ,	<i>Black Box Testing, White Box Testing</i>	<ul style="list-style-type: none"> • <i>Login</i> admin/kasir • manajemen menu • transaksi pesanan • pencetakan bukti transaksi • laporan penjualan 	Web
(Hafidz & Ikhwan, 2024)	Aplikasi Kasir	<i>Waterfall</i> ,	<i>Black-box Testing</i>	<ul style="list-style-type: none"> • Pemesanan menu • transaksi dengan <i>QRIS/cash</i> • laporan penjualan • notifikasi ke dapur • status pesanan 	Android

Berdasarkan Tabel 2.1 pada kajian literatur, terdapat sejumlah temuan penting yang menjadi dasar dan keunikan dalam penelitian ini. Secara umum, mayoritas aplikasi kasir yang dikembangkan dalam penelitian sebelumnya berfokus pada usaha tertentu, seperti *coffee shop*, apotek, UMKM, hingga perusahaan besar. Penelitian ini mengambil studi kasus pada CV Arjuna Jati Jepara, sebuah perusahaan distribusi minuman. Meskipun konteks distribusi minuman ini bukan merupakan fokus baru dalam pengembangan aplikasi kasir, penelitian ini menawarkan pendekatan teknis dan inovasi fitur yang lebih mendalam, yang menjadi nilai tambah dibandingkan penelitian sebelumnya.

Dari segi metode pengembangan, pendekatan *Waterfall* banyak digunakan dalam penelitian terdahulu, dan penelitian ini juga mengadopsi pendekatan serupa. Namun, kelebihan penelitian ini terletak pada tambahan fitur mendalam yang ditawarkan, seperti integrasi pencetakan struk menggunakan *Bluetooth* printer, yang meningkatkan fungsionalitas aplikasi. Kebanyakan penelitian sebelumnya cenderung menawarkan fitur dasar seperti manajemen transaksi, pencetakan struk, dan laporan keuangan. Penelitian ini melangkah lebih jauh dengan menambahkan fitur inovatif berupa pengaturan target karyawan dan pelacakan target penjualan, memberikan nilai tambah yang signifikan dibandingkan aplikasi kasir serupa.

Selain itu, penelitian ini mengoptimalkan penggunaan platform Android dengan memanfaatkan *Firebase Realtime Database*. Teknologi ini memungkinkan sinkronisasi data secara *real-time*, memberikan keunggulan dalam hal kecepatan dan keakuratan data dibandingkan pendekatan basis data lokal yang sering digunakan dalam penelitian sebelumnya.

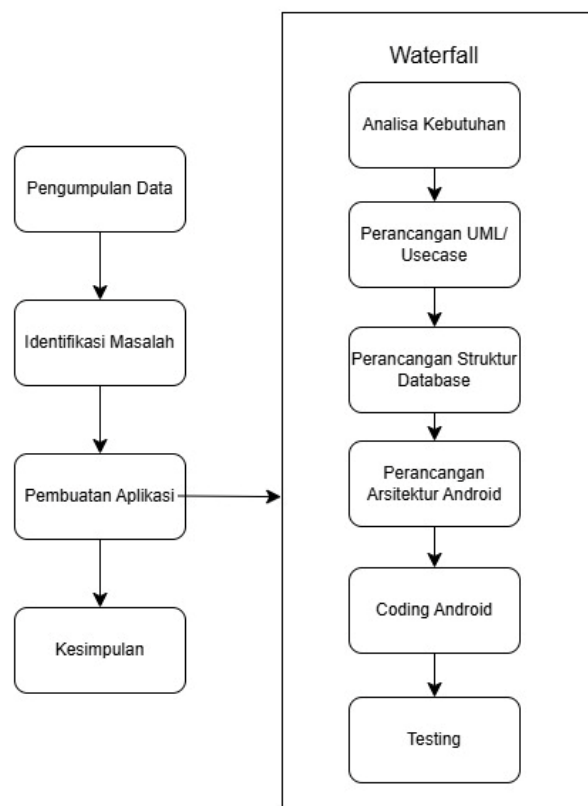
Penelitian ini juga menunjukkan kebaruan yang mencolok dibandingkan penelitian terdahulu. Dari segi teknologi, aplikasi ini menggunakan *Firebase* untuk penyimpanan data secara *real-time*, memastikan kecepatan dan keandalan data yang lebih baik. Selain itu, penelitian ini tidak hanya mengandalkan *Black-box Testing* sebagai metode pengujian, tetapi juga menggunakan *System Usability Scale (SUS)* untuk mengukur tingkat kepuasan pengguna terhadap aplikasi.

Keunggulan lainnya adalah fokus pada fleksibilitas dan keamanan. Penelitian ini mengembangkan sistem akses berbasis hak pengguna, yang membedakan fitur antara karyawan dan pemilik perusahaan sesuai dengan kebutuhan operasional mereka. Fitur ini memberikan keunggulan dalam hal keamanan dan fleksibilitas. Terakhir, aplikasi ini juga menyertakan fitur manajemen dan pelacakan target karyawan, sebuah inovasi unik yang belum ditemukan dalam kajian literatur sebelumnya. Dengan semua kebaruan ini, penelitian ini berhasil menawarkan solusi yang lebih modern dan relevan bagi kebutuhan operasional bisnis CV Arjuna Jati Jepara.

BAB III

METODOLOGI PENELITIAN

Penelitian ini dilaksanakan melalui beberapa tahapan sistematis yang dimulai dengan proses pengumpulan data melalui metode wawancara dan observasi. Setelah data terkumpul, dilanjutkan dengan tahap identifikasi masalah untuk menganalisis permasalahan yang ditemukan dari hasil pengumpulan data tersebut. Tahap selanjutnya adalah pembuatan aplikasi yang terbagi menjadi beberapa fase pengembangan, meliputi perancangan *UML/Usecase* untuk memodelkan sistem, perancangan struktur *database* untuk pengelolaan data, perancangan arsitektur android sebagai platform pengembangan, pengembangan aplikasi android, dan testing untuk memastikan kualitas aplikasi. Seluruh proses pengembangan aplikasi ini menggunakan metodologi *Waterfall* sebagai kerangka kerja pengembangan perangkat lunak. Penelitian kemudian diakhiri dengan penarikan kesimpulan dari keseluruhan proses yang telah dilakukan. Alur ini menggambarkan pendekatan sistematis dalam pengembangan aplikasi berbasis Android, mulai dari identifikasi kebutuhan hingga implementasi dan pengujian sistem. Alur penelitian terdapat pada Gambar 3. 1.



Gambar 3. 1 Alur Penelitian

3.1 Pengumpulan Data

Tahap Pengumpulan data bertujuan untuk memahami kebutuhan sistem dan mengidentifikasi informasi yang diperlukan dalam pengembangan Aplikasi Kasir Agen Portable Berbasis Mobile. Pada tahap ini, dilakukan serangkaian proses untuk mengumpulkan data yang menjadi dasar dalam merancang sistem. Proses tersebut meliputi observasi langsung dan wawancara dengan pihak terkait di CV Arjuna Jati Jepara.

3.1.1 Wawancara

Wawancara dilakukan dengan pihak pengelola CV Arjuna Jati Jepara untuk menggali informasi lebih dalam terkait kebutuhan sistem dan permasalahan yang dihadapi. Proses wawancara ini bertujuan untuk mengetahui preferensi pengguna terhadap fitur-fitur yang dibutuhkan dalam aplikasi kasir, serta memperoleh data penting yang tidak dapat diperoleh melalui observasi saja. Pertanyaan yang diajukan meliputi proses pencatatan transaksi, metode penyimpanan data saat ini, serta harapan pengelola terhadap sistem yang akan dikembangkan.

Wawancara dilakukan secara langsung pada tanggal 11 Agustus 2024 dengan pemilik CV Arjuna Jati Jepara untuk mengidentifikasi permasalahan utama yang dihadapi dalam proses pencatatan transaksi penjualan. Dari wawancara ini, beberapa poin penting berhasil dikumpulkan, antara lain:

a. Permasalahan Pencatatan Transaksi Manual

Pencatatan transaksi masih dilakukan secara manual menggunakan buku dan lembar kerja *Excel*, yang menyebabkan beberapa kendala, seperti:

1. Ketidakteraturan data yang tercatat.
2. Risiko kehilangan data akibat *human error* atau kerusakan dokumen.
3. Kesulitan dalam pembuatan laporan penjualan yang cepat dan akurat.

b. Kebutuhan akan Sistem Kasir Berbasis *Mobile*

Pemilik mengungkapkan kebutuhan akan sistem kasir berbasis *mobile* yang mampu:

1. Mencatat transaksi secara otomatis dan terstruktur.
2. Menyediakan laporan penjualan secara *real-time*.
3. Memiliki antarmuka yang mudah digunakan oleh pengguna dengan berbagai tingkat keterampilan teknologi.

c. Harapan dari Aplikasi yang Dikembangkan

Pemilik berharap aplikasi ini tidak hanya membantu dalam efisiensi pencatatan transaksi tetapi juga mampu meningkatkan kecepatan pelayanan kepada pelanggan. Selain itu, sistem diharapkan dapat diakses secara fleksibel melalui perangkat *mobile* untuk mendukung mobilitas operasional.

Hasil wawancara ini menjadi dasar dalam merumuskan kebutuhan sistem pada tahap berikutnya. Semua informasi yang diperoleh difokuskan untuk menghasilkan solusi yang relevan dengan kebutuhan operasional CV Arjuna Jati Jepara.

3.1.2 Observasi

Observasi langsung dilakukan dengan mendatangi lokasi CV Arjuna Jati Jepara pada tanggal 12 Agustus 2024 untuk mengamati aktivitas operasional, khususnya proses pencatatan transaksi. Melalui observasi ini, peneliti berusaha memahami secara nyata bagaimana prosedur yang digunakan dalam pencatatan penjualan, alat atau media yang digunakan, serta tantangan yang dihadapi dalam proses tersebut. Observasi ini juga membantu peneliti dalam mengetahui kebiasaan dan pola kerja yang nantinya akan diakomodasi dalam pengembangan sistem kasir berbasis *mobile*.

Berdasarkan pengamatan, transaksi penjualan masih dicatat secara manual menggunakan buku kas atau *Microsoft Excel*. Proses ini menghadirkan beberapa kendala, seperti ketidakkonsistenan format pencatatan, waktu yang lebih lama untuk mencatat transaksi, terutama saat terjadi lonjakan transaksi, serta risiko kehilangan data akibat kerusakan buku kas atau file yang tidak tersimpan dengan baik. Selain itu, penyajian laporan penjualan, baik harian, mingguan, maupun bulanan, masih dilakukan secara manual, yang memakan waktu dan rentan terhadap kesalahan perhitungan.

Observasi juga mengungkapkan bahwa karyawan memiliki akses ke perangkat *mobile* berbasis Android, sehingga memungkinkan pengembangan sistem kasir berbasis *mobile*. Namun, variasi tingkat pemahaman teknologi di antara karyawan menjadi faktor yang harus diperhatikan dalam pengembangan aplikasi, terutama terkait dengan desain antarmuka yang sederhana dan mudah digunakan. Informasi yang diperoleh dari observasi ini melengkapi hasil wawancara sebelumnya dan menjadi dasar penting dalam menyusun spesifikasi kebutuhan sistem.

3.2 Identifikasi Masalah

Berdasarkan hasil wawancara dan observasi yang telah dilakukan, beberapa permasalahan utama dalam pengelolaan transaksi di CV Arjuna Jati Jepara berhasil diidentifikasi. Proses pencatatan transaksi masih dilakukan secara manual menggunakan buku kas atau *Microsoft Excel*, yang membutuhkan waktu lama dan kurang efisien, terutama saat *volume* transaksi tinggi. Selain itu, sering terjadi kesalahan dalam pencatatan dan perhitungan, yang dapat memengaruhi akurasi laporan keuangan. Pencatatan yang tersebar di berbagai dokumen fisik juga menyulitkan pembuatan laporan penjualan, karena memerlukan waktu lebih banyak untuk mengumpulkan dan merekap data.

Permasalahan lain yang teridentifikasi adalah risiko keamanan data, karena penyimpanan data masih dalam bentuk fisik yang rentan terhadap kerusakan atau kehilangan. Di sisi lain, pengelolaan stok barang dan pelacakan performa penjualan juga menjadi tantangan, karena belum ada sistem terintegrasi yang dapat membantu memantau secara *real-time*. Menghadapi berbagai kendala ini, CV Arjuna Jati Jepara membutuhkan solusi berbasis teknologi yang dapat meningkatkan efisiensi, akurasi, dan keamanan dalam pengelolaan transaksi.

3.3 Perancangan aplikasi

Tahap ini bertujuan untuk merancang sistem berdasarkan kebutuhan yang telah diidentifikasi pada tahap sebelumnya. Pada tahap ini, dilakukan perancangan komponen sistem yang meliputi struktur *database*, arsitektur sistem berbasis Android, dan diagram *UML/Use Case* untuk memvisualisasikan fungsionalitas sistem secara jelas.

3.3.1 Analisis kebutuhan

Terdapat 3 bagian dalam tahapan analisis ini, yakni analisis kebutuhan masukan, analisis kebutuhan proses, dan analisis kebutuhan keluaran. Setiap aktor memiliki kebutuhan masing-masing yang nantinya akan dijelaskan dengan tabel yang berbeda. Berikut penjelasan analisis kebutuhan masukan, proses, dan keluaran.

a. Perusahaan

Perusahaan merupakan pemilik atau pengelola CV Arjuna Jati Jepara. Didapatkan analisis kebutuhan untuk perusahaan yaitu, proses *login*, kelola karyawan, kelola produk, kelola pembayaran, kelola target, kelola penjualan, kelola mutasi, dan pengaturan. Berikut ini penjelasan lebih rinci mengenai proses, masukan (*input*), dan keluaran (*output*) pada Tabel 3.

Tabel 3. 1 Analisis Kebutuhan Sistem (Perusahaan)

No.	Proses	Masukan (<i>Input</i>)	Keluaran (<i>Output</i>)
1	Login	Nama pengguna dan kata sandi	Halaman utama Perusahaan (<i>Dashboard</i>)
2	Kelola Karyawan (menambah, menghapus, mengubah, melihat)	Data karyawan (nama, nomor telepon, nama pengguna, kata sandi)	Informasi daftar karyawan
3	Kelola Produk (menambah, menghapus, mengubah, melihat Data produk (nama, stok, harga))	Data produk (nama, stok, harga)	Informasi daftar produk
4	Kelola Pembayaran	Data pembeli, produk yang dibeli	Struk pembayaran
5	Kelola Target	Data target penjualan karyawan	Informasi target karyawan
6	Kelola Penjualan	Data pembeli, produk yang dibeli	Laporan penjualan
7	Kelola Mutasi	Filter periode laporan	File CSV laporan keuangan
8	Pengaturan Perusahaan	Data perusahaan (nama, alamat, nomor telepon)	Informasi profil perusahaan
9	Pengaturan Akun	Data akun (nama, kata sandi)	Informasi profil akun

b. Karyawan

Karyawan merupakan pegawai CV Arjuna Jati Jepara. Didapatkan analisis kebutuhan untuk karyawan yaitu, proses *login*, lihat produk, kelola pembayaran, lihat target, kelola penjualan, dan pengaturan akun. Berikut ini penjelasan lebih rinci mengenai proses, masukan (*input*), dan keluaran (*output*) pada Tabel 3. 2

Tabel 3. 2 Analisis Kebutuhan (Karyawan)

No.	Proses	Masukan (<i>Input</i>)	Keluaran (<i>Output</i>)
1	Login	Nama pengguna dan kata sandi	Halaman utama Perusahaan (<i>Dashboard</i>)
2	Lihat Produk	Kata kunci pencarian (opsional)	Informasi stok produk
3	Kelola Pembayaran	Data pembeli, produk yang dibeli	Struk pembayaran
4	Lihat Target	-	Informasi target dan pencapaian
5	Kelola Penjualan	Data Transaksi	Laporan Penjualan
6	Pengaturan Akun	Data akun (nama, kata sandi)	Informasi profil akun

3.3.2 Perancangan Struktur *Database*

Perancangan struktur *database* dilakukan untuk menyimpan data transaksi secara terstruktur dan efisien. Struktur *database* ini mencakup tabel-tabel yang diperlukan, seperti data produk, data transaksi penjualan, dan data pengguna. Perancangan dilakukan dengan mempertimbangkan relasi antar tabel agar memudahkan proses pengolahan dan pengambilan data.

3.3.3 Perancangan Arsitektur Android

Perancangan arsitektur Android dilakukan untuk menentukan alur kerja sistem dalam aplikasi berbasis mobile. Arsitektur ini mencakup desain logis dari komponen-komponen aplikasi, seperti tampilan antarmuka pengguna (*UI*), komunikasi dengan *database*, serta pengolahan data transaksi. Arsitektur ini memastikan aplikasi dapat berjalan secara optimal dan memenuhi kebutuhan pengguna.

3.3.4 Perancangan *UML/Use Case Diagram*

Use Case Diagram digunakan untuk memvisualisasikan fungsionalitas sistem dan interaksi antara pengguna dengan sistem. Diagram ini menggambarkan bagaimana pengguna (*admin* atau kasir) berinteraksi dengan fitur-fitur utama aplikasi, seperti input data transaksi, pengelolaan produk, penyimpanan data, dan pembuatan laporan penjualan.

3.4 Coding Android

Tahap ini merupakan proses implementasi dari hasil perancangan sistem yang telah dilakukan pada tahap sebelumnya. Pada tahap ini, sistem mulai dikembangkan menggunakan teknologi dan tools yang sesuai untuk menghasilkan Aplikasi Kasir Agen Portable Berbasis *Mobile*.

3.4.1 Coding Menggunakan Kotlin dan Android Studio

Pengembangan aplikasi dilakukan dengan menggunakan bahasa pemrograman *Kotlin* yang dijalankan melalui *Android Studio* sebagai *Integrated Development Environment (IDE)*. *Kotlin* dipilih karena kompatibilitasnya yang tinggi dengan Android dan kemampuannya untuk menghasilkan kode yang lebih ringkas, efisien, dan mudah dipahami.

