

**PERANCANGAN APLIKASI KASIR BERBASIS *ANDROID*
UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN
PADA APAN CELL**

SKRIPSI



**MAULIDIA ROSINTA
2022230012**

**PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH
2026**

**PERANCANGAN APLIKASI KASIR BERBASIS *ANDROID*
UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN
PADA APAN CELL**

Disusun Untuk Memenuhi Salah Satu Syarat Memperoleh Gelar Sarjana
Program Studi Informatika Universitas Prabumulih



**MAULIDIA ROSINTA
2022230012**

**PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH
2026**

HALAMAN PENGESAHAN

PERANCANGAN APLIKASI KASIR BERBASIS *ANDROID* UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN PADA APAN CELL

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana
Program Studi Informatika
Universitas Prabumulih

Oleh :

Maulidia Rosinta
2022230012

Prabumulih, 10 Juni 2026

Pembimbing I

Pembimbing II



Nur Aini H, S.Kom., M.Si., M.Kom.
NUPTK. 2434764664300022



Iwan Setiawan, S.Kom., M.Kom.
NUPTK. 1541769670130303

Ketua Program Studi
Informatika



Iwan Setiawan, S.Kom., M.Kom.
NUPTK. 1541769670130303

HALAMAN PERSETUJUAN

Skripsi ini dengan judul “Perancangan Aplikasi Kasir Berbasis *Android* Untuk Pengelolaan Penjualan dan Persediaan Pada Apan Cell” telah dipertahankan dan disetujui dihadapan Tim Penguji Skripsi Program Studi Informatika Fakultas Ilmu Komputer Universitas Prabumulih pada tanggal 22 Juni 2026.

Prabumulih, 22 Juni 2026

Tim Penguji Tugas Akhir

Ketua:

Nur Aini H, S.Kom., M.Si., M.Kom
NUPTK. 2434764664300022

Anggota:

1. Iwan Setiawan, S.Kom., M.Kom
NUPTK. 1541769670130303
2. Fajriyah, S.Kom. M.Kom
NUPTK. 1159766667237103

**Mengetahui,
Dekan Fakultas Ilmu Komputer**



(Khana Wijaya, S.Kom., M.Kom)
NUPTK. 7146768669130363

**Ketua Program Studi
Informatika**



(Iwan Setiawan, S.Kom., M.Kom)
NUPTK. 1541769670130303

PERNYATAAN INTEGRITAS

Saya yang bertanda tangan di bawah ini

Nama : Maulidia Rosinta

NIM : 2022230012

Program Studi : Informatika

Menyatakan dengan sebenarnya bahwa, Skripsi ini benar-benar merupakan hasil karya saya dan tidak terdapat karya orang lain, apabila di kemudian hari terbukti atau dapat dibuktikan bahwa Skripsi ini hasil *plagiasi*, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku atas perbuatan tersebut.



Prabumulih, 22 Juni 2026

Yang membuat pernyataan,



Maulidia Rosinta

NIM. 2022230012

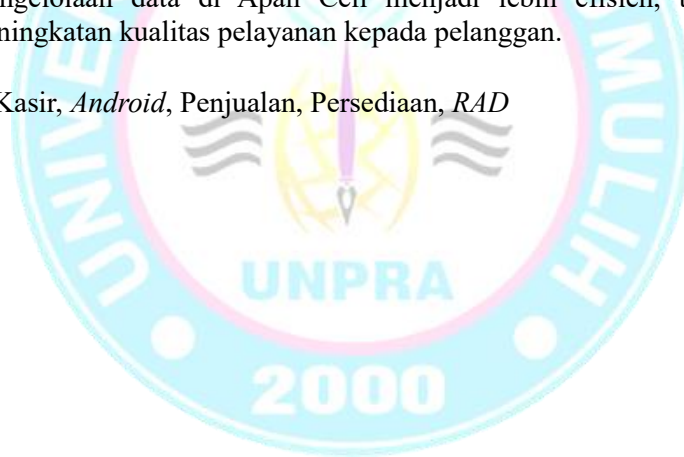
ABSTRAK

Perancangan Aplikasi Kasir Berbasis *Android* Untuk Pengelolaan Penjualan dan Persediaan pada Apan Cell

Maulidia Rosinta, 2022230012, 2026

Pengelolaan penjualan dan persediaan barang merupakan kegiatan penting dalam operasional toko. Di Apan Cell, proses pencatatan transaksi dan stok barang masih dilakukan secara manual sehingga sering menimbulkan kesalahan data, keterlambatan pelayanan, dan kesulitan dalam penyusunan laporan. Penelitian ini bertujuan merancang aplikasi kasir berbasis *Android* yang dapat membantu pencatatan penjualan serta pemantauan persediaan barang secara cepat dan akurat. Pengembangan sistem menggunakan metode *Rapid Application Development (RAD)* agar sesuai dengan kebutuhan pengguna dan mempercepat proses implementasi. Dengan adanya aplikasi ini, diharapkan pengelolaan data di Apan Cell menjadi lebih efisien, terstruktur, serta mendukung peningkatan kualitas pelayanan kepada pelanggan.

Kata Kunci : Kasir, *Android*, Penjualan, Persediaan, *RAD*



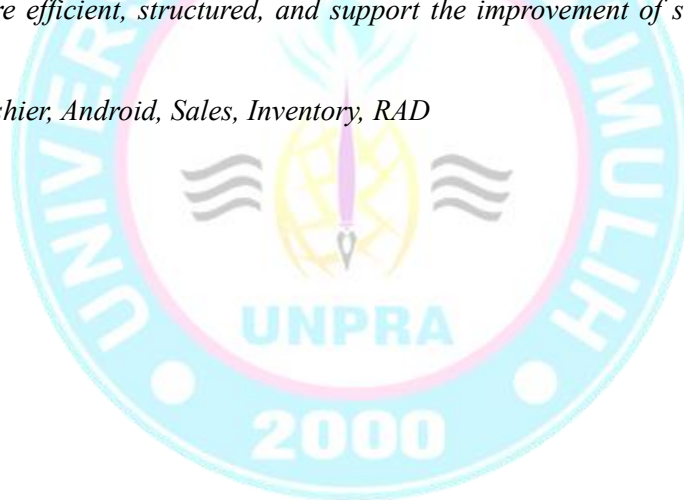
ABSTRACT

Design of an Android-Based Cashier Application for Sales and Inventory Management at Apan Cell

Maulidia Rosinta, 2022230012, 2026

Managing sales and inventory is a crucial activity in store operations. At Apan Cell, the process of recording transactions and stock is still done manually, which often leads to data errors, service delays, and difficulties in preparing reports. This research aims to design an Android-based cashier application that can assist in recording sales and monitoring inventory quickly and accurately. The system development utilizes the Rapid Application Development (RAD) method to ensure it meets user needs and accelerates the implementation process. With this application, data management at Apan Cell is expected to become more efficient, structured, and support the improvement of service quality to customers.

Keywords: *Cashier, Android, Sales, Inventory, RAD*



KATA PENGANTAR

Puji syukur saya panjatkan atas kehadiran Allah SWT yang telah memberikan segala nikmat dan karunia-Nya kepada kita semua, sehingga Skripsi dengan judul “Perancangan Aplikasi Kasir Berbasis *Android* untuk Pengelolaan Penjualan dan Persediaan pada Apan Cell” dapat diselesaikan sesuai dengan rencana dan waktu yang ditargetkan.

Skripsi ini merupakan salah satu syarat untuk mendapatkan gelar Sarjana di Program Studi S1 Informatika Fakultas Ilmu Komputer Universitas Prabumulih. Penulis banyak menerima bantuan, arahan, dan bimbingan dari berbagai pihak. Penulis mengucapkan terima kasih kepada:

1. Ibu Dr. Yuniar Pratiwi, S.Si., M.Si selaku Rektor Universitas Prabumulih.
2. Bapak Andi Christian, S.Kom., M.Kom selaku Wakil Rektor Universitas Prabumulih.
3. Bapak Khana Wijaya, S.Kom., M.Kom selaku Dekan Fakultas Universitas Prabumulih.
4. Ibu Suhartini, S.Kom., M.Kom selaku Wakil Dekan Bidang Keuangan Fakultas Ilmu Komputer Universitas Prabumulih.
5. Ibu Nur Aini H, S.Kom., M.Si., M.Kom selaku Wakil Dekan Bidang Kemahasiswaan Sekaligus Pembimbing I yang telah memberikan bimbingan, saran, serta waktu selama proses penyusunan Skripsi ini.
6. Bapak Iwan Setiawan S.Kom., M.Kom selaku Ketua Program Studi Informatika Universitas Prabumulih sekaligus Pembimbing II yang telah memberikan bimbingan dan arahan selama penyusunan Skripsi ini.
7. Ibu Riza Kartina, S.Kom., M.Kom selaku Pembimbing Akademik yang telah memberikan bimbingan dan masukan kepada penulis selama perkuliahan.
8. Seluruh Bapak dan Ibu Dosen Program Studi Informatika yang telah memberikan ilmu dan wawasan selama perkuliahan.
9. Bapak Andika Putra Abdi Negara selaku Pemilik Apan Cell yang telah memberikan izin serta membantu dalam pengumpulan data penelitian.

10. Mama tercinta, terima kasih atas doa-doa yang tidak pernah putus, kasih sayang yang tak terhingga, serta dukungan moral dan material yang menjadi penguat utama bagi penulis. Untuk Almarhum Papa, meski raga tak lagi ada, kasih sayang dan nilai-nilai kehidupan yang Papa tanamkan akan selalu hidup dalam setiap langkah penulis. Karya ini adalah wujud bakti dan cinta untuk Mama dan Papa.
11. Kakak tersayang, Imam Ardi Riansyah, terima kasih telah menjadi sosok kakak yang selalu memberikan dukungan, motivasi, dan bantuan di setiap masa sulit selama proses penyusunan ini.
12. Seluruh keluarga besar dari pihak Papa dan Mama, terima kasih atas dukungan, semangat, dan doa yang senantiasa diberikan kepada penulis.
13. Teman-teman kelas, terima kasih atas kebersamaan, bantuan, dan keceriaan yang telah kita bagi selama masa studi. Secara khusus, penulis ingin menyampaikan terima kasih kepada Ajeng Anjani, dan Eci Wulandari yang telah menjadi rekan diskusi yang hebat, pemberi semangat, dan sahabat yang selalu ada di saat suka maupun duka.
14. Anggi Saputra, sebagai pasangan penulis, yang telah memberikan doa, dukungan, semangat, dan motivasi selama proses penyusunan skripsi ini.
15. Diri sendiri, terima kasih telah bertahan, berjuang, dan tidak menyerah hingga sampai di titik ini. Pencapaian ini adalah buah dari kerja keras dan kesabaran diri.

Penulis menyadari bahwa Skripsi ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi penyempurnaan penelitian dan pengembangan aplikasi ini di masa mendatang. Semoga Skripsi ini dapat memberikan manfaat bagi penulis serta semua pihak yang berkepentingan.

Prabumulih, 22 Juni 2026

Penulis

MAULIDIA ROSINTA

DAFTAR ISI

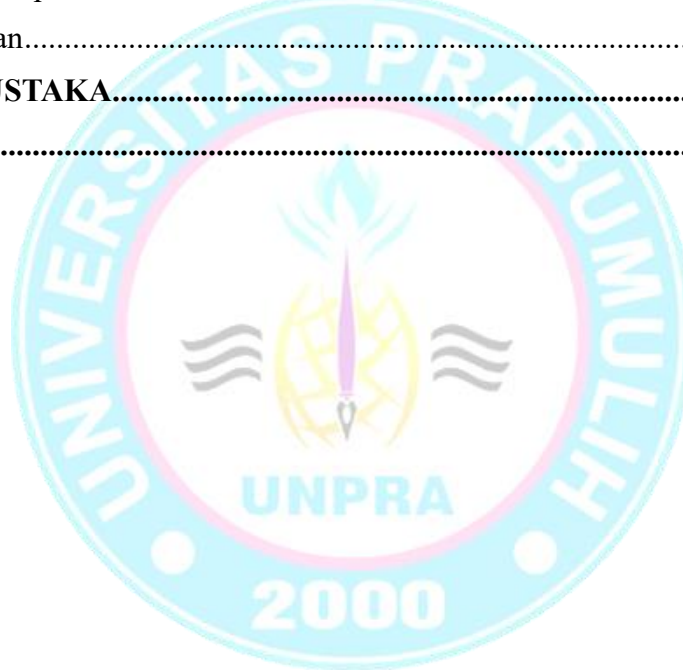
Halaman

HALAMAN JUDUL DALAM	i
HALAMAN PENGESAHAN	ii
HALAMAN PERSETUJUAN	iii
PERNYATAAN INTEGRITAS	iv
ABSTRAK	v
<i>ABSTRACT</i>	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xv
DAFTAR LAMPIRAN	xviii
DAFTAR SINGKATAN.....	xix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian	4
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA	6
2.1 Pengertian Perancangan	6
2.2 Pengertian Aplikasi	6
2.3 Pengertian Kasir	6
2.4 Pengertian <i>Android</i>	7
2.5 Pengertian Penjualan	7
2.6 Pengertian Persediaan	8
2.7 Pengertian <i>Visual Studio Code</i>	8
2.8 Pengertian <i>Dart</i>	9
2.9 Pengertian <i>Flutter</i>	9

2.10	Pengertian <i>Firestore</i>	10
2.11	Pengertian <i>Figma</i>	10
2.12	Penelitian Terdahulu.....	10
2.13	Kerangka Berfikir Penelitian.....	12
BAB III METODE PENELITIAN		14
3.1	Objek dan Jadwal Penelitian	14
3.1.1	Objek.....	14
3.1.2	Jadwal Penelitian	14
3.2	Metode Penelitian.....	15
3.3	Sumber Data.....	16
3.3.1	Data Primer	16
3.3.2	Data Sekunder.....	16
3.4	Teknik Pengumpulan Data	17
3.4.1	Observasi	17
3.4.2	Wawancara.....	17
3.4.3	Dokumentasi	17
3.4.4	Studi Literatur	18
3.5	Metode Pengembangan Sistem	18
3.6	Alat Bantu Perancangan.....	20
3.6.1	Alat Bantu Utama	20
3.6.2	Alat Bantu Pemodelan Visual	21
3.6.3	Alat Bantu Desain.....	23
3.6.4	Alat Bantu <i>Debugging</i> dan Pengembangan	24
3.7	Pengujian Sistem.....	24
BAB IV ANALISA DAN PERANCANGAN.....		26
4.1	Analisa Sistem.....	26
4.1.1	Analisa dan Evaluasi Sistem Berjalan	26
4.1.2	Analisa Kebutuhan.....	28
4.2	Perancangan Sistem (<i>Desain</i>) yang diusulkan	31
4.2.1	Perancangan Sistem (Logis)	31
4.2.2	Perancangan Database (Fisik).....	54
4.2.3	Perancangan Antar Muka.....	60
BAB V HASIL DAN PEMBAHASAN		98
5.1	Spesifikasi Perangkat	98

5.1.1	Perangkat Keras yang digunakan.....	98
5.1.2	Perangkat Lunak yang digunakan.....	99
5.2	Implementasi <i>Database</i>	100
5.2.1	Implementasi Tabel Admin	100
5.2.2	Implementasi Tabel Barang	101
5.2.3	Implementasi Tabel Kasir	102
5.2.4	Implementasi Tabel Transaksi Barang	103
5.2.5	Implementasi Tabel Transaksi Layanan.....	104
5.3	Tampilan Antar Muka	106
5.3.1	Tampilan Halaman Register.....	106
5.3.2	Tampilan Halaman <i>Login</i>	108
5.3.3	Tampilan Halaman <i>Dashboard</i> Admin	110
5.3.4	Tampilan Halaman Transaksi Admin.....	111
5.3.5	Tampilan Halaman <i>Checkout</i> Barang Admin	113
5.3.6	Tampilan Halaman <i>Scan</i> Barang Admin.....	114
5.3.7	Tampilan Halaman Transaksi Layanan Admin	116
5.3.8	Tampilan Halaman Daftar Barang	118
5.3.9	Tampilan Halaman Tambah Barang.....	119
5.3.10	Tampilan Halaman Edit Barang.....	121
5.3.11	Tampilan Halaman Laporan Penjualan	122
5.3.12	Tampilan Halaman Cetak PDF	124
5.3.13	Tampilan Halaman Pengaturan	126
5.3.14	Tampilan Halaman Profil Toko	127
5.3.15	Tampilan Halaman Edit Profil Toko	129
5.3.16	Tampilan Halaman Kelola Kasir.....	130
5.3.17	Tampilan Halaman Tambah Kasir.....	132
5.3.18	Tampilan Halaman Edit Kasir.....	133
5.3.19	Tampilan Halaman <i>Dashboard</i> Kasir	135
5.3.20	Tampilan Halaman Transaksi Kasir	136
5.3.21	Tampilan Halaman <i>Checkout</i> Barang Kasir.....	137
5.3.22	Tampilan Halaman <i>Scan</i> Barang Kasir	139
5.3.23	Tampilan Halaman Transaksi Layanan Kasir	140
5.4	Hasil Pengujian Sistem	141
5.4.1	Pengujian Fungsi Register	141

5.4.2	Pengujian Fungsi <i>Login</i>	147
5.4.3	Pengujian Fungsi Tambah Barang	152
5.4.4	Pengujian Fungsi Edit Barang	158
5.4.5	Pengujian Fungsi Transaksi Barang.....	165
5.4.6	Pengujian Fungsi Transaksi Layanan	170
5.4.7	Pengujian Fungsi Tambah Kasir	174
5.4.8	Pengujian Fungsi Edit Kasir	180
5.4.9	Pengujian Fungsi Edit Profil Toko.....	185
5.4.10	Pengujian Fungsi <i>Logout</i>	189
BAB VI KESIMPULAN DAN SARAN.....		194
6.1	Kesimpulan	194
6.2	Saran.....	195
DAFTAR PUSTAKA.....		196
LAMPIRAN.....		200



DAFTAR TABEL

	Halaman
Tabel 2.1 Penelitian Terdahulu.....	11
Tabel 3.1 Jadwal Penelitian.....	15
Tabel 3.2 Simbol <i>Use Case</i>	21
Tabel 3.3 Simbol <i>Activity Diagram</i>	22
Tabel 3.4 Simbol <i>Class Diagram</i>	22
Tabel 4.1 Admin	54
Tabel 4.2 Barang	55
Tabel 4.3 Kasir	56
Tabel 4.4 Transaksi Barang	57
Tabel 4.5 Transaksi Layanan.....	58
Tabel 5.1 Identifikasi <i>Statement</i> Fungsi Register.....	142
Tabel 5.2 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Register.....	143
Tabel 5.3 Identifikasi <i>Statement</i> fungsi <i>Login</i>	148
Tabel 5.4 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi <i>Login</i>	149
Tabel 5.5 Identifikasi <i>Statement</i> fungsi Tambah Barang.....	154
Tabel 5.6 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Tambah Barang	155
Tabel 5.7 Identifikasi <i>Statement</i> fungsi Edit Barang.....	161
Tabel 5.8 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Edit Barang	161
Tabel 5.9 Identifikasi <i>Statement</i> fungsi Transaksi Barang	167
Tabel 5.10 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Transaksi Barang	167
Tabel 5.11 Identifikasi <i>Statement</i> fungsi Transaksi Layanan	171
Tabel 5.12 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Transaksi Layanan	172
Tabel 5.13 Identifikasi <i>Statement</i> fungsi Tambah Kasir	176
Tabel 5.14 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Tambah Kasir.....	177
Tabel 5.15 Identifikasi <i>Statement</i> fungsi Edit Kasir.....	182
Tabel 5.16 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Edit Kasir.....	182
Tabel 5.17 Identifikasi <i>Statement</i> fungsi Edit Profil Toko	186
Tabel 5.18 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi Edit Profil Toko	186

Tabel 5.19 Identifikasi <i>Statement</i> fungsi <i>Logout</i>	190
Tabel 5.20 <i>Test Case</i> untuk <i>Statement Coverage</i> fungsi <i>Logout</i>	190



DAFTAR GAMBAR

	Halaman
Gambar 2.1 Kerangka Berfikir Penelitian.....	12
Gambar 3.1 Tahapan RAD.....	19
Gambar 4.1 <i>Usecase</i> Diagram.....	33
Gambar 4.2 Registrasi <i>Activity</i>	34
Gambar 4.3 <i>Login Activity</i>	35
Gambar 4.4 Tambah Barang <i>Activity</i>	36
Gambar 4.5 Edit Barang <i>Activity</i>	38
Gambar 4.6 Hapus Barang <i>Activity</i>	40
Gambar 4.7 Tambah Data Kasir <i>Activity</i>	42
Gambar 4.8 Edit Data Kasir <i>Activity</i>	44
Gambar 4.9 Hapus Data Kasir <i>Activity</i>	46
Gambar 4.10 Transaksi Barang <i>Activity</i>	48
Gambar 4.11 Transaksi Layanan <i>Activity</i>	50
Gambar 4.12 Laporan <i>Activity</i>	52
Gambar 4.13 <i>Class</i> Diagram yang diusulkan.....	53
Gambar 4.14 Halaman Registrasi	60
Gambar 4.15 Halaman <i>Login</i>	62
Gambar 4.16 Halaman <i>Dashboard</i> Admin.....	64
Gambar 4.17 Halaman Pilihan Transaksi Admin.....	66
Gambar 4.18 Halaman <i>Checkout</i> Barang Admin.....	67
Gambar 4.19 Halaman <i>Scan</i> Barang Admin	69
Gambar 4.20 Halaman Transaksi Layanan Admin	70
Gambar 4.21 Halaman Daftar Barang.....	71
Gambar 4.22 Halaman Tambah Barang	73
Gambar 4.23 Halaman Edit Barang	75
Gambar 4.24 Halaman Laporan Transaksi.....	77
Gambar 4.25 Halaman Cetak PDF	79
Gambar 4.26 Halaman Pengaturan	80

Gambar 4.27 Halaman Profil Toko	82
Gambar 4.28 Halaman Edit Profil Toko.....	84
Gambar 4.29 Halaman Manajemen Kasir.....	86
Gambar 4.30 Halaman Tambah Kasir	87
Gambar 4.31 Halaman Edit Data Kasir.....	89
Gambar 4.32 Halaman <i>Dashboard</i> Kasir	90
Gambar 4.33 Halaman Pilihan Transaksi Kasir	91
Gambar 4.34 Halaman <i>Checkout</i> Barang Kasir	93
Gambar 4.35 Halaman <i>Scan</i> Barang Kasir	95
Gambar 4.36 Halaman Transaksi Layanan Kasir.....	96
Gambar 5.1 Tabel Admin	100
Gambar 5.2 Tabel Barang.....	101
Gambar 5.3 Tabel Kasir	102
Gambar 5.4 Tabel Transaksi Barang	103
Gambar 5.5 Tabel Transaksi Layanan.....	104
Gambar 5.6 Halaman Register	106
Gambar 5.7 Halaman <i>Login</i>	108
Gambar 5.8 Halaman <i>Dashboard</i> Admin.....	110
Gambar 5.9 Halaman Transaksi Admin	111
Gambar 5. 10 Halaman <i>Checkout</i> Barang Admin.....	113
Gambar 5.11 Halaman <i>Scan</i> Barang Admin	114
Gambar 5.12 Halaman Transaksi Layanan Admin	116
Gambar 5.13 Halaman Daftar Barang.....	118
Gambar 5.14 Halaman Tambah Barang	119
Gambar 5.15 Halaman Edit Barang	121
Gambar 5.16 Halaman Laporan Barang.....	122
Gambar 5.17 Halaman Cetak PDF	124
Gambar 5.18 Halaman Pengaturan	126
Gambar 5.19 Halaman Profil Toko	127
Gambar 5.20 Halaman Edit Profil Toko.....	129
Gambar 5.21 Halaman Kelola Kasir	130
Gambar 5.22 Halaman Tambah Kasir	132

Gambar 5.23 Halaman Edit Kasir	133
Gambar 5.24 Halaman <i>Dashboard</i> Kasir	135
Gambar 5.25 Halaman Transaksi Kasir	136
Gambar 5.26 Halaman <i>Checkout</i> Kasir.....	137
Gambar 5.27 Halaman <i>Scan</i> Barang Kasir	139
Gambar 5.28 Halaman Transaksi Layanan Kasir.....	140
Gambar 5.29 <i>Flowchart</i> Register	145
Gambar 5.30 <i>Flowgraph</i> Register.....	146
Gambar 5.31 <i>Flowchart</i> Login.....	150
Gambar 5.32 <i>Flowgraph</i> Login.....	151
Gambar 5.33 <i>Flowchart</i> Tambah Barang	156
Gambar 5.34 <i>Flowgraph</i> Tambah Barang.....	157
Gambar 5.35 <i>Flowchart</i> Edit Barang	162
Gambar 5.36 <i>Flowgraph</i> Edit Barang.....	164
Gambar 5.37 <i>Flowchart</i> Transaksi Barang	168
Gambar 5.38 <i>Flowgraph</i> Transaksi Barang	169
Gambar 5.39 <i>Flowchart</i> Transaksi Layanan	172
Gambar 5.40 <i>Flowgraph</i> Transaksi Layanan	174
Gambar 5.41 <i>Flowchart</i> Tambah Kasir.....	178
Gambar 5.42 <i>Flowgraph</i> Tambah Kasir.....	179
Gambar 5.43 <i>Flowchart</i> Edit Kasir.....	183
Gambar 5.44 <i>Flowgraph</i> Edit Kasir.....	184
Gambar 5.45 <i>Flowchart</i> Edit Profil Toko	187
Gambar 5.46 <i>Flowgraph</i> Edit Profil Toko	188
Gambar 5.47 <i>Flowchart</i> Logout.....	191
Gambar 5.48 <i>Flowgraph</i> Logout.....	192

DAFTAR LAMPIRAN

Halaman

Lampiran 1 Surat Permohonan Penelitian.....	200
Lampiran 2 Surat Balasan Permohonan Penelitian.....	201
Lampiran 3 Dokumentasi Wawancara & Observasi	202
Lampiran 4 Daftar Pertanyaan Wawancara.....	203
Lampiran 5 Daftar Wawancara Penyerahan Aplikasi	205



DAFTAR SINGKATAN



IT	: <i>Information Technology</i>
IDE	: <i>Integrated Development Environment</i>
UI	: <i>User Interface</i>
UML	: <i>Unified Modeling Language</i>
SQL	: <i>Structured Query Language</i>
RAD	: <i>Rapid Application Development</i>
FIFO	: <i>First In-First Out</i>
QR CODE	: <i>Quick Response Code</i>
CLI	: <i>Command Line Interface</i>
UKM	: <i>Usaha Kecil dan Menengah</i>
NUPTK	: <i>Nomor Unik Pendidik dan Tenaga Kependidikan</i>
API	: <i>Application Programming Interface</i>
SDK	: <i>Software Development Kit</i>
CRUD	: <i>Create, Read, Update, Delete</i>
PDF	: <i>Portable Document Format</i>
VS CODE	: <i>Visual Studio Code</i>
NoSQL	: <i>Not Only SQL</i>
UID	: <i>User Identifier</i>
OTP	: <i>One Time Password</i>

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi dan komunikasi di era Revolusi Industri 4.0 membawa perubahan besar dalam berbagai aspek kehidupan manusia. Kemajuan teknologi telah mendorong banyak perusahaan untuk mengadopsi penggunaan internet untuk meningkatkan proses bisnis mereka, baik secara internal maupun eksternal. Hal ini dilakukan untuk menciptakan sistem kerja yang lebih praktis dan fleksibel dalam memenuhi kebutuhan pelanggan (Assegaf et al., 2024: 43). Hampir semua bidang termasuk pendidikan, kesehatan, pemerintahan, hingga perdagangan dan ritel mengalami transformasi ke arah digital. Teknologi menjadi kunci utama dalam meningkatkan kecepatan, akurasi, serta efisiensi dalam proses kerja, sehingga banyak sektor yang meninggalkan sistem manual dan beralih menggunakan sistem berbasis komputer atau aplikasi.