3.4.2 Layout Menggunakan XML

Desain antarmuka pengguna (UI) dikembangkan menggunakan XML (*Extensible Markup Language*). Layout dirancang agar memiliki tampilan yang responsif, intuitif, dan *user-friendly*, sehingga memudahkan pengguna dalam melakukan input data transaksi dan mengelola aplikasi. Desain ini mencakup tampilan untuk fitur-fitur utama, seperti input produk, transaksi penjualan, dan tampilan laporan.

3.4.3 Database Menggunakan Firebase

Untuk penyimpanan data, aplikasi menggunakan *Firebase Realtime Database*. *Firebase* dipilih karena kemampuannya menyimpan dan mengelola data secara *real-time* serta kemudahan dalam integrasi dengan aplikasi Android. Data yang disimpan meliputi informasi produk, transaksi penjualan, dan data pengguna, sehingga memungkinkan pengambilan dan pengolahan data yang cepat dan akurat.

3.5 Testing

Tahap *Testing* bertujuan untuk memastikan bahwa aplikasi yang dikembangkan dapat berfungsi dengan baik dan sesuai dengan kebutuhan yang telah diidentifikasi pada tahap sebelumnya. Pengujian dilakukan untuk mengevaluasi fungsionalitas, performa, dan kegunaan aplikasi secara menyeluruh.

3.5.1 Black-Box Testing

Pengujian pertama yang dilakukan adalah *black-box testing*, yang fokus pada pengujian fungsionalitas aplikasi tanpa melihat kode sumber atau struktur internal aplikasi. Dalam pengujian ini, setiap fitur aplikasi diuji berdasarkan bagaimana aplikasi berperilaku sesuai dengan input yang diberikan. *Black-box testing* dilakukan untuk memastikan bahwa aplikasi dapat berjalan sesuai dengan yang diharapkan tanpa ada bug atau kesalahan dalam fungsionalitas utama.

Tabel 3. 3 Tabel skenario pengujian *black-box* pada akun Perusahaan

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan
1	Login	Memasukkan nama pengguna () dan kata sandi pengguna yang valid	Sistem menampilkan halaman utama

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan
			sesuai dengan jenis akun (perusahaan)
2	Login	Memasukkan nama pengguna dan kata sandi pengguna yang salah	Sistem menampilkan pesan kesalahan "Nama pengguna belum terdaftar"
3	Login	Mengosongkan <i>form</i> nama pengguna dan kata sandi atau mengisi salah satu	Sistem menampilkan pesan kesalahan "Nama pengguna tidak boleh kosong" "Kata sandi tidak boleh kosong"
4	Halaman Home	Masuk pada halaman <i>home</i> yang menampilkan pendapatan hari ini dan beberapa menu yang sesuai dengan jenis akun perusahaan	Sistem menampilkan pendapatan hari ini dan menampilkan menu yang sesuai dengan jenis akun
5	Karyawan (Tambah karyawan)	Mengisi semua <i>form</i> pada tambah karyawan dengan data yang valid dan menekan tombol "Daftar"	Sistem menyimpan data karyawan baru dan memperbarui daftar karyawan di halaman terkait.
6	Karyawan (Tambah karyawan)	Mengosongkan salah satu form dan menekan tombol "Daftar"	Sistem menampilkan pesan kesalahan "Harap lengkapi <i>form</i> terlebih dahulu."
7	Karyawan (Edit karyawan)	Mengedit data karyawan dan menekan tombol "simpan"	Sistem menyimpan data karyawan yang sudah diperbaharui.
8	Karyawan (Hapus karyawan)	Menghapus Karyawan dengan menekan tombol hapus.	Sistem menampilkan peringatan kepada pengguna apakah yakin untuk menghapus data karyawan jika yakin maka pengguna menekan tombol hapus dan data karyawan terhapus.
9	Produk (Tambah Produk)	Menambahkan produk dengan mengisi form pada halaman tambah produk dan menekan	Sistem menyimpan data produk yang baru ditambahkan dan

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan
		tombol “Tambah Produk”	menampilkannya di halaman produk.
10	Produk (Edit Produk)	Mengedit data produk dengan mengisi atau mengganti data produk pada halaman edit produk dengan <i>update</i> terbaru dan menekan tombol “Edit Produk”	Sistem menyimpan <i>update</i> data produk terbaru dan menampilkannya pada halaman produk
11	Pembayaran	Memasukkan data pembeli dan produk yang dibeli, lalu menekan tombol “Bayar.”	Sistem menyimpan transaksi, memperbarui stok produk, dan mencetak struk melalui printer yang terhubung.
12	Target	Menambahkan target penjualan bulanan dan menekan tombol “Simpan.”	Sistem menyimpan target penjualan dan menampilkan progres target tiap karyawan.
13	Penjualan	Menampilkan daftar penjualan, mengedit transaksi, menghapus transaksi, dan mencetak struk.	Sistem mampu menampilkan data transaksi, mengedit atau menghapus transaksi, serta mencetak ulang struk.
14	Mutasi	Memilih filter tertentu untuk melihat laporan penjualan, lalu menekan tombol "Unduh".	Sistem menghasilkan file CSV sesuai dengan filter yang dipilih dan mengunduhnya ke perangkat pengguna.
15	Pengaturan Perusahaan	Mengedit data perusahaan, seperti nama atau alamat, lalu menekan tombol "Simpan".	Sistem memperbarui data perusahaan yang ditampilkan di halaman terkait.
16	Pengaturan Akun	Mengubah kata sandi atau mengatur printer <i>Bluetooth</i> , lalu menekan tombol "Simpan."	Sistem memperbarui kata sandi dan pengaturan printer, serta menampilkan pesan "Perubahan berhasil disimpan."

Pada Tabel 3. 3 merupakan sekenario pengujian *Black-box* untuk akun Perusahaan yang mencakup berbagai fitur utama seperti login, manajemen karyawan, pengelolaan produk,

sistem pembayaran, pengaturan target penjualan, pencatatan transaksi, pembuatan laporan mutasi, serta pengaturan profil perusahaan dan akun. Setiap fitur diuji dengan berbagai skenario untuk memastikan sistem dapat berjalan sesuai dengan hasil yang diharapkan dan dapat menangani berbagai kondisi *input* dengan tepat.

Tabel 3. 4 Tabel skenario pengujian *black-box* pada akun karyawan

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan
1	Login	Memasukkan nama pengguna dan kata sandi yang <i>valid</i>	Sistem menampilkan halaman utama untuk akun karyawan sesuai dengan hak aksesnya
2	Login	Memasukkan nama pengguna atau kata sandi yang salah	Sistem menampilkan pesan kesalahan "Nama pengguna atau kata sandi salah"
3	Login	Mengosongkan form nama pengguna atau kata sandi	Sistem menampilkan pesan "Nama pengguna/kata sandi tidak boleh kosong"
4	Halaman Home	Masuk ke halaman utama yang menampilkan informasi pendapatan harian dan progres target bulanan	Sistem menampilkan pendapatan harian dan progres target sesuai data karyawan
5	Produk (View)	Mengakses menu produk untuk melihat stok produk yang tersedia	Sistem menampilkan daftar produk dengan informasi stok terkini
6	Produk (Search)	Menggunakan fitur pencarian untuk mencari produk tertentu	Sistem menampilkan produk yang sesuai dengan kata kunci pencarian
7	Target (View)	Mengakses menu target untuk melihat target bulanan karyawan	Sistem menampilkan target bulanan karyawan, rata-rata pencapaian, dan persentase

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan
8	Pembayaran	Memasukkan data pembeli dan produk yang dibeli, lalu menekan tombol "Bayar"	Sistem menyimpan transaksi dan mencetak struk pembayaran
9	Penjualan (View)	Melihat riwayat transaksi yang telah dilakukan	Sistem menampilkan daftar transaksi dengan detail seperti waktu, total, dan kasir
10	Penjualan (Print)	Menekan tombol <i>print</i> untuk mencetak ulang struk pembayaran dari transaksi tertentu	Sistem mencetak ulang struk transaksi menggunakan printer yang terhubung
11	Pengaturan Akun	Mengubah kata sandi atau pengaturan printer lalu menekan tombol "Simpan"	Sistem memperbarui pengaturan akun dan menampilkan pesan "Perubahan berhasil"

Pada Tabel 3. 4 merupakan skenario pengujian *black-box* untuk fitur yang ada pada jenis akun Karyawan yang mencakup fitur-fitur penting seperti *login*, akses halaman utama untuk melihat pendapatan dan target, manajemen produk termasuk pencarian dan pemantauan stok, pengelolaan target penjualan, sistem pembayaran dan transaksi, serta pengaturan akun pribadi. Pengujian ini dilakukan untuk memastikan bahwa setiap fitur yang tersedia untuk akun Karyawan dapat berfungsi dengan baik dan sesuai dengan hak akses yang diberikan.

3.5.2 Usability Testing dengan System Usability Scale (SUS)

Selain pengujian fungsionalitas, dilakukan juga *usability testing* menggunakan metode *System Usability Scale (SUS)* untuk menilai kemudahan penggunaan aplikasi oleh pengguna akhir. *SUS* adalah metode pengukuran yang melibatkan pengisian kuesioner oleh pengguna setelah mereka menggunakan aplikasi, untuk mengukur kepuasan pengguna terhadap desain antarmuka dan kemudahan penggunaan aplikasi. Berdasarkan hasil pengujian ini, dapat diketahui seberapa baik aplikasi dalam memberikan pengalaman pengguna yang memuaskan.

Tabel 3. 5 Tabel pernyataan *SUS*

No.	Kode	Pernyataan
1	Pt1	Saya merasa aplikasi kasir ini akan sangat membantu pekerjaan sehari-hari
2	Pt2	Saya merasa tampilan menu dalam aplikasi ini membingungkan
3	Pt3	Saya dapat menggunakan fitur-fitur aplikasi ini tanpa kesulitan
4	Pt4	Saya kesulitan memahami cara mengoperasikan aplikasi ini sendirian
5	Pt5	Saya merasa proses transaksi dalam aplikasi ini berjalan dengan lancar
6	Pt6	Saya sering mengalami error saat menggunakan aplikasi ini
7	Pt7	Saya yakin rekan kerja saya dapat menggunakan aplikasi ini dengan mudah
8	Pt8	Saya merasa prosedur dalam aplikasi ini terlalu berbelit-belit
9	Pt9	Saya merasa nyaman menggunakan semua fitur dalam aplikasi ini
10	Pt10	Saya membutuhkan waktu lama untuk mahir menggunakan aplikasi ini

Pada Tabel 3. 5 telah disusun sepuluh pernyataan *System Usability Scale (SUS)* yang digunakan untuk mengukur tingkat kegunaan dan kemudahan penggunaan aplikasi kasir. Pernyataan ini dirancang untuk mengevaluasi berbagai aspek pengalaman pengguna, mulai dari kemudahan pengoperasian, efisiensi dalam penggunaan, hingga tingkat kenyamanan pengguna saat berinteraksi dengan aplikasi. Setiap pertanyaan disusun dengan menggunakan bahasa yang mudah dipahami dan mencakup baik pernyataan positif maupun negatif untuk mendapatkan penilaian yang objektif dari pengguna.

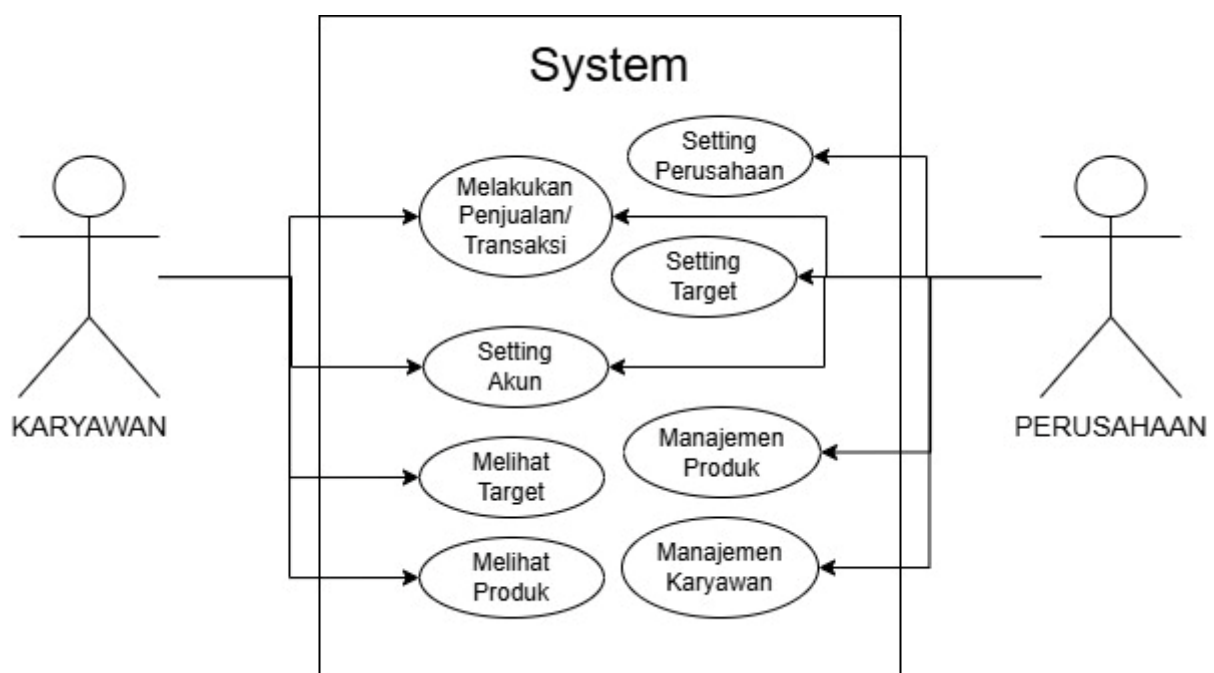
BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Perancangan Aplikasi

4.1.1 Hasil Perancangan UML/Use Case Diagram

Perancangan sistem diawali dengan membuat *Use Case Diagram* untuk memvisualisasikan kebutuhan fungsional dari aplikasi kasir berbasis mobile yang dikembangkan. *Use Case Diagram* ini digunakan untuk menggambarkan interaksi antara pengguna (aktor) dengan sistem, serta fungsi-fungsi utama yang tersedia dalam aplikasi.



Gambar 4. 1 *Use Case Diagram*

Pada Gambar 4. 1 terlihat ada 2 aktor utama yaitu Perusahaan dan karyawan yang berperan pada proses system tersebut dengan beberapa fitur utama yaitu:

a. *Register* perusahaan

Fitur ini memungkinkan calon perusahaan (*Customer* baru) untuk mendaftar ke dalam aplikasi dan memulai proses penggunaan sistem.

b. *Login*

Fitur *login* memberikan akses bagi Perusahaan maupun Karyawan untuk masuk ke dalam aplikasi dengan menggunakan akun yang telah terdaftar.

c. Manajemen Perusahaan

Fitur ini memungkinkan perusahaan untuk mengatur profil perusahaan, seperti alamat perusahaan, nomor perusahaan, dan nama perusahaan.

d. Manajemen Karyawan

Fitur ini berfungsi untuk memudahkan perusahaan dalam mengelola data karyawan, termasuk menambahkan akun karyawan baru, mengedit data karyawan, melihat data karyawan yang terdaftar, dan menghapus karyawan yang tidak aktif.

e. Manajemen produk

Fitur ini memungkinkan perusahaan untuk menambahkan produk, *update* stok produk, edit harga produk, edit produk, dan melihat daftar produk yang ada dalam sistem.

f. Fitur transaksi

Fitur ini meliputi input penjualan, lihat penjualan, cetak struk, dan edit penjualan, yang memungkinkan karyawan untuk memproses transaksi penjualan dengan mudah dan cepat.

g. Manajemen Akun

Fitur ini memungkinkan pengguna untuk mengganti kata sandi, mengganti nama, serta mengelola printer untuk mencetak struk transaksi.

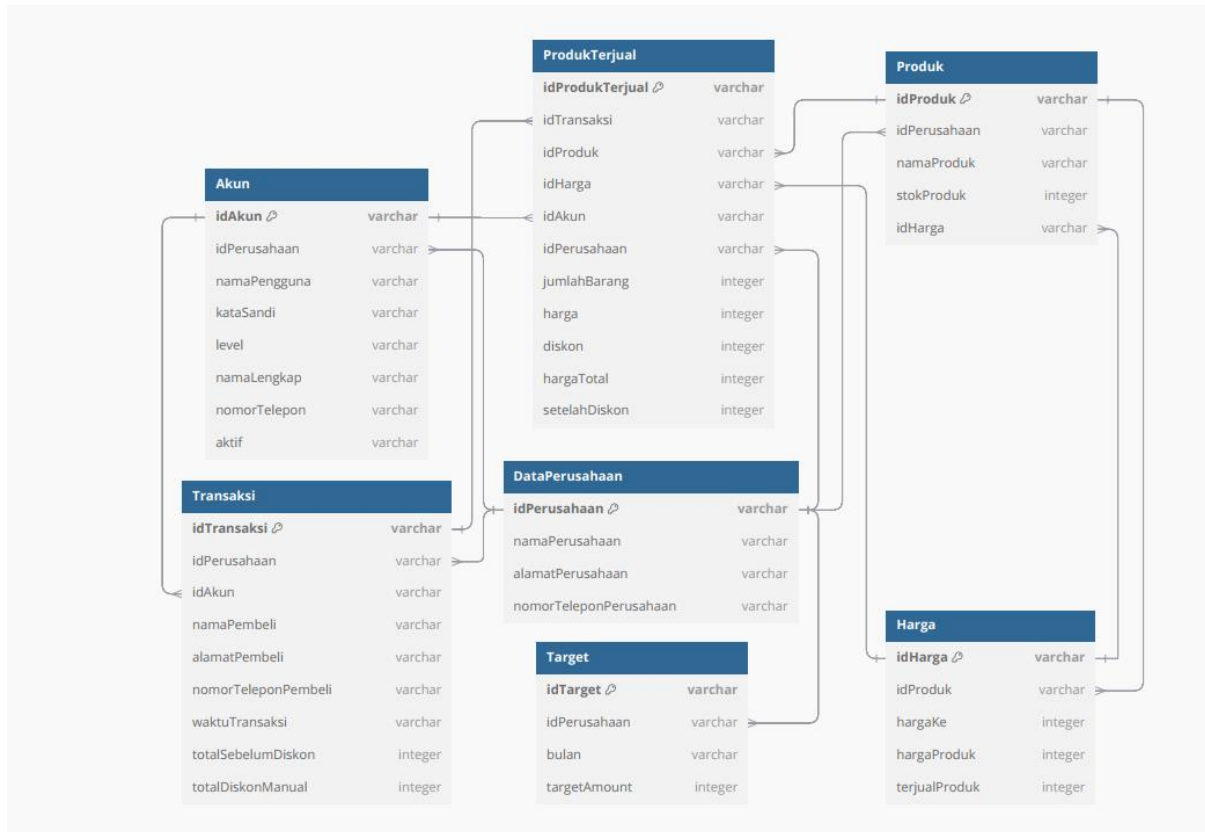
h. Edit Target

Fitur ini berfungsi bagi perusahaan untuk mengatur dan mengedit target penjualan yang diberikan kepada masing-masing karyawan, sesuai dengan kebutuhan dan tujuan perusahaan.

i. Lihat Target

Fitur ini memberikan akses bagi karyawan untuk melihat target penjualan yang telah ditetapkan oleh perusahaan, sehingga mereka dapat memonitor kinerja mereka secara lebih jelas.