Salah satu bentuk pemanfaatan teknologi yang berkembang pesat adalah penggunaan aplikasi berbasis *Android*. *Android* menjadi *platform* yang paling banyak digunakan karena bersifat terbuka (*open source*), mudah dioperasikan, serta dapat diakses oleh berbagai kalangan dengan biaya yang relatif murah. Berdasarkan pengamatan penulis, kemudahan tersebut menjadi alasan utama mengapa *Android* cocok dijadikan sarana pengembangan aplikasi bisnis, termasuk sistem kasir digital. Dengan menggunakan aplikasi berbasis *Android*, pemilik usaha dapat melakukan pencatatan transaksi, memantau persediaan barang, dan membuat laporan penjualan secara *real-time* hanya melalui perangkat *mobile*, tanpa bergantung pada kasir manual. Hal ini membuat proses pelayanan menjadi lebih cepat, praktis dan efisien.

Berdasarkan hasil penelitian, dapat diketahui bahwa proses transaksi penjualan dan pencatatan persediaan di toko tersebut masih dilakukan secara manual menggunakan buku. Apan Cell menjual produk seperti *handphone*, aksesoris *handphone*, rokok, minuman dan es krim, serta layanan digital seperti pengisian pulsa, paket data, pembayaran tagihan, *top-up e-wallet*, transfer bank dan tarik tunai. Sistem manual ini sering menimbulkan beberapa permasalahan, antara lain risiko kesalahan penulisan data transaksi, lambatnya pelayanan kepada

konsumen, serta proses pengecekan persediaan yang membutuhkan waktu cukup lama.

Permasalahan tersebut berdampak pada tidak optimalnya pengelolaan operasional toko. Kesalahan pencatatan maupun data transaksi yang tidak terdokumentasi dengan baik dapat menyebabkan ketidaksesuaian antara jumlah persediaan barang secara nyata dengan data yang tertulis di buku. Selain itu penyusunan laporan penjualan harian, mingguan, maupun bulanan juga menjadi kurang efektif karena seluruh data harus direkap kembali secara manual. Kondisi ini pada akhirnya dapat memperlambat proses pelayanan kepada pelanggan dan mengurangi efisiensi kerja.

Hal ini menunjukkan bahwa, diperlukan suatu sistem yang mampu membantu pengelolaan transaksi dan persediaan barang secara lebih terstruktur. Salah satu solusi yang tepat adalah penerapan aplikasi kasir berbasis *Android* yang dapat mencatat transaksi secara otomatis, memantau persediaan barang secara *real-time*, serta menghasilkan laporan transaksi dengan cepat dan akurat. Dengan adanya aplikasi ini, diharapkan risiko kesalahan pencatatan dapat diminimalkan, proses pelayanan menjadi lebih efektif, dan pengelolaan usaha di Apan Cell dapat berjalan lebih efisien.

Berdasarkan perkembangan teknologi, potensi *Android* sebagai *platform*, bukti nyata dari penelitian sebelumnya, serta permasalahan yang dihadapi oleh Apan Cell, maka penelitian ini diarahkan untuk merancang sebuah aplikasi kasir berbasis *Android* yang sesuai kebutuhan. Penelitian ini tidak hanya bertujuan untuk memberikan manfaat langsung bagi Apan Cell, tetapi juga menambah referensi dalam bidang pengembangan aplikasi kasir dan manajemen persediaan barang. Oleh karena itu, penulis mengambil judul ini, **Perancangan Aplikasi Kasir Berbasis *Android* untuk Pengelolaan Penjualan dan Persediaan pada Apan Cell**.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka rumusan masalah dalam penelitian ini adalah sebagai berikut :

1. Bagaimana merancang dan membangun aplikasi kasir berbasis *Android* yang dapat mencatat transaksi penjualan dengan cepat dan akurat di Apan Cell?.
2. Bagaimana aplikasi tersebut dapat memantau persediaan barang seperti *handphone*, aksesoris *handphone*, rokok, minuman dan es krim yang masuk dan keluar secara *real-time*?.
3. Bagaimana aplikasi kasir dapat menghasilkan laporan transaksi dan persediaan barang yang rapi, sistematis, serta mudah dipahami oleh pemilik toko ataupun karyawan?.
4. Sejauh mana penerapan aplikasi kasir berbasis *Android* dapat meningkatkan efisiensi operasional toko dibandingkan dengan sistem manual?.

1.3 Batasan Masalah

Batasan penelitian ini ditetapkan agar penelitian lebih terarah dan fokus, yaitu sebagai berikut :

1. Penelitian ini difokuskan pada pengelolaan transaksi dan persediaan barang di Apan Cell, yang meliputi proses transaksi penjualan, pencatatan barang yang masuk dan keluar, serta pemantauan persediaan secara *real-time* melalui aplikasi kasir berbasis *Android*.
2. Laporan dibuat mencakup laporan transaksi dan persediaan dalam periode harian, mingguan dan bulanan, serta tidak mencakup integrasi dengan sistem pembayaran atau aplikasi lain di luar sistem yang dirancang.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Merancang dan mengembangkan aplikasi kasir berbasis *Android* yang dapat mencatat transaksi dengan cepat dan akurat di Apan Cell.

2. Membangun aplikasi yang dapat memantau persediaan barang seperti *handphone*, aksesoris *handphone*, rokok, minuman, dan es krim secara *real-time*.
3. Menghasilkan laporan transaksi dan persediaan yang rapi, sistematis, informatif, dan mudah dipahami oleh pemilik toko ataupun karyawan.
4. Melihat sejauh mana aplikasi kasir berbasis *Android* dapat membantu toko bekerja lebih cepat dan efisien dibandingkan pencatatan manual.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan beberapa manfaat sebagai berikut :

1. Membantu pemilik dan karyawan Apan Cell dalam mengelola transaksi dan persediaan barang secara cepat, akurat dan *real-time* melalui aplikasi kasir berbasis *Android*, serta menghasilkan laporan transaksi dan persediaan yang rapi, sistematis, informatif, dan mudah dipahami.
2. Menjadi referensi atau bahan acuan bagi peneliti lain yang ingin mengembangkan penelitian serupa dibidang aplikasi kasir berbasis *Android* atau sistem informasi penjualan.

1.6 Sistematika Penulisan

Sistematika penulisan ini disusun agar pembahasan penelitian lebih terarah dan tidak menyimpang dari tujuan yang telah ditetapkan. Adapun sistematika penulisan Skripsi ini adalah sebagai berikut :

BAB I

PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II

TINJAUAN PUSTAKA

Bab ini berisi teori-teori pendukung, hasil penelitian terdahulu, serta komponen yang digunakan dalam Perancangan Aplikasi Kasir Berbasis *Android* untuk

Pengelolaan Penjualan dan Persediaan pada Apan Cell yang akan dijadikan dasar penulisan Skripsi ini.

BAB III

METODE PENELITIAN

Bab ini berisi tentang metode penelitian yang digunakan, tahapan perancangan sistem, serta langkah-langkah dalam proses pengembangan aplikasi kasir berbasis *Android* untuk Apan Cell.

BAB IV

ANALISIS DAN PERANCANGAN

Bab ini membahas proses analisis serta perancangan sistem yang dilakukan berdasarkan permasalahan yang diteliti. Tahapan analisis digunakan untuk mengidentifikasi kebutuhan sistem sebagai landasan dalam merancang sistem yang akan dikembangkan.

BAB V

HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil penelitian yang telah diperoleh serta pembahasannya. Pembahasan dilakukan berdasarkan analisis data secara kualitatif dan kuantitatif guna memberikan gambaran yang jelas terhadap hasil penelitian.

BAB VI

KESIMPULAN DAN SARAN

Bab ini memuat kesimpulan dan saran berdasarkan hasil penelitian yang telah dilaksanakan. Kesimpulan disusun untuk menjawab rumusan masalah serta menggambarkan hasil penyelesaian penelitian secara objektif. Saran diberikan sebagai masukan atau rekomendasi untuk perbaikan dan pengembangan selanjutnya sesuai dengan ruang lingkup penelitian.

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Perancangan

Perancangan adalah sebuah proses atau tahapan untuk membuat atau merencanakan sesuatu dengan menggunakan teknik untuk merumuskan tujuan yang akan dicapai (Fauzi et al., 2022). Perancangan adalah suatu kegiatan yang berhubungan berdasarkan evaluasi yang telah dilakukan pada kegiatan analisis (Mulyana et al., 2021).

Dari kedua pendapat tersebut, dapat disimpulkan bahwa perancangan merupakan suatu proses untuk menyatukan analisis, tujuan, dan hasil yang ingin dicapai. Melalui perancangan, ide dan hasil analisis dapat diwujudkan menjadi suatu yang lebih jelas, terarah, dan mudah diterapkan sesuai dengan kebutuhan.

2.2 Pengertian Aplikasi

Aplikasi merupakan perangkat lunak komputer yang menggunakan sistem komputerisasi dan bermanfaat untuk mempermudah pengguna dalam melakukan suatu tugas yang diinginkan (Surahmat, 2023). Aplikasi dapat diartikan sebagai suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia (Arianto et al., 2022).

Berdasarkan kedua pendapat tersebut, dapat disimpulkan bahwa aplikasi merupakan perangkat lunak yang digunakan untuk mendukung, mempermudah, dan mempercepat pekerjaan manusia. Aplikasi tidak hanya berperan sebagai alat bantu, tetapi juga sebagai solusi praktis yang mengintegrasikan teknologi komputerisasi dengan kebutuhan pengguna, sehingga setiap kegiatan yang dilakukan dapat terlaksana dengan lebih efisien dan efektif.

2.3 Pengertian Kasir

Kasir merupakan salah satu peran penting dalam transaksi jual-beli. Kasir memiliki tugas untuk menerima pembayaran, mengkonfirmasi adanya pembelian, memberikan uang kembalian dan bukti pembayaran serta memberikan barang yang

telah dibeli oleh seseorang (Siregar, 2025). Kasir adalah tempat melakukan transaksi atau pembayaran yang menjadi akhir dari sebuah transaksi jual beli antara konsumen dan produsen sehingga menghasilkan kepuasan kepada masing-masing pihak (Hafidh et al., 2021).

Dari kedua pendapat tersebut, dapat disimpulkan bahwa kasir adalah suatu fasilitas penting dalam kegiatan jual beli yang berfungsi sebagai titik akhir transaksi. Kasir tidak hanya menjadi tempat pembayaran, tetapi juga menjadi bagian dari pelayanan yang menentukan kepuasan konsumen terhadap produsen. Dengan adanya kasir, proses jual beli dapat berjalan lebih teratur, transparan, dan memberikan kepastian bagi kedua belah pihak dalam melakukan pertukaran barang atau jasa dengan nilai tukar yang berupa pembayaran.

2.4 Pengertian *Android*

Android merupakan sistem operasi berbasis *Linux* yang sedang berkembang ditengah operasi sistem lainnya yang menyediakan *platform* terbuka bagi para pengembang dalam menciptakan aplikasi mereka sendiri (Johari, 2024). *Android* merupakan sebuah sistem operasi yang dirancang untuk perangkat seluler seperti telepon pintar (*smartphone*) dan komputer tablet. Sistem operasi *Android* ini berbasis *Linux*, yang mengartikan bahwa pondasi dasar sistem operasi ini adalah *Linux*. *Linux* sendiri merupakan sistem operasi yang dibuat untuk komputer, namun dikembangkan lagi menjadi versi *mobile* berbasis *touchscreen*, agar bisa digunakan untuk *smartphone* dan tablet (Suhendri, 2024: 2).

Berdasarkan kedua pendapat di atas, dapat disimpulkan bahwa *Android* merupakan sistem operasi berbasis *Linux* yang bersifat terbuka (*open source*) dan dikembangkan secara khusus untuk perangkat seluler seperti *smartphone* dan tablet. *Android* memberikan kebebasan bagi para pengembang untuk membuat serta mengembangkan aplikasi mereka sendiri, sehingga menjadikannya salah satu sistem operasi yang paling banyak digunakan di dunia.

2.5 Pengertian Penjualan

Penjualan adalah suatu aktifitas perusahaan yang utama dalam memperoleh pendapatan, baik untuk perusahaan kecil ataupun perusahaan besar, penjualan

merupakan sasaran akhir dari kegiatan pemasaran, karena pada bagian ini ada penetapan harga, diadakan perundingan dan perjanjian (Cahyono & Jayanti, 2022). Penjualan adalah aktivitas maupun kegiatan yang dilakukan manusia yang saling menguntungkan satu sama lain, dimana penjual menawarkan produk yang mereka miliki kepada konsumen, sehingga mampu menarik kecenderungan konsumen tersebut untuk sedia mengeluarkan uang guna memberi suatu produk yang telah ditawarkan oleh penjual (Ardian et al., 2021).

Berdasarkan kedua pendapat tersebut, dapat disimpulkan bahwa penjualan merupakan kegiatan penting dalam dunia usaha yang berfungsi untuk menghasilkan pendapatan sekaligus menjadi tujuan akhir dari pemasaran. Penjualan bukan hanya sekedar proses pertukaran barang dengan uang, tetapi juga melibatkan strategi penawaran, penetapan harga, hingga perundingan agar tercapai kesepakatan yang saling menguntungkan antara penjual dan pembeli.

2.6 Pengertian Persediaan

Persediaan adalah barang dagang yang dimiliki perusahaan untuk dijual kembali atau diproses terlebih dahulu kemudian baru dijual (Puspitasari, 2024). Persediaan adalah stok dari suatu item atau sumber daya-sumber daya organisasi perusahaan yang disimpan dalam antisipasi terhadap pemenuhan permintaan para konsumen (Tangkere et al., 2024).

Dari kedua pendapat diatas, bisa disimpulkan bahwa persediaan adalah barang atau stok yang dimiliki sebuah usaha untuk dijual kepada pelanggan. Persediaan disiapkan agar perusahaan dapat memenuhi permintaan pembeli dengan baik dan menjaga kegiatan usaha tetap berjalan lancar.

2.7 Pengertian *Visual Studio Code*

Visual Studio Code adalah salah satu *Integrated Development Environment (IDE)* yang sangat efektif dan populer dikalangan pengembang (Gunawan, 2024: 1). *Visual Studio Code* adalah sebuah teks editor ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform* yang artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows* (Martin & Dewanto, 2023).

Dari kedua pendapat di atas, dapat disimpulkan bahwa *Visual Studio Code* merupakan sebuah aplikasi editor kode yang dikembangkan oleh *Microsoft* dan dapat digunakan di berbagai sistem operasi seperti *Windows*, *Linux*, dan *Mac*. Aplikasi ini digunakan oleh pengembang untuk menulis, mengedit, dan mengelola kode program dengan mudah karena memiliki tampilan yang sederhana serta dukungan berbagai ekstensi yang mempermudah proses pembuatan aplikasi.

2.8 Pengertian Dart

Dart adalah bahasa pemrograman *open source* untuk membuat aplikasi *web* berbasis *browser* yang kompleks. Pengguna dapat menjalankan aplikasi yang dibuat di *Dart* baik dengan menggunakan *browser* yang secara langsung mendukung kode *Dart* atau dengan mengkompilasi kode *Dart* pengguna ke *JavaScript* (Sofi & Dharmawan, 2022). *Dart* adalah bahasa pemrograman yang dikembangkan oleh *Google* untuk kebutuhan dalam membuat aplikasi *Android* atau *mobile*, *front-end*, *web*, *IoT*, *back-end (CLI)*, dan *game* (Suryana, 2021: 1).

Dari kedua pengertian di atas, dapat disimpulkan bahwa *Dart* adalah bahasa pemrograman yang dikembangkan oleh *Google* dan digunakan untuk membuat berbagai jenis aplikasi, seperti *mobile*, *web*, maupun *desktop*. Bahasa ini bersifat fleksibel, mudah digunakan, serta menjadi dasar utama dalam pengembangan aplikasi menggunakan *framework Flutter*.

2.9 Pengertian Flutter

Flutter adalah *framework open source* yang memungkinkan pengembang untuk membuat aplikasi *mobile* dengan cepat dan mudah (Gunawan, 2024: 7). *Flutter* adalah sebuah *framework open-source* yang dikembangkan oleh *Google* untuk membangun antar muka (*user interface*) aplikasi *Android* dan *iOS* (Setiawan et al., 2022).

Dari kedua pendapat di atas, dapat disimpulkan bahwa *Flutter* adalah *framework open source* yang dikembangkan oleh *Google* untuk membangun aplikasi *Android* dan *iOS*. *Flutter* memudahkan pengembang dalam membuat tampilan antarmuka yang menarik, responsif, serta memungkinkan pembuatan aplikasi lintas *platform* dengan satu basis kode.

2.10 Pengertian *Firebase*

Firebase sendiri memiliki keunggulan dengan memiliki fitur lengkap seperti *authentication*, *firestore database*, *realtime database*, dan *google cloud messaging* (Setyawan, 2024). *Firebase* merupakan suatu layanan yang dimiliki oleh *google* dan digunakan untuk mempermudah para developer aplikasi dalam mengembangkan aplikasinya. *Firebase cloud service provider* dan *backend as a service* ini merupakan solusi yang ditawarkan oleh *google* untuk mempermudah pekerjaan developer dalam pengembangan aplikasi *mobile* maupun *web* (Aziz, 2021: 25).

Berdasarkan pendapat di atas, dapat disimpulkan bahwa *firebase* merupakan layanan yang dikembangkan oleh *google* untuk mempermudah pengembang dalam membangun aplikasi berbasis *mobile* maupun *web*. *Firebase* menyediakan berbagai fitur seperti *authentication*, *cloud firestore*, *realtime database*, dan *cloud messaging* yang mendukung proses pengelolaan data serta pengembangan aplikasi menjadi lebih cepat, mudah, dan efisien.

2.11 Pengertian *Figma*

Figma adalah salah satu *tools* atau aplikasi desain yang dapat digunakan pada *Windows* untuk membuat *prototype* aplikasi serta berbagai desain lainnya (Suparman et al., 2023). *Figma* merupakan perangkat desain yang umumnya dipakai untuk mendesain antarmuka aplikasi *mobile*, *desktop*, dan *website* (Pamungkas, 2023).

Dari kedua pendapat di atas, dapat disimpulkan bahwa *Figma* adalah aplikasi desain berbasis *web* yang digunakan untuk membuat *prototype* dan tampilan antarmuka aplikasi. *Figma* memudahkan desainer dalam membuat rancangan karena dapat digunakan secara kolaboratif, mudah diakses, serta mendukung berbagai jenis desain antarmuka.

2.12 Penelitian Terdahulu

Penelitian terdahulu merupakan kajian terhadap hasil penelitian sebelumnya yang memiliki keterkaitan dengan topik yang sedang dibahas. Kajian ini bertujuan untuk mengetahui sejauh mana penelitian serupa telah dilakukan, metode apa yang

digunakan, serta hasil yang telah dicapai oleh peneliti lain sebagai bahan perbandingan dan pengembangan penelitian ini. Beberapa penelitian yang relevan dengan topik perancangan aplikasi kasir berbasis *Android* dapat dilihat pada Tabel 2.1 di bawah ini.

Tabel 2.1 Penelitian Terdahulu

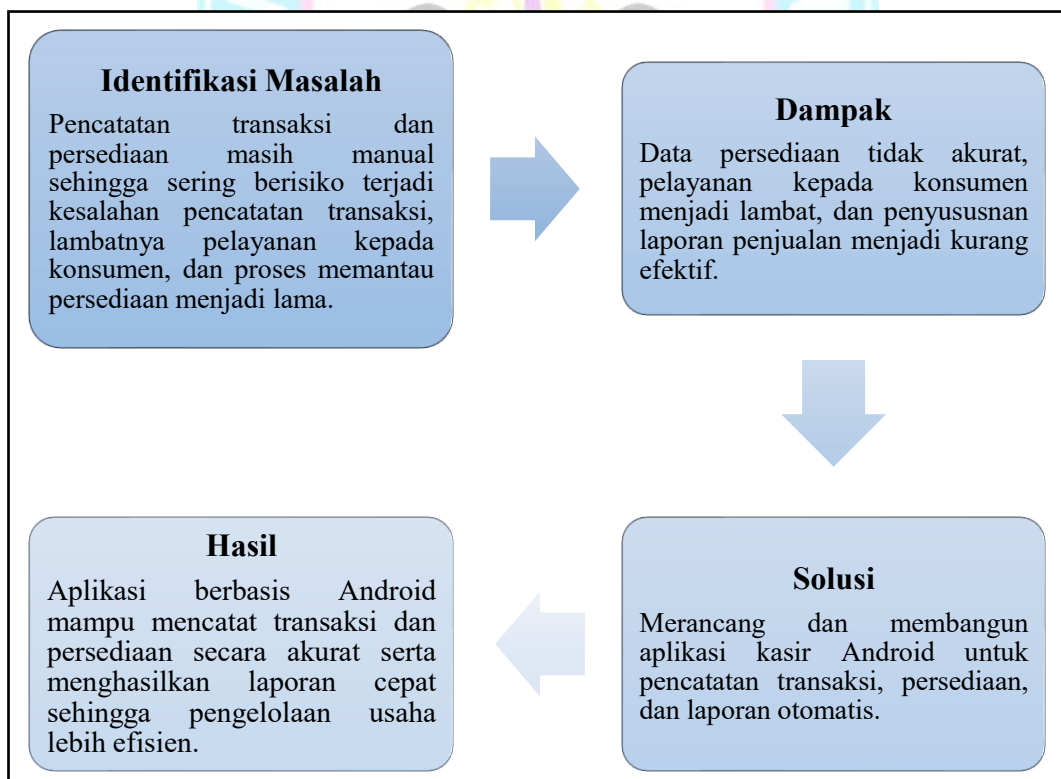
No	Judul	(Penulis, Tahun)	Hasil
1	Aplikasi Kasir Agen <i>Portable</i> Berbasis <i>Mobile</i>	(Hasim & Arifin, 2025)	Menargetkan kemudahan transaksi bisnis lewat perangkat <i>mobile</i> , mengurangi kesalahan pencatatan dan mempercepat laporan dibanding metode manual.
2	Sistem Informasi Manajemen Persediaan Barang Berbasis <i>Website</i> di PT. Rejo Mulyo Solution	(Lisa et al., 2025)	Sistem ini menyelesaikan masalah manual persediaan, kurang akurasi data, kesulitan pelacakan barang, mengambil keputusan pengadaan ulang. Relevan sebagai acuan modul persediaan agar akurat dan praktis.
3	Aplikasi <i>E-Stock Mobile</i> dengan <i>QR Code</i> untuk Pengelolaan Persediaan Barang Elektronik	(Nugraha & Kalifia, 2024)	Membantu proses manajemen stok barang elektronik dengan akurasi dan kecepatan, sebagai contoh integrasi aplikasi kasir dengan modul stok di perangkat <i>mobile</i> .
4	Perancangan Sistem Manajemen Kasir Berbasis <i>Android</i> (<i>Blessing of Shoes</i>)	(Nurwicaksono, 2024)	Sistem ini memungkinkan pengelolaan produk, stok, transaksi, dan keuangan dalam satu aplikasi <i>Android</i> menggunakan metode <i>RAD</i> .
5	Perancangan Aplikasi Kasir Aira Motor Berbasis <i>Android</i> Menggunakan <i>Firebase Realtime Database</i>	(Riswanto & Fahrudin, 2024)	Data transaksi disimpan <i>real-time</i> , mempercepat proses pembayaran dan memungkinkan pemilik melihat laporan secara langsung. Menjadi contoh penggunaan basis data <i>online</i> di aplikasi kasir.
6	Sistem Manajemen Persediaan Stok Barang Berbasis <i>Web</i> yang Meningkatkan Efisiensi Stok Barang	(Ardian Nugraha & Aziz, 2024)	Sistem ini meningkatkan efisiensi persediaan barang melalui pencatatan masuk dan keluar secara otomatis dan meminimalkan kesalahan persediaan. Menjadi referensi bagaimana modul stok harus dirancang agar efisien.
7	Perancangan Aplikasi <i>Point of Sales</i> Berbasis <i>Android</i> pada Komedi Cafe	(Krischnan et al., 2024)	Aplikasi kasir <i>mobile</i> menggantikan sistem desktop, mempermudah transaksi dan laporan penjualan. Menjadi acuan bahwa aplikasi kasir <i>Android</i> bisa meningkatkan efisiensi sistem penjualan.
8	Rancang Bangun Sistem Informasi Manajemen Stok Barang pada Toko Plastik Bimiu dengan Metode <i>First In-First Out (FIFO)</i>	(Sadiyah et al., 2024)	Sistem stok meningkatkan akurasi pencatatan persediaan dan mempermudah pelacakan produk, sekaligus meningkatkan efisiensi operasional gudang.
9	Implementasi Aplikasi Manajemen Stok dengan Metode <i>FIFO</i> untuk	(Darip et al., 2024)	Membantu UMKM dalam mengendalikan persediaan barang masuk dan keluar, agar stok tidak terjadi

Optimalisasi <i>Inventori</i> : Studi Kasus Aziz Helmet <i>Store</i>		kelebihan ataupun kekurangan. Relevan untuk modul stok pada aplikasi kasir anda.
10	Aplikasi Kasir Berbasis <i>Android</i> pada Warung Soto Sehat Mak Jum (Hidayat, 2022)	Aplikasi ini mempermudah pengguna menghitung transaksi, menyimpan data laporan penjualan, dan menghasilkan nota lebih cepat serta efisien dibanding metode manual.

Sumber : data diolah oleh penulis (2026)

2.13 Kerangka Berfikir Penelitian

Kerangka berfikir penelitian merupakan alur pemikiran yang menjelaskan hubungan logis antara masalah yang terjadi di lapangan dengan solusi yang ditawarkan melalui penelitian. Pada penelitian ini, kerangka berfikir disusun berdasarkan kondisi nyata yang ditemukan pada Apan Cell, di mana proses pencatatan transaksi penjualan dan pengelolaan persediaan barang masih dilakukan secara manual. Dari kondisi tersebut kemudian dianalisis dampaknya terhadap kegiatan operasional toko, sehingga ditetapkan solusi berupa perancangan aplikasi kasir berbasis *Android*. Alur kerangka berfikir penelitian ini dapat dilihat pada Gambar 2.1 berikut.



Sumber : data diolah oleh penulis (2026)

Gambar 2.1 Kerangka Berfikir Penelitian

Pada Gambar 2.1 di atas, kerangka berfikir penelitian ini diawali dengan proses identifikasi masalah yang dilakukan melalui observasi dan wawancara di Apan Cell. Dari hasil pengamatan tersebut dapat diketahui bahwa proses pencatatan transaksi penjualan dan persediaan barang masih dilakukan secara manual menggunakan buku. Kondisi ini menyebabkan data transaksi tidak terdokumentasi dengan baik, berpotensi terjadi kesalahan pencatatan, serta menyulitkan pemilik usaha dalam memantau persediaan barang.

Permasalahan tersebut menimbulkan beberapa dampak terhadap kegiatan operasional toko, seperti ketidaksesuaian antara data persediaan dengan kondisi barang yang sebenarnya, keterlambatan pelayanan kepada pelanggan, serta kesulitan dalam menyusun laporan penjualan secara cepat dan tepat. Hal ini menunjukkan bahwa sistem pencatatan manual sudah kurang efektif digunakan dalam pengelolaan usaha yang dimiliki berbagai jenis transaksi.

Berdasarkan permasalahan dan dampak yang ditimbulkan, maka diperlukan suatu solusi berupa pengembangan aplikasi berbasis *Android*. Aplikasi ini dirancang untuk membantu pencatatan transaksi penjualan, pengelolaan persediaan barang, serta penyusunan laporan penjualan secara otomatis dan terstruktur. Dengan adanya aplikasi ini, diharapkan pengelolaan usaha di Apan Cell menjadi lebih efisien, data transaksi tercatat dengan baik, serta proses pelayanan kepada pelanggan dapat berjalan lebih optimal.

Berdasarkan permasalahan dan dampak yang ada, maka dikembangkan solusi berupa aplikasi penjualan berbasis *Android* yang dirancang untuk membantu proses pencatatan transaksi penjualan, pengelolaan persediaan barang, serta penyusunan laporan penjualan secara otomatis dan terstruktur. Hasil dari penerapan aplikasi ini diharapkan mampu mengatasi permasalahan sebelumnya, yaitu mencatat seluruh transaksi secara sistematis dalam database, tersedianya informasi stok barang secara *real-time* dan akurat, serta disusunnya laporan penjualan dengan cepat dan tepat. Dengan demikian, pengelolaan usaha di Apan Cell menjadi lebih efisien, meminimalkan kesalahan pencatatan, serta meningkatkan kualitas pelayanan kepada pelanggan.

BAB III

METODE PENELITIAN

3.1 Objek dan Jadwal Penelitian

3.1.1 Objek

Objek penelitian ini adalah Apan Cell yang berlokasi di Sido Makmur, Kecamatan Air Kumbang, Kabupaten Banyuasin, Provinsi Sumatera Selatan. Toko ini bergerak di bidang penjualan berbagai kebutuhan telekomunikasi seperti *handphone*, kartu perdana, aksesoris, serta layanan isi ulang pulsa dan paket data. Dalam kegiatan sehari-harinya, Apan Cell juga menyediakan produk lain seperti rokok, minuman dan es krim. Apan Cell masih menggunakan sistem manual dalam mencatat transaksi penjualan maupun pengelolaan persediaan barang. Hal ini sering menimbulkan permasalahan seperti risiko kesalahan penulisan data transaksi, lambatnya pelayanan kepada konsumen, serta proses pengecekan persediaan yang membutuhkan waktu cukup lama. Oleh sebab itu, penelitian ini dilakukan untuk merancang dan membangun aplikasi kasir berbasis *Android* yang dapat membantu pemilik toko dalam melakukan pencatatan transaksi dan pengelolaan persediaan secara lebih cepat, efisien, dan akurat.

Pemilihan Apan Cell sebagai objek penelitian didasarkan pada relevansinya terhadap topik penelitian ini, yaitu perancangan sistem aplikasi kasir berbasis *Android* untuk mendukung proses bisnis. Selain itu, pihak pemilik toko juga memberikan izin dan dukungan penuh dalam proses penelitian, sehingga peneliti dapat melakukan observasi langsung terhadap kegiatan operasional dan mendapatkan data yang akurat untuk proses analisis kebutuhan sistem.

3.1.2 Jadwal Penelitian

Penelitian ini direncanakan berlangsung selama 6 bulan dengan rincian kegiatan sebagai berikut:

Tabel 3.1 Jadwal Penelitian

No	Kegiatan	Bulan Ke 1	Bulan Ke 2	Bulan Ke 3	Bulan Ke 4	Bulan Ke 5	Bulan Ke 6
1	Studi Pustaka dan Analisis Kebutuhan	✓	✓				
2	Perancangan Sistem (UML dan <i>Database</i>)		✓	✓			
3	Pengembangan Aplikasi			✓	✓		
4	Pengujian dan Evaluasi Sistem				✓	✓	
5	Analisis Hasil dan Penyusunan Laporan					✓	✓

Sumber: Penulis, 2026

Penelitian ini direncanakan berlangsung selama 6 bulan dengan rincian kegiatan yang disusun secara bertahap dan sistematis. Pembagian waktu penelitian dilakukan untuk memastikan setiap tahapan, mulai dari studi pustaka, analisis kebutuhan, perancangan sistem, pengembangan aplikasi, hingga pengujian dan penyusunan laporan, dapat terlaksana secara terarah dan sesuai dengan target yang telah ditetapkan.

3.2 Metode Penelitian

Metode penelitian merupakan serangkaian kegiatan dalam mencari kebenaran suatu studi penelitian, yang diawali dengan suatu pemikiran yang membentuk rumusan masalah sehingga menimbulkan hipotesis awal, dengan dibantu dan persepsi penelitian terdahulu, sehingga penelitian bisa diolah dan dianalisis yang akhirnya membentuk suatu kesimpulan (Sahir, 2021: 1). Sejalan dengan tujuan penelitian ini yang berfokus pada perancangan dan pengembangan aplikasi kasir berbasis *Android* untuk pengelolaan penjualan dan persediaan pada Apan Cell, metode yang digunakan adalah metode deskriptif kualitatif karena mampu menggambarkan secara mendalam proses dan hasil dari pengembangan aplikasi kasir tersebut secara kontekstual mulai dari analisis kebutuhan pengguna, perancangan sistem, hingga implementasi dan pengujian aplikasi agar sesuai dengan kebutuhan operasional toko.

Metode deskriptif adalah metode penelitian yang bertujuan untuk menggambarkan karakteristik suatu fenomena atau populasi (Irawan, 2025: 98). Metode penelitian kualitatif adalah metode penelitian yang menekankan pada pengamatan dan pemahaman yang alamiah dan mendalam, disajikan secara deskriptif serta diinterpretasikan secara komprehensif (Waruwu, 2024). Dengan menggabungkan kedua pendekatan ini, penelitian dapat menghasilkan gambaran jelas mengenai kondisi pencatatan penjualan di Apan Cell sekaligus dasar dalam pengembangan aplikasi kasir yang sesuai dengan permasalahan yang ditemukan.

3.3 Sumber Data

Dalam penelitian ini, sumber data merupakan komponen penting yang digunakan untuk memperoleh informasi yang relevan dengan permasalahan penelitian. Sumber data diperoleh dari dua jenis, yaitu data primer dan data sekunder, yang keduanya saling melengkapi dalam mendukung hasil penelitian.

3.3.1 Data Primer

Data primer adalah data yang diperoleh atau dikumpulkan langsung oleh peneliti dari orang-orang yang bersangkutan (Sinar et al., 2024). Dalam penelitian ini, data primer diperoleh dari hasil observasi langsung ke toko Apan Cell, wawancara langsung ke pemilik toko Apan Cell, Bapak Andika Putra Abdi Negara, pada tanggal 21 November 2025, serta melakukan dokumentasi terkait pencatatan data transaksi dan persediaan mereka yang masih manual. Data ini digunakan untuk memahami kebutuhan pengguna, alur pencatatan transaksi, serta kendala yang terjadi dalam sistem manual sebelum diterapkannya aplikasi kasir berbasis *Android*. Melalui data primer ini, peneliti dapat merancang aplikasi yang sesuai dengan kondisi toko dan kebutuhan pengguna sebenarnya.

3.3.2 Data Sekunder

Data sekunder adalah data yang telah dikumpulkan oleh lembaga pengumpul data dan dipublikasikan kepada masyarakat pengguna data (Asriani et al., 2022: 39). Data ini biasanya berasal dari berbagai referensi seperti e-book,

jurnal ilmiah, artikel penelitian, laporan sebelumnya, maupun sumber internet yang relevan dengan topik penelitian.

3.4 Teknik Pengumpulan Data

Pengumpulan data merupakan langkah penting dalam penelitian ini karena kualitas data yang diperoleh akan sangat mempengaruhi ketepatan hasil perancangan aplikasi yang dikembangkan. Oleh karena itu, digunakan beberapa teknik pengumpulan data untuk memperoleh informasi yang akurat dan relevan terhadap objek penelitian, yaitu observasi, wawancara, dokumentasi dan studi literatur.

3.4.1 Observasi

Observasi dilakukan dengan cara mengamati langsung proses transaksi penjualan dan pengelolaan persediaan barang di Apan Cell. Melalui observasi ini, peneliti dapat mengetahui secara nyata bagaimana sistem pencatatan dilakukan secara manual, alur proses transaksi, serta kendala yang sering muncul seperti keterlambatan pencatatan, kesalahan pencatatan persediaan barang dan kesulitan dalam membuat laporan penjualan. Hasil observasi ini menjadi dasar penting dalam menentukan kebutuhan sistem pada aplikasi kasir berbasis *Android* yang akan dirancang.

3.4.2 Wawancara

Wawancara dilakukan dengan pemilik toko Apan Cell, Bapak Andika Putra Abdi Negara, pada bulan ke 2 penelitian. Wawancara ini bertujuan untuk menggali informasi lebih mendalam terkait prosedur pencatatan transaksi dan pengelolaan persediaan barang. Data yang diperoleh dari wawancara digunakan untuk memahami kebutuhan pengguna secara langsung, agar sistem yang dikembangkan benar-benar sesuai dengan permasalahan yang ada di Apan Cell.

3.4.3 Dokumentasi

Teknik dokumentasi dilakukan dengan mengumpulkan berbagai dokumen dan catatan yang berkaitan dengan penjualan dan persediaan barang, seperti nota

transaksi, laporan penjualan, dan daftar persediaan barang. Dokumen-dokumen ini digunakan untuk menganalisis format data yang diperlukan dalam aplikasi, serta menjadi acuan dalam merancang *database* dan laporan penjualan pada sistem yang dikembangkan.

3.4.4 Studi Literatur

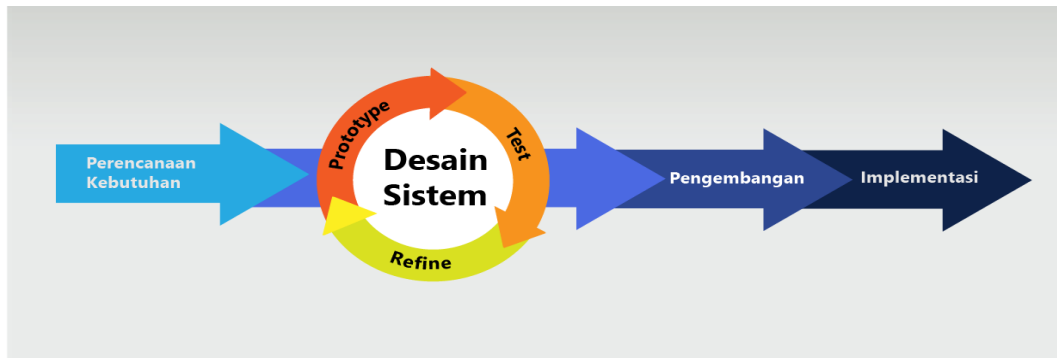
Studi literatur dilakukan dengan menelaah berbagai sumber akademik maupun non-akademik, seperti *e-book*, jurnal ilmiah, dan penelitian sebelumnya yang berkaitan dengan pengembangan aplikasi kasir berbasis *Android*, teknologi informasi, serta sistem manajemen penjualan dan persediaan. Hal ini bertujuan untuk memperoleh landasan teori yang kuat mengenai konsep sistem informasi penjualan, pengelolaan persediaan barang secara digital, serta pemanfaatan teknologi *mobile* dalam meningkatkan efisiensi operasional usaha kecil dan menengah (UKM).

3.5 Metode Pengembangan Sistem

Metode pengembangan sistem merupakan langkah-langkah sistematis yang digunakan untuk merancang dan membangun aplikasi agar sesuai dengan kebutuhan pengguna. Dalam penelitian ini, peneliti menggunakan metode *Rapid Application Development (RAD)*. Metode ini dipilih karena sesuai dengan kebutuhan Apan Cell yang memerlukan sistem kasir yang dapat dikembangkan dengan cepat, fleksibel, serta mampu menyesuaikan fitur aplikasi berdasarkan kebutuhan pengguna di lapangan. Hal ini penting mengingat selama ini proses transaksi dan persediaan barang masih dilakukan secara manual, sehingga dibutuhkan solusi berupa aplikasi kasir yang dapat segera diterapkan.

Rapid Application Development (RAD) adalah metode yang berfokus kepada pengembangan sistem secara cepat, melalui pengulangan dan *feedback* berulang-ulang. Ciri utama dari metode *Rapid Application Development (RAD)* adalah proses perangkat lunak yang lebih menekankan kepada daur pengembangan hidup yang singkat, serta versi adaptasi cepat dari *waterfall* dan *prototype* (Sintawati, 2022).

Berikut ini adalah tahapan-tahapan pada metode ini, yang bisa dilihat pada Gambar 3.1 di bawah ini :



Sumber : Agus Hermanto (2025)

Gambar 3.1 Tahapan RAD

Berikut ini adalah beberapa penjelasan dari tahapan metode *Rapid Application Development (RAD)* pada Gambar 3.1 di atas :

1. Perencanaan Kebutuhan

Ini adalah tahap awal dalam pengembangan sistem. Pada tahap ini, masalah diidentifikasi dan data dikumpulkan dari *stakeholder* pengguna atau pengguna untuk mengidentifikasi tujuan akhir atau tujuan sistem serta kebutuhan informasi yang diinginkan. Keterlibatan kedua belah pihak sangat penting untuk menentukan kebutuhan sistem.

2. Perancangan Sistem

Untuk mencapai tujuan pada tahap desain sistem, keaktifan pengguna yang terlibat sangat penting karena pada tahap ini proses desain dan perbaikan dilakukan berulang kali jika terdapat ketidaksesuaian desain dengan kebutuhan pengguna yang diidentifikasi pada tahap sebelumnya. Sebagai hasil dari langkah ini, spesifikasi *software* dibuat, yang mencakup struktur data, organisasi sistem secara keseluruhan, dan elemen lainnya.

3. Pengembangan

Pada titik ini, desain sistem yang telah dibuat dan diputuskan, diubah menjadi aplikasi versi beta sebelum mencapai versi akhir. Pada tahap ini, peneliti harus terus melakukan pekerjaan pengembangan dan berintegrasi dengan komponen lainnya sambil mempertimbangkan

umpan balik pengguna. Jika proses berjalan lancar, maka dapat berlanjut ke tahapan berikutnya. Jika aplikasi yang dikembangkan belum memenuhi kebutuhan, peneliti akan kembali ke tahap desain sistem.

4. Implementasi

Pada titik ini, peneliti menerapkan desain sistem yang telah disetujui pada tahap sebelumnya. Sebelum sistem diterapkan, proses pengujian dilakukan terhadap program untuk mengidentifikasi kesalahan. Pada tahap ini, peneliti biasanya memberikan tanggapan terhadap sistem yang telah dibuat dan mendapat persetujuan.

3.6 Alat Bantu Perancangan

Dalam proses penelitian dan pengembangan aplikasi kasir berbasis *Android* ini, peneliti menggunakan beberapa alat bantu untuk mendukung kegiatan perancangan dan implementasi sistem. Alat bantu ini digunakan agar proses pembuatan aplikasi menjadi lebih mudah, cepat, dan hasilnya sesuai dengan kebutuhan pengguna. Secara umum, alat bantu dibagi menjadi tiga bagian utama, yaitu alat bantu utama, alat bantu desain, serta alat bantu *debugging* dan pengembangan.

3.6.1 Alat Bantu Utama

Alat bantu utama merupakan perangkat atau *software* yang digunakan untuk menulis, menjalankan, serta mengelola kode program. Pada penelitian ini, peneliti menggunakan *Visual Studio Code* sebagai editor utama karena ringan, mudah digunakan, serta mendukung berbagai ekstensi untuk pengembangan aplikasi.

Bahasa pemrograman yang digunakan adalah *Dart*, dengan *framework Flutter*. *Flutter* dipilih karena memiliki keunggulan dalam membuat aplikasi lintas *platform*, memiliki tampilan yang menarik, serta proses pengembangannya cepat berkat fitur *hot reload*. Untuk pengelolaan basis data, peneliti menggunakan *Firebase* untuk mendukung aplikasi kasir dalam menyimpan dan mengelola basis data.

3.6.2 Alat Bantu Pemodelan Visual

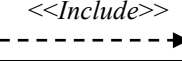
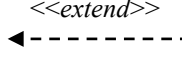
Pemodelan visual digunakan untuk menggambarkan proses bisnis, alur sistem dan hubungan antara pengguna dengan aplikasi yang dikembangkan. Pada penelitian ini, peneliti menggunakan *Unified Modeling Language (UML)* sebagai alat bantu pemodelan visual.

Berikut adalah beberapa jenis diagram *Unified Modeling Language (UML)* yang digunakan dalam penelitian ini :

a. *Use Case Diagram*

Use Case Diagram digunakan untuk menggambarkan interaksi antara pengguna (aktor) dengan sistem, serta fungsionalitas utama yang disediakan oleh sistem. Dalam konteks aplikasi kasir berbasis *Android*, *Use Case Diagram* memvisualisasikan peran pengguna seperti kasir dan admin dalam melakukan proses *login*, transaksi penjualan, pengelolaan data barang dan pembuatan laporan penjualan. Simbol-simbol yang digunakan dalam *Use Case Diagram* dapat dilihat pada Tabel 3.1 di bawah ini :

Tabel 3.2 Simbol *Use Case*

No	Gambar	Nama	Keterangan
1		Aktor	Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>Use Case</i> .
2		<i>Use Case</i>	Fungsionalitas atau layanan yang diberikan sistem kepada aktor.
3		<i>Association</i>	Garis yang menghubungkan aktor dengan <i>Use Case</i> yang digunakan.
4		<i>Generalization</i>	Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>Use Case</i> .
5		<i>Include</i>	Relasi yang menunjukkan bahwa suatu <i>Use Case</i> selalu memanggil <i>Use Case</i> lain.
6		<i>Extend</i>	Relasi yang menunjukkan bahwa <i>Use Case</i> bisa memanggil <i>Use Case</i> lain secara opsional.
7		<i>System</i>	Kotak yang menampung semua <i>Use Case</i> dan menunjukkan batas sistem.

Sumber : Riswan (2024)

b. *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan alur aktivitas atau proses kerja sistem yang terjadi dalam aplikasi kasir berbasis *Android*. Diagram ini

menunjukkan langkah-langkah logis dari awal hingga akhir dalam suatu proses, seperti proses *login*, transaksi penjualan, pengelolaan persediaan barang, dan pembuatan laporan. Dengan *Activity Diagram*, peneliti dapat memahami bagaimana sistem beroperasi secara dinamis dan bagaimana interaksi antar aktivitas berlangsung di dalam sistem. Simbol-simbol *Activity Diagram* dapat dilihat pada Tabel 3.2 di bawah ini :

Tabel 3.3 Simbol *Activity Diagram*

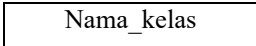
No	Gambar	Nama	Keterangan
1		<i>Initial Node</i> (Titik Awal)	Simbol yang menunjukkan awal dari suatu aktivitas atau proses.
2		<i>Final Node</i> (Titik Akhir)	Menunjukkan akhir dari suatu aktivitas atau proses.
3		<i>Activity/Action</i>	Menunjukkan aktivitas atau tindakan yang dilakukan dengan sistem maupun pengguna.
4		<i>Decision</i> (Keputusan)	Menunjukkan percabangan kondisi dalam proses seperti ya/tidak atau benar/salah.
5		<i>Control Flow</i> (Alur Kendali)	Panah yang menunjukkan urutan eksekusi antar aktivitas.
6		<i>Fork/Join</i> (Pararel)	Menunjukkan aktivitas yang sedang berjalan bersamaan (<i>fork</i>) atau bergabung kembali (<i>join</i>).
7		<i>Swimlane</i>	Area yang menunjukkan siapa pelaku dari suatu aktivitas, misal kasir atau sistem.

Sumber : M. Hernawan (2023)

c. *Class Diagram*

Class Diagram digunakan untuk menggambarkan struktur data dan hubungan antar kelas dalam aplikasi kasir berbasis *Android*. Diagram ini menunjukkan atribut dan fungsi yang dimiliki setiap kelas, seperti kelas pengguna, transaksi, persediaan serta laporan. Dengan *Class Diagram*, struktur sistem dapat dipahami dengan lebih jelas sebelum aplikasi dikembangkan. Simbol-simbol *Class Diagram* ditampilkan pada Tabel 3.3 dibawah ini :

Tabel 3.4 Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Class</i>	Menggambarkan sebuah kelas pada sistem yang terbagi menjadi 3 bagian.

	+atribut +operasi()		Bagian atas adalah nama kelas. Bagian tengah adalah atribut kelas. Bagian bawah adalah <i>methode</i> dari kelas.
2		<i>Association</i>	Hubungan statis antar kelas. Menggambarkan kelas yang memiliki atribut berupa kelas lain atau kelas yang harus mengetahui eksistensi kelas lain.
3		<i>Agregation</i>	Hubungan yang menyatakan bahwa suatu kelas menjadi atribut bagi kelas lain.
4		<i>Composition</i>	Bentuk khusus dari <i>agregation</i> dimana kelas yang menjadi bagian diciptakan setelah kelas <i>whole</i> dibuat.
5		<i>Generalization</i>	Relasi antar kelas dengan makna <i>generalisasi-spesialisasi</i> (umum-khusus).
6		<i>Directed Association</i>	Asosiasi dengan makna kelas yang satu digunakan oleh kelas yang lain.