4.1.2 Hasil Perancangan Struktur *Database*



Gambar 4. 2 Struktur *Database Relasional*

Gambar 4. 2 menunjukkan struktur *database* yang terdiri dari beberapa tabel yang saling terhubung untuk mendukung operasional sistem. Berikut adalah rincian dari masing-masing tabel:

a. Tabel Akun

Tabel ini berfungsi untuk menyimpan informasi pengguna sistem, baik perusahaan maupun karyawan.

- idAkun: *Primary Key*, identitas unik setiap akun.
- idPerusahaan: *Foreign Key*, menghubungkan dengan tabel DataPerusahaan.
- namaPengguna: Nama pengguna untuk *login*.
- kataSandi: Kata sandi pengguna.
- level: Menentukan level akses (misalnya karyawan atau perusahaan).
- namaLengkap: Nama lengkap pengguna.
- nomorTelepon: Kontak pengguna.
- aktif: Status aktif akun.

b. Tabel DataPerusahaan

Tabel ini menyimpan informasi mengenai perusahaan yang menggunakan sistem.

- idPerusahaan: *Primary Key*, identitas unik perusahaan.
- namaPerusahaan: Nama perusahaan.
- alamatPerusahaan: Alamat perusahaan.
- nomorTeleponPerusahaan: Kontak perusahaan.

c. Tabel Produk

Tabel ini menyimpan data produk yang dikelola dalam sistem.

- idProduk: *Primary Key*, identitas unik produk.
- idPerusahaan: *Foreign Key*, menghubungkan dengan DataPerusahaan.
- namaProduk: Nama produk.
- stokProduk: Jumlah stok produk tersedia.
- idHarga: *Foreign Key*, menghubungkan dengan tabel Harga.

d. Tabel Harga

Tabel ini menyimpan informasi harga dari setiap produk.

- idHarga: *Primary Key*, identitas unik harga.
- idProduk: *Foreign Key*, menghubungkan dengan Produk.
- hargaKe: Indikator versi harga (misalnya harga 1, 2, dsb).
- hargaProduk: Harga jual produk.
- terjualProduk: Jumlah produk yang terjual.

e. Tabel Transaksi

Tabel ini menyimpan informasi transaksi yang dilakukan dalam sistem.

- idTransaksi: *Primary Key*, identitas unik transaksi.
- idPerusahaan: *Foreign Key*, menghubungkan dengan DataPerusahaan.
- idAkun: *Foreign Key*, menghubungkan dengan Akun (pengguna yang melakukan transaksi).
- namaPembeli: Nama pembeli dalam transaksi.
- alamatPembeli: Alamat pembeli.
- nomorTeleponPembeli: Kontak pembeli.
- waktuTransaksi: Waktu transaksi dilakukan.
- totalSebelumDiskon: Total harga sebelum diskon diterapkan.
- totalDiskonManual: Diskon manual yang diberikan.

f. Tabel ProdukTerjual

Tabel ini mencatat detail produk yang terjual dalam setiap transaksi.

- idProdukTerjual: *Primary Key*, identitas unik produk terjual.
- idTransaksi: *Foreign Key*, menghubungkan dengan Transaksi.
- idProduk: *Foreign Key*, menghubungkan dengan Produk.
- idHarga: *Foreign Key*, menghubungkan dengan Harga.
- idAkun: *Foreign Key*, menghubungkan dengan Akun.
- idPerusahaan: *Foreign Key*, menghubungkan dengan DataPerusahaan.
- jumlahBarang: Jumlah produk yang terjual.
- harga: Harga per unit produk.
- diskon: Diskon yang diberikan.
- hargaTotal: Total harga sebelum diskon.
- setelahDiskon: Total harga setelah diskon diterapkan.

g. Tabel Target

Tabel ini menyimpan target penjualan yang ditetapkan perusahaan untuk masing-masing karyawan.

- idTarget: *Primary Key*, identitas unik target.
- idPerusahaan: *Foreign Key*, menghubungkan dengan DataPerusahaan.
- bulan: Periode target (bulan).
- targetAmount: Nilai target penjualan.

```

"Akun": {
  "aktif": "",
  "idAkun": "",
  "idPerusahaan": "",
  "kataSandi": "",
  "level": "",
  "namaLengkap": "",
  "namaPengguna": "",
  "nomorTelepon": ""
},
"Transaksi": {
  "alamatPembeli": "",
  "idAkun": "",
  "idPerusahaan": "",
  "idTransaksi": "",
  "namaPembeli": "",
  "nomorTeleponPembeli": "",
  "totalDiskonManual": "",
  "totalSebelumDiskon": "",
  "waktuTransaksi": ""
},
"DataPerusahaan": {
  "alamatPerusahaan": "",
  "idPerusahaan": "",
  "namaPerusahaan": "",
  "nomorTeleponPerusahaan": ""
},
"Produk": {
  "idHarga": "",
  "idPerusahaan": "",
  "idProduk": "",
  "namaProduk": "",
  "stokProduk": ""
},
"Harga": {
  "hargaKe": "",
  "hargaProduk": "",
  "idHarga": "",
  "idProduk": "",
  "terjualProduk": ""
},
"Target": {
  "bulan": "",
  "idPerusahaan": "",
  "idTarget": "",
  "targetAmount": ""
},
"ProdukTerjual": {
  "diskon": "",
  "harga": "",
  "hargaTotal": "",
  "idAkun": "",
  "idHarga": "",
  "idPerusahaan": "",
  "idProduk": "",
  "idProdukTerjual": "",
  "idTransaksi": "",
  "jumlahBarang": "",
  "setelahDiskon": ""
}

```

Gambar 4. 2. 1 Struktur Database Non-Relasional

Struktur database relasional tersebut kemudian ditransformasikan menjadi struktur *non-relasional* seperti yang ditunjukkan pada Gambar 4.2.1. Berikut adalah proses transformasinya:

A. Proses Transformasi ke *Non-Relasional*

Dalam transformasi struktur dasar, setiap tabel dalam database relasional dikonversi menjadi node di *Firebase* dengan menggunakan *UUID* sebagai *key* utama untuk identifikasi unik setiap data. Relasi antar tabel yang sebelumnya dihubungkan melalui *foreign key* diubah menjadi sistem referensi ID yang disimpan sebagai properti dalam masing-masing node. Seluruh struktur tabel kemudian dikonversi ke dalam format *JSON tree* yang memungkinkan penyimpanan data secara hierarkis dan lebih fleksibel dalam *Firebase*.

Dalam proses migrasi data dari sistem relasional ke *Firebase*, langkah pertama yang dilakukan adalah mengekstrak data dari tabel *SQL* menggunakan query *SELECT*. Data yang diperoleh kemudian ditransformasi ke format *JSON* yang sesuai dengan struktur *Firebase*. Misalnya, data dari tabel akun akan diubah menjadi node "Akun" di *Firebase* dengan *UUID* sebagai *key* dan properti-properti akun sebagai *value*. Proses ini dilakukan untuk semua tabel yang ada dalam database relasional.

Untuk menangani relasi antar data, *foreign key* yang ada pada tabel *SQL* ditransformasi menjadi referensi ID dalam properti node *Firebase*. Query *JOIN* yang biasa digunakan dalam database relasional digantikan dengan *multiple path queries* di *Firebase*. Sebagai contoh, untuk mendapatkan data produk beserta harganya, alih-alih menggunakan *JOIN*, aplikasi akan melakukan query bertingkat dengan mengakses node Produk terlebih dahulu untuk mendapatkan *idHarga*, kemudian menggunakan *idHarga* tersebut untuk mengakses node Harga.

B. Hasil Transformasi per Node

1. Node Akun:

- *Key*: *UUID* sebagai *identifier* unik akun
- *Value*: Objek berisi properti akun (aktif, *idAkun*, *idPerusahaan*, dst)
- Relasi dengan perusahaan melalui *idPerusahaan*

2. Node DataPerusahaan:

- *Key*: *UUID* perusahaan
- *Value*: Data perusahaan (alamat, nama, nomor telepon)
- Menjadi referensi untuk node lain

3. Node Transaksi:

- *Key*: *UUID* transaksi
- *Value*: Detail transaksi termasuk referensi ke akun dan perusahaan
- *Timestamp* untuk *tracking* waktu transaksi

4. Node Produk:

- *Key: UUID* produk
- *Value:* Informasi produk dengan referensi ke harga
- *Tracking* stok secara *realtime*

5. Node Harga:

- *Key: UUID* harga
- *Value:* Detail harga dan jumlah terjual
- *Versioning* harga melalui properti *hargaKe*

6. Node ProdukTerjual:

- *Key: UUID* produk terjual
- *Value:* Detail penjualan dengan referensi ke transaksi dan produk
- Perhitungan diskon dan total harga

7. Node Target:

- *Key: UUID* target
- *Value:* Target penjualan per periode
- *Tracking* performa melalui *targetAmount*

Transformasi ini menghasilkan struktur database yang lebih fleksibel dan mendukung operasi *realtime*, sekaligus mempertahankan integritas data dan relasi antar entitas. Struktur non-relasional ini memungkinkan aplikasi untuk melakukan operasi *CRUD* dengan lebih efisien dan mendukung skalabilitas sistem sesuai pertumbuhan bisnis.

4.1.3 Hasil Perancangan Arsitektur Android

Pada bagian ini, ditampilkan rancangan diagram aktivitas dan rancangan antarmuka aplikasi sebagai panduan dalam pengembangan sistem. Rancangan ini mencakup alur aktivitas pengguna dan desain tata letak antarmuka untuk memastikan aplikasi dapat berfungsi sesuai dengan kebutuhan yang telah diidentifikasi.

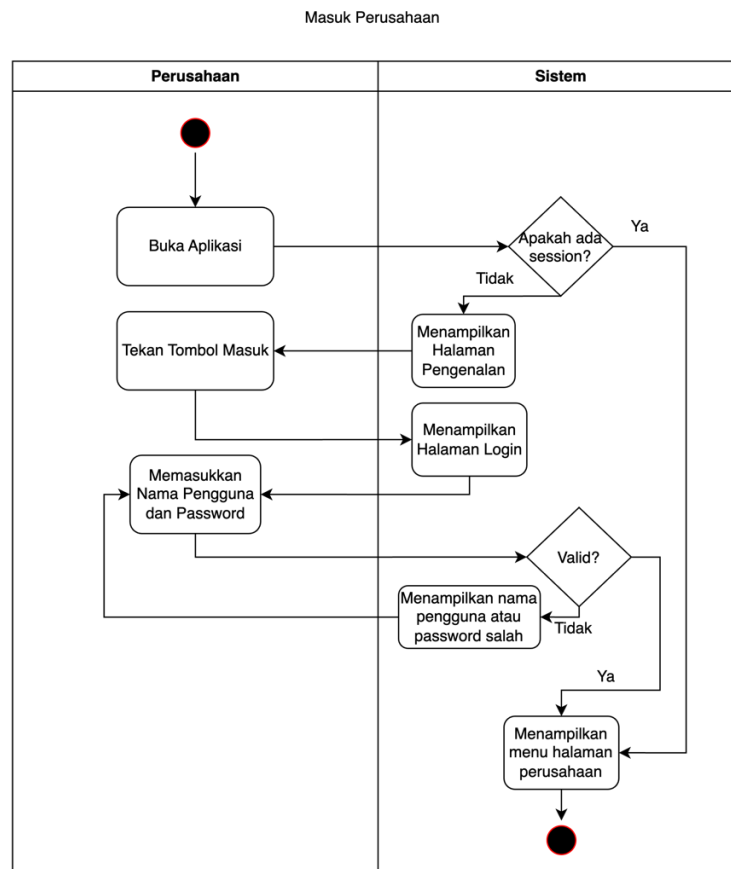
Diagram aktivitas perusahaan

Bagian ini memuat rancangan diagram aktivitas yang menggambarkan alur proses yang dilakukan oleh perusahaan.

1. Diagram aktivitas *login*

Gambar 4. 3 menunjukkan proses login yang dilakukan oleh perusahaan untuk masuk ke dalam aplikasi. Proses login ini merupakan langkah awal yang harus

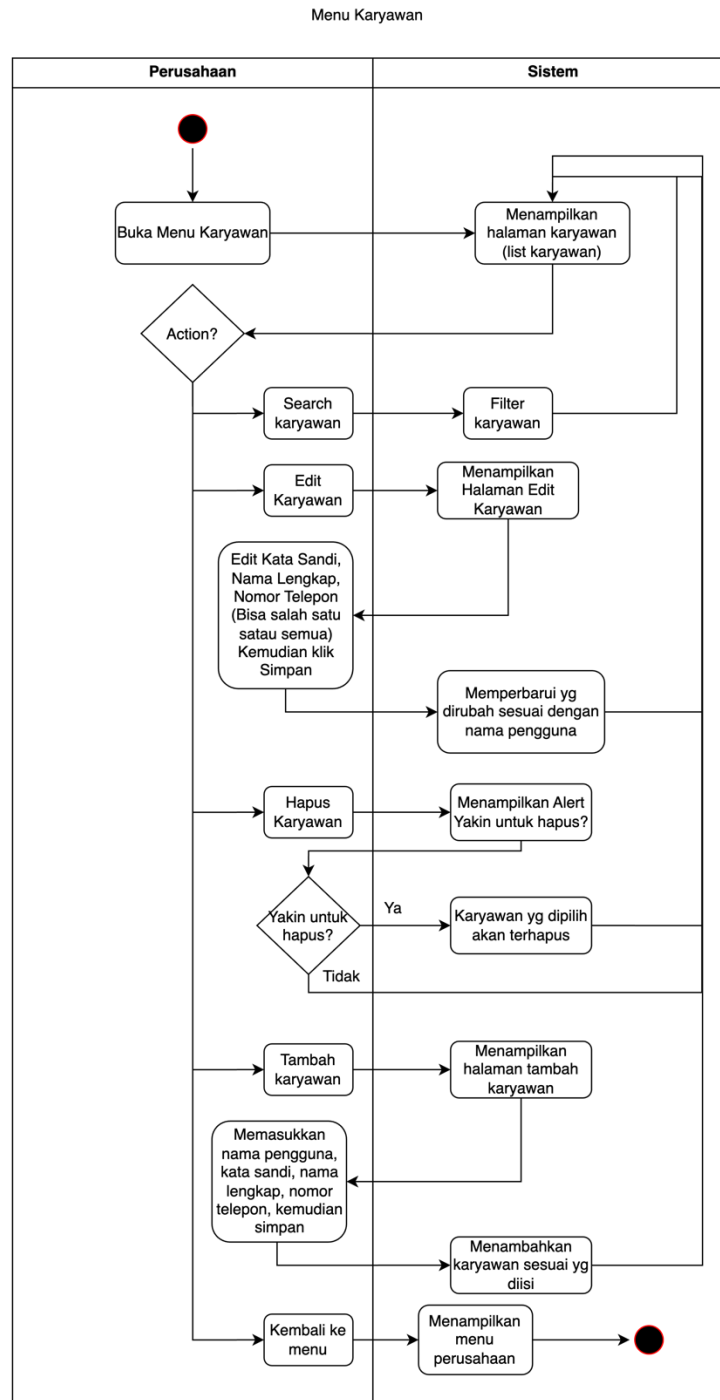
dilakukan agar perusahaan dapat mengakses fitur-fitur yang tersedia dalam aplikasi.



Gambar 4. 3 Diagram aktivitas *login*

2. Diagram aktivitas menu karyawan

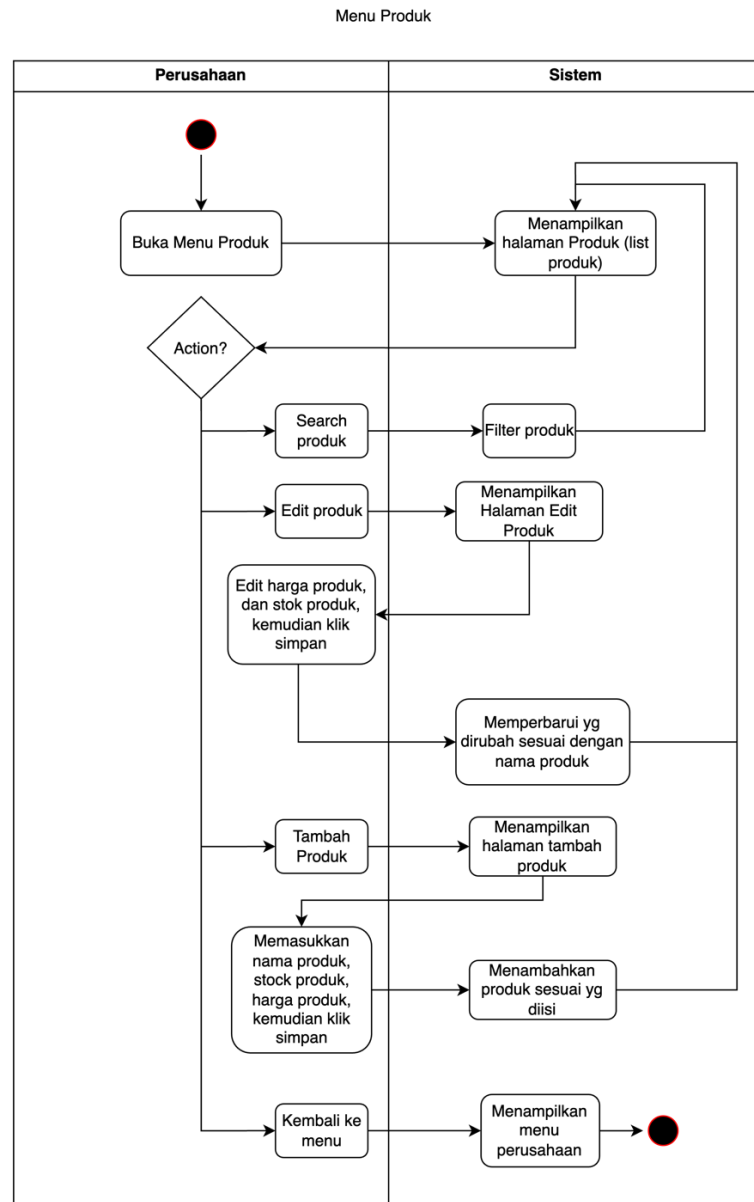
Gambar 4. 4 menggambarkan alur proses yang dilakukan oleh perusahaan dalam mengelola karyawan melalui menu karyawan. Proses ini mencakup penambahan, pengeditan, penghapusan, dan peninjauan data karyawan yang terdaftar dalam sistem.



Gambar 4. 4 Diagram aktivitas menu karyawan

3. Diagram aktivitas menu produk

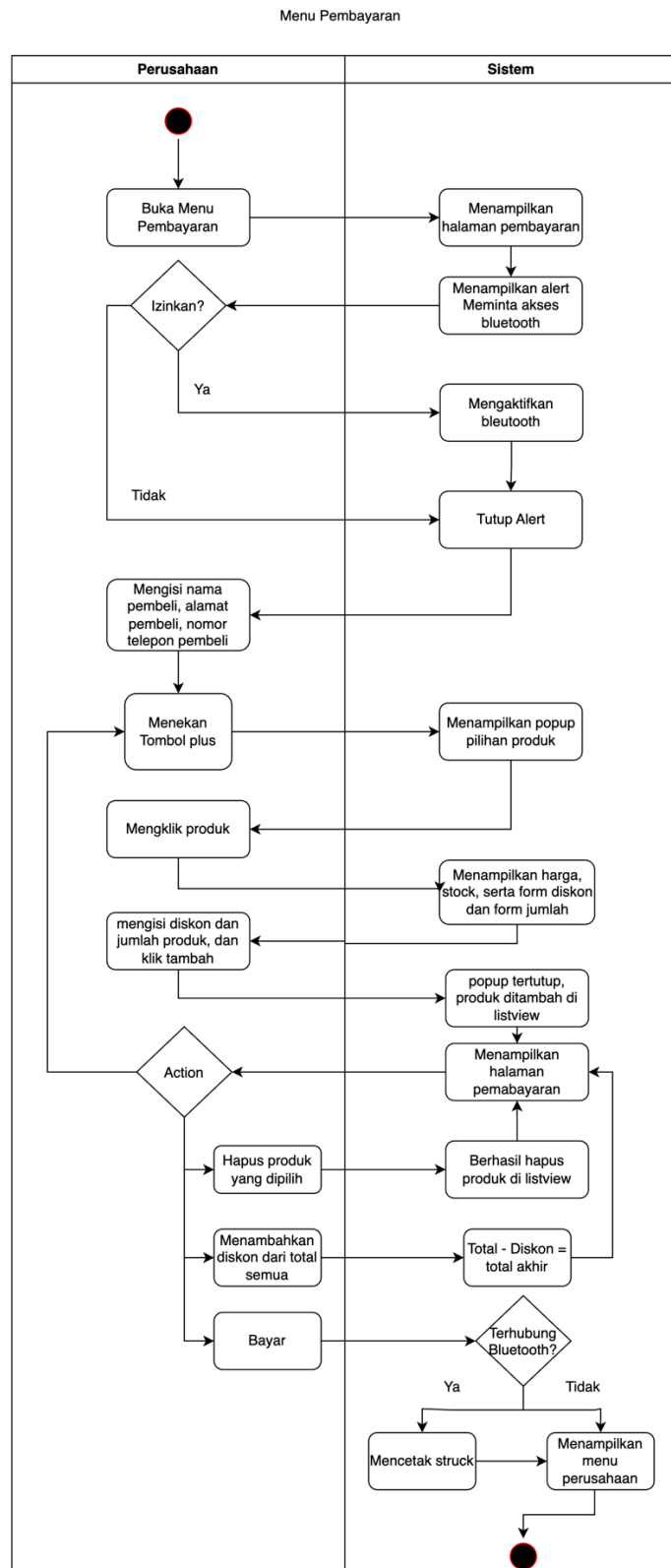
Gambar 4. 5 menampilkan alur proses yang dilakukan oleh perusahaan saat mengakses menu produk. Proses ini mencakup pengelolaan data produk, seperti menambahkan produk baru, memperbarui stok, mengubah informasi produk, dan melihat daftar produk yang tersedia.



Gambar 4. 5 Diagram aktivitas menu produk perusahaan

4. Diagram aktivitas menu Pembayaran

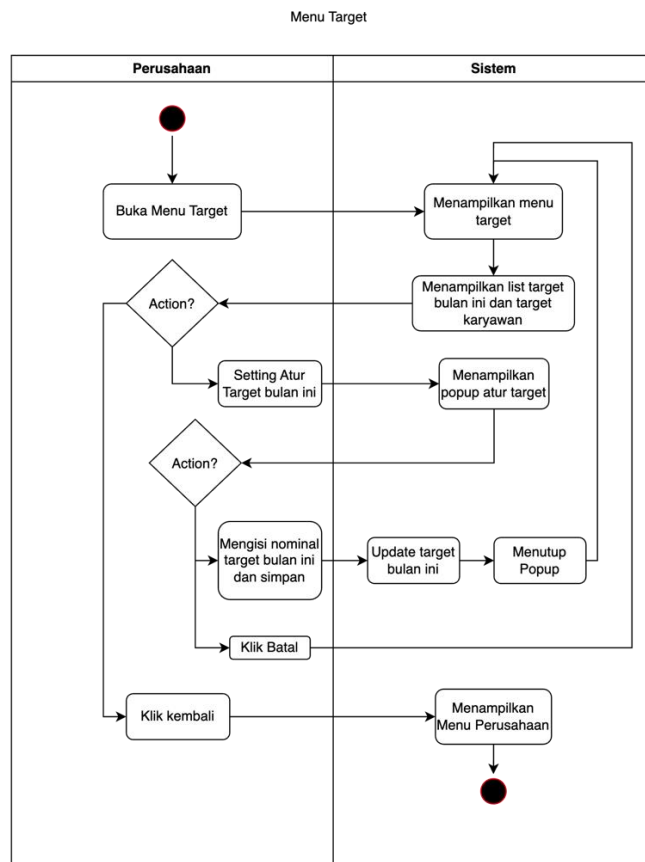
Gambar 4. 6 menampilkan alur proses yang dilakukan oleh Perusahaan saat mengakses dan menggunakan fitur menu pembayaran.



Gambar 4. 6 Diagram aktivitas menu pembayaran perusahaan

5. Diagram aktivitas menu Target

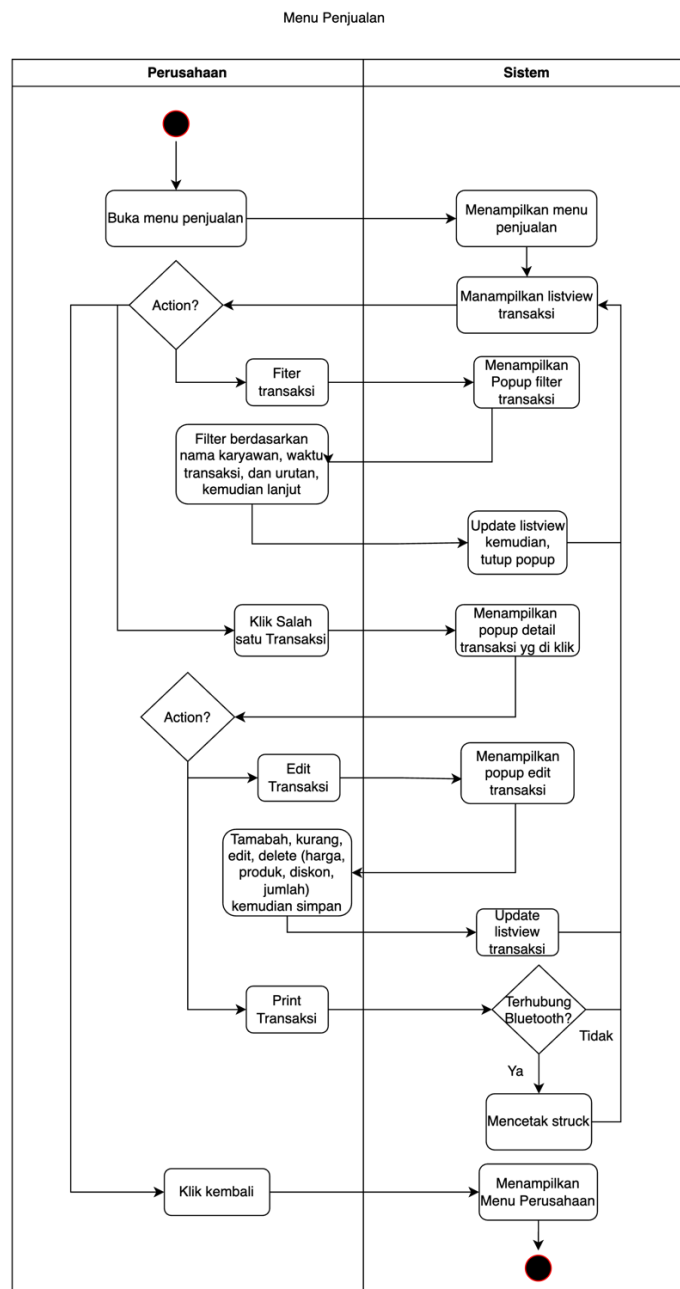
Pada Gambar 4. 7 merupakan Gambaran proses Perusahaan menggunakan fitur untuk mengatur target setiap karyawannya.



Gambar 4. 7 Diagram aktivitas menu target perusahaan

6. Diagram aktivitas menu Penjualan

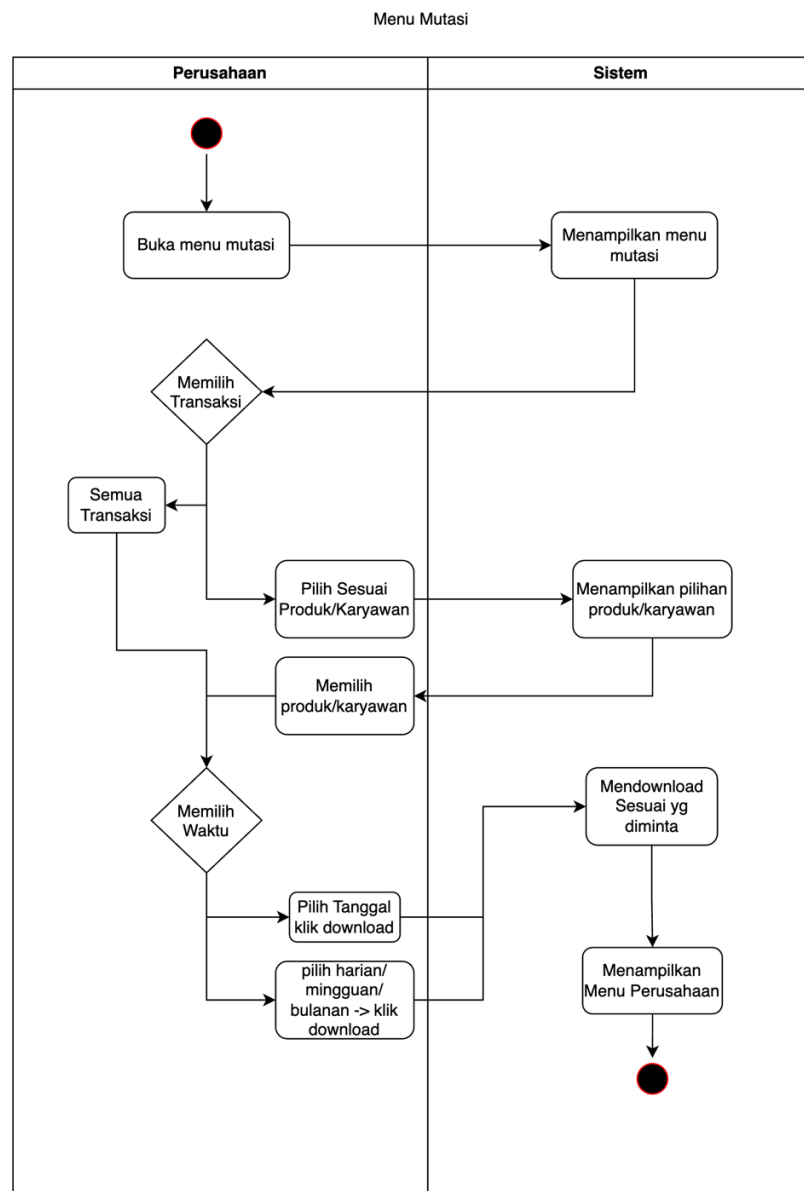
Pada Gambar 4. 8 disajikan Gambaran proses Perusahaan mengelola penjualan yang telah mereka lakukan sebelumnya.



Gambar 4. 8 Diagram aktivitas menu penjualan perusahaan

7. Diagram aktivitas menu mutasi

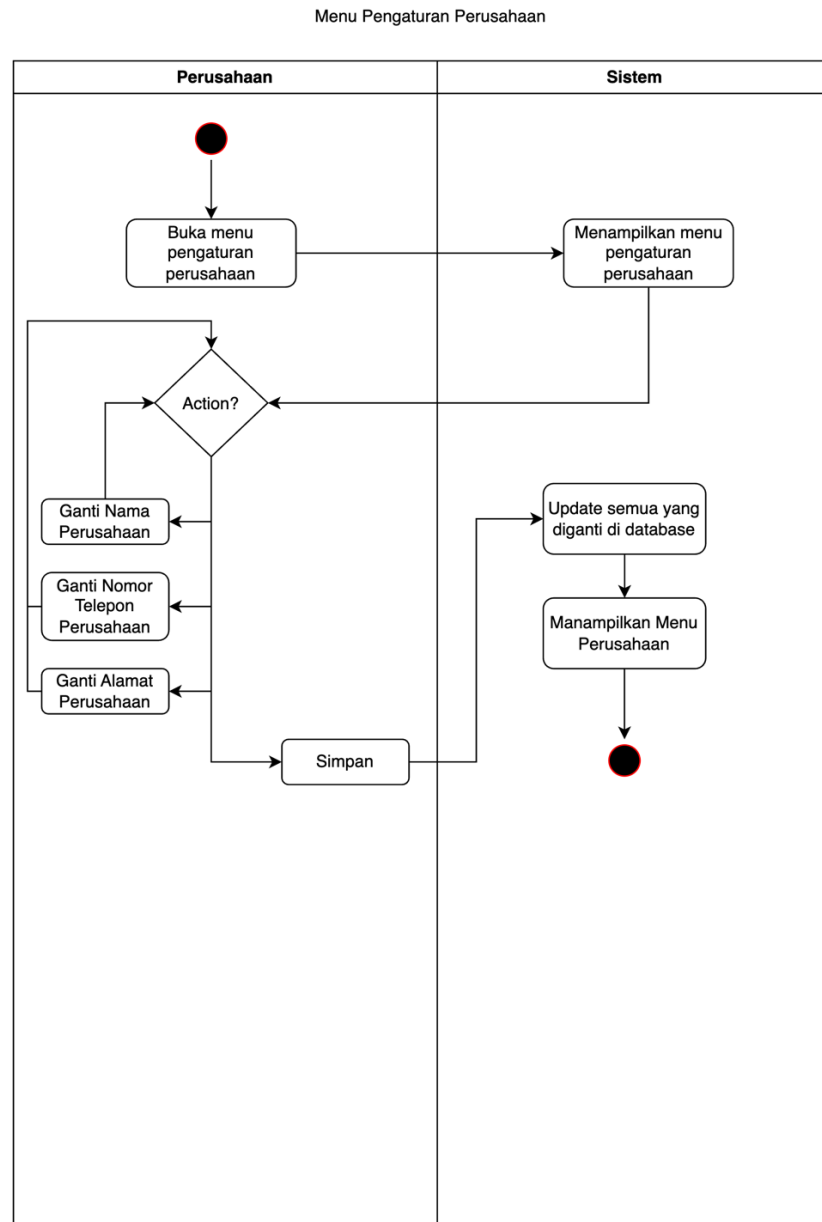
Gambar 4. 9 menunjukkan Gambaran proses Perusahaan untuk mengunduh laporan keuangan mereka melalui menu mutasi.