Sumber : Syarif Hidayatullah (2023)

3.6.3 Alat Bantu Desain

Alat bantu desain merupakan perangkat atau aplikasi yang digunakan untuk menggambarkan rancangan tampilan antarmuka sebuah sistem sebelum diimplementasikan ke dalam bentuk aplikasi. Dengan alat bantu desain, pengembang dapat menyusun konsep tampilan secara lebih jelas, mulai dari penempatan menu, warna, ikon, hingga alur interaksi antara pengguna dengan sistem. Hal ini bertujuan untuk memastikan bahwa aplikasi yang dibuat memiliki tampilan yang nyaman, mudah digunakan, dan sesuai dengan kebutuhan pengguna. Dalam penelitian ini, peneliti menggunakan *Figma* sebagai alat bantu desain antarmuka aplikasi kasir berbasis *Android*. *Figma* dipilih karena memiliki fitur yang mendukung pembuatan *wireframe*, desain *User Interface (UI)*, dan *prototype* secara interaktif. *Figma* juga berbasis *web* sehingga mudah diakses di berbagai perangkat tanpa instalasi kompleks serta mendukung kolaborasi jika diperlukan. Dengan penggunaan *Figma*, desain antarmuka dapat dibuat lebih terstruktur dan

estetis sehingga hasil implementasi nantinya sesuai dengan rancangan yang telah dibuat dan mampu meningkatkan kenyamanan penggunaan aplikasi bagi kasir di Apan Cell.

3.6.4 Alat Bantu *Debugging* dan Pengembangan

Alat bantu *debugging* dan pengembangan digunakan untuk menguji, menemukan, dan memperbaiki kesalahan yang terjadi selama proses pembuatan aplikasi. Dalam penelitian ini, peneliti menggunakan perangkat keras seperti *handphone* untuk menguji aplikasi yang dibuat. Cara ini dipilih agar hasil pengujian lebih akurat dan sesuai dengan kondisi nyata pengguna.

Peneliti menggunakan *Visual Studio Code* dengan bantuan ekstensi *Flutter DevTools* untuk memantau performa aplikasi, melihat *error* yang muncul, serta memastikan aplikasi berjalan lancar. Selain itu, *console log* dari bahasa *Dart* juga digunakan untuk melacak proses yang sedang dijalankan dan mempermudah menemukan kesalahan logika.

Dengan menggunakan kombinasi alat bantu tersebut, proses *debugging* dan pengembangan dapat berjalan dengan lebih efisien dan cepat, sehingga aplikasi kasir yang dibuat dapat berfungsi dengan baik tanpa banyak kesalahan.

3.7 Pengujian Sistem

Pengujian sistem dalam penelitian ini menggunakan *White Box Testing*, yaitu pengujian yang dilakukan dengan memeriksa alur logika pada kode program. Pengujian ini bertujuan untuk memastikan bahwa setiap proses di dalam aplikasi kasir berbasis *Android* berjalan dengan benar, terutama pada transaksi penjualan, pembaruan persediaan barang, dan pembuatan laporan penjualan. Pengujian ini dilakukan pada fungsi-fungsi utama, seperti menambah dan menghapus barang, menyimpan transaksi, menghitung total pembayaran, serta memperbarui persediaan barang. Selain itu, integrasi antar bagian aplikasi, misal modul transaksi, persediaan, dan laporan juga diuji supaya data bisa berjalan dengan lancar dan sesuai yang diharapkan.

Alur *White Box* Testing pada aplikasi kasir ini adalah sebagai berikut :

1. Analisis kode program
Pada tahap ini dilakukan identifikasi terhadap alur logika program, khususnya pada proses transaksi penjualan, pembaruan persediaan, dan pembuatan laporan.
2. Perancangan *test case*
Berdasarkan hasil analisis, disusun *test case* untuk menguji setiap jalur logika, kondisi, dan perulangan pada fungsi-fungsi aplikasi.
3. Eksekusi pengujian
Test case yang telah disusun, dijalankan pada aplikasi kasir untuk melihat apakah hasilnya sudah sesuai dengan yang diharapkan.
4. Analisis hasil
Hasil keluaran sistem dibandingkan dengan hasil yang seharusnya terjadi. Jika ditemukan kesalahan logika atau ketidaksesuaian data, maka dicatat sebagai *error*.
5. Perbaikan dan pengujian ulang
Kode program yang bermasalah diperbaiki, kemudian dilakukan pengujian ulang hingga seluruh fungsi berjalan dengan benar.
Alat bantu yang digunakan antara lain *Visual Studio Code*, *Flutter DevTools*, dan *console log* bahasa *Dart* untuk mengecek kode dan menemukan kesalahan. Pengujian ini juga dilakukan di perangkat *Android* agar aplikasi bisa dicoba seperti kondisi pengguna sebenarnya. Dengan *White Box Testing*, peneliti dapat memastikan aplikasi kasir bekerja sesuai desain, bebas dari kesalahan, dan semua fitur dapat berjalan dengan baik serta saling terhubung.

BAB IV

ANALISA DAN PERANCANGAN

4.1 Analisa Sistem

4.1.1 Analisa dan Evaluasi Sistem Berjalan

a. Analisa Sistem Berjalan

Analisa sistem berjalan dilakukan untuk mengetahui kondisi nyata proses pengelolaan penjualan dan persediaan barang yang saat ini diterapkan di Apan Cell. Berdasarkan hasil observasi dan wawancara yang telah dilakukan, diketahui bahwa sistem yang digunakan masih bersifat manual, yaitu dengan mencatat seluruh transaksi penjualan dan data persediaan barang ke dalam buku. Proses transaksi penjualan dimulai ketika pelanggan memilih barang yang akan dibeli, kemudian kasir mencatat nama barang, jumlah, dan harga ke dalam buku transaksi. Setelah itu, kasir menghitung total pembayaran secara manual dan menyerahkan barang kepada pelanggan.

Proses ini dilakukan secara berulang setiap hari tanpa adanya sistem terkomputerisasi yang membantu pengolahan data. Pengelolaan persediaan barang juga dilakukan secara manual dengan mencatat barang masuk dan barang keluar ke dalam buku stok. Pengecekan jumlah barang dilakukan dengan cara menghitung langsung stok fisik di toko. Proses ini membutuhkan waktu yang cukup lama, terutama ketika jumlah barang yang tersedia cukup banyak dan jenis barang yang beragam.

Kondisi tersebut menimbulkan beberapa permasalahan dalam kegiatan operasional toko. Salah satu permasalahan utama adalah tingginya risiko kesalahan pencatatan data transaksi akibat faktor *human error*. Kesalahan tersebut dapat berupa salah tulis, data yang terlewat, maupun ketidaksesuaian antara jumlah barang yang tercatat dengan kondisi sebenarnya. Selain itu, proses pelayanan kepada pelanggan menjadi kurang optimal, terutama saat kondisi toko ramai. Proses pencatatan manual dan perhitungan yang dilakukan secara langsung menyebabkan pelayanan menjadi lebih lambat. Hal ini dapat berdampak pada kepuasan pelanggan.

Dari sisi kelemahan sistem, dapat diketahui bahwa sistem yang berjalan saat ini belum memanfaatkan perkembangan teknologi informasi, khususnya aplikasi berbasis *mobile* yang dapat membantu proses pencatatan dan pengolahan data secara otomatis. Sistem yang ada juga belum mampu menyediakan informasi stok barang secara *real-time* serta belum memiliki fitur pembuatan laporan secara cepat dan akurat. Berdasarkan permasalahan dan kelemahan tersebut, diperlukan suatu sistem yang mampu memberikan solusi berupa aplikasi kasir berbasis *Android*. Sistem yang dikembangkan diharapkan mampu membantu proses pencatatan transaksi secara otomatis, mengelola persediaan barang secara *real-time*, serta menghasilkan laporan penjualan dengan cepat dan terstruktur.

Dalam penelitian ini, analisa sistem berjalan dikaitkan dengan metode *Rapid Application Development (RAD)* yang menekankan pada pengembangan sistem secara cepat dan berorientasi pada kebutuhan pengguna. Dengan pendekatan ini, proses analisa tidak hanya berfokus pada permasalahan yang ada, tetapi juga pada kebutuhan pengguna sebagai dasar dalam pengembangan sistem yang akan dibangun.

b. Evaluasi Sistem Bejalan

Evaluasi sistem berjalan dilakukan untuk menilai efektivitas sistem pengelolaan penjualan dan persediaan barang yang saat ini diterapkan di Apan Cell. Evaluasi ini dilakukan berdasarkan hasil analisa sistem berjalan yang menunjukkan bahwa sistem yang digunakan masih bersifat manual dan belum memanfaatkan teknologi secara optimal. Berdasarkan kondisi yang ada, sistem manual yang digunakan saat ini memiliki keterbatasan dalam hal kecepatan, ketepatan, dan efisiensi pengolahan data.

Pencatatan yang dilakukan secara manual berisiko menimbulkan kesalahan serta membutuhkan waktu yang lebih lama dalam proses transaksi maupun pembuatan laporan. Selain itu, sistem yang berjalan belum mampu menyediakan informasi yang terintegrasi antara data transaksi dan persediaan barang. Hal ini menyebabkan kesulitan dalam melakukan pemantauan stok barang secara akurat. Proses pengecekan stok juga harus dilakukan secara langsung, sehingga kurang efisien dan memerlukan waktu yang cukup lama.

Berdasarkan hasil evaluasi tersebut, solusi yang ditawarkan dalam penelitian ini adalah pengembangan aplikasi kasir berbasis *Android* yang dapat membantu proses pencatatan transaksi dan pengelolaan persediaan barang secara otomatis. Aplikasi ini dirancang untuk meningkatkan efisiensi kerja, mengurangi kesalahan pencatatan, serta mempercepat proses pelayanan kepada pelanggan.

Sistem yang diusulkan memiliki beberapa keunggulan, seperti pencatatan transaksi secara otomatis, perhitungan total pembayaran yang lebih akurat, pembaruan stok barang secara *real-time*, serta penyajian laporan penjualan secara cepat dan terstruktur. Dengan adanya sistem ini, diharapkan pengelolaan operasional di Apan Cell menjadi lebih efektif dan efisien.

4.1.2 Analisa Kebutuhan

a. Kebutuhan Pengguna (*User*)

Kebutuhan pengguna dalam sistem ini mengacu pada pihak-pihak yang akan berinteraksi langsung dengan aplikasi kasir berbasis *Android*. Berdasarkan hasil analisis, pengguna sistem terbagi menjadi dua kategori utama dengan rincian sebagai berikut:

1. Admin

Admin adalah pengguna dengan otoritas tertinggi yang bertanggung jawab penuh atas manajemen operasional toko di dalam sistem. Kebutuhan utama admin mencakup pengelolaan data inventaris secara menyeluruh seperti menambah atau mengubah stok dan harga barang, mengatur hak akses akun kasir, serta memantau laporan transaksi harian guna mengambil keputusan bisnis yang tepat berdasarkan data yang tersimpan di *database*.

2. Kasir

Kasir adalah pengguna operasional yang fokus utamanya adalah menjalankan aktivitas transaksi penjualan kepada pelanggan. Kebutuhan kasir dititikberatkan pada kemudahan penggunaan antarmuka aplikasi untuk mencatat barang yang terjual dan memproses total pembayaran secara cepat, sehingga kegiatan operasional di kasir

dapat berjalan lancar tanpa memerlukan pemahaman teknis yang mendalam mengenai pengelolaan *database* atau konfigurasi sistem.

b. Kebutuhan Data

Dalam pengembangan aplikasi kasir berbasis *Android*, sistem membutuhkan beberapa jenis data untuk mendukung proses operasional. Data yang dibutuhkan meliputi:

1. Data Barang

Digunakan untuk menyimpan informasi barang seperti nama barang, harga jual, harga beli, stok, dan kategori. Data ini berfungsi sebagai dasar dalam proses transaksi penjualan dan pengelolaan persediaan.

2. Data Transaksi

Digunakan untuk mencatat setiap transaksi penjualan yang terjadi. Data ini meliputi tanggal transaksi, daftar barang yang dibeli, jumlah barang, serta total pembayaran.

3. Data Pengguna

Digunakan untuk menyimpan informasi login pengguna seperti *username*, *password*, dan hak akses. Data ini berfungsi untuk menjaga keamanan sistem.

4. Data Laporan

Digunakan untuk menyusun laporan penjualan harian, mingguan, dan bulanan yang dapat digunakan oleh pemilik toko dalam pengambilan keputusan.

Sumber data diperoleh dari hasil observasi, wawancara, serta dokumentasi yang dilakukan di Apan Cell. Tujuan pengelolaan data ini adalah untuk mempermudah proses pencatatan, meningkatkan akurasi data, serta mendukung pengelolaan usaha secara lebih terstruktur.

c. Kebutuhan Fungsional

Kebutuhan fungsional merupakan fungsi utama yang harus tersedia dalam sistem agar dapat berjalan sesuai dengan tujuan. Adapun fungsi yang harus ada dalam aplikasi ini meliputi:

1. Sistem harus dapat melakukan *registrasi* dan *login* pengguna
2. Sistem harus dapat menampilkan data barang
3. Sistem harus dapat melakukan transaksi penjualan
4. Sistem harus dapat menghitung total pembayaran secara otomatis
5. Sistem harus dapat mengelola data barang (tambah, ubah, hapus)
6. Sistem harus dapat memperbarui stok barang secara otomatis
7. Sistem harus dapat menampilkan laporan penjualan
8. Sistem harus dapat melakukan *logout* untuk keamanan pengguna

d. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional berkaitan dengan kualitas sistem yang harus dipenuhi agar sistem dapat berjalan dengan baik. Beberapa kebutuhan non-fungsional dalam sistem ini antara lain:

1. Keamanan (*Security*)

Sistem harus mampu melindungi data pengguna agar tidak dapat diakses oleh pihak yang tidak berwenang.

2. Performa (*Performance*)

Aplikasi harus dapat berjalan dengan cepat dan responsif tanpa mengalami gangguan.

3. Kemudahan Penggunaan (*Usability*)

Antarmuka aplikasi harus sederhana dan mudah digunakan oleh pengguna.

4. Keandalan (*Reliability*)

Sistem harus dapat berjalan secara stabil tanpa mengalami *error*.

5. Kompatibilitas (*Compatibility*)

Aplikasi harus dapat berjalan pada berbagai perangkat *Android*.

e. Analisa Perangkat Keras dan Lunak (*Hardware/Software*)

Agar aplikasi dapat dikembangkan dan dijalankan dengan stabil, diperlukan standar minimum perangkat keras dan perangkat lunak sebagai berikut:

1. Perangkat Keras (*Hardware*)

Untuk pengguna:

- a. *Smartphone* berbasis *Android*
- b. *RAM* minimal 2 GB
- c. *Storage*: 32 GB

Untuk pengembang:

- a. *Laptop/PC* dengan prosesor minimal *Intel Core i3*
- b. *RAM* minimal 4 GB
- c. *Storage*: 128 GB

Printer *Bluetooth* minimal: *Thermal Receipt Printer* 58 mm (*USB* + *Bluetooth*)

2. Perangkat Lunak (*Software*)

1. Sistem Operasi minimal *Windows* 10 64-bit
2. *Code Editor* minimal *Visual Studio Code* versi 1.120.0
3. Bahasa pemrograman minimal *Dart SDK* versi 3.0.0
4. *Framework* minimal *Flutter* versi 3.22.0
5. *Firebase* minimal:
 - a. *Firebase_core* versi 2.25.0
 - b. *Cloud_firestore* versi 4.13.0
 - c. *Firebase_auth* versi 4.15.0
6. *Android SDK* minimal versi 33.0.0
7. Sistem Operasi *Android* minimal versi 8.0 (*Oreo*)
8. Alat bantu desain menggunakan *Figma*

Dengan analisa kebutuhan perangkat keras dan perangkat lunak ini, diharapkan sistem dapat dikembangkan dan diimplementasikan secara optimal sesuai dengan tujuan penelitian.

4.2 Perancangan Sistem (*Desain*) yang diusulkan

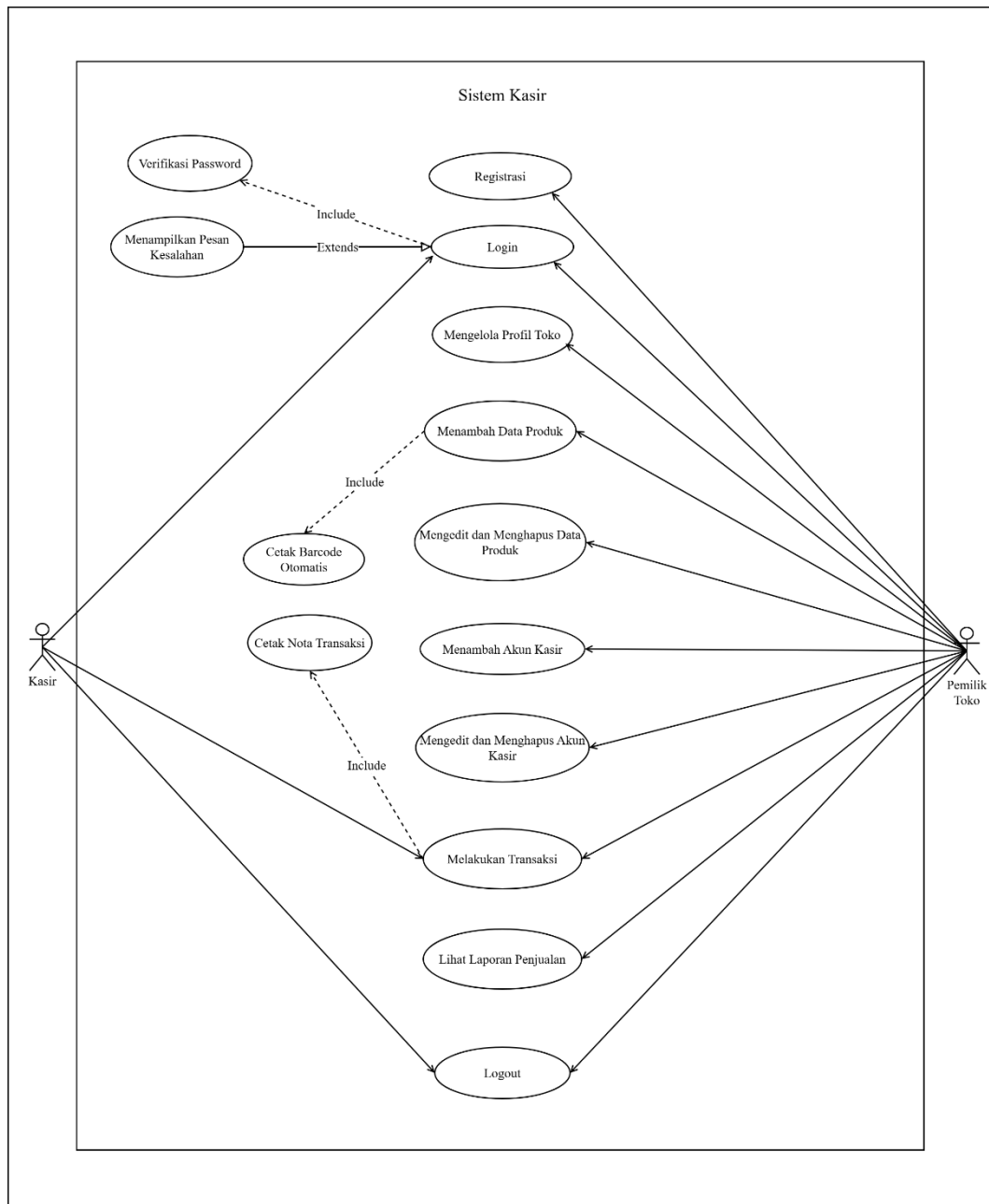
4.2.1 Perancangan Sistem (*Logis*)

Perancangan sistem secara logis merupakan tahap yang menjelaskan bagaimana sistem aplikasi kasir berbasis *Android* untuk pengelolaan penjualan dan

persediaan pada Apan Cell bekerja dari sisi fungsional tanpa memperhatikan implementasi teknis perangkat keras maupun perangkat lunak yang digunakan. Pada tahap ini, fokus utama adalah pada alur proses transaksi penjualan, pengelolaan data barang, serta interaksi antara pengguna dengan sistem, termasuk admin dan kasir, serta fungsi-fungsi utama yang harus dijalankan oleh sistem agar sesuai dengan kebutuhan yang telah dianalisis sebelumnya. Perancangan sistem logis ini digambarkan melalui model-model seperti *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram* yang menunjukkan hubungan antara aktor dan sistem secara konseptual.

a. *Use Case Diagram*

Use Case Diagram digunakan untuk memetakan hubungan antara aktor dengan sistem, mencakup proses registrasi, *login*, pengelolaan profil toko, data produk, akun kasir, hingga transaksi penjualan. Selain fungsionalitas utama, diagram ini juga menggambarkan fitur pendukung seperti verifikasi keamanan, pesan kesalahan, serta pencetakan *barcode* dan nota transaksi. Secara keseluruhan, diagram ini memberikan gambaran umum mengenai interaksi antara pemilik toko dan kasir terhadap sistem yang dikembangkan.



Sumber : penulis 2026

Gambar 4.1 Usecase Diagram

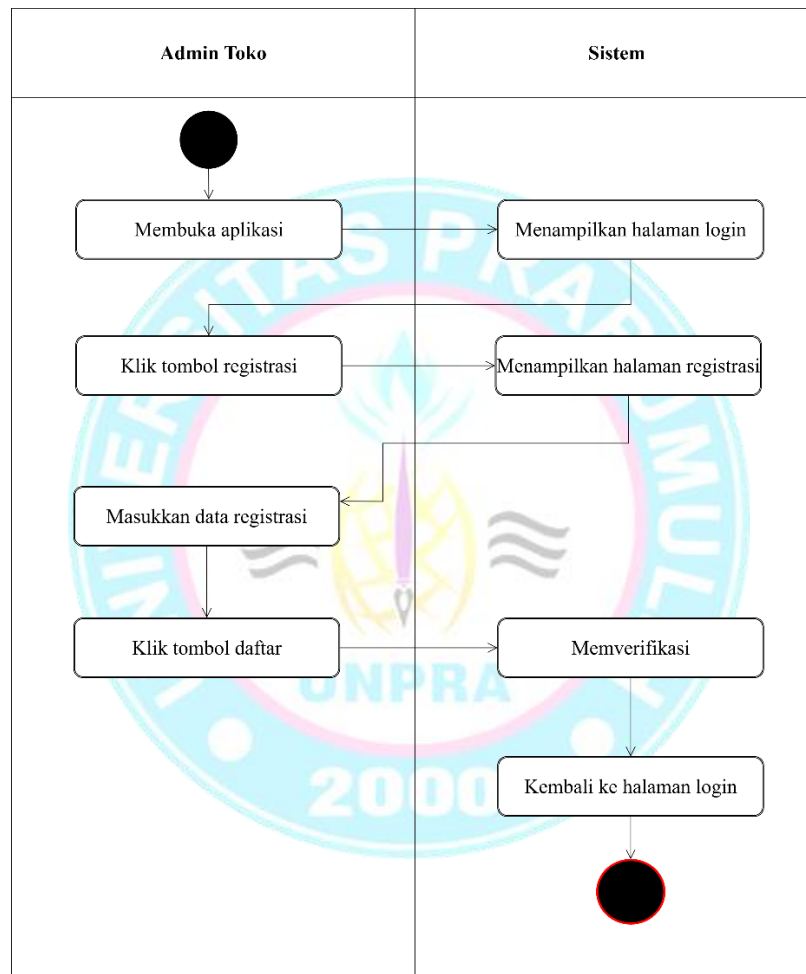
Use Case Diagram menggambarkan interaksi antara aktor dengan sistem aplikasi kasir berbasis *Android* untuk pengelolaan penjualan dan persediaan pada *Apan Cell*. Sistem ini memiliki dua aktor utama yaitu:

1. Pemilik Toko (Admin) bertugas mengelola keseluruhan sistem seperti mengelola profil toko, menambah, mengedit, dan menghapus data produk, mengelola akun kasir, serta melihat laporan penjualan.