Gambar 4. 9 Diagram aktivitas menu mutasi

8. Diagram aktivitas menu pengaturan Perusahaan

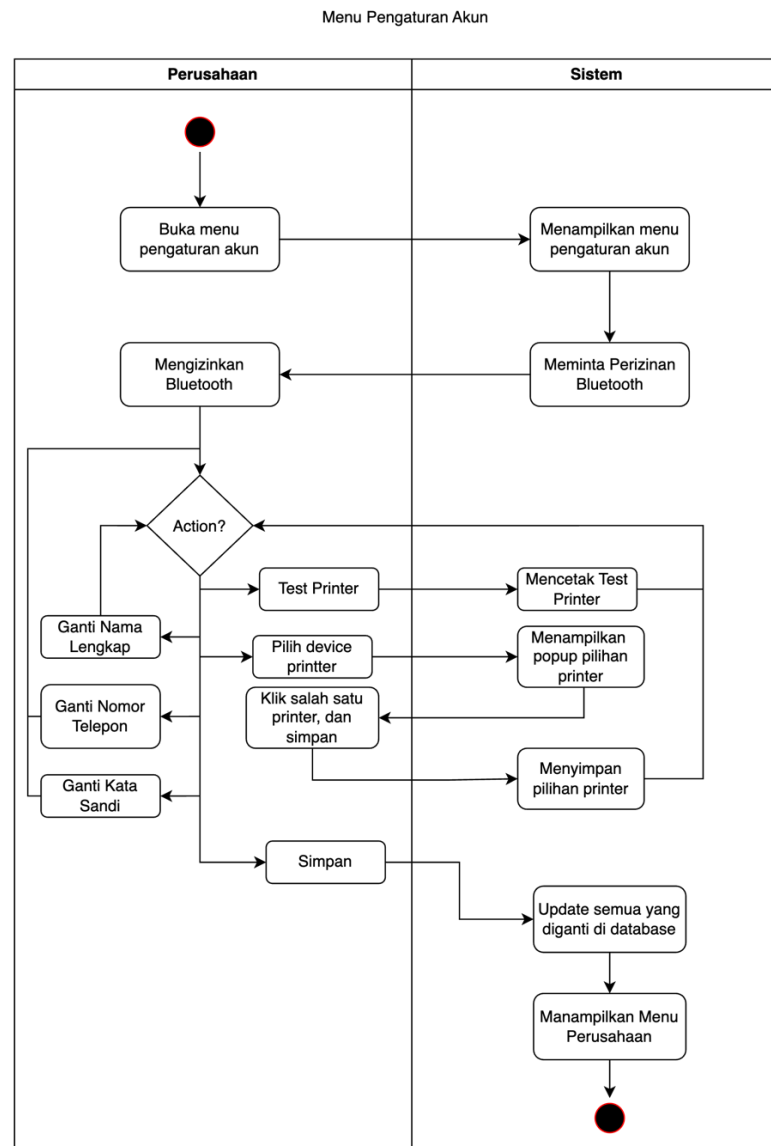
Gambar 4. 10 merupakan Gambaran proses Perusahaan untuk mengatur profil Perusahaan mereka.



Gambar 4. 10 Diagram menu pengaturan perusahaan

9. Diagram aktivitas menu pengaturan akun

Gambar 4. 11 merupakan Gambaran proses Perusahaan untuk mengatur akun Perusahaan mereka.



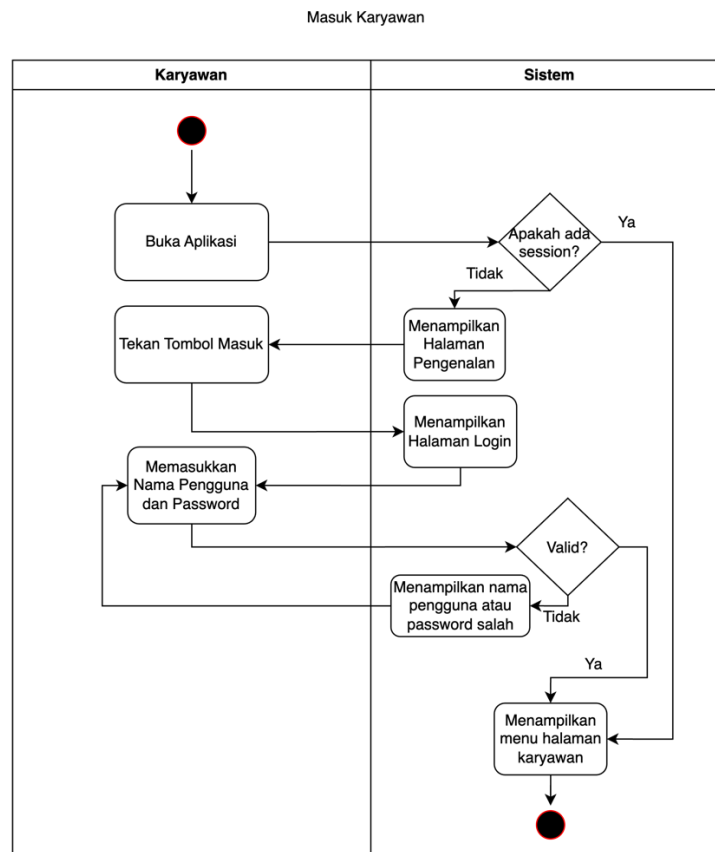
Gambar 4. 11 Diagram pengaturan akun perusahaan

Diagram aktivitas Karyawan

Bagian ini memuat rancangan diagram aktivitas yang dilakukan oleh karyawan.

1. Diagram aktivitas *login*

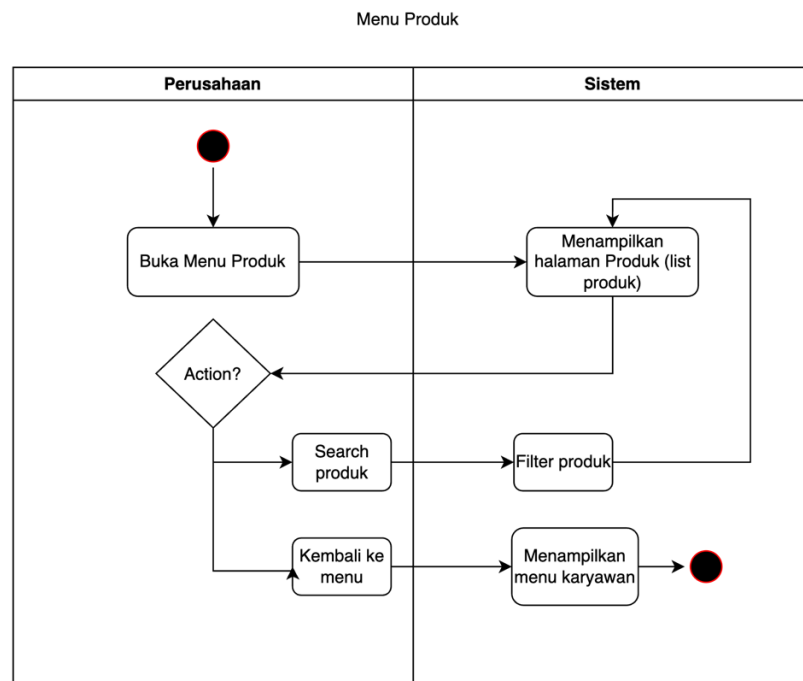
Gambar 4. 12 menunjukkan alur proses login bagi karyawan, yang memungkinkan mereka untuk masuk ke dalam sistem dan menggunakan fitur-fitur yang sesuai dengan hak akses mereka.



Gambar 4. 12 Diagram aktivitas *login* karyawan

2. Diagram aktivitas menu produk

Gambar 4. 13 menunjukkan proses karyawan mengakses menu produk untuk melihat *stock* barang yang tersedia.

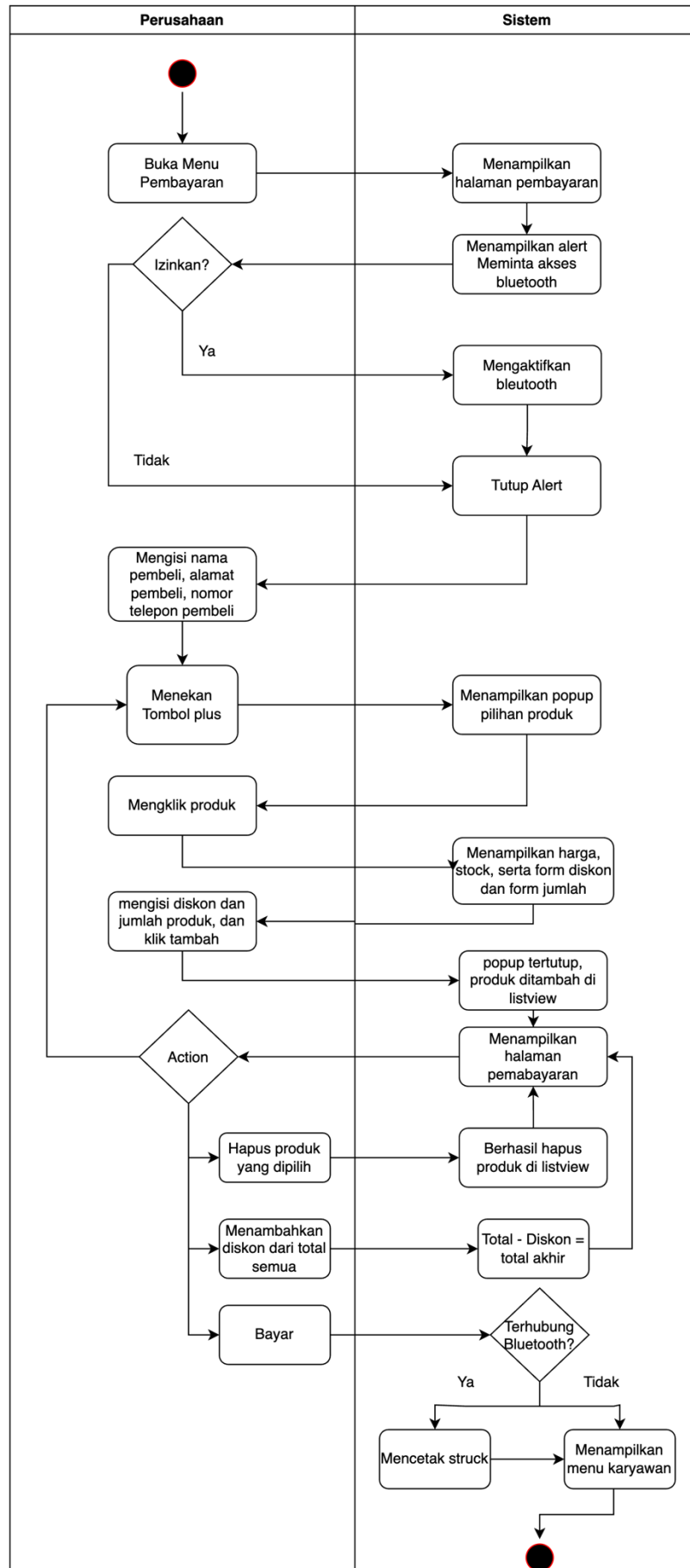


Gambar 4. 13 Diagram aktivitas menu produk karyawan

3. Diagram aktivitas menu pembayaran

Gambar 4. 14 menunjukkan alur aktivitas karyawan melakukan transaksi pada menu pembayaran.

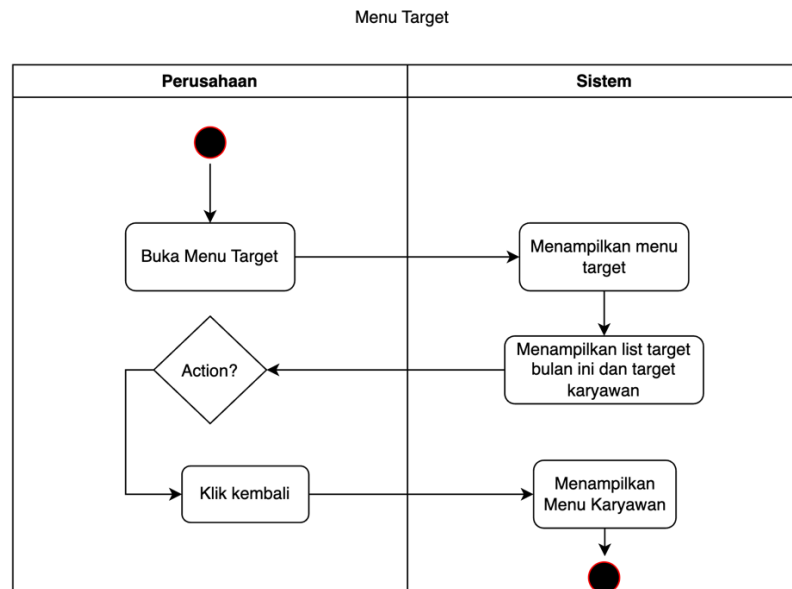
Menu Pembayaran



Gambar 4. 14 Diagram aktivitas menu pembayaran karyawan

4. Diagram aktivitas menu target

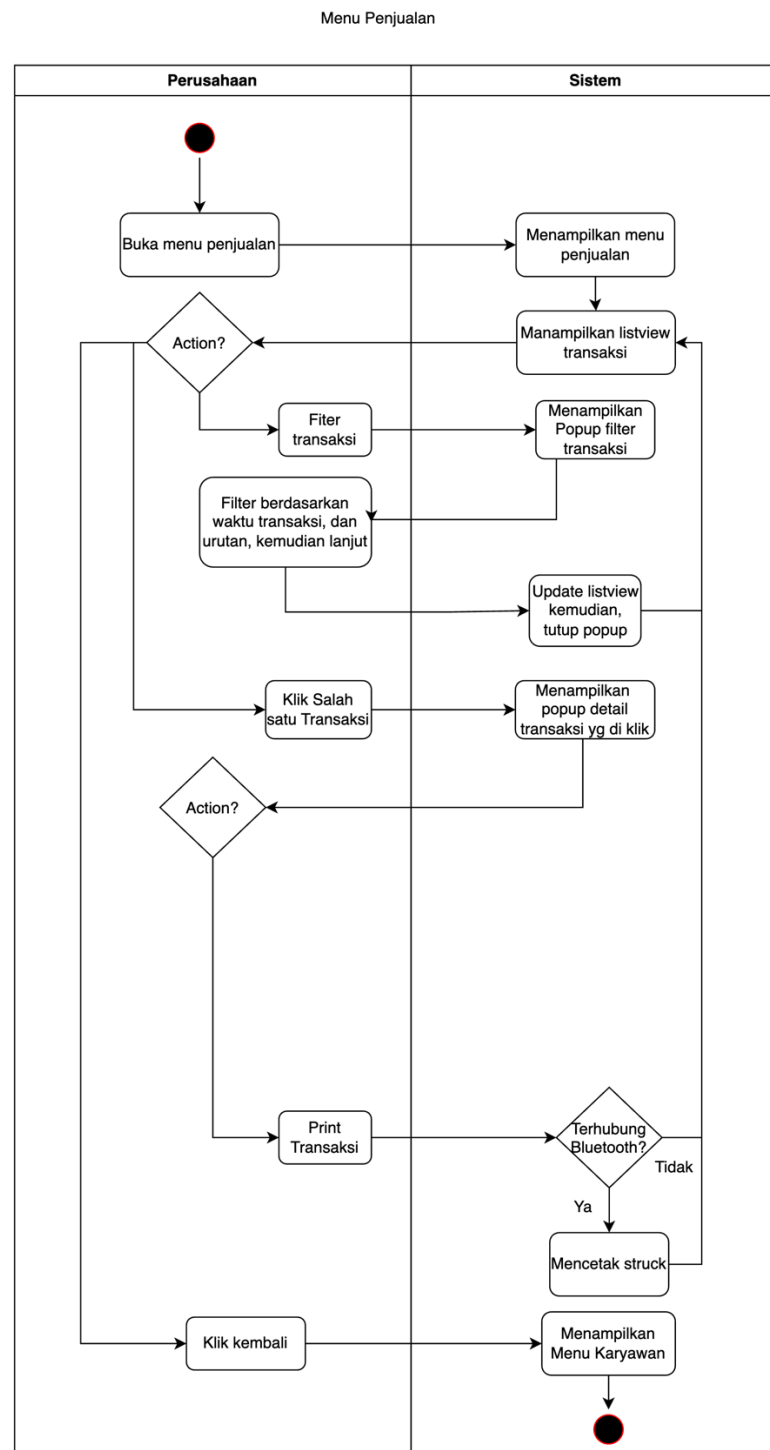
Pada Gambar 4. 15 ditampilkan aktivitas karyawan untuk melihat progress target yang ditargetkan oleh Perusahaan.



Gambar 4. 15 Diagram aktivitas menu target karyawan

5. Diagram aktivitas menu penjualan

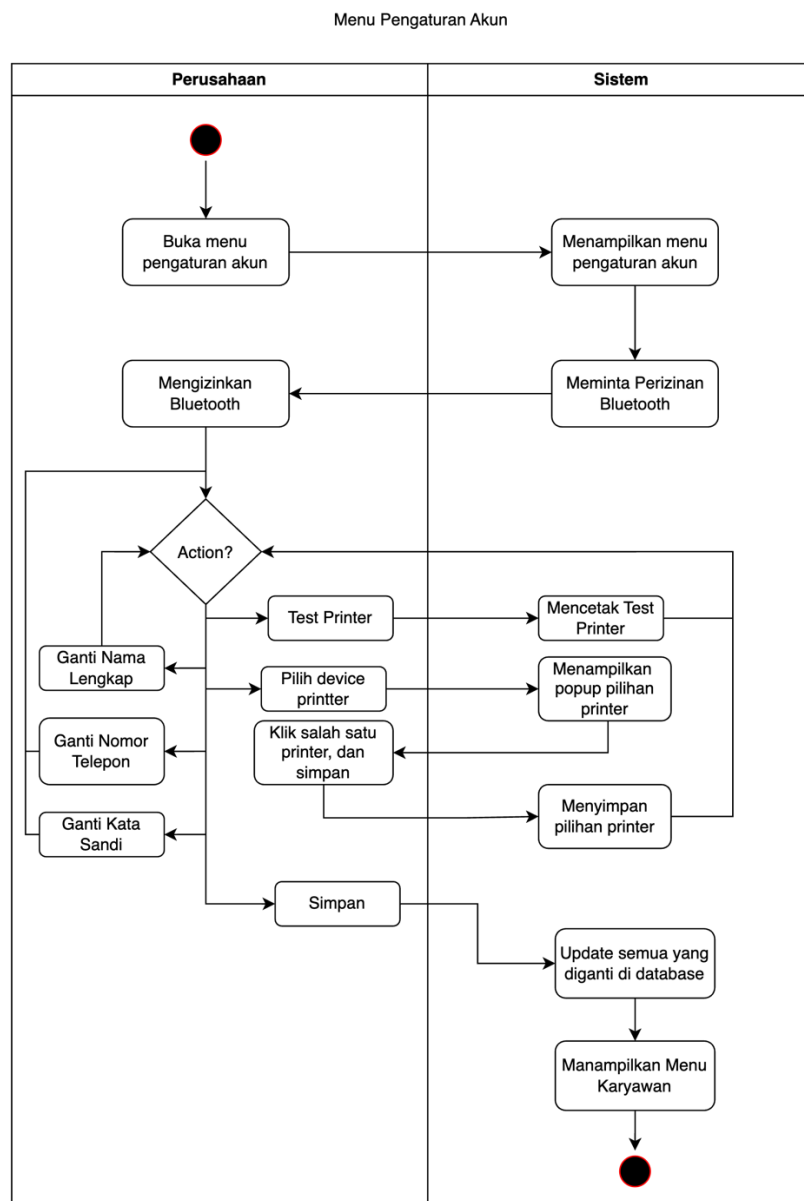
Gambar 4. 16 menunjukkan aktivitas yang bisa dilakukan oleh karyawan pada menu penjualan.