2. Kasir bertugas menjalankan operasional transaksi penjualan, seperti melakukan *login*, melakukan transaksi penjualan, mencatat barang yang dibeli oleh pelanggan, mencetak nota transaksi, serta melakukan *logout* dari sistem.

b. Activity Diagram

1. Activity Registrasi Admin



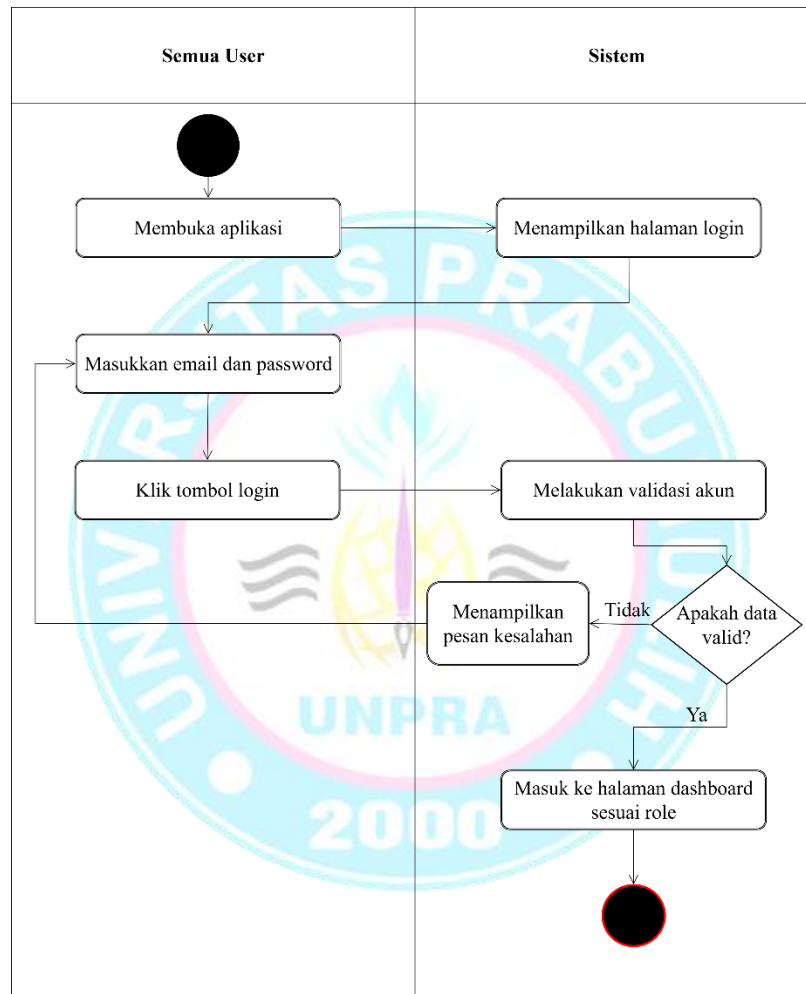
Sumber : penulis, 2026

Gambar 4.2 Registrasi Activity

Diagram *activity* pertama menggambarkan alur proses registrasi yang dapat dilakukan oleh pengguna sistem. Proses dimulai ketika pengguna membuka aplikasi, kemudian sistem menampilkan halaman *login*. Selanjutnya pengguna menekan tombol registrasi dan sistem akan menampilkan halaman registrasi. Pengguna kemudian memasukkan data registrasi, lalu menekan tombol daftar.

Sistem melakukan verifikasi terhadap data yang diinputkan. Apabila data yang dimasukkan tidak valid, maka sistem akan meminta pengguna untuk menginput ulang data registrasi. Namun jika data valid, sistem akan mengarahkan pengguna kembali ke halaman *login*. Proses berakhir ketika halaman *login* berhasil ditampilkan.

2. Activity Login



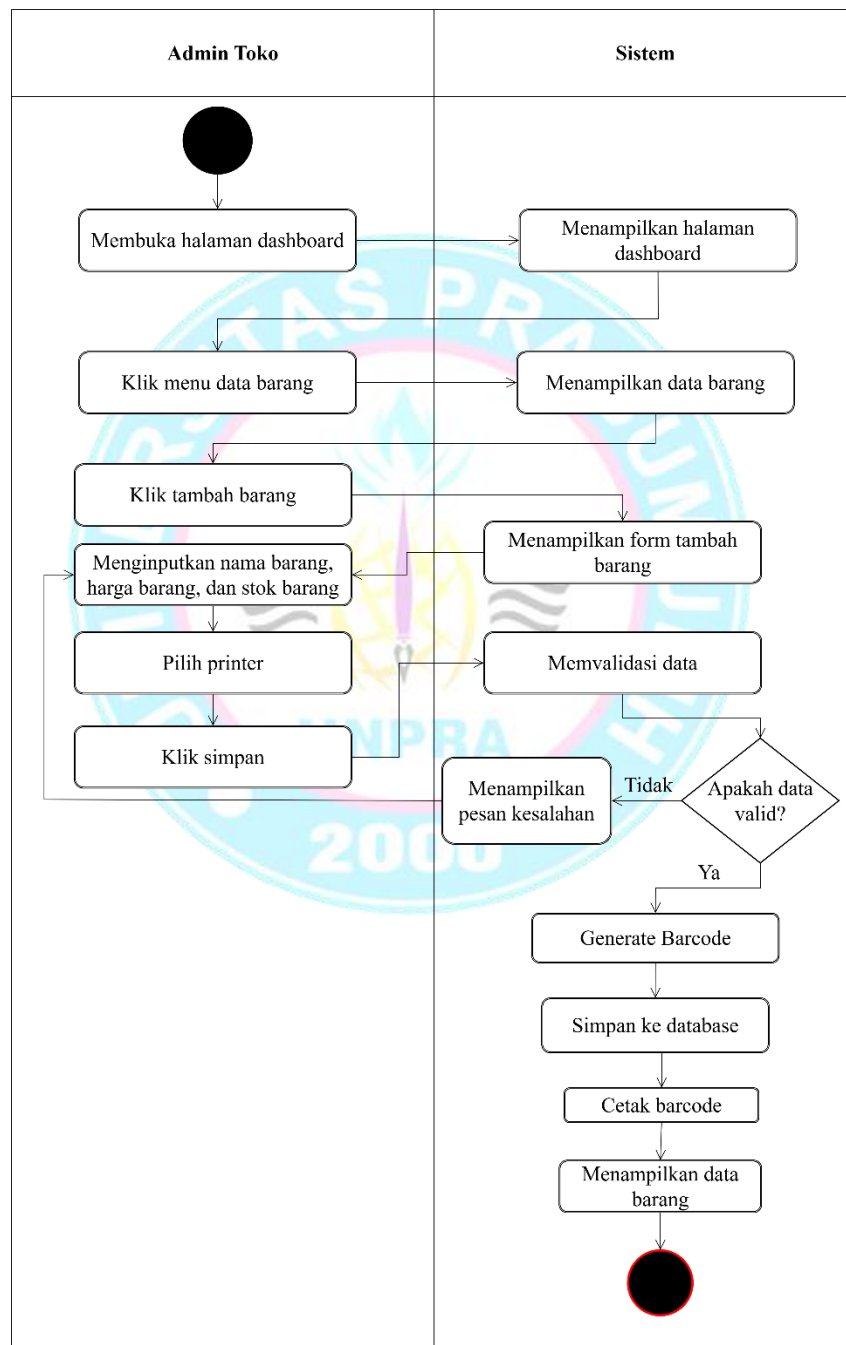
Sumber: Penulis, 2026

Gambar 4.3 Login Activity

Diagram *activity* berikutnya menggambarkan alur proses login yang dapat dilakukan oleh seluruh pengguna sistem. Proses dimulai ketika pengguna membuka aplikasi, kemudian sistem menampilkan halaman *login*. Selanjutnya pengguna memasukkan *email* dan *password*, lalu menekan tombol *login*. Sistem melakukan validasi terhadap data akun yang diinputkan. Apabila data yang dimasukkan tidak

valid, maka sistem akan menampilkan pesan kesalahan dan pengguna diminta untuk menginput ulang data *login*. Namun jika data valid, sistem akan mengarahkan pengguna ke halaman *dashboard* sesuai dengan *role* masing-masing. Proses berakhir ketika halaman *dashboard* berhasil ditampilkan.

3. Activity Tambah Barang



Sumber: Penulis, 2026

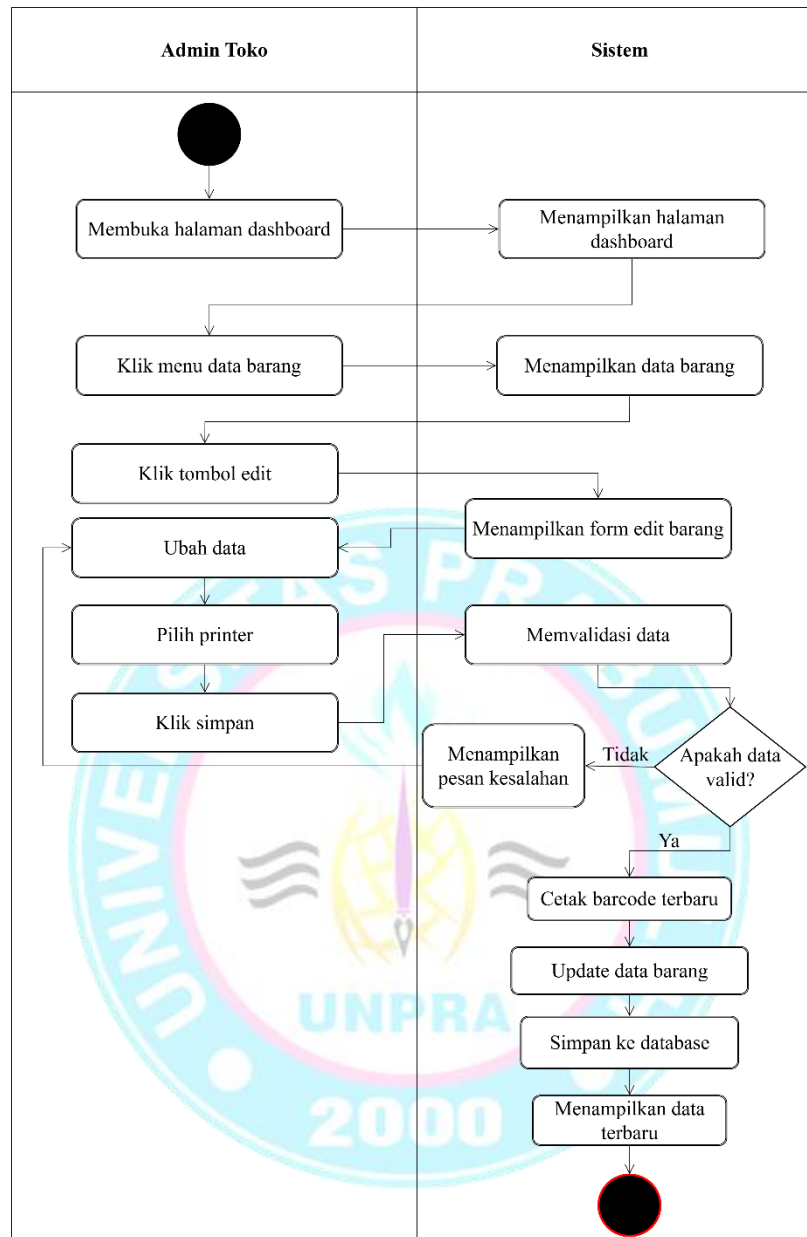
Gambar 4.4 Tambah Barang Activity

Activity diagram ini menggambarkan proses ketika pengguna menambahkan data barang ke dalam sistem aplikasi kasir. Proses dimulai ketika pengguna membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya pengguna menekan menu data barang, sehingga sistem menampilkan daftar data barang yang telah tersimpan di dalam sistem. Pengguna kemudian dapat melihat daftar barang yang tersedia dan memilih untuk menambahkan data barang baru dengan menekan tombol tambah barang.

Untuk menambahkan data barang, pengguna menekan tombol tambah barang sehingga sistem menampilkan *form* tambah barang. Pada tahap ini pengguna diminta untuk menginputkan data barang seperti nama barang, harga jual, harga beli, dan jumlah stok barang. Setelah data diinputkan, pengguna akan memilih printer dan menekan tombol simpan untuk menyimpan data tersebut ke dalam sistem. Selanjutnya sistem akan melakukan proses validasi terhadap data yang telah dimasukkan oleh pengguna.

Apabila data yang dimasukkan tidak lengkap atau tidak sesuai, maka sistem akan menampilkan pesan kesalahan dan pengguna diminta untuk memperbaiki atau menginput ulang data barang. Namun apabila data yang dimasukkan sudah valid, maka sistem akan melanjutkan proses dengan melakukan *generate barcode* secara otomatis untuk barang tersebut. *Barcode* yang dihasilkan kemudian disimpan bersamaan dengan data barang ke dalam *database*. Setelah proses berhasil, sistem menyediakan fitur untuk mencetak *barcode* sebagai label barang. Selanjutnya, sistem menampilkan kembali daftar barang yang telah diperbarui, termasuk data barang yang baru saja ditambahkan. Proses berakhir ketika data barang berhasil tersimpan dan ditampilkan kembali oleh sistem.

4. Activity Edit Barang



Sumber: Penulis, 2026

Gambar 4.5 Edit Barang Activity

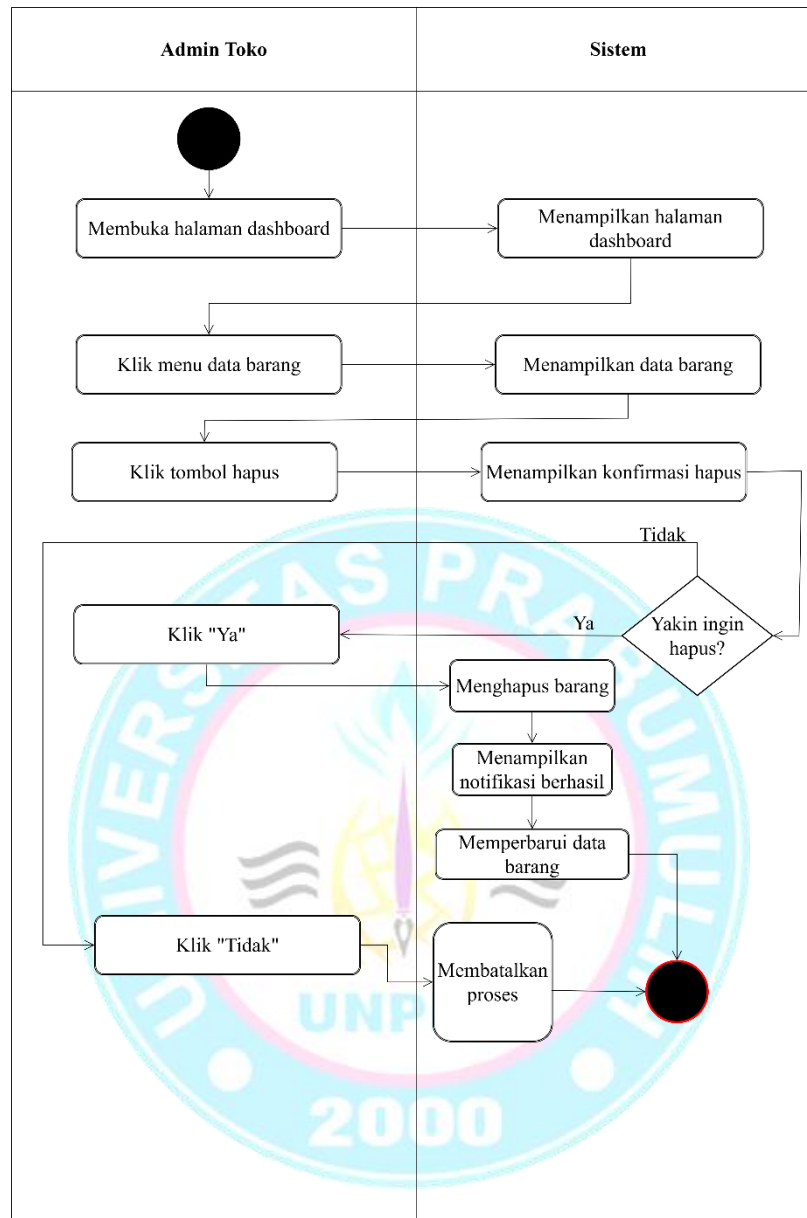
Activity diagram ini menggambarkan proses ketika pengguna melakukan pengeditan data barang yang telah tersimpan di dalam sistem aplikasi kasir. Proses dimulai ketika pengguna membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya pengguna menekan menu data barang, sehingga sistem menampilkan daftar data barang yang tersedia di dalam sistem. Pengguna kemudian dapat melihat daftar barang dan memilih salah satu

data barang yang akan diubah dengan menekan tombol edit. Untuk mengubah data barang, pengguna menekan tombol edit sehingga sistem menampilkan form edit barang. Pada tahap ini pengguna dapat melakukan perubahan data seperti nama barang, harga jual, harga beli, maupun jumlah stok barang sesuai dengan kebutuhan. Setelah melakukan perubahan data, pengguna memilih printer yang akan digunakan, kemudian menekan tombol simpan untuk menyimpan perubahan tersebut.

Selanjutnya sistem akan melakukan proses validasi terhadap data yang telah diinputkan. Apabila data yang dimasukkan tidak valid atau tidak sesuai, maka sistem akan menampilkan pesan kesalahan dan pengguna diminta untuk memperbaiki data yang diinputkan. Namun apabila data yang dimasukkan sudah valid, maka sistem akan melanjutkan proses dengan mencetak *barcode* terbaru sebagai pembaruan informasi barang.

Setelah itu sistem akan melakukan proses *update* data barang dan menyimpan perubahan tersebut ke dalam *database*. Setelah proses penyimpanan berhasil, sistem akan menampilkan kembali data barang yang telah diperbarui. Dengan demikian, pengguna dapat melihat perubahan data yang telah dilakukan. Proses berakhir ketika data barang terbaru berhasil ditampilkan pada sistem.

5. Activity Hapus Barang



Sumber: Penulis, 2026

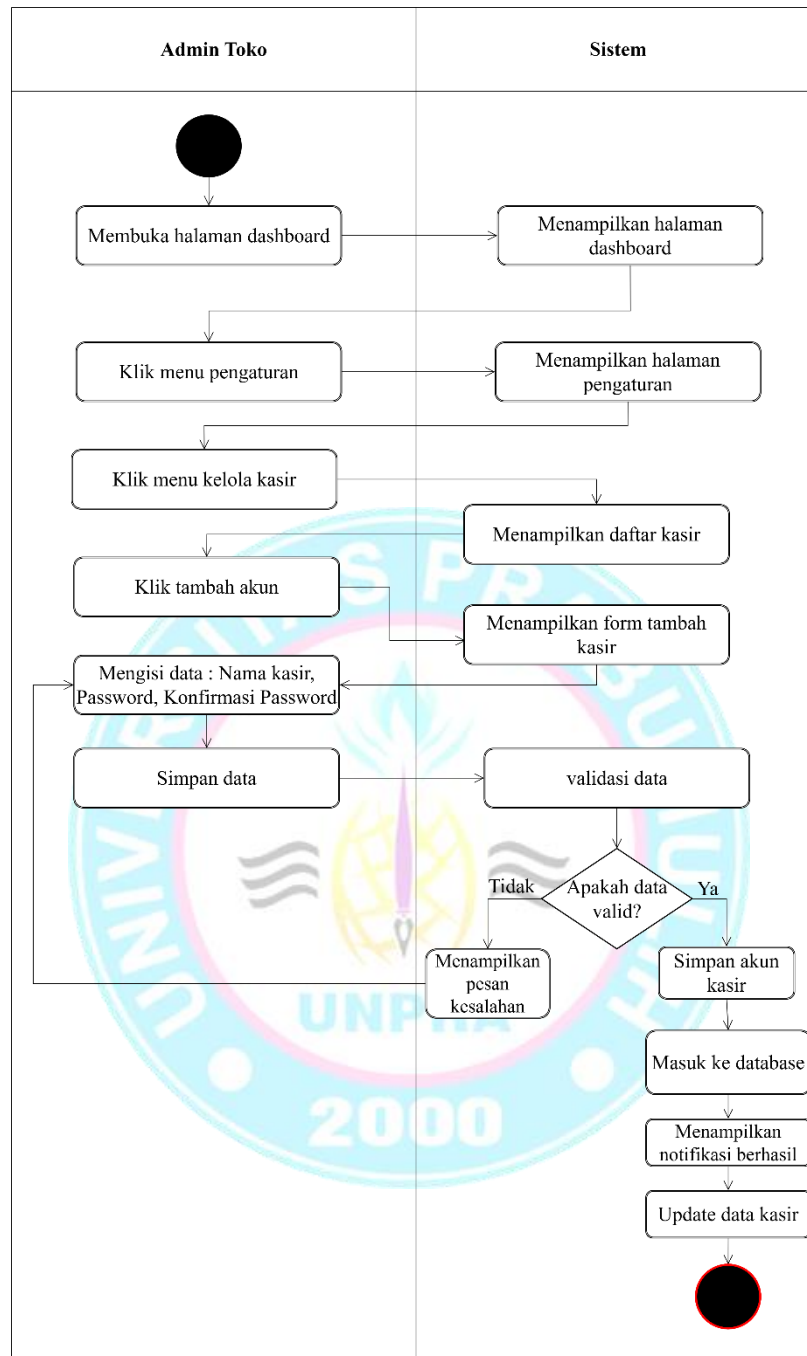
Gambar 4.6 Hapus Barang Activity

Activity diagram ini menggambarkan proses ketika pengguna melakukan penghapusan data barang yang telah tersimpan di dalam sistem aplikasi kasir. Proses dimulai ketika pengguna membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya pengguna menekan menu data barang, sehingga sistem menampilkan daftar data barang yang tersedia di dalam sistem. Pengguna kemudian dapat melihat daftar barang dan memilih salah satu data barang yang akan dihapus dengan menekan tombol hapus.

Untuk menghapus data barang, pengguna menekan tombol hapus sehingga sistem akan menampilkan konfirmasi penghapusan data. Pada tahap ini pengguna diminta untuk memastikan apakah data barang benar-benar akan dihapus atau tidak. Jika pengguna membatalkan proses, maka sistem akan kembali menampilkan daftar data barang tanpa adanya perubahan. Namun apabila pengguna menyetujui penghapusan, maka sistem akan melakukan proses penghapusan data barang dari *database*. Setelah proses penghapusan berhasil, sistem akan menampilkan kembali daftar data barang yang telah diperbarui tanpa data yang telah dihapus. Proses berakhir ketika data barang terbaru berhasil ditampilkan pada sistem.



6. Activity Tambah Data Kasir



Sumber: Penulis, 2026

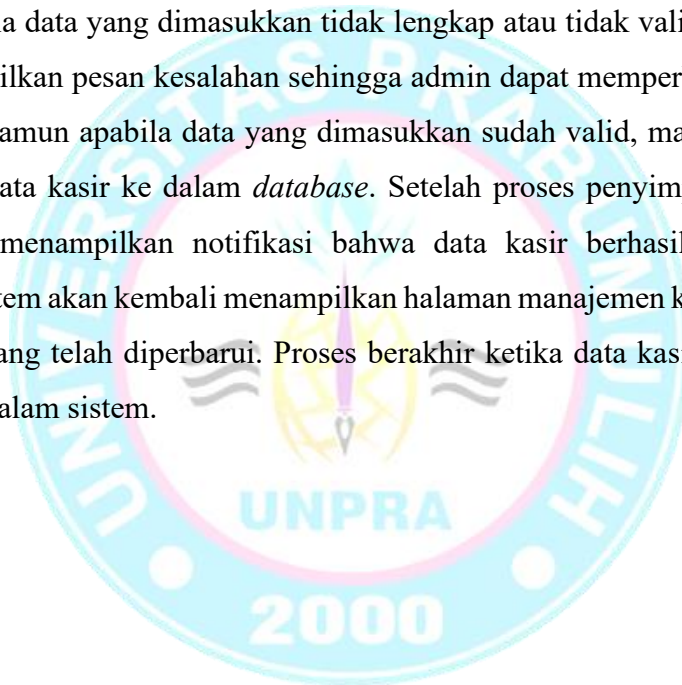
Gambar 4.7 Tambah Data Kasir Activity

Activity diagram ini menggambarkan alur proses ketika admin menambahkan data kasir baru ke dalam sistem aplikasi kasir berbasis *Android*. Proses dimulai ketika admin membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard* tersebut. Selanjutnya admin memilih menu

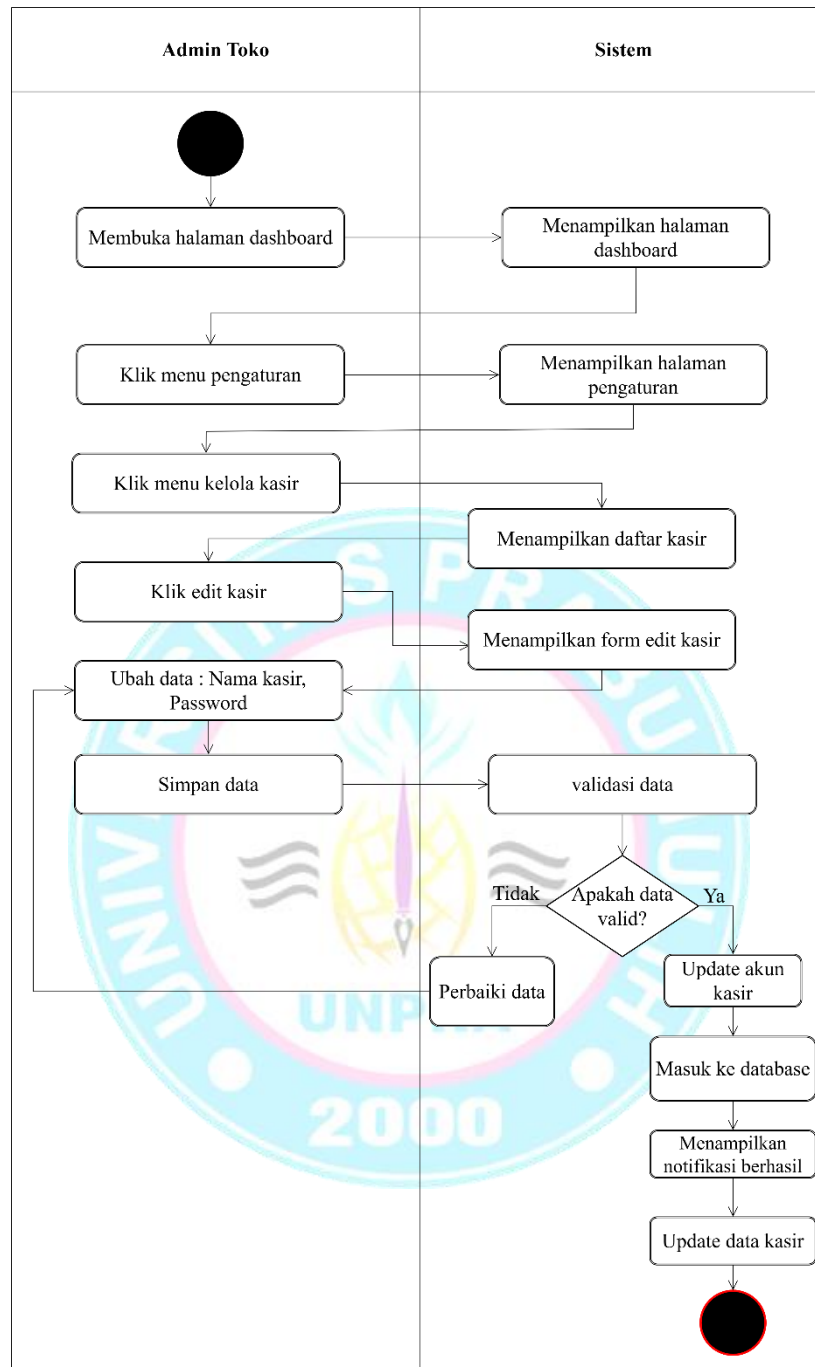
manajemen kasir sehingga sistem menampilkan daftar data kasir yang telah terdaftar di dalam sistem.