Gambar 4. 16 Diagram aktivitas menu penjualan karyawan

6. Diagram aktivitas menu pengaturan akun

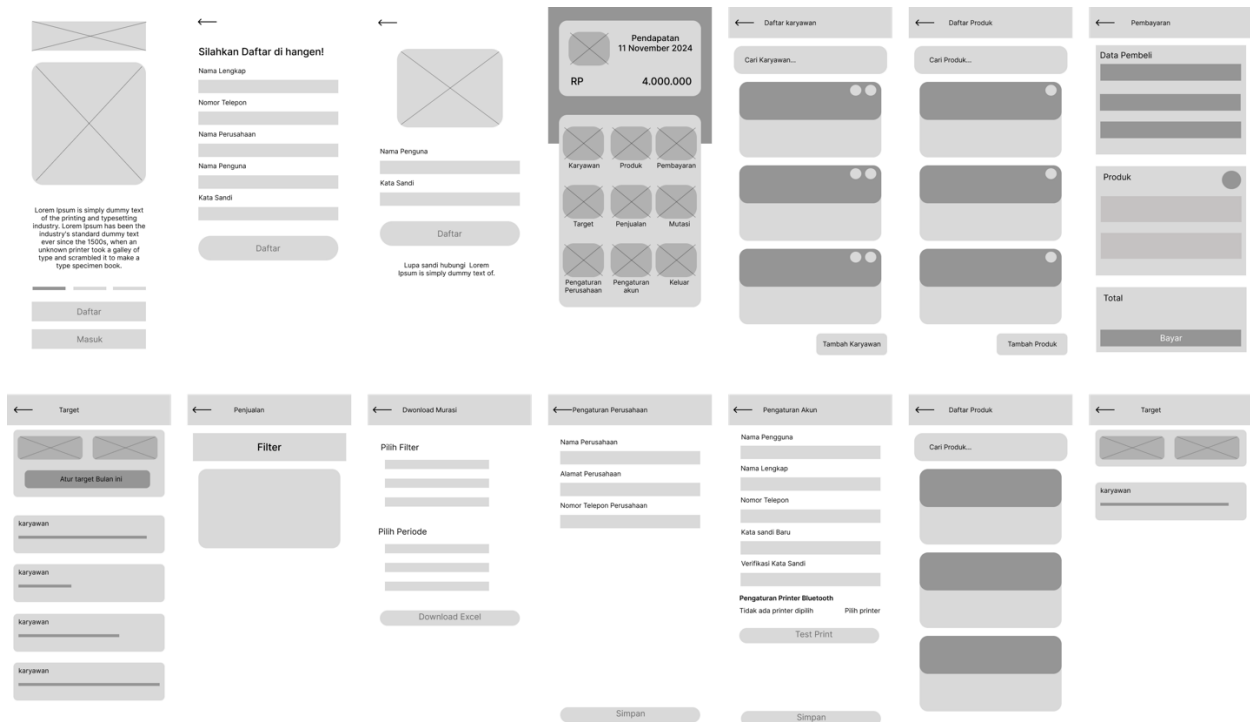
Pada Gambar 4. 17 merupakan Gambaran proses karyawan untuk mengatur akun mereka.



Gambar 4. 17 Diagram aktivitas menu Pengaturan akun karyawan

Rancangan Antarmuka

Bagian ini menyajikan rancangan antarmuka aplikasi, yang mencakup desain tata letak dan elemen visual yang dirancang untuk memberikan pengalaman pengguna yang intuitif, responsif, dan efisien. Rancangan ini disusun untuk memudahkan pengguna dalam mengoperasikan aplikasi serta mendukung aktivitas yang dilakukan oleh perusahaan maupun karyawan.



Gambar 4. 18 Wireframe low-fidelity

Pada Gambar 4. 18, ditampilkan desain *wireframe low-fidelity* yang menggambarkan struktur halaman-halaman utama dalam aplikasi. *Wireframe* ini berfungsi sebagai rancangan awal dalam proses pengembangan antarmuka pengguna, memungkinkan perancang untuk memvisualisasikan struktur dan tata letak elemen antarmuka tanpa harus terlalu terfokus pada detail grafis. Pendekatan ini memberikan ruang untuk menyusun navigasi yang jelas, memastikan pengguna dapat mengakses fitur-fitur utama dengan mudah, serta menentukan tata letak elemen antarmuka yang optimal sesuai dengan prioritas fungsi dan kenyamanan pengguna. Selain itu, rancangan ini juga mengakomodasi kebutuhan pengguna berdasarkan peran mereka dalam aplikasi, baik sebagai perusahaan maupun karyawan.

4.2 Hasil Pengembangan Aplikasi

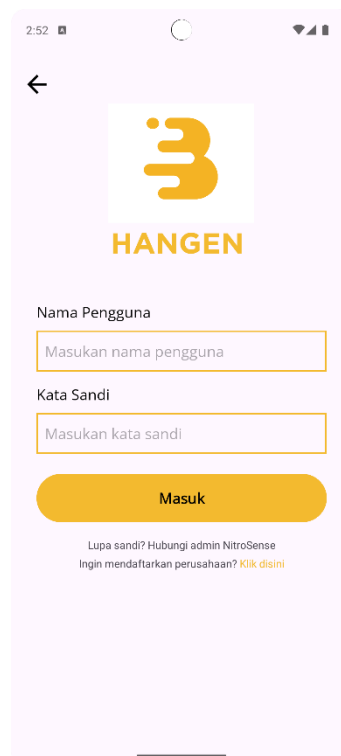
Sistem diimplementasikan berdasarkan rancangan yang telah dijabarkan pada tahap sebelumnya, meliputi perancangan *UML/Use Case Diagram*, struktur *database*, dan fitur-fitur yang dirancang untuk memenuhi kebutuhan pengguna. Implementasi sistem ini dilakukan menggunakan bahasa pemrograman *Kotlin* sebagai teknologi utama dalam pengembangan aplikasi Android, dengan *XML* digunakan untuk merancang antarmuka pengguna agar responsif dan mudah digunakan. *Firebase* dipilih sebagai solusi basis data karena

kemampuannya menyimpan data secara *real-time* dan kemudahan integrasinya dengan aplikasi *mobile*.

Proses pengembangan aplikasi ini berfokus pada fitur-fitur utama yang telah dirancang, seperti manajemen produk, penginputan transaksi penjualan, pencetakan struk, manajemen akun pengguna, hingga pemantauan target penjualan. Hasil dari proses implementasi ini adalah sebuah aplikasi kasir berbasis *mobile* yang tidak hanya mempermudah proses pencatatan transaksi dan pengelolaan data, tetapi juga mampu meningkatkan efisiensi operasional, meminimalkan kesalahan pencatatan manual, serta membantu perusahaan dalam melakukan analisis penjualan dan manajemen stok barang secara terintegrasi.

4.2.1 Halaman *Login*

Gambar 4. 19 menunjukkan tampilan halaman *login* pada aplikasi. Halaman ini digunakan untuk masuk ke dalam sistem baik oleh akun perusahaan maupun akun karyawan. Pada halaman ini terdapat form input yang terdiri dari nama pengguna dan kata sandi. Setelah pengguna mengisi kedua kolom tersebut dengan benar, mereka dapat menekan tombol "*Login*" untuk mengakses halaman utama aplikasi.



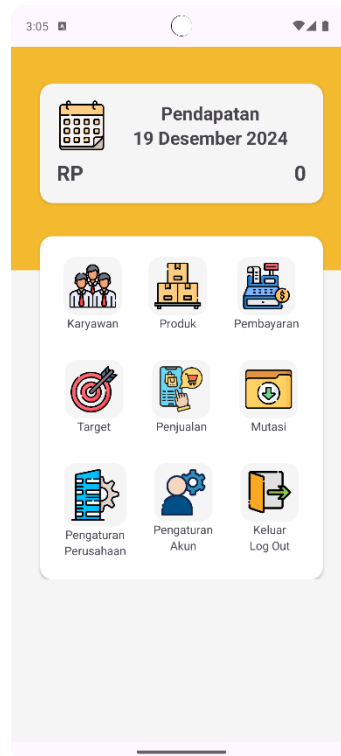
Gambar 4. 19 Halaman *login*

4.2.2 Halaman *Home* (Tampilan Perusahaan)

Gambar 4. 20 menunjukkan tampilan halaman *Home* yang dirancang khusus untuk pengelola perusahaan dalam aplikasi. Pada halaman ini, bagian header menampilkan informasi mengenai jumlah pendapatan perusahaan pada hari ini secara ringkas. Di bawah *header*, terdapat beberapa *button* fitur yang memudahkan pengelola dalam mengakses fungsi-fungsi utama, di antaranya:

- a. *Button* Karyawan: Untuk masuk ke halaman manajemen karyawan.
- b. *Button* Produk: Untuk mengelola data produk, termasuk stok dan harga.
- c. *Button* Pembayaran: Untuk mencatat dan memproses transaksi pembayaran.
- d. *Button* Target: Untuk mengatur target performa karyawan.
- e. *Button* Penjualan: Untuk melihat dan mengelola data penjualan.
- f. *Button* Mutasi: Untuk memantau pergerakan data atau transaksi.
- g. *Button* Pengaturan Perusahaan: Untuk mengatur profil dan informasi perusahaan.
- h. *Button* Pengaturan Akun: Untuk mengelola akun pengguna, seperti kata sandi dan nama pengguna.
- i. *Button* Keluar: Untuk keluar dari akun atau aplikasi.

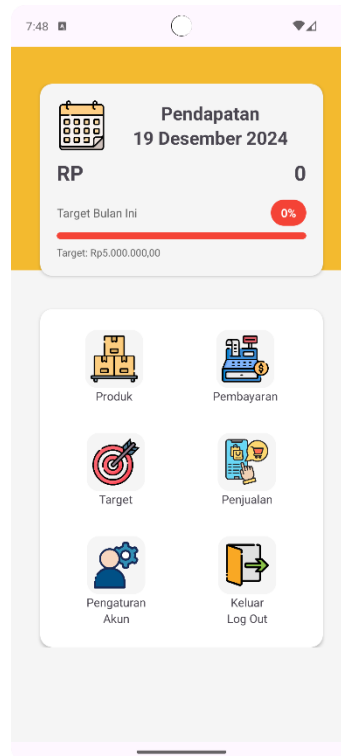
Tampilan ini dirancang agar pengelola perusahaan dapat dengan mudah mengakses berbagai fitur penting yang mendukung pengelolaan operasional perusahaan.



Gambar 4. 20 Halaman *Home* Perusahaan

4.2.3 Halaman *Home* (Tampilan Karyawan)

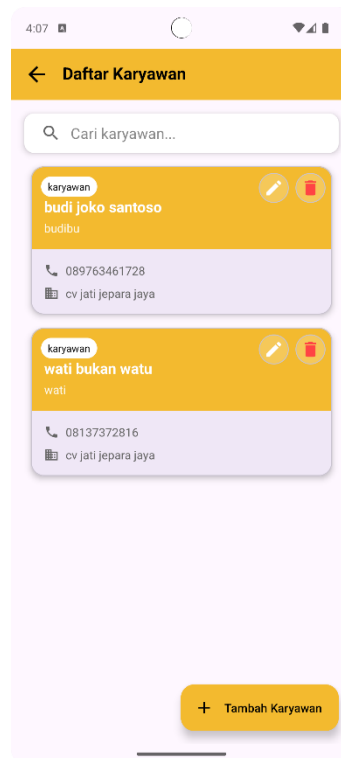
Gambar 4. 21 memperlihatkan halaman menu utama untuk karyawan, yang dirancang untuk memberikan informasi penting seperti pendapatan harian dan pencapaian target bulanan. Fitur-fitur pada menu ini lebih terbatas dibandingkan dengan menu pada akun perusahaan, mencakup akses ke menu produk, pembayaran, target, penjualan, pengaturan akun, dan tombol keluar. Pembatasan fitur ini disesuaikan dengan peran dan tanggung jawab karyawan, memastikan mereka hanya memiliki akses ke fitur-fitur yang relevan dengan tugas mereka.



Gambar 4. 21 Halaman *home* karyawan

4.2.4 Halaman menu karyawan

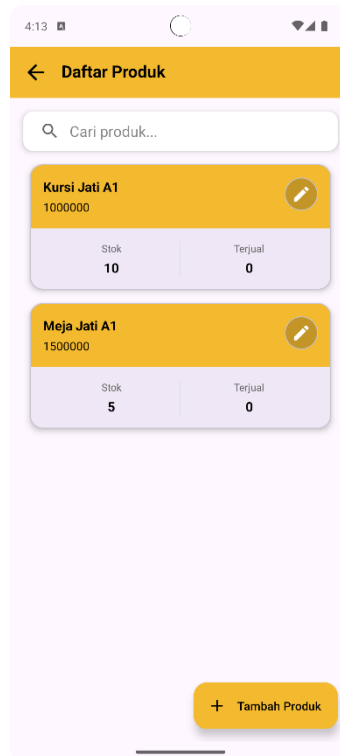
Gambar 4. 22 menampilkan halaman menu karyawan yang berisi informasi lengkap mengenai karyawan yang terdaftar. Pada halaman ini, perusahaan dapat melakukan berbagai tindakan, seperti mengedit data karyawan, mencari karyawan berdasarkan nama atau informasi tertentu, serta menambahkan karyawan baru ke dalam sistem.



Gambar 4. 22 Halaman menu karyawan

4.2.5 Halaman Menu Produk (Tampilan Perusahaan)

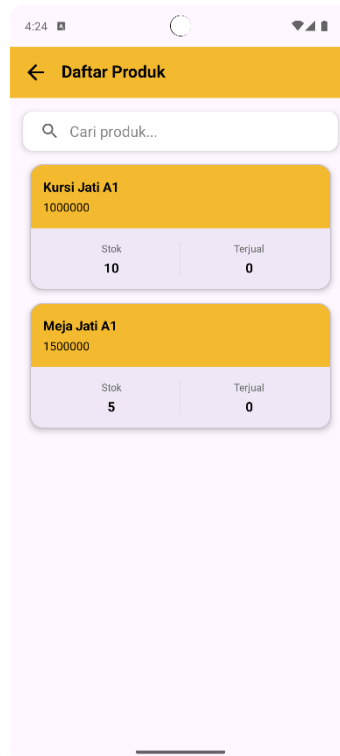
Gambar 4. 23 menampilkan halaman menu produk yang berisi daftar produk milik perusahaan. Pada halaman ini, pengguna dapat melakukan pencarian produk, mengedit informasi produk yang sudah ada, serta menambahkan produk baru ke dalam sistem.



Gambar 4. 23 Halaman menu produk perusahaan

4.2.6 Halaman Menu Produk (Tampilan Karyawan)

Gambar 4. 24 menampilkan halaman menu produk dari perspektif karyawan. Pada halaman ini, karyawan hanya dapat melihat informasi mengenai stok produk yang tersedia di perusahaan, termasuk jumlah produk yang telah terjual. Fitur pencarian produk juga disediakan untuk mempermudah karyawan dalam menemukan produk tertentu, sehingga mereka dapat dengan cepat mengecek ketersediaan stok sesuai kebutuhan. Halaman ini dirancang untuk memberikan akses terbatas yang relevan dengan tugas karyawan tanpa mengizinkan pengeditan data produk.



Gambar 4. 24 Halaman menu produk karyawan

4.2.7 Halaman menu Pembayaran

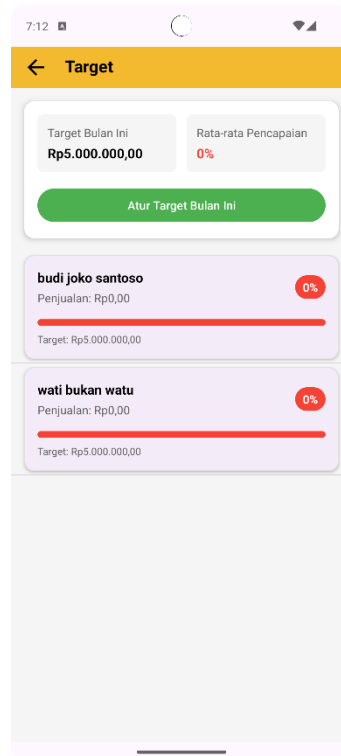
Pada Gambar 4. 25 menunjukkan tampilan halaman menu pembayaran yang digunakan untuk melakukan transaksi. Pada halaman ini, terdapat form untuk mengisi data pembeli, seperti nama, alamat, dan nomor telepon. Selain itu, ditampilkan daftar produk yang dibeli, lengkap dengan informasi harga satuan, jumlah yang dibeli, dan total harga per produk. Pengguna dapat menambahkan produk ke daftar dengan menekan tombol "+" atau menghapus produk dari daftar menggunakan tombol hapus. Di bagian bawah halaman, tersedia rincian total pembayaran yang mencakup total harga, diskon yang diterapkan, dan total akhir pembayaran. Setelah tombol "Bayar" ditekan, sistem akan secara otomatis mencetak struk transaksi ke perangkat printer yang telah tersambung.

The screenshot shows a mobile application interface for a payment screen. At the top, there is a yellow header bar with a back arrow and the title 'Pembayaran'. Below this, the screen is divided into three main sections. The first section, 'Data Pembeli', contains three input fields: 'Nama Pembeli' with the value 'Dodi Satria', 'Alamat Pembeli' with the value 'Jl Merdeka 21', and 'Nomor Telepon Pembeli' with the value '08135261527'. The second section, 'Produk', has a yellow '+' button at the top right. It lists two items: 'Kursi Jati A1' with a 'HAPUS' button, showing a discount of 'Rp. 10000' and a calculation 'Rp. 990000 x 2 =' resulting in '1980000'; and 'Meja Jati A1' with a 'HAPUS' button. The third section, 'TOTAL', shows a calculation: '3480000 - Diskon: 0 = 3480000'. At the bottom of this section is a large yellow button labeled 'BAYAR'.

Gambar 4. 25 Halaman menu pembayaran

4.2.8 Halaman menu Target (Tampilan Perusahaan)

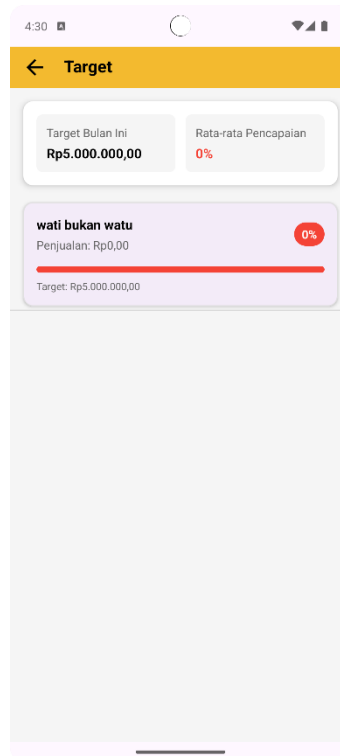
Gambar 4. 26 menunjukkan tampilan halaman menu target, di mana Perusahaan dapat mengatur target untuk setiap karyawan sesuai dengan kebutuhan operasional. Selain itu, halaman ini juga memungkinkan Perusahaan untuk memantau progres pencapaian target oleh masing-masing karyawan secara real-time. Halaman ini dirancang untuk mendukung pengelolaan performa karyawan dengan lebih efektif dan terorganisir.



Gambar 4. 26 Halaman menu target perusahaan

4.2.9 Halaman Menu Target (Tampilan Karyawan)

Gambar 4. 27 menampilkan halaman menu target dari sudut pandang karyawan. Halaman ini dirancang untuk memberikan informasi mengenai target individu yang harus dicapai oleh karyawan dalam periode tertentu. Pada halaman ini, karyawan dapat melihat detail target bulanan mereka, termasuk rata-rata pencapaian hingga saat ini dan persentase target yang telah terpenuhi. Tampilan ini membantu karyawan untuk memantau progres mereka secara mandiri dan tetap fokus pada pencapaian target yang telah ditetapkan oleh perusahaan.



Gambar 4. 27 Halaman target karyawan

4.2.10 Halaman menu Penjualan

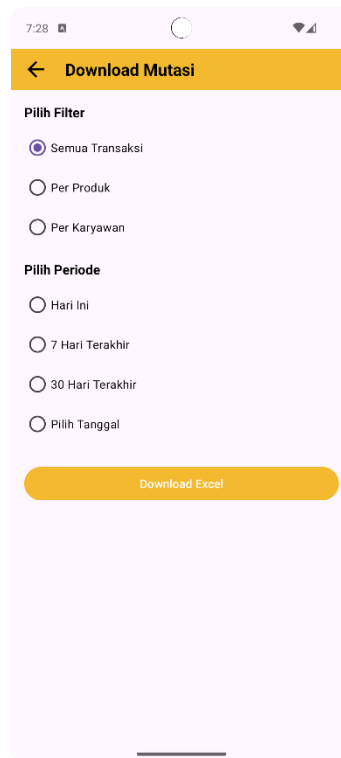
Gambar 4. 28 memperlihatkan tampilan halaman riwayat transaksi yang mencatat seluruh penjualan yang telah dilakukan. Setiap *entri* transaksi dilengkapi dengan informasi seperti waktu transaksi, nama kasir yang bertugas, alamat pembeli, nomor telepon, serta total pembayaran. Pada bagian atas halaman terdapat tombol *filter* yang memungkinkan pengguna untuk mencari transaksi berdasarkan rentang tanggal tertentu. Selain itu, di bagian bawah halaman ditampilkan detail perangkat *printer* yang terhubung untuk memudahkan pencetakan ulang struk jika diperlukan.



Gambar 4. 28 Halaman menu penjualan

4.2.11 Halaman menu Mutasi

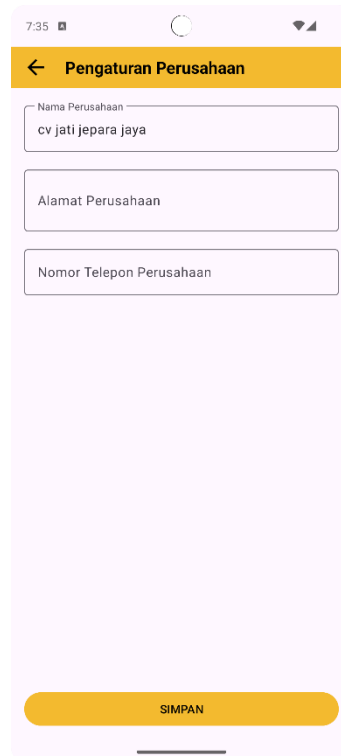
Gambar 4. 29 menampilkan halaman yang dirancang untuk mencetak laporan keuangan perusahaan dalam format CSV. Halaman ini dilengkapi dengan fitur *filter* yang memungkinkan pengguna memilih laporan berdasarkan berbagai kategori, seperti semua transaksi, transaksi per produk, atau transaksi per karyawan. Selain itu, pengguna juga dapat menyaring laporan berdasarkan periode waktu, seperti hari ini, 7 hari terakhir, 30 hari terakhir, atau memilih tanggal tertentu sesuai kebutuhan.



Gambar 4. 29 Halaman menu mutasi

4.2.12 Halaman menu pengaturan perusahaan

Gambar 4. 30 menampilkan halaman pengaturan yang memungkinkan pengguna untuk mengubah detail informasi perusahaan. Pada halaman ini, pengguna dapat mengedit data seperti nama perusahaan, alamat perusahaan, dan nomor telepon perusahaan sesuai kebutuhan.



7:35

← Pengaturan Perusahaan

Nama Perusahaan
cv jati jepara jaya

Alamat Perusahaan

Nomor Telepon Perusahaan

SIMPAN

Gambar 4. 30 Halaman menu pengaturan perusahaan

4.2.13 Halaman pengaturan akun

Pada Gambar 4. 31 menampilkan halaman pengaturan akun, yang dirancang untuk memudahkan pengguna dalam mengelola informasi akun mereka. Halaman ini memuat data seperti nama pengguna, nama lengkap, dan nomor telepon yang dapat diperbarui sesuai kebutuhan. Selain itu, halaman ini menyediakan fitur untuk mengganti kata sandi guna menjaga keamanan akun. Tidak hanya itu, pengguna juga dapat mengatur perangkat printer yang terhubung melalui opsi pengaturan printer, memastikan kompatibilitas perangkat dengan aplikasi. Halaman ini bertujuan untuk memberikan fleksibilitas dan kenyamanan dalam pengelolaan data pengguna sekaligus menjaga keamanan dan efisiensi operasional.

Gambar 4. 31 Halaman pengaturan akun

4.3 Testing

4.3.1 Black box testing

Pengujian fungsional pada aplikasi dilakukan menggunakan metode *black box testing*. Metode ini menitikberatkan pada pengamatan keluaran aplikasi berdasarkan skenario uji yang telah dirancang, tanpa melihat kode atau struktur internal aplikasi. Pengujian bertujuan memastikan bahwa semua fitur bekerja sesuai dengan fungsinya dan memberikan respons yang sesuai. Seluruh proses pengujian dilakukan langsung oleh peneliti untuk memverifikasi bahwa setiap fungsi memenuhi kebutuhan yang telah ditentukan. Pengujian ini juga dilakukan oleh owner CV Arjuna Jati secara langsung dengan dipandu untuk menerapkan skenario yang sudah dibuat.

Hasil Black Box Testing akun Perusahaan

Hasil black box testing untuk aplikasi dengan peran perusahaan menunjukkan bahwa seluruh fitur telah berfungsi sesuai dengan skenario yang dirancang. Detail hasil pengujian ini dapat dilihat pada Tabel 4. 1.

Tabel 4. 1 Tabel pengujian akun perusahaan

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Login	Memasukkan nama pengguna () dan kata sandi pengguna yang valid	Sistem menampilkan halaman utama sesuai dengan jenis akun (perusahaan)	Sesuai
2	Login	Memasukkan nama pengguna dan kata sandi pengguna yang salah	Sistem menampilkan pesan kesalahan "Nama pengguna belum terdaftar"	Sesuai
3	Login	Mengosongkan <i>form</i> nama pengguna dan kata sandi atau mengisi salah satu	Sistem menampilkan pesan kesalahan "Nama pengguna tidak boleh kosong" "Kata sandi tidak boleh kosong"	Sesuai
4	Halaman Home	Masuk pada halaman <i>home</i> yang menampilkan pendapatan hari ini dan beberapa menu yang sesuai dengan jenis akun perusahaan	Sistem menampilkan pendapatan hari ini dan menampilkan menu yang sesuai dengan jenis akun	Sesuai
5	Karyawan (Tambah karyawan)	Mengisi semua <i>form</i> pada tambah karyawan dengan data yang valid dan menekan tombol "Daftar"	Sistem menyimpan data karyawan baru dan memperbarui daftar karyawan di halaman terkait.	Sesuai
6	Karyawan (Tambah karyawan)	Mengosongkan salah satu <i>form</i> dan menekan tombol "Daftar"	Sistem menampilkan pesan kesalahan "Harap lengkapi <i>form</i> terlebih dahulu."	Sesuai
7	Karyawan (Edit karyawan)	Mengedit data karyawan dan menekan tombol "simpan"	Sistem menyimpan data karyawan yang sudah diperbaharui.	Sesuai
8	Karyawan (Hapus karyawan)	Menghapus Karyawan dengan menekan tombol hapus.	Sistem menampilkan peringatan kepada pengguna apakah yakin untuk menghapus data karyawan jika yakin maka pengguna menekan tombol hapus dan data karyawan terhapus.	Sesuai

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
9	Produk (Tambah Produk)	Menambahkan produk dengan mengisi <i>form</i> pada halaman tambah produk dan menekan tombol “Tambah Produk”	Sistem menyimpan data produk yang baru ditambahkan dan menampilkannya di halaman produk.	Sesuai
10	Produk (Edit Produk)	Mengedit data produk dengan mengisi atau mengganti data produk pada halaman edit produk dengan update terbaru dan menekan tombol “Edit Produk”	Sistem menyimpan <i>update</i> data produk terbaru dan menampilkannya pada halaman produk	Sesuai
11	Pembayaran	Memasukkan data pembeli dan produk yang dibeli, lalu menekan tombol “Bayar.”	Sistem menyimpan transaksi, memperbarui stok produk, dan mencetak struk melalui printer yang terhubung.	Sesuai
12	Target	Menambahkan target penjualan bulanan dan menekan tombol “Simpan.”	Sistem menyimpan target penjualan dan menampilkan progres target tiap karyawan.	Sesuai
13	Penjualan	Menampilkan daftar penjualan, mengedit transaksi, menghapus transaksi, dan mencetak struk.	Sistem mampu menampilkan data transaksi, mengedit atau menghapus transaksi, serta mencetak ulang struk.	Sesuai
14	Mutasi	Memilih filter tertentu untuk melihat laporan penjualan, lalu menekan tombol "Unduh".	Sistem menghasilkan file CSV sesuai dengan filter yang dipilih dan mengunduhnya ke perangkat pengguna.	Sesuai
15	Pengaturan Perusahaan	Mengedit data perusahaan, seperti nama atau alamat, lalu menekan tombol "Simpan".	Sistem memperbarui data perusahaan yang ditampilkan di halaman terkait.	Sesuai
16	Pengaturan Akun	Mengubah kata sandi atau mengatur printer Bluetooth, lalu menekan tombol "Simpan."	Sistem memperbarui kata sandi dan pengaturan printer, serta menampilkan pesan "Perubahan berhasil disimpan."	Sesuai

Hasil *Black Box Testing* akun Karyawan

Hasil *black box testing* pada aplikasi untuk peran karyawan menunjukkan bahwa semua fitur berjalan sesuai dengan skenario yang telah dirancang. Pengujian dilakukan untuk memastikan setiap fungsi dapat diakses dan digunakan dengan baik oleh pengguna dengan peran karyawan. Rincian hasil pengujian dapat dilihat pada Tabel 4. 2.

Tabel 4. 2 Tabel pengujian akun karyawan

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1	Login	Memasukkan nama pengguna dan kata sandi yang <i>valid</i>	Sistem menampilkan halaman utama untuk akun karyawan sesuai dengan hak aksesnya	Sesuai
2	Login	Memasukkan nama pengguna atau kata sandi yang salah	Sistem menampilkan pesan kesalahan "Nama pengguna atau kata sandi salah"	Sesuai
3	Login	Mengosongkan <i>form</i> nama pengguna atau kata sandi	Sistem menampilkan pesan "Nama pengguna/kata sandi tidak boleh kosong"	Sesuai
4	Halaman <i>Home</i>	Masuk ke halaman utama yang menampilkan informasi pendapatan harian dan progres target bulanan	Sistem menampilkan pendapatan harian dan progres target sesuai data karyawan	Sesuai
5	Produk (<i>View</i>)	Mengakses menu produk untuk melihat stok produk yang tersedia	Sistem menampilkan daftar produk dengan informasi stok terkini	Sesuai
6	Produk (<i>Search</i>)	Menggunakan fitur pencarian untuk mencari produk tertentu	Sistem menampilkan produk yang sesuai dengan kata kunci pencarian	Sesuai
7	Target (<i>View</i>)	Mengakses menu target untuk melihat target bulanan karyawan	Sistem menampilkan target bulanan karyawan,	Sesuai

No.	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
			rata-rata pencapaian, dan persentase	
8	Pembayaran	Memasukkan data pembeli dan produk yang dibeli, lalu menekan tombol "Bayar"	Sistem menyimpan transaksi dan mencetak struk pembayaran	Sesuai
9	Penjualan (<i>View</i>)	Melihat riwayat transaksi yang telah dilakukan	Sistem menampilkan daftar transaksi dengan detail seperti waktu, total, dan kasir	Sesuai
10	Penjualan (<i>Print</i>)	Menekan tombol <i>print</i> untuk mencetak ulang struk pembayaran dari transaksi tertentu	Sistem mencetak ulang struk transaksi menggunakan printer yang terhubung	Sesuai
11	Pengaturan Akun	Mengubah kata sandi atau pengaturan printer lalu menekan tombol "Simpan"	Sistem memperbarui pengaturan akun dan menampilkan pesan "Perubahan berhasil"	Sesuai

4.3.2 Pengujian Usabilitas

Pengujian usabilitas dilakukan untuk mengevaluasi sejauh mana aplikasi memenuhi kebutuhan pengguna dalam hal kemudahan penggunaan dan efisiensi. Metode yang digunakan adalah *System Usability Scale (SUS)*, yang memberikan skor berdasarkan penilaian pengguna terhadap pengalaman mereka dalam menggunakan aplikasi. Pernyataan pengujian di sajikan pada tabel Tabel 4. 3.

Tabel 4. 3 Tabel pernyataan pengujian *sus*

No.	Kode	Pernyataan	Pengelompokan
1	Pt1	Saya merasa aplikasi kasir ini akan sangat membantu pekerjaan sehari-hari	<i>Satisfaction</i>
2	Pt2	Saya merasa tampilan menu dalam aplikasi ini membingungkan	<i>Efficiency</i>
3	Pt3	Saya dapat menggunakan fitur-fitur aplikasi ini tanpa kesulitan	<i>Learnability</i>
4	Pt4	Saya kesulitan memahami cara mengoperasikan aplikasi ini sendirian	<i>Learnability</i>
5	Pt5	Saya merasa proses transaksi dalam aplikasi ini berjalan dengan lancar	<i>Efficiency</i>
6	Pt6	Saya sering mengalami error saat menggunakan aplikasi ini	<i>Learnability</i>
7	Pt7	Saya yakin rekan kerja saya dapat menggunakan aplikasi ini dengan mudah	<i>Learnability</i>
8	Pt8	Saya merasa prosedur dalam aplikasi ini terlalu berbelit-belit	<i>Efficiency</i>
9	Pt9	Saya merasa nyaman menggunakan semua fitur dalam aplikasi ini	<i>Satisfaction</i>
10	Pt10	Saya membutuhkan waktu lama untuk mahir menggunakan aplikasi ini	<i>Learnability</i>

Pengujian ini melibatkan 1 pemilik CV Arjuna Jati Distributor dan 10 karyawan CV Arjuna Jati Distributor sebagai responden. Partisipan diminta untuk menjawab pernyataan yang mencerminkan pengalaman mereka dalam menggunakan fitur-fitur aplikasi. Hasil pengujian SUS ditampilkan pada Tabel 4. 4 Penilaian *SUS*.

Tabel 4. 4 Tabel penilaian pengujian *sus*

Responden	Jabatan	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
R1	Pemilik	5	1	4	1	4	1	5	1	5	2
R2	Karyawan	5	2	3	2	4	2	4	2	5	3
R3	Karyawan	5	1	4	2	3	1	5	1	5	1
R4	Karyawan	5	2	4	2	4	1	4	2	5	2
R5	Karyawan	4	1	3	1	5	1	4	1	4	1
R6	Karyawan	5	1	4	2	4	2	4	2	5	1
R7	Karyawan	5	2	3	2	4	1	4	1	5	1
R8	Karyawan	4	1	4	2	4	2	3	2	4	2
R9	Karyawan	5	1	3	1	3	1	4	1	5	1
R10	Karyawan	5	2	4	2	4	2	4	3	4	2
R11	Karyawan	5	1	4	2	5	1	3	2	4	2

Berdasarkan hasil penilaian tersebut, skor *SUS* untuk masing-masing *responden* dihitung dan disajikan pada Tabel 4. 5 Hasil Pengujian *SUS*.