Setelah itu, admin menekan tombol tambah kasir untuk menambahkan data kasir baru. Sistem kemudian menampilkan *form* tambah kasir yang harus diisi oleh admin. Pada tahap ini admin menginputkan data kasir seperti nama kasir, *password*, konfirmasi *password* serta informasi lainnya yang dibutuhkan oleh sistem. Setelah seluruh data diisi, admin menekan tombol simpan data sehingga sistem menerima input data tersebut. Selanjutnya sistem melakukan proses validasi data untuk memastikan bahwa seluruh *field* telah terisi dengan benar dan sesuai dengan format yang ditentukan.

Apabila data yang dimasukkan tidak lengkap atau tidak valid, maka sistem akan menampilkan pesan kesalahan sehingga admin dapat memperbaiki data yang diinputkan. Namun apabila data yang dimasukkan sudah valid, maka sistem akan menyimpan data kasir ke dalam *database*. Setelah proses penyimpanan berhasil, sistem akan menampilkan notifikasi bahwa data kasir berhasil ditambahkan. Kemudian sistem akan kembali menampilkan halaman manajemen kasir yang berisi daftar kasir yang telah diperbarui. Proses berakhir ketika data kasir baru berhasil ditampilkan dalam sistem.



7. Activity Edit Data Kasir



Sumber: Penulis, 2026

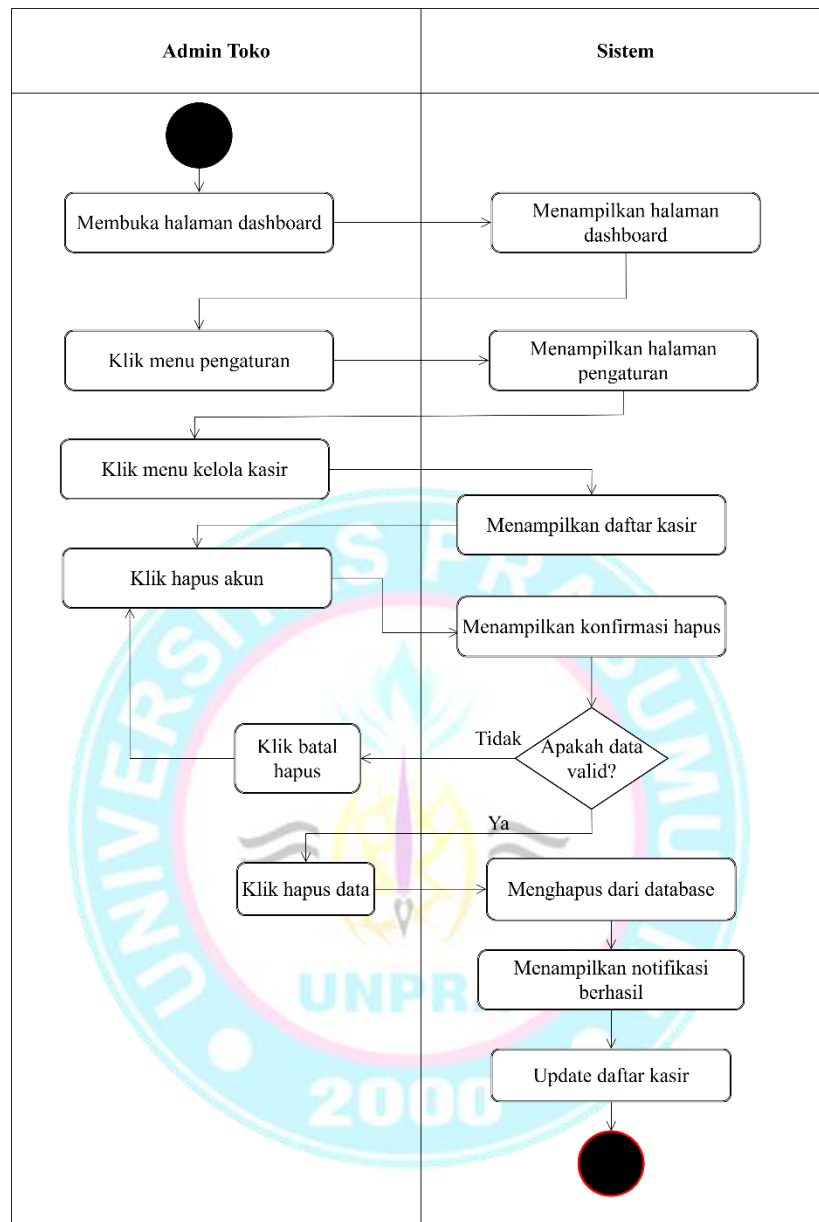
Gambar 4.8 Edit Data Kasir Activity

Activity diagram ini menggambarkan alur proses ketika admin melakukan perubahan atau edit data kasir pada sistem aplikasi kasir berbasis *Android*. Proses dimulai ketika admin membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya admin menekan menu pengaturan

sehingga sistem menampilkan halaman pengaturan. Setelah itu admin memilih menu kelola kasir, sehingga sistem menampilkan daftar kasir yang telah terdaftar di dalam sistem.

Admin kemudian memilih salah satu data kasir yang akan diubah dengan menekan tombol edit akun. Sistem selanjutnya menampilkan form edit kasir yang berisi data kasir yang dapat diperbarui oleh admin. Pada tahap ini admin melakukan perubahan data kasir seperti nama kasir dan *password* sesuai kebutuhan. Setelah selesai melakukan perubahan, admin menekan tombol simpan data sehingga sistem menerima input perubahan tersebut. Selanjutnya sistem melakukan validasi data untuk memastikan bahwa data yang dimasukkan telah sesuai dan tidak terdapat kesalahan. Apabila data yang dimasukkan tidak valid, maka sistem akan meminta admin untuk memperbaiki data yang telah diinputkan.

Namun apabila data yang dimasukkan sudah valid, maka sistem akan melakukan proses *update* data kasir dan menyimpan perubahan tersebut ke dalam *database*. Setelah proses berhasil, sistem akan menampilkan notifikasi bahwa data kasir berhasil diperbarui. Selanjutnya sistem akan memperbarui daftar kasir dan menampilkan data kasir yang telah diperbarui. Proses berakhir ketika perubahan data kasir berhasil ditampilkan pada sistem.

8. *Activity Hapus Data Kasir*

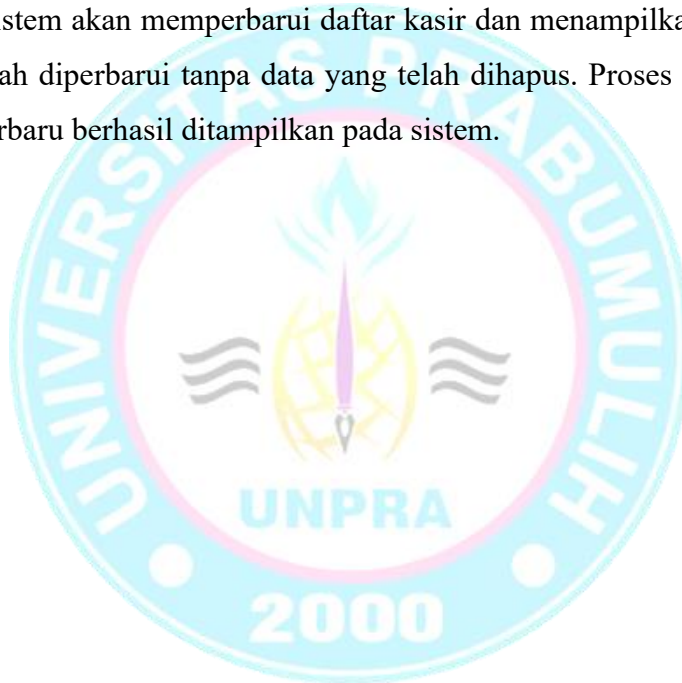
Sumber: Penulis, 2026

Gambar 4.9 Hapus Data Kasir *Activity*

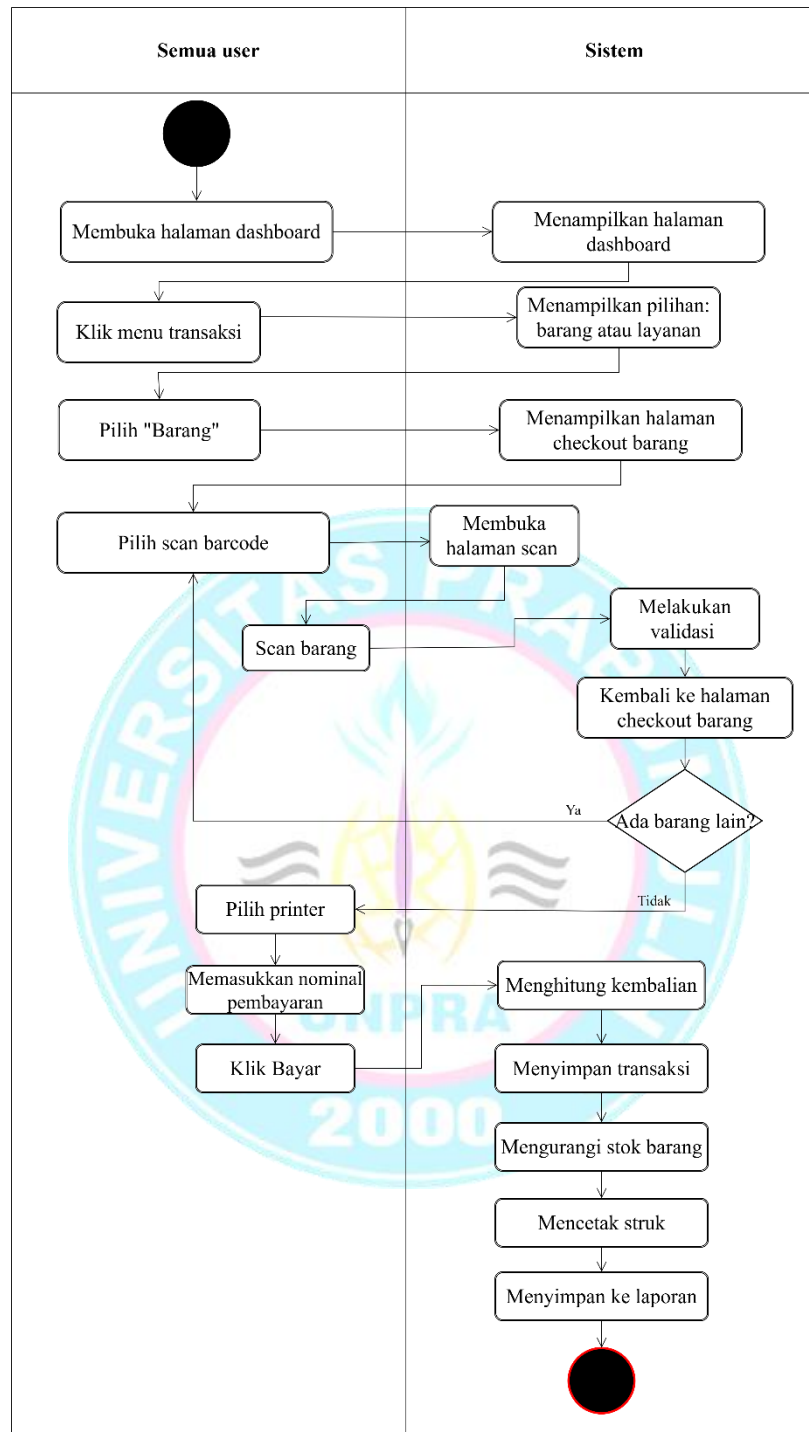
Activity diagram ini menggambarkan alur proses ketika admin melakukan penghapusan data kasir pada sistem aplikasi kasir berbasis *Android*. Proses dimulai ketika admin membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya admin menekan menu pengaturan sehingga sistem menampilkan halaman pengaturan. Setelah itu admin memilih menu kelola kasir, sehingga sistem menampilkan daftar kasir yang terdaftar di dalam sistem. Admin

kemudian memilih salah satu data kasir yang akan dihapus dengan menekan tombol hapus akun. Sistem selanjutnya menampilkan konfirmasi penghapusan data untuk memastikan tindakan yang dilakukan oleh admin.

Pada tahap ini, admin memiliki dua pilihan yaitu membatalkan atau melanjutkan proses penghapusan. Jika admin menekan tombol batal hapus, maka sistem akan kembali menampilkan daftar kasir tanpa adanya perubahan data. Namun apabila admin menekan tombol hapus data, maka sistem akan memproses penghapusan tersebut dengan melakukan validasi. Jika data dinyatakan valid, maka sistem akan menghapus data kasir dari *database*. Setelah proses penghapusan berhasil, sistem akan menampilkan notifikasi bahwa data kasir berhasil dihapus. Selanjutnya sistem akan memperbarui daftar kasir dan menampilkan kembali data kasir yang telah diperbarui tanpa data yang telah dihapus. Proses berakhir ketika daftar kasir terbaru berhasil ditampilkan pada sistem.



9. Activity Transaksi Barang



Sumber: Penulis, 2026

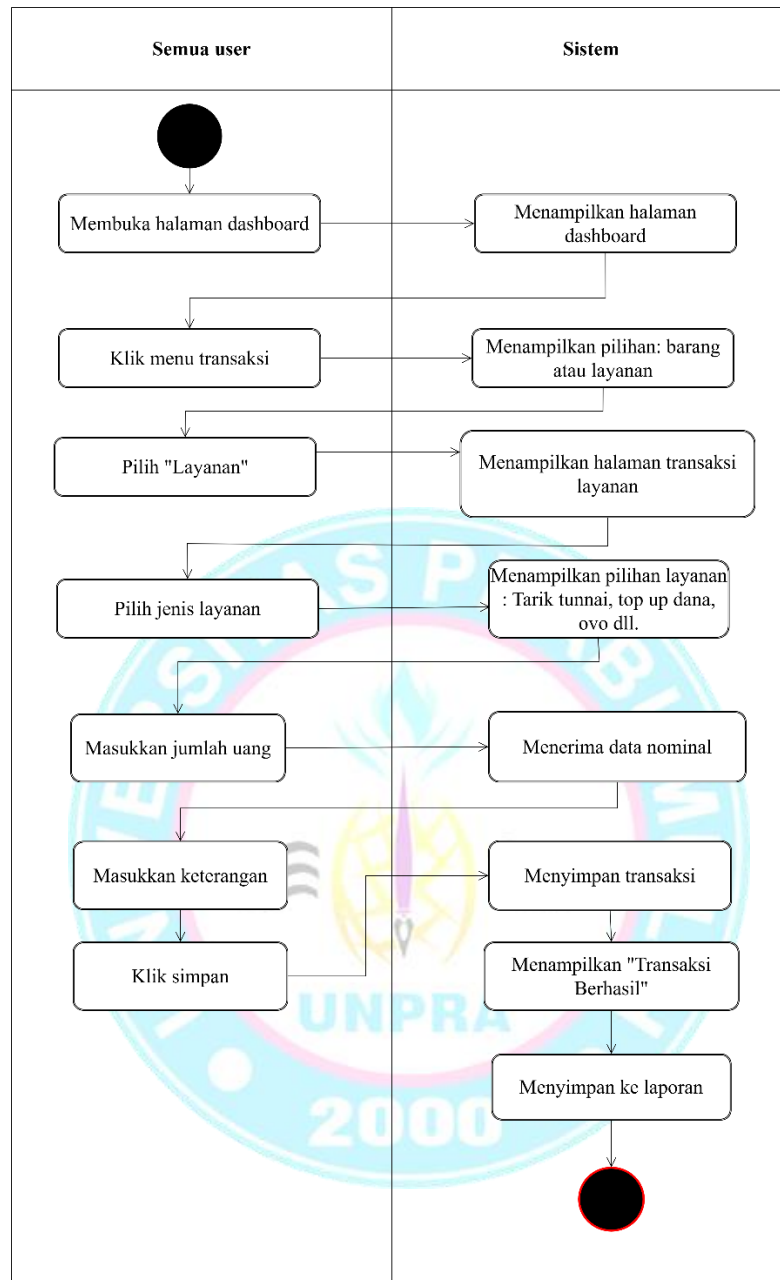
Gambar 4.10 Transaksi Barang Activity

Activity diagram ini menggambarkan alur proses transaksi barang pada sistem aplikasi kasir berbasis *Android*. Proses dimulai ketika admin membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*.

Selanjutnya admin memilih menu transaksi sehingga sistem menampilkan pilihan transaksi berupa barang atau layanan. Pada proses ini admin memilih opsi “barang”, sehingga sistem menampilkan halaman *checkout* barang. Setelah itu admin memilih fitur *scan barcode*, dan sistem membuka fitur kamera untuk melakukan pemindaian *barcode*. Admin kemudian melakukan *scan barcode* pada barang yang akan dibeli, dan sistem akan membaca *barcode* tersebut serta melakukan validasi data barang. Setelah validasi berhasil, sistem kembali ke halaman *checkout* barang dan menambahkan barang ke dalam daftar transaksi. Apabila terdapat barang lain yang akan ditambahkan, maka admin dapat melakukan *scan barcode* kembali hingga seluruh barang berhasil dimasukkan ke dalam daftar transaksi. Setiap barang yang berhasil dipindai akan ditampilkan sebagai daftar pembelian. Ketika semua barang telah dimasukkan, sistem akan menampilkan total harga transaksi dan admin dapat melanjutkan proses pembayaran.

Selanjutnya admin memasukkan nominal pembayaran yang diberikan pelanggan dan memilih *printer* yang akan digunakan untuk mencetak struk. Setelah itu admin menekan tombol bayar, sehingga sistem memproses transaksi dengan menghitung kembalian secara otomatis. Sistem kemudian menyimpan data transaksi ke *database*, mengurangi stok barang sesuai jumlah yang terjual, serta mencetak struk sebagai bukti transaksi. Sistem menyimpan seluruh data transaksi ke dalam laporan. Proses transaksi dinyatakan selesai setelah seluruh data berhasil diproses, disimpan, dan dicatat oleh sistem.

10. Activity Transaksi Layanan



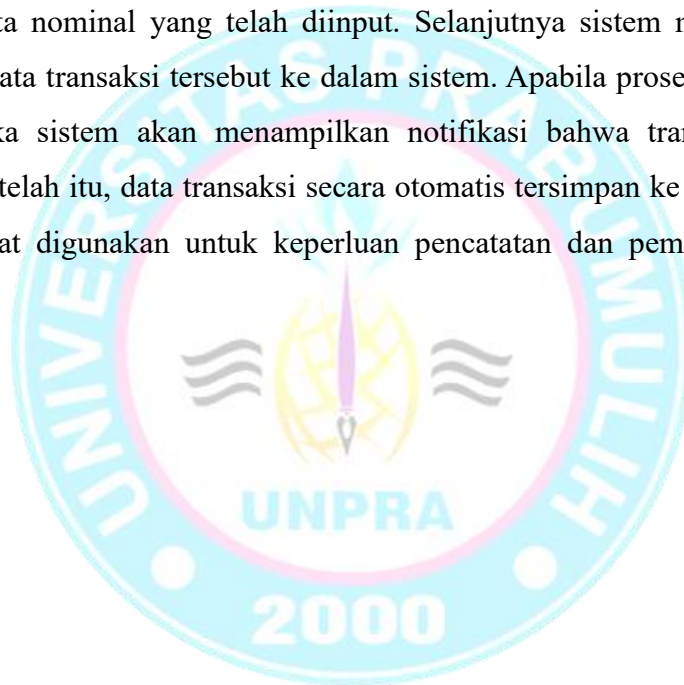
Sumber: Penulis, 2026

Gambar 4.11 Transaksi Layanan Activity

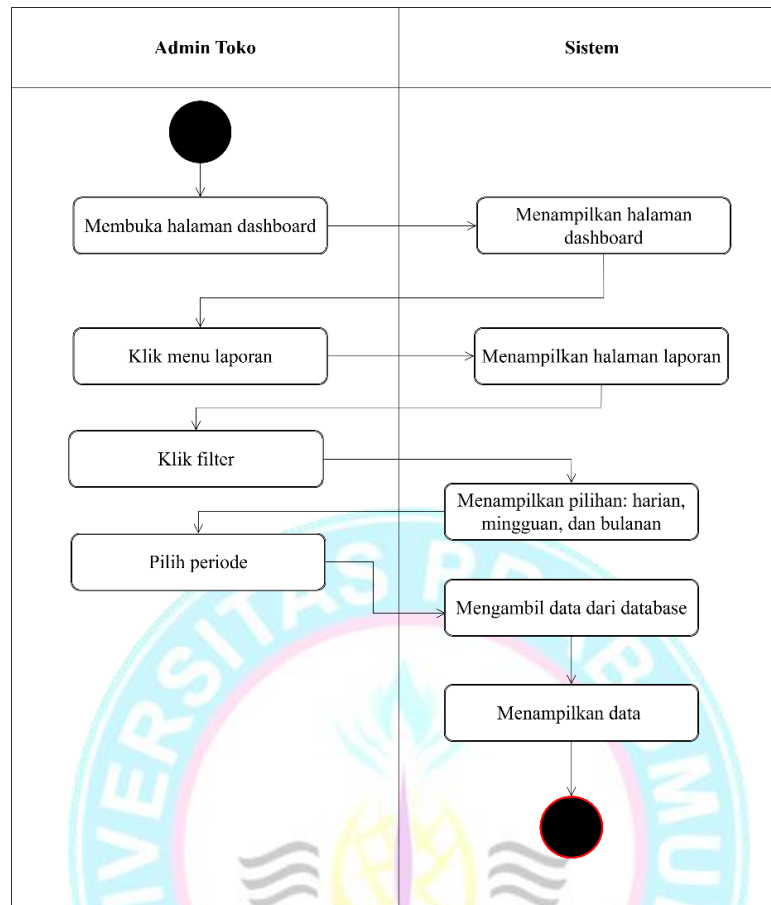
Activity diagram ini menggambarkan alur proses transaksi layanan pada aplikasi kasir berbasis *Android* di Apan Cell. Proses dimulai ketika pengguna (admin atau kasir) membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard* sebagai tampilan utama aplikasi. Selanjutnya, pengguna memilih menu transaksi, sehingga sistem menampilkan pilihan jenis transaksi yang tersedia, yaitu barang atau layanan. Pada proses ini, pengguna memilih opsi

layanan, sehingga sistem akan menampilkan halaman transaksi layanan, yang berisi jenis layanan yang harus di pilih, nominal dan keterangan yang harus diisi. Pada filter jenis layanan, ada berbagai jenis layanan yang tersedia seperti *top up*, tarik tunai, *transfer*, dan layanan lainnya. Setelah itu, pengguna memilih salah satu jenis layanan yang diinginkan. Sistem kemudian menampilkan form input nominal yang harus diisi oleh pengguna.

Pada tahap ini, pengguna memasukkan jumlah nominal transaksi sesuai dengan kebutuhan pelanggan. Pengguna juga di haruskan memasukkan keterangan dari layanan, misalnya nomor *handphone* pelanggan. Setelah semuanya data sudah terisi dengan benar, pengguna menekan tombol "Simpan", sehingga sistem menerima data nominal yang telah diinput. Selanjutnya sistem memproses dan menyimpan data transaksi tersebut ke dalam sistem. Apabila proses penyimpanan berhasil, maka sistem akan menampilkan notifikasi bahwa transaksi berhasil dilakukan. Setelah itu, data transaksi secara otomatis tersimpan ke dalam laporan, sehingga dapat digunakan untuk keperluan pencatatan dan pembuatan laporan penjualan.



11. Activity Laporan



Sumber: Penulis, 2026

Gambar 4.12 Laporan Activity

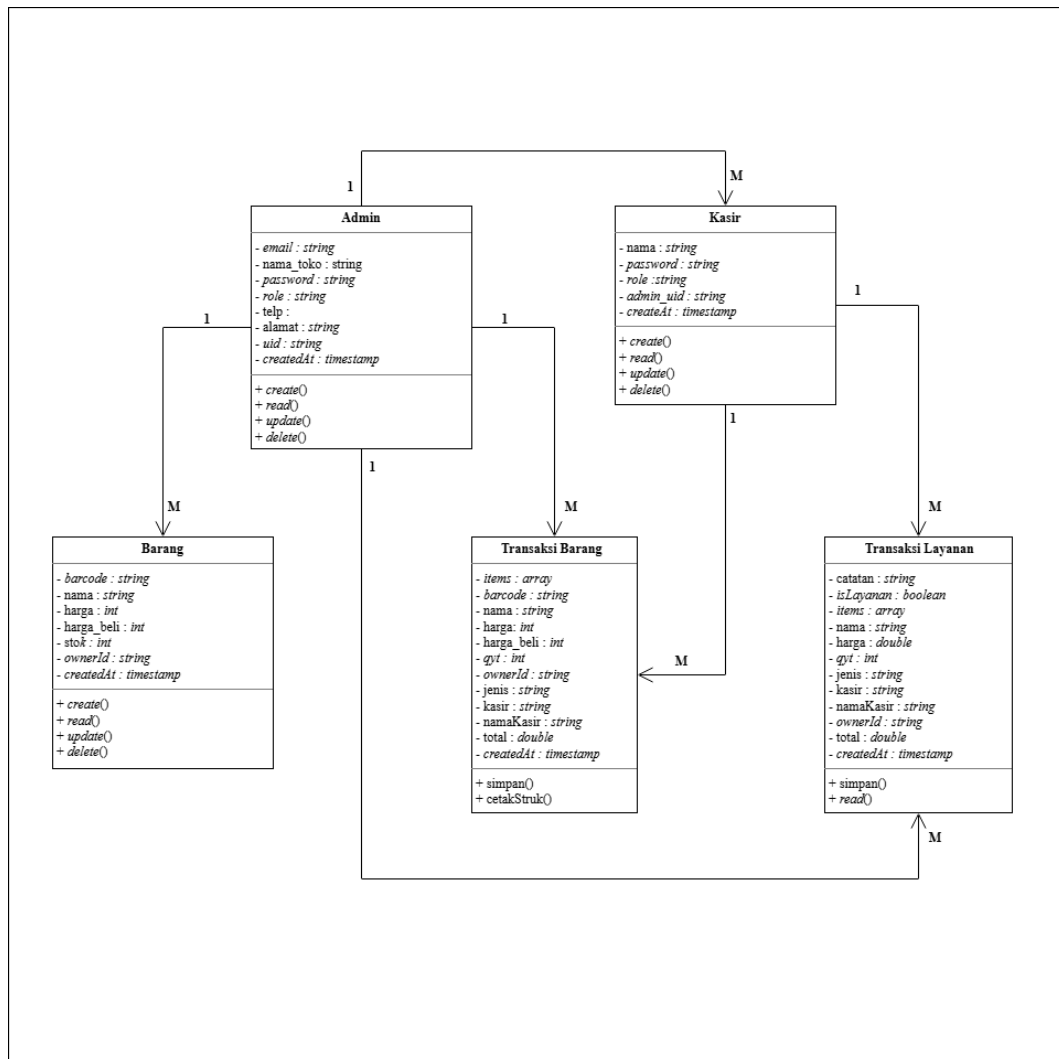
Activity diagram ini menggambarkan alur proses dalam melihat laporan pada aplikasi kasir berbasis *Android* di Apan Cell. Proses dimulai ketika admin membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard* sebagai tampilan awal aplikasi. Selanjutnya, pengguna memilih menu laporan, sehingga sistem menampilkan halaman laporan yang berisi data transaksi yang telah tersimpan. Untuk menampilkan data sesuai kebutuhan, pengguna dapat menekan tombol *filter*.

Setelah tombol *filter* dipilih, sistem akan menampilkan beberapa pilihan periode laporan, yaitu harian, mingguan, dan bulanan. Pada tahap ini, pengguna memilih periode laporan yang diinginkan. Setelah periode dipilih, sistem akan mengambil data laporan dari *database* sesuai dengan periode yang telah ditentukan. Data yang diambil meliputi informasi transaksi penjualan dan aktivitas yang telah

dilakukan pada rentang waktu tersebut. Selanjutnya, sistem menampilkan data laporan kepada pengguna sesuai dengan filter yang dipilih. Dengan demikian, pengguna dapat melihat dan menganalisis laporan transaksi secara lebih mudah, cepat, dan terstruktur.

c. *Class Diagram*

Berdasarkan dari *usecase* dan *activity* diagram yang diusulkan, maka *class* diagram dari perancangan sistem kasir yang diusulkan sebagai berikut:



Sumber : penulis, 2026

Gambar 4. 13 *Class Diagram* yang diusulkan

4.2.2 Perancangan Database (Fisik)

a. Tabel Admin

Tabel 4.1 Admin

Nama Field	Tipe Data	Keterangan
<i>email</i>	<i>string</i>	Alamat email pengguna sebagai identitas login admin.
<i>nama_toko</i>	<i>string</i>	Nama bisnis/toko yang dikelola oleh admin.
<i>password</i>	<i>string</i>	Kata sandi akun admin untuk keamanan sistem.
<i>role</i>	<i>string</i>	Hak akses pengguna dengan nilai “admin”.
<i>telp</i>	<i>string</i>	Nomor telepon admin yang digunakan sebagai kontak.
<i>alamat</i>	<i>string</i>	Alamat tempat usaha atau alamat admin.
<i>uid</i>	<i>string</i>	ID unik pengguna yang digunakan sebagai identitas akun admin pada sistem.
<i>createdAt</i>	<i>timestamp</i>	Waktu pendaftaran atau pembuatan akun admin.

Sumber: Penulis, 2026

Koleksi admin merupakan pusat penyimpanan data identitas dan kredensial bagi pengguna yang memiliki hak akses sebagai pengelola utama pada aplikasi kasir berbasis *Android* di Apan Cell. Berbeda dengan basis data relasional konvensional, koleksi ini bersifat fleksibel karena setiap dokumen mewakili satu akun admin yang dapat mengelola seluruh fitur sistem, seperti pengelolaan barang, transaksi, layanan, serta laporan penjualan dan persediaan.

Di dalam koleksi ini terdapat *field email* dan *telp* yang digunakan sebagai data kontak utama sekaligus identitas saat melakukan proses masuk (*login*) ke dalam aplikasi. *Field password* berfungsi sebagai kunci keamanan akun agar data dan aktivitas pada sistem tetap terlindungi. Selain itu, *field role* digunakan untuk menentukan hak akses pengguna, di mana nilai yang digunakan pada koleksi ini adalah “admin” sehingga pengguna memiliki akses penuh terhadap seluruh menu dan fitur aplikasi.

Field nama_toko digunakan untuk menyimpan identitas usaha atau toko yang dikelola oleh admin, sedangkan *field alamat* digunakan untuk menyimpan informasi lokasi atau alamat admin maupun toko. Sistem juga memanfaatkan *field createdAt* bertipe *timestamp* untuk mencatat waktu pembuatan akun admin secara

otomatis, sehingga memudahkan proses pendataan dan pengelolaan pengguna di dalam sistem.

Field uid digunakan sebagai identitas unik dari setiap akun admin yang tersimpan di *Firebase Authentication*. Nilai uid dibuat secara otomatis oleh sistem *Firebase* dan berfungsi untuk membedakan setiap pengguna secara spesifik. Selain itu, uid juga digunakan untuk menghubungkan data akun admin dengan data lainnya di dalam *database Firestore* sehingga proses autentikasi, pengelolaan data, serta validasi pengguna dapat dilakukan dengan lebih aman dan terstruktur.

b. Tabel Barang

Tabel 4.2 Barang

Nama Field	Tipe Data	Keterangan
<i>barcode</i>	<i>string</i>	Kode unik produk (<i>Primary Key</i> dalam bentuk <i>Document ID</i>).
<i>nama</i>	<i>string</i>	Nama lengkap barang/produk.
<i>harga</i>	<i>int</i>	Harga jual satuan produk.
<i>harga_beli</i>	<i>int</i>	Harga modal atau harga beli produk dari <i>supplier</i> .
<i>stok</i>	<i>int</i>	Jumlah persediaan barang yang tersedia.
<i>ownerId</i>	<i>string</i>	Relasi ID ke admin/pemilik yang mengelola barang tersebut.
<i>createAt</i>	<i>timestamp</i>	Waktu data barang pertama kali dibuat.

Sumber: Penulis, 2026

Koleksi barang dirancang untuk mengelola seluruh data inventaris produk secara sistematis dan real-time pada aplikasi kasir berbasis *Android* di Apan Cell. Setiap dokumen pada koleksi ini diidentifikasi melalui *field barcode* yang berfungsi sebagai identitas unik atau *Primary Key* dalam bentuk *Document ID*, sehingga dapat mencegah terjadinya duplikasi data barang.

Field nama digunakan untuk menyimpan nama lengkap produk atau barang yang dijual, seperti aksesoris handphone, makanan, minuman, maupun produk lainnya. *Field* harga digunakan untuk menyimpan nilai harga jual produk kepada konsumen. Selanjutnya, *field* harga_beli digunakan untuk mencatat harga modal

atau harga beli barang dari *supplier* sebagai dasar perhitungan keuntungan penjualan.

Field stok digunakan untuk menyimpan jumlah persediaan barang yang tersedia di toko. Nilai stok akan berubah secara otomatis setiap kali terjadi transaksi penjualan maupun penambahan stok barang. Selain itu, *field ownerId* digunakan sebagai relasi identitas admin atau pemilik toko yang mengelola data barang tersebut agar data tetap terhubung dengan pengguna yang sesuai.

Field createAt bertipe timestamp digunakan untuk mencatat waktu pertama kali data barang dibuat ke dalam sistem. Struktur data pada koleksi barang ini membantu proses pengelolaan inventaris menjadi lebih teratur, cepat, dan akurat sehingga memudahkan pemilik toko dalam memantau kondisi stok barang secara *real-time*.

c. Tabel Kasir

Tabel 4.3 Kasir

Nama <i>Field</i>	Tipe Data	Keterangan
nama	<i>string</i>	Nama lengkap kasir yang terdaftar pada sistem.
<i>password</i>	<i>string</i>	Kata sandi akun kasir untuk proses login aplikasi.
<i>role</i>	<i>string</i>	Hak akses pengguna dengan nilai “kasir”.
admin_uid	<i>string</i>	ID admin yang menjadi pemilik atau pengelola akun kasir.
<i>createAt</i>	<i>timestamp</i>	Waktu pertama kali akun kasir dibuat.

Sumber: Penulis, 2026

Koleksi kasir digunakan untuk menyimpan data pengguna yang memiliki hak akses sebagai kasir pada aplikasi kasir berbasis *Android* di Apan Cell. Setiap dokumen pada koleksi ini mewakili satu akun kasir yang dapat digunakan untuk melakukan transaksi penjualan sesuai dengan hak akses yang diberikan oleh admin.

Field nama digunakan untuk menyimpan nama lengkap kasir yang terdaftar pada sistem. *Field password* digunakan sebagai kata sandi akun kasir untuk proses *login* aplikasi agar keamanan data pengguna tetap terjaga. Selain itu, *field role* digunakan untuk menentukan hak akses pengguna dengan nilai “kasir”, sehingga pengguna hanya dapat mengakses fitur sesuai dengan perannya sebagai kasir.

Field `admin_uid` digunakan untuk menyimpan ID admin yang menjadi pemilik atau pengelola akun kasir tersebut. Relasi ini membantu sistem dalam menghubungkan data kasir dengan admin yang sesuai sehingga pengelolaan pengguna dapat dilakukan dengan lebih teratur.

Field `createAt` bertipe `timestamp` digunakan untuk mencatat waktu pertama kali akun kasir dibuat di dalam sistem. Struktur data pada koleksi kasir membantu proses pengelolaan pengguna menjadi lebih aman dan terstruktur sehingga setiap aktivitas transaksi dapat dikontrol berdasarkan hak akses masing-masing pengguna.

d. Tabel Transaksi Barang

Tabel 4.4 Transaksi Barang

Nama <i>Field</i>	Tipe Data	Keterangan
<code>items</code>	<code>array</code>	Menyimpan daftar barang yang dibeli dalam satu transaksi.
<code>barcode</code>	<code>string</code>	Kode unik barang yang ditransaksikan.
<code>nama</code>	<code>string</code>	Nama barang yang dibeli oleh pelanggan.
<code>harga</code>	<code>int</code>	Harga jual barang per satuan.
<code>harga_beli</code>	<code>int</code>	Harga modal barang untuk perhitungan keuntungan.
<code>qty</code>	<code>int</code>	Jumlah barang yang dibeli dalam transaksi.
<code>ownerId</code>	<code>string</code>	ID admin atau pemilik toko yang memiliki data transaksi.
<code>jenis</code>	<code>string</code>	Jenis transaksi yang dilakukan, yaitu “barang”.
<code>kasir</code>	<code>string</code>	Nama pengguna atau akun kasir yang melakukan transaksi.
<code>namaKasir</code>	<code>string</code>	Nama lengkap kasir yang bertugas saat transaksi berlangsung.
<code>total</code>	<code>double</code>	Total keseluruhan pembayaran transaksi barang.
<code>createdAt</code>	<code>timestamp</code>	Waktu transaksi barang dilakukan dan disimpan ke sistem.

Sumber: Penulis, 2026

Koleksi `transaksi_barang` digunakan untuk menyimpan seluruh data transaksi penjualan barang pada aplikasi kasir berbasis *Android* di Apan Cell. Setiap dokumen transaksi berisi informasi barang yang dibeli pelanggan beserta detail transaksi yang dilakukan oleh kasir pada waktu tertentu.

Field items bertipe *array* digunakan untuk menyimpan daftar barang yang dibeli dalam satu transaksi. Data di dalam *array* tersebut terdiri dari *field barcode*, nama, harga, harga_beli, qty, dan *ownerId* yang berfungsi untuk menyimpan detail setiap produk yang terlibat pada transaksi penjualan. *Field barcode* digunakan sebagai kode unik barang yang ditransaksikan. *Field nama* digunakan untuk menyimpan nama barang yang dibeli oleh pelanggan. *Field harga* digunakan untuk menyimpan harga jual barang per satuan, sedangkan *field harga_beli* digunakan untuk menyimpan harga modal barang untuk perhitungan keuntungan. *Field qty* digunakan untuk menyimpan jumlah barang yang dibeli dalam transaksi.

Field ownerId digunakan untuk menyimpan ID admin atau pemilik toko yang memiliki data transaksi tersebut. Selain itu, *field jenis* digunakan untuk menandai jenis transaksi yang dilakukan, yaitu “barang”. *Field kasir* digunakan untuk menyimpan nama pengguna atau akun kasir yang melakukan transaksi, sedangkan *field namaKasir* digunakan untuk menyimpan nama lengkap kasir yang bertugas saat transaksi berlangsung.

Field total bertipe *double* digunakan untuk menyimpan total keseluruhan pembayaran transaksi barang. Selanjutnya, *field createdAt* bertipe *timestamp* digunakan untuk mencatat waktu transaksi barang dilakukan dan disimpan ke dalam sistem secara otomatis. Struktur data pada koleksi transaksi_barang membantu proses pencatatan transaksi menjadi lebih cepat, teratur, dan akurat sehingga laporan penjualan dapat dihasilkan secara otomatis dan real-time.

e. Tabel Transaksi Layanan

Tabel 4.5 Transaksi Layanan

Nama <i>Field</i>	Tipe Data	Keterangan
catatan	<i>string</i>	Nomor tujuan atau keterangan tambahan layanan transaksi.
<i>isLayanan</i>	<i>boolean</i>	Penanda bahwa transaksi termasuk jenis layanan.
<i>items</i>	<i>array</i>	Menyimpan daftar layanan yang diproses dalam transaksi.
nama	<i>string</i>	Nama layanan yang dipilih pelanggan.
harga	<i>double</i>	Harga layanan yang dibayarkan pelanggan.

<i>qty</i>	<i>int</i>	Jumlah layanan yang diproses dalam transaksi.
<i>jenis</i>	<i>string</i>	Jenis transaksi dengan nilai “layanan”.
<i>kasir</i>	<i>string</i>	Nama akun pengguna yang melakukan transaksi layanan.
<i>namaKasir</i>	<i>string</i>	Nama lengkap kasir yang bertugas saat transaksi berlangsung.
<i>ownerId</i>	<i>string</i>	ID admin atau pemilik toko yang memiliki data transaksi.
<i>total</i>	<i>double</i>	Total pembayaran transaksi layanan.
<i>createdAt</i>	<i>timestamp</i>	Waktu transaksi layanan dilakukan dan disimpan ke sistem.

Sumber: Penulis, 2026

Koleksi *transaksi_layanan* digunakan untuk menyimpan seluruh data transaksi layanan digital pada aplikasi kasir berbasis *Android* di Apan Cell. Transaksi layanan yang dimaksud meliputi pengisian pulsa, paket data, *top up e-wallet*, pembayaran tagihan, dan berbagai layanan digital lainnya yang tersedia pada toko.

Field catatan digunakan untuk menyimpan nomor tujuan atau keterangan tambahan layanan transaksi. *Field* *isLayanan* bertipe *boolean* digunakan sebagai penanda bahwa transaksi yang dilakukan termasuk jenis layanan. Selain itu, *field items* bertipe *array* digunakan untuk menyimpan daftar layanan yang diproses dalam transaksi. Data di dalam *array* tersebut terdiri dari *field* nama, harga, dan *qty* yang digunakan untuk menyimpan detail layanan yang dipilih pelanggan.

Field nama digunakan untuk menyimpan nama layanan yang dipilih pelanggan. *Field* harga bertipe *double* digunakan untuk menyimpan harga layanan yang dibayarkan pelanggan. Selanjutnya, *field qty* digunakan untuk menyimpan jumlah layanan yang diproses dalam transaksi.

Field jenis digunakan untuk menandai jenis transaksi dengan nilai “layanan”. *Field* kasir digunakan untuk menyimpan nama akun pengguna yang melakukan transaksi layanan, sedangkan *field* *namaKasir* digunakan untuk menyimpan nama lengkap kasir yang bertugas saat transaksi berlangsung. Selain itu, *field* *ownerId* digunakan untuk menyimpan ID admin atau pemilik toko yang memiliki data transaksi tersebut.

Field total bertipe *double* digunakan untuk menyimpan total pembayaran transaksi layanan. Selanjutnya, *field createdAt* bertipe *timestamp* digunakan untuk mencatat waktu transaksi layanan dilakukan dan disimpan ke dalam sistem secara otomatis. Struktur data pada koleksi *transaksi_layanan* membantu proses pencatatan layanan digital menjadi lebih cepat, teratur, dan akurat sehingga laporan transaksi dapat dihasilkan secara otomatis dan *real-time*.

4.2.3 Perancangan Antar Muka

a. Halaman Registrasi

Sumber: Penulis, 2026

Gambar 4.14 Halaman Registrasi

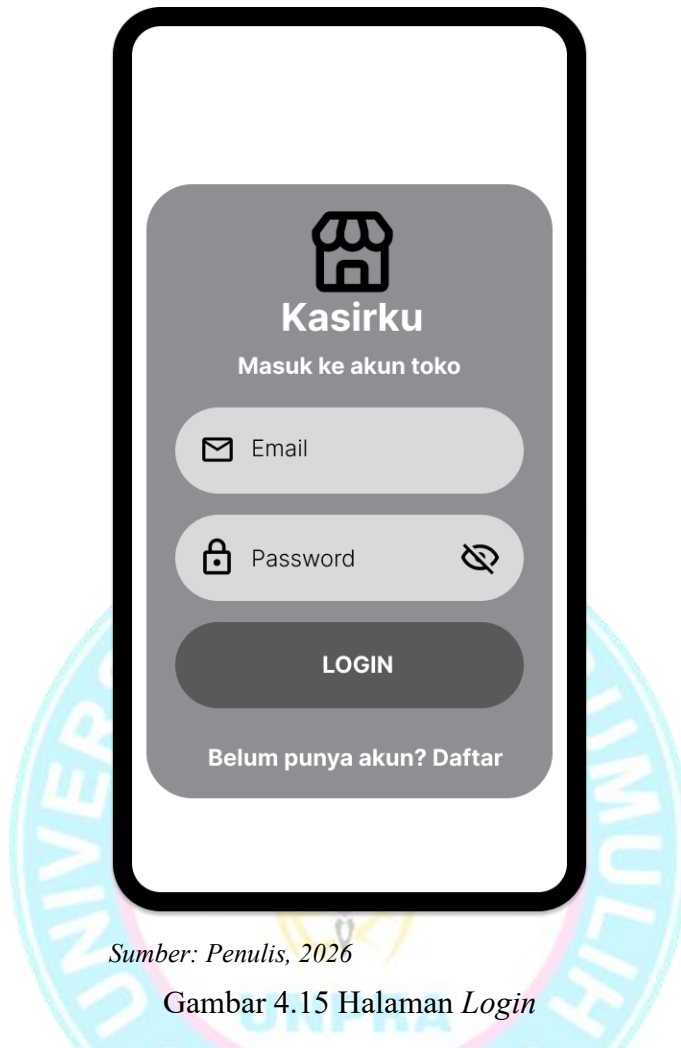
Berdasarkan tampilan antarmuka pada gambar di atas, halaman daftar toko merupakan halaman yang digunakan oleh pengguna untuk melakukan pendaftaran akun admin pada aplikasi kasir berbasis *Android*. Halaman ini berfungsi sebagai sarana input data identitas toko agar pengguna dapat memperoleh akses ke dalam

sistem aplikasi. Pada tampilan tersebut terdapat beberapa kolom isian data yang harus dilengkapi oleh pengguna, yaitu nama toko, alamat, no telepon, *email*, *password*, dan konfirmasi *password*.

Kolom *nama toko* digunakan untuk memasukkan identitas usaha yang akan dikelola di dalam aplikasi. Kolom *alamat* digunakan untuk menyimpan lokasi usaha pengguna. Kolom *no telepon* digunakan sebagai data kontak pengguna. Kolom *email* digunakan sebagai identitas akun saat proses *login* ke aplikasi. Kolom *password* dan konfirmasi *password* digunakan sebagai keamanan akun agar akses sistem tetap aman dari pihak yang tidak bertanggung jawab. Pada bagian kolom *password* juga tersedia ikon mata yang berfungsi untuk menampilkan atau menyembunyikan karakter *password* yang diinputkan pengguna.

Pada bagian bawah kolom *password* terdapat informasi ketentuan *password*, yaitu minimal terdiri dari 8 karakter yang mengandung huruf besar, huruf kecil, angka, dan simbol. Setelah seluruh data terisi dengan lengkap, pengguna dapat menekan tombol “Daftar” untuk melakukan proses pendaftaran akun. Sistem akan melakukan validasi terhadap seluruh data yang dimasukkan pengguna. Jika data telah sesuai, maka akun akan berhasil dibuat dan pengguna dapat melanjutkan proses *login* ke dalam aplikasi. Pada bagian bawah halaman juga terdapat navigasi “Sudah punya akun? Login” yang digunakan untuk mengarahkan pengguna menuju halaman *login* apabila telah memiliki akun sebelumnya.

b. Halaman *Login*



Sumber: Penulis, 2026

Gambar 4.15 Halaman *Login*

Berdasarkan tampilan antarmuka pada gambar di atas, halaman *login* toko merupakan halaman autentikasi yang digunakan oleh administrator dan kasir untuk masuk ke dalam aplikasi kasir berbasis *Android*. Halaman ini menjadi akses utama sebelum pengguna dapat menggunakan fitur-fitur yang tersedia di dalam sistem. Pada bagian atas halaman terdapat ikon toko, nama aplikasi “Kasirku”, serta keterangan “Masuk ke akun toko” sebagai identitas aplikasi.

Pada halaman ini terdapat *form login* yang terdiri dari kolom *Email* dan *Password*. Kolom *email* digunakan untuk memasukkan alamat *email* toko yang telah terdaftar pada sistem. Dalam implementasinya, administrator dan kasir menggunakan *email* toko yang sama untuk proses *login*. Perbedaan hak akses ditentukan berdasarkan data akun dan *password* yang digunakan. Akun administrator diperoleh melalui proses pendaftaran (*register*) pada aplikasi,

sedangkan akun kasir dibuat dan dikelola langsung oleh administrator melalui fitur manajemen pengguna yang tersedia pada sistem. Kolom *password* digunakan untuk memasukkan kata sandi sesuai dengan akun yang dimiliki. Setiap kasir memiliki *password* yang berbeda meskipun menggunakan *email* toko yang sama. Pada bagian kolom *password* terdapat ikon mata yang berfungsi untuk menampilkan atau menyembunyikan karakter *password* sehingga pengguna dapat memastikan data yang dimasukkan sudah benar.

Setelah *email* dan *password* diisi dengan benar, pengguna dapat menekan tombol “Login” untuk masuk ke dalam aplikasi. Sistem akan melakukan proses validasi terhadap data *login* yang dimasukkan. Jika data sesuai dengan data yang tersimpan di dalam basis data, maka sistem akan mengidentifikasi jenis akun yang digunakan. Apabila yang *login* adalah administrator, pengguna akan diarahkan ke halaman utama administrator yang memiliki akses penuh terhadap pengelolaan toko, data kasir, barang, layanan, dan laporan. Apabila yang *login* adalah kasir, pengguna akan diarahkan ke halaman utama kasir yang hanya memiliki akses sesuai dengan hak yang diberikan oleh administrator.