Tabel 4. 5 Tabel hasil pengujian

Responden	Skor Konversi Pernyataan	Total Skor	SUS Score
R1	4+4+3+4+3+4+4+4+3	37	92,5
R2	4+3+2+3+3+3+3+4+2	30	75
R3	4+4+3+3+2+4+4+4+4	36	90
R4	4+3+3+3+3+4+3+3+4+3	33	82,5
R5	3+4+2+4+4+4+3+4+3+4	35	87,5
R6	4+4+3+3+3+3+3+3+4+4	34	85
R7	4+3+2+3+3+4+3+4+4+4	34	85
R8	3+4+3+3+3+3+2+3+3+3	30	75
R9	4+4+2+4+2+4+3+4+4+4	35	87,5
R10	4+3+3+3+3+3+3+2+3+3	30	75
R11	4+4+3+3+4+4+2+3+3+3	33	82,5

Dari hasil pengujian, skor *SUS* rata-rata yang diperoleh adalah 83,4. Hal ini menunjukkan tingkat kepuasan pengguna terhadap aplikasi sangat tinggi. Berdasarkan skala *SUS*, aplikasi kasir ini mendapatkan kategori A. Pengguna menganggap aplikasi ini mudah digunakan, mendukung kelancaran proses kerja, dan memiliki fitur yang relevan untuk kebutuhan mereka.

Untuk memberikan analisis lebih mendalam, skor *SUS* dianalisis berdasarkan pengelompokan pernyataan pada Tabel 4. 6 berikut:

Tabel 4. 6 Tabel analisis berdasarkan pengelompokan Pernyataan

Pengelompokan	Pernyataan	Skor Rata-rata
<i>Learnability</i>	Pt3, Pt4, Pt6, Pt7, Pt10	79,54545455
<i>Efficiency</i>	Pt2, Pt5, Pt8	83,33333333
<i>Satisfaction</i>	Pt1, Pt9	93,18181818

Skor rata-rata untuk *learnability* adalah 79,54. Hal ini menunjukkan bahwa pengguna umumnya merasa aplikasi mudah dipahami, meskipun terdapat beberapa responden yang merasa kesulitan di awal. Skor rata-rata untuk *efficiency* adalah 83,33. Proses transaksi dan prosedur dalam aplikasi dinilai cukup efisien, tetapi ada ruang untuk menyederhanakan prosedur tertentu. Skor rata-rata untuk *satisfaction* adalah 93,18. Pengguna merasa sangat puas dengan pengalaman menggunakan aplikasi, terutama dalam kenyamanan menggunakan fitur-fiturnya.

Hasil pengujian menunjukkan bahwa pengelompokan dengan skor rata-rata terendah adalah *Learnability* (79,54). Hal ini mengindikasikan adanya potensi perbaikan dalam aspek pembelajaran aplikasi, meskipun skor ini masih menunjukkan kategori "baik." Beberapa responden menyebutkan bahwa tahap awal pembelajaran aplikasi dapat lebih dipermudah untuk meningkatkan pengalaman pengguna.

Berdasarkan hasil pengujian dan analisis, dapat disimpulkan bahwa rumusan masalah dan tujuan penelitian telah tercapai. Aplikasi kasir berbasis *mobile* yang dikembangkan berhasil memenuhi rumusan masalah pertama, yaitu melakukan pencatatan transaksi secara otomatis serta pelacakan performa penjualan dan stok barang di CV Arjuna Jati Jepara. Tingkat kebergunaan (*usability*) aplikasi, yang diukur melalui metode *SUS*, menunjukkan skor rata-rata 83,4, yang termasuk dalam kategori "sangat baik."

Adapun tujuan penelitian, yaitu menghasilkan aplikasi kasir agen *portable* berbasis *mobile* yang dapat mengoptimalkan efisiensi proses transaksi, juga telah tercapai. Analisis pengaruh kebergunaan aplikasi menunjukkan tingkat kepuasan pengguna sangat tinggi (kategori A). Meskipun demikian, hasil pengujian juga menunjukkan adanya ruang untuk penyempurnaan pada aspek *learnability* aplikasi.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut:

- a. Pengembangan aplikasi kasir berbasis *mobile* untuk CV Arjuna Jati Jepara telah berhasil dilakukan menggunakan bahasa pemrograman *Kotlin* untuk pengembangan *Android*, *Firebase Realtime Database* untuk manajemen data, dan *XML* untuk perancangan antarmuka. Aplikasi berhasil mengimplementasikan sistem pencatatan transaksi otomatis yang dilengkapi dengan fitur cetak struk melalui printer *Bluetooth*. Sistem manajemen stok *real-time* memungkinkan pelacakan persediaan barang secara akurat, sementara sistem target penjualan memfasilitasi pemantauan performa karyawan secara efektif. Fitur pembedaan hak akses antara pemilik perusahaan dan karyawan meningkatkan keamanan data, didukung dengan sistem pelaporan transaksi terintegrasi yang memudahkan analisis bisnis. Keberhasilan implementasi ini dibuktikan melalui pengujian *black-box* yang menunjukkan semua fitur berfungsi sesuai kebutuhan.
- b. Evaluasi tingkat kebergunaan (*usability*) aplikasi menunjukkan hasil sangat memuaskan dengan perolehan skor rata-rata *System Usability Scale (SUS)* mencapai 83,4 dari skala 100. Hasil ini mencerminkan tingkat penerimaan yang sangat positif dari pengguna, dengan rincian aspek *learnability* mencapai skor 79,54, aspek *efficiency* memperoleh skor 83,33, dan aspek *satisfaction* mendapatkan skor tertinggi 93,18. Pengujian yang dilakukan terhadap 11 responden, terdiri dari 1 pemilik perusahaan dan 10 karyawan, menghasilkan skor individual yang konsisten berkisar antara 75,0 hingga 92,5, membuktikan bahwa aplikasi ini berhasil memenuhi kebutuhan pengguna dari berbagai level. Pencapaian ini menunjukkan bahwa aplikasi memiliki tingkat *usability* yang sangat baik dan sangat diterima oleh pengguna. Dengan hasil tersebut, dapat disimpulkan bahwa penelitian ini telah berhasil menjawab rumusan masalah dan mencapai tujuan penelitian dalam mengembangkan aplikasi kasir berbasis *mobile* yang efisien dan memiliki tingkat kebergunaan yang tinggi untuk CV Arjuna Jati Jepara.

5.2 Saran

Berdasarkan hasil penelitian, saran yang dapat diberikan untuk pengembangan lebih lanjut adalah sebagai berikut:

- c. Pelatihan Penggunaan Aplikasi

Memberikan pelatihan bagi karyawan agar mereka dapat memanfaatkan fitur aplikasi secara optimal, sehingga semua fungsi aplikasi dapat digunakan dengan efektif.

- d. Pengembangan Fitur Tambahan

Disarankan untuk menambahkan fitur seperti laporan penjualan berbasis grafik, integrasi pembayaran digital, dan pengelolaan inventaris untuk meningkatkan fungsionalitas aplikasi.

e. Peningkatan Keamanan Data

Mengimplementasikan teknologi enkripsi data serta penyimpanan berbasis cloud untuk menjamin keamanan dan aksesibilitas data transaksi.

f. Evaluasi dan Pembaharuan Berkala

Melakukan evaluasi rutin terhadap kinerja aplikasi dan kebutuhan pengguna. Pembaruan sistem secara berkala juga diperlukan untuk menjaga aplikasi tetap relevan dengan perkembangan teknologi.

Dengan menerapkan saran-saran ini, aplikasi kasir berbasis mobile dapat terus berkembang menjadi solusi yang lebih lengkap dan mendukung kebutuhan operasional CV Arjuna Jati Jepara secara berkelanjutan.

DAFTAR PUSTAKA

- Agustina, T. A., Aisyah, N. A., & Wirawan, R. (2023). PERANCANGAN APLIKASI ANDROID E-LEARN ARMETA DENGAN PENDEKATAN MEODE WATERFALL. *Jurnal Sistem Informasi dan Aplikasi (JSIA)*, 1(1), 34–49.
- Andilala, A., Gunawan, G., & Kirman, K. (2021). Aplikasi Informasi Lowongan Pekerjaan Menggunakan Firebase Application Programming Interface Berbasis Android. *Journal of Technopreneurship and Information System*, 4(2), 12–18.
- Arafat, A. F. R., Rizkiansah, F., & Rosyani, P. (2021). Penerapan Aplikasi Kasir Berbasis Android Pada UMKM ‘Nangkringan’ Bintaro Pesanggrahan. *Jurnal Kreativitas Mahasiswa Informatika*, 2(1), 7–10.
- Developer Training, G. (2016). *Android Developer Fundamentals Course – Concepts*.
- Fahnun Putri, F., Karim, S. A., & Muis Mappalotteng, A. (2019). *PENGEMBANGAN APLIKASI LATIHAN TOEIC (TEST OF ENGLISH FOR INTERNATIONAL COMMUNICATION) BERBASIS ANDROID*.
- Faizal, B., Yudha, S., & Nuryanto, N. (2023). APLIKASI KASIR APOTIK BERBASIS ANDROID. *Seminar Nasional Teknologi & Sains*, 2(1), 65–72.
- Grier, R. A., Bangor, A., Kortum, P., & Peres, S. C. (2013). The system usability scale: Beyond standard usability testing. *Proceedings of the human factors and ergonomics society annual meeting*, 57(1), 187–191.
- Gurning, L., Malabay, M., Simorangkir, H., & Yulhendri, Y. (2022). Aplikasi Kasir Katering Mama Suci Berbasis Android. *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, 6(3), 1–10.
- Hafidz, K., & Ikhwan, A. (2024). The Android-Based Cashier Service Information System Utilizes the Waterfall Method. *Journal of Information Systems and Informatics*, 6(2), 959–978.
- Kamilla, A. C., Priyani, N., Pranatawijaya, V. H., & Sari, N. N. K. (2024). PENGEMBANGAN APLIKASI KASIR MOBILE YANG EFISIEN. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4), 5966–5971.
- Nasrullah, A., Muslim, B., Wijaya, C. H., Pirmantara, D., & Saifudin, A. (2020). Penerapan Teknik Equivalence Partitions pada Pengujian Aplikasi Seleksi Kenaikan Jabatan di PT Maju Makmur. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 3(4), 193–199.

- Nielsen, J., & Levy, J. (1994). Measuring usability: preference vs. performance. *Communications of the ACM*, 37(4), 66–75.
- Nurudin, M., Jayanti, W., Saputro, R. D., Saputra, M. P., & Yulianti, Y. (2019). Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis. *J. Inform. Univ. Pamulang*, 4(4), 143.
- Nurwicaksono, A. B., & Ikmah, I. (2024). PERANCANGAN SISTEM MANAJEMEN KASIR BERBASIS ANDROID PADA BLESSING OF SHOES. *Journal of Information System Management (JOISM)*, 5(2), 108–114.
- Nuryamin, Y., & Risyda, F. (2024). PERANCANGAN APLIKASI KASIR PADA KEDAI KOPI BERBASIS WEB MENGGUNAKAN MODEL WATERFALL. *JSI (Jurnal sistem Informasi) Universitas Suryadarma*, 11(1), 191–198.
- Pangestu, S. D., & Astutik, I. R. I. (2024). RANCANGAN APLIKASI KASIR TOKO KELONTONG BERBASIS WEBSITE MENGGUNAKAN METODE WATERFALL. *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 9(1), 125–135.
- Parlika, R., Nisaaâ, T. A., Ningrum, S. M., & Haque, B. A. (2020). Studi Literatur Kekurangan dan Kelebihan Pengujian Black Box. *Teknomatika*, 10(2), 131–140.
- Pratama, R. Y., & Somya, R. (2021). Perancangan Aplikasi Point Of Sales (POS) Berbasis Android (Studi Kasus: Warkop Vape Salatiga). *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 8(4), 1923–1938.
- Pressman, R. S. (2005). Software Engineering: a practitioner's approach. *Pressman and Associates*.
- Rachman NIM, E. (2017). *Implementasi Enkripsi dan Dekripsi Data Sensitif pada SharedPreferences Aplikasi Android menggunakan ECC dan AES*.
- Rahayu, E. F., & Fandhika, S. S. (2021). *Aplikasi Kasir Digital dan Perhitungan Pembayaran Pajak Penghasilan UMKM Berbasis Android*.
- Rayhan, A., & Fatonah, N. S. (2024). PERANCANGAN APLIKASI KASIR KANTIN BERBASIS WEBSITE MENGGUNAKAN METODE WATERFALL (STUDI KASUS: SMKN 15 JAKARTA). *Kohesi: Jurnal Sains dan Teknologi*, 5(2), 61–70.
- Riswanto, B., & Fahrudin, I. (2024). Perancangan Aplikasi Kasir Aira Motor Berbasis Android Menggunakan Firebase Realtime Database. *Digital Transformation Technology*, 4(2), 776–784.

- Sanad, E. A. W., Achmad, A., & Dewiani, D. (2018). Pemanfaatan Realtime Database di Platform Firebase Pada Aplikasi E-Tourism Kabupaten Nabire. *Jurnal Penelitian Enjiniring*, 22(1), 20–26.
- Saputra, A., Safitri, C. F. I., Fitriyani, Y. G., & Desyani, T. (2021). Pengembangan Aplikasi Kasir Menggunakan Model Waterfall. *Jurnal Teknologi Sistem Informasi dan Aplikasi ISSN*, 2654, 3788.
- Sasmito, G. W. (2017). Penerapan metode Waterfall pada desain sistem informasi geografis industri kabupaten Tegal. *Jurnal Informatika: Jurnal Pengembangan IT*, 2(1), 6–12.
- Sholeh, M., Sinduningrum, E., & Putri, A. N. (2022). Aplikasi Kasir Portabel Berbasis Android Point of Sale Terintegrasi dengan Pencetak. *InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan*, 6(2), 227–236.
- Sibarani, N. S., Munawar, G., & Wisnuadhi, B. (2018). Analisis performa aplikasi android pada bahasa pemrograman java dan kotlin. *Prosiding Industrial Research Workshop and National Seminar*, 9, 319–324.
- Sibuea, S., Saputro, M. I., Annan, A., & Widodo, Y. B. (2022). Aplikasi Mobile Collection Berbasis Android Pada Pt. Suzuki Finance Indonesia. *Jurnal Informatika Dan Tekonologi Komputer (JITEK)*, 2(1), 31–42.
- Sonita, A., & Fardianitama, R. F. (2018). Aplikasi E-Order Menggunakan Firebase dan Algoritme Knuth Morris Pratt Berbasis Android. *Pseudocode*, 5(2), 38–45.
- Suprianto, S., Fadlan, M., & Prayogi, D. (2021). Perancangan Aplikasi Point of Sale Berbasis Web Pada Toko Project Salfa Tarakan. *Sebatik*, 25(2), 624–631.
- Susanto Anna Dara Andriana, R. (2016). Perbandingan model waterfall dan prototyping untuk pengembangan sistem informasi. *Majalah Ilmiah UNIKOM*.
- Wiratno, A. R., & Hastuti, K. (2017). Implementation of Firebase Realtime Database to track BRT Trans Semarang. *Scientific Journal of Informatics*, 4(2), 95-103.
- Zahara, T., & Widodo, T. (2024). Pengembangan Aplikasi Kasir Menggunakan User Centered Design (UCD) Berbasis Mobile: Development of Cashier Applications Using Mobile-Based User Centered Design (UCD). *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 5(1).

LAMPIRAN



Stock Tanggal : 2 - 11 - 2024

No	Produk	Barang	Saluran	Unit	Total	No	Produk	Barang	Saluran	Unit	Total
1	AQUA 1.5		61	2	61	46	KOPITAN				
2	AQUA 600		55	6	116	47	CLEO GELAS 250 ML				
3	NEAR BREAND					48	CHU BIG				
4	CINCAU	7	7		7	49	CHU JELI DRINK				
5	COCACOLA HESBERU	7			12	50	POKASS				
6	COCA COLA 1.5		21	2	28	51	POKASS 100				
7	KOKO DRINK		0	1	1	52	POKASS ACTION				
8	COCA COLA 250		0	1	1	53	TEH KELAS CUP				
9	KRATINGDENS	5	5	10	50	54	POKASS MINI 330				
10	FANTA 1.5					55	PULPY ORANGE				
11	FANTA 250		8	2	10	56	SPRITE 1.5				
12	FANTA 400					57	SPRITE 400				
13	FLORONA 5000					58	SPRITE 250				
14	FLORONA ORANGE		15	4	14	59	TANGGO DRINK				
15	PRESTIA 350		12	1	1	60	TANGGO KID0				
16	PRESTIA CUP					61	COOL TIME COCONUT				
17	KOPHOLATE 18		5	3	10	62	TESS 500 (1.5)				
18	GOLDA COFFE		16	3	16	63	REE SIAATI HESBERU				
19	KEH GOHA		50	2	32	64	JIT BOTOL 120ML				
20	REOCUP					65	TEH BANDEULAN CUP				
21	NAKATI WAFER					66	TEH GELAS BOTOL 250 ML				
22	GERY BEG ROL					67	TEH JAWANA				
23	KOPHAP					68	TEH PULCOK 250				
24	KOPHOLUCKY DAY					69	TEH PULCOK 350				
25	CLEO GELAS 115 ML					70	TORA CAFE CUP				
26	REOPLUS		75	3	77	71	LENNI GALON				
27	LARUTAN (1.5)		100	6	146						
28	LENNI 1.5		99	7	106						
29	LENNI 330										
30	LE MIN 600		100		100						
31	PELANGI		321	5	52						
32	RELUK										
33	YOU C 1000										
34	WIZINE										
35	MOUNTEA		45	4	52						