Pada bagian bawah halaman terdapat tautan “Belum punya akun? Daftar” yang digunakan untuk mengarahkan pengguna ke halaman pendaftaran akun. Fitur pendaftaran ini hanya digunakan untuk membuat akun administrator toko. Sementara itu, akun kasir tidak dapat melakukan pendaftaran secara mandiri karena harus dibuat terlebih dahulu oleh administrator melalui sistem manajemen pengguna. Dengan mekanisme tersebut, keamanan dan pengelolaan akses pengguna dalam aplikasi dapat dilakukan secara lebih terstruktur dan terkontrol.

c. Halaman *Dashboard Admin*



Sumber: Penulis, 2026

Gambar 4.16 Halaman *Dashboard Admin*

Berdasarkan tampilan antarmuka pada gambar di atas, halaman beranda administrator merupakan halaman utama yang tampil setelah admin berhasil melakukan proses *login* ke dalam aplikasi kasir berbasis *Android*. Halaman ini berfungsi sebagai pusat informasi dan navigasi utama untuk membantu administrator memantau aktivitas penjualan serta mengakses berbagai fitur yang tersedia di dalam sistem. Pada bagian atas halaman terdapat nama toko “Apan Cell”, informasi hak akses sebagai administrator, serta ikon keranjang yang digunakan untuk mengakses menu transaksi penjualan.

Pada bagian tengah halaman terdapat informasi jumlah produk dan jumlah transaksi hari ini dalam bentuk kartu informasi. Informasi jumlah produk digunakan untuk menampilkan total data barang yang tersimpan di dalam sistem, sedangkan informasi transaksi hari ini digunakan untuk menampilkan jumlah transaksi yang

telah dilakukan pada hari tersebut. Data tersebut membantu administrator dalam memantau aktivitas operasional toko secara lebih mudah.

Pada halaman ini juga terdapat grafik penjualan hari ini yang digunakan untuk menampilkan visualisasi data transaksi berdasarkan waktu penjualan pada hari berjalan. Grafik tersebut membantu administrator dalam melihat perkembangan aktivitas penjualan toko secara lebih cepat dan terstruktur.

Pada bagian bawah halaman terdapat menu navigasi utama aplikasi yang terdiri dari menu beranda, stok, laporan dan setting. Menu beranda digunakan untuk kembali ke halaman utama aplikasi. Menu stok digunakan untuk mengelola data barang dan persediaan produk. Menu laporan digunakan untuk melihat data transaksi dan laporan penjualan. Menu setting digunakan untuk mengelola informasi akun serta pengaturan aplikasi.



d. Halaman Pilihan Transaksi Admin



Sumber: Penulis, 2026

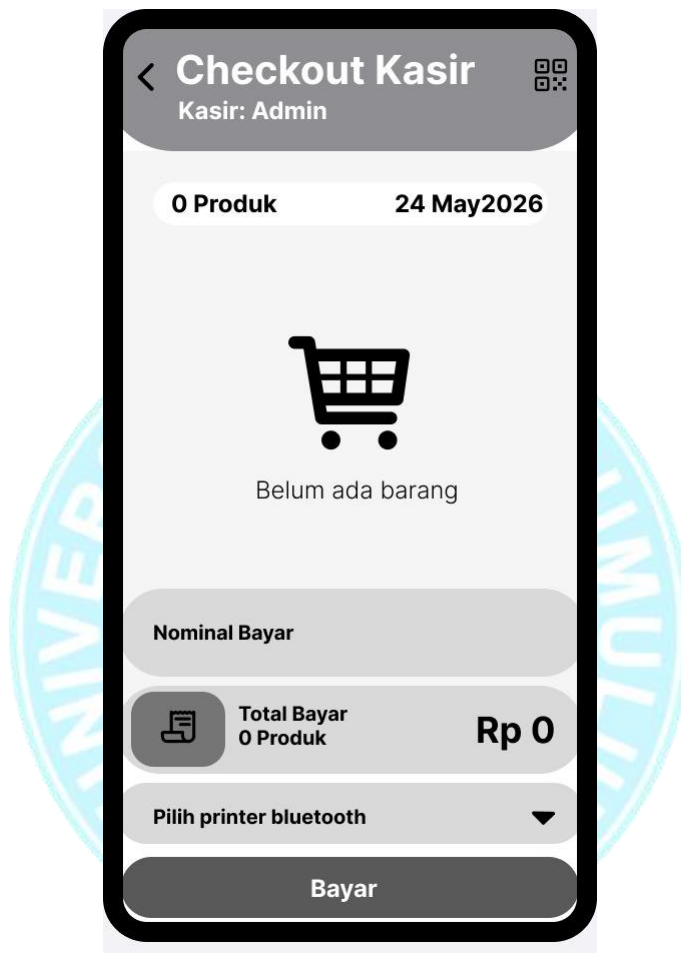
Gambar 4.17 Halaman Pilihan Transaksi Admin

Berdasarkan tampilan antarmuka pada gambar di atas, halaman pilihan transaksi administrator merupakan halaman yang digunakan oleh admin untuk memilih jenis transaksi yang akan dilakukan pada aplikasi kasir berbasis *Android*. Halaman ini ditampilkan ketika administrator menekan ikon keranjang pada halaman beranda aplikasi. Pada bagian atas halaman tetap ditampilkan nama toko “Apan Cell”, informasi hak akses sebagai administrator, jumlah produk, jumlah transaksi hari ini, serta grafik penjualan hari ini.

Pada bagian bawah halaman terdapat menu “Pilih transaksi” yang berisi dua pilihan transaksi, yaitu barang dan layanan. Menu “Barang” digunakan untuk melakukan transaksi penjualan produk yang tersedia pada toko, seperti barang atau persediaan produk yang dijual. Menu “Layanan” digunakan untuk melakukan transaksi jasa atau layanan yang tersedia pada toko. Halaman transaksi

administrator ini membantu admin dalam memilih jenis transaksi yang akan diproses sehingga kegiatan transaksi penjualan barang maupun layanan dapat dilakukan secara lebih mudah dan terstruktur.

e. Halaman *Checkout* Barang Admin



Sumber: Penulis, 2026

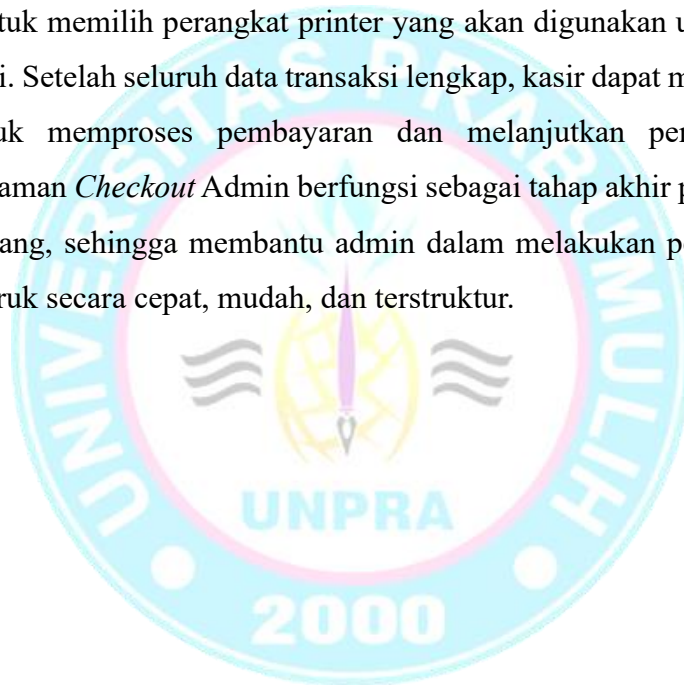
Gambar 4.18 Halaman *Checkout* Barang Admin

Berdasarkan tampilan antarmuka pada gambar di atas, halaman *checkout* admin merupakan halaman yang digunakan oleh administrator untuk melakukan proses transaksi penjualan barang pada aplikasi kasir berbasis *Android*. Halaman ini akan ditampilkan setelah admin memilih menu “Barang” pada halaman transaksi. Pada bagian atas halaman terdapat judul “*Checkout Kasir*”, informasi pengguna yang sedang *login* sebagai Admin, tombol kembali ke halaman

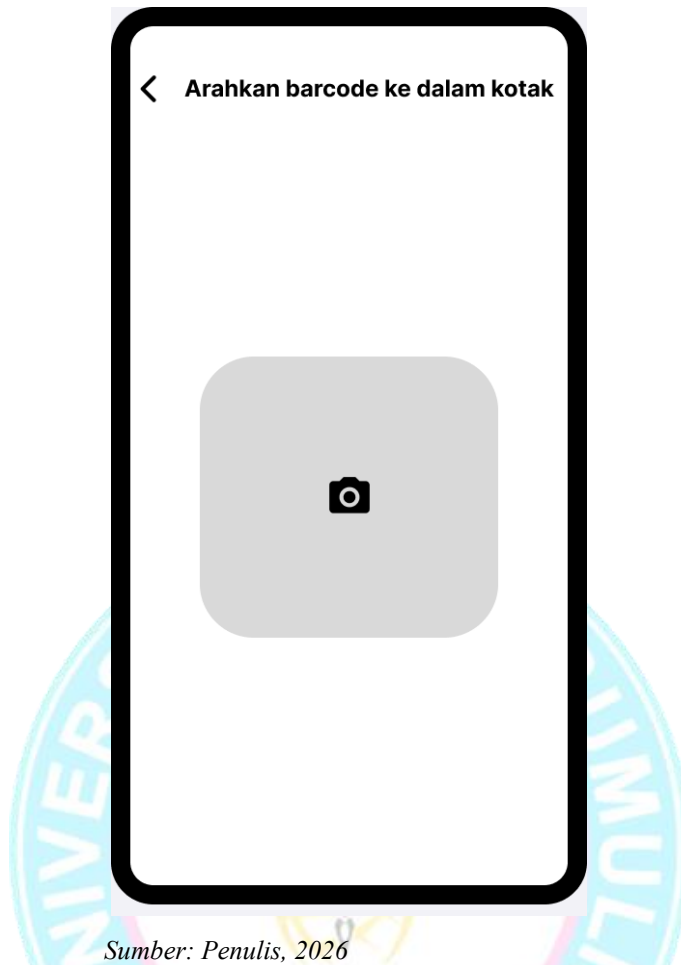
sebelumnya, serta ikon *scan barcode* yang digunakan untuk masuk ke halaman scan barang.

Pada halaman ini ditampilkan informasi jumlah produk yang dipilih dalam transaksi serta tanggal transaksi yang sedang berlangsung. Apabila belum terdapat barang yang dipilih, maka sistem akan menampilkan informasi “Belum ada barang”. Pada bagian bawah halaman tersedia kolom “Input Nominal” yang digunakan untuk memasukkan jumlah uang yang dibayarkan oleh pelanggan. Selanjutnya terdapat panel “Total Bayar” yang menampilkan jumlah produk yang dibeli serta total harga transaksi yang harus dibayarkan.

Halaman ini juga menyediakan fitur “Pilih printer *bluetooth*” yang digunakan untuk memilih perangkat printer yang akan digunakan untuk mencetak struk transaksi. Setelah seluruh data transaksi lengkap, kasir dapat menekan tombol “Bayar” untuk memproses pembayaran dan melanjutkan pencetakan struk transaksi. Halaman *Checkout Admin* berfungsi sebagai tahap akhir proses transaksi penjualan barang, sehingga membantu admin dalam melakukan pembayaran dan pencetakan struk secara cepat, mudah, dan terstruktur.



f. Halaman *Scan* Barang Admin



Sumber: Penulis, 2026

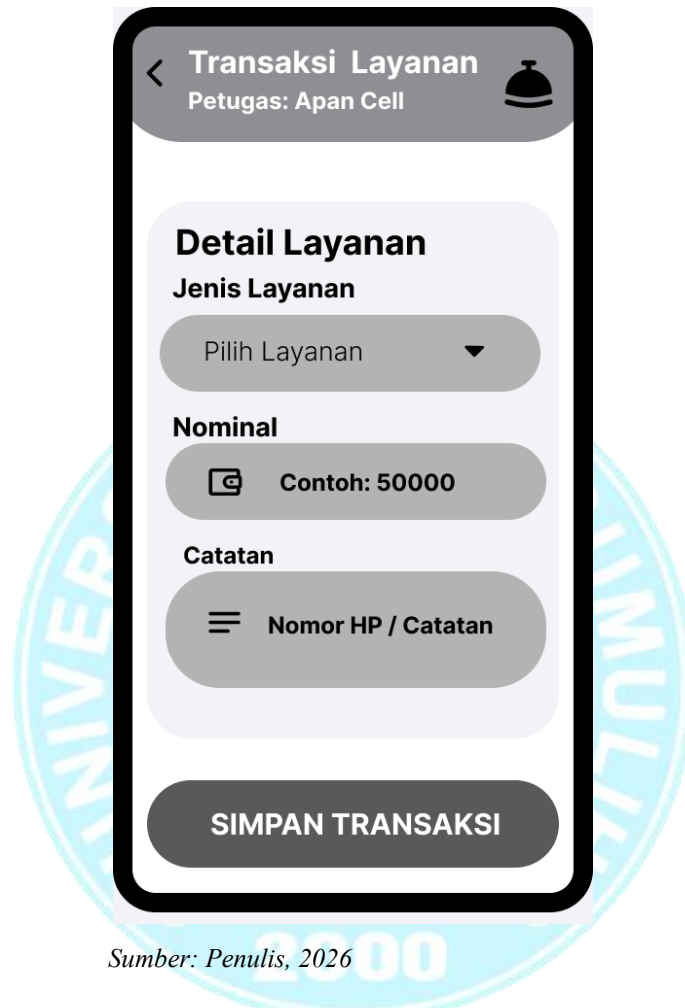
Gambar 4.19 Halaman *Scan* Barang Admin

Berdasarkan tampilan antarmuka pada gambar di atas, halaman *scan* barang merupakan halaman yang digunakan untuk melakukan proses pemindaian *barcode* produk pada aplikasi kasir berbasis *Android*. Halaman ini akan ditampilkan setelah pengguna menekan ikon *scan barcode* pada halaman *checkout* kasir. Fitur ini digunakan untuk membantu kasir memasukkan data barang ke dalam transaksi secara otomatis menggunakan kamera perangkat.

Pada bagian atas halaman terdapat informasi “Arahkan *barcode* ke dalam kotak” sebagai petunjuk penggunaan fitur pemindaian *barcode*. Di bagian tengah halaman terdapat area pemindaian yang digunakan untuk mendeteksi dan membaca *barcode* barang yang diarahkan ke kamera perangkat. Halaman *scan* barang ini membantu proses transaksi penjualan menjadi lebih cepat, praktis, dan akurat

karena data barang dapat dimasukkan secara otomatis tanpa perlu melakukan *input* data secara manual.

g. Halaman Transaksi Layanan Admin



Sumber: Penulis, 2026

Gambar 4.20 Halaman Transaksi Layanan Admin

Berdasarkan tampilan antarmuka pada gambar di atas, halaman transaksi layanan merupakan halaman yang digunakan oleh administrator untuk melakukan transaksi layanan atau jasa pada aplikasi kasir berbasis *Android*. Halaman ini akan ditampilkan setelah administrator memilih menu “Layanan” pada halaman transaksi administrator. Pada bagian atas halaman terdapat judul “Transaksi Layanan”, informasi nama petugas, serta ikon keranjang yang digunakan untuk melihat data transaksi layanan yang telah dipilih.

Pada halaman ini tersedia beberapa *field input*, yaitu jenis layanan, nominal, dan catatan. *Field* jenis layanan digunakan untuk memilih layanan yang akan

diproses, sedangkan *field* nominal digunakan untuk memasukkan jumlah pembayaran transaksi layanan. *Field* catatan digunakan untuk menambahkan informasi tambahan, seperti nomor *handphone* atau keterangan lain yang berkaitan dengan transaksi layanan. Pada bagian bawah halaman terdapat tombol “Simpan transaksi” yang digunakan untuk menyimpan data transaksi layanan yang telah diinput. Halaman ini membantu administrator dalam melakukan proses transaksi layanan secara lebih mudah, cepat, dan terstruktur.

h. Halaman Daftar Barang



Sumber: Penulis, 2026

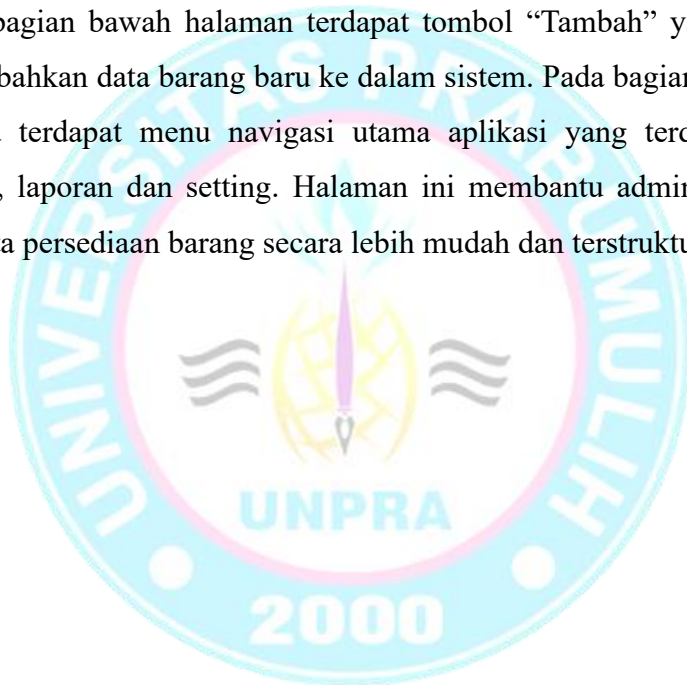
Gambar 4.21 Halaman Daftar Barang

Berdasarkan tampilan antarmuka pada gambar di atas, halaman daftar barang merupakan halaman yang digunakan oleh administrator untuk melihat dan mengelola data produk pada aplikasi kasir berbasis *Android*. Halaman ini menampilkan seluruh data barang yang telah tersimpan di dalam sistem sehingga

administrator dapat memantau informasi persediaan barang dengan lebih mudah. Pada bagian atas halaman terdapat judul “Daftar barang”, keterangan “Data produk”, serta ikon keranjang yang digunakan untuk mengakses menu transaksi.

Pada halaman ini ditampilkan beberapa informasi barang, seperti nama barang, harga beli, harga jual, jumlah stok, dan status stok barang. Informasi tersebut digunakan untuk membantu administrator mengetahui kondisi persediaan barang yang tersedia pada toko. Setiap data barang juga dilengkapi dengan ikon *barcode*, ikon edit, dan ikon hapus. Ikon *barcode* digunakan untuk melihat atau mencetak *barcode* barang, ikon edit digunakan untuk mengubah data barang, sedangkan ikon hapus digunakan untuk menghapus data barang dari sistem.

Pada bagian bawah halaman terdapat tombol “Tambah” yang digunakan untuk menambahkan data barang baru ke dalam sistem. Pada bagian paling bawah halaman juga terdapat menu navigasi utama aplikasi yang terdiri dari menu beranda, stok, laporan dan setting. Halaman ini membantu administrator dalam mengelola data persediaan barang secara lebih mudah dan terstruktur.



i. Halaman Tambah Data Barang

Sumber: Penulis, 2026

Gambar 4.22 Halaman Tambah Barang

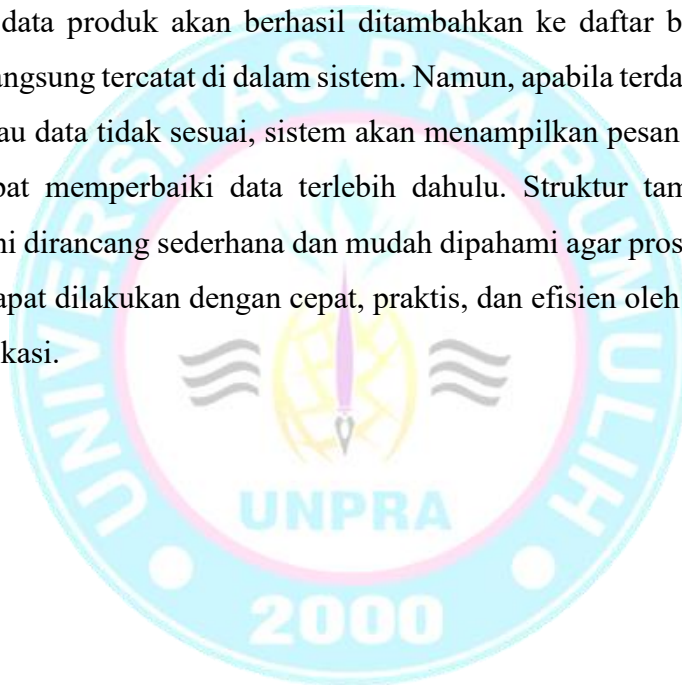
Berdasarkan tampilan antarmuka pada gambar di atas, halaman produk toko merupakan halaman yang digunakan untuk menambahkan data barang baru ke dalam aplikasi kasir berbasis *Android*. Halaman ini berfungsi sebagai sarana input data produk agar seluruh informasi barang dapat tersimpan ke dalam sistem dan digunakan dalam proses pengelolaan stok maupun transaksi penjualan.

Pada halaman ini terdapat beberapa kolom *input* data yang harus diisi oleh pengguna. Kolom nama barang digunakan untuk memasukkan nama produk yang akan dijual di toko. Kolom harga beli digunakan untuk mencatat harga modal barang dari *supplier* atau *distributor*. Kolom harga jual digunakan untuk memasukkan harga penjualan produk kepada pelanggan. Kolom jumlah stok

digunakan untuk menentukan jumlah persediaan awal barang yang tersedia di dalam toko.

Bagian *printer* digunakan untuk memilih perangkat *printer* yang akan digunakan dalam proses pencetakan *barcode* produk. Fitur ini membantu pengguna dalam mencetak *barcode* secara langsung setelah data barang berhasil disimpan ke dalam sistem. Pada bagian tersebut tersedia pilihan *printer* yang dapat dipilih sesuai perangkat yang terhubung dengan aplikasi.

Tombol “Simpan Data” digunakan untuk menyimpan seluruh data barang yang telah diinputkan pengguna ke dalam *database* aplikasi. Sistem akan melakukan validasi terhadap seluruh data yang dimasukkan. Jika seluruh data telah sesuai, maka data produk akan berhasil ditambahkan ke daftar barang dan stok barang akan langsung tercatat di dalam sistem. Namun, apabila terdapat kolom yang belum diisi atau data tidak sesuai, sistem akan menampilkan pesan kesalahan agar pengguna dapat memperbaiki data terlebih dahulu. Struktur tampilan halaman produk toko ini dirancang sederhana dan mudah dipahami agar proses penambahan data barang dapat dilakukan dengan cepat, praktis, dan efisien oleh admin maupun pengguna aplikasi.



j. Halaman Edit Barang

Sumber: Penulis, 2026

Gambar 4.23 Halaman Edit Barang

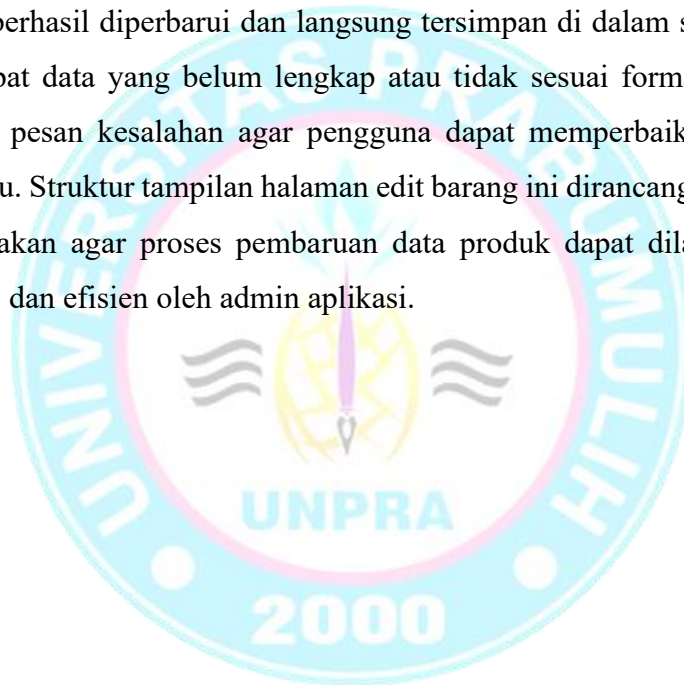
Berdasarkan tampilan antarmuka pada gambar di atas, halaman edit barang merupakan halaman yang digunakan untuk mengubah atau memperbarui data produk yang telah tersimpan di dalam aplikasi kasir berbasis *Android*. Halaman ini membantu admin dalam melakukan pengelolaan data barang apabila terdapat perubahan informasi produk, seperti perubahan harga maupun jumlah stok barang.

Pada halaman ini terdapat beberapa kolom input yang telah terisi otomatis berdasarkan data barang sebelumnya. Kolom nama barang digunakan untuk menampilkan dan mengubah nama produk yang tersimpan di dalam sistem. Kolom harga beli digunakan untuk memperbarui harga modal barang dari *supplier*. Kolom harga jual digunakan untuk mengubah harga penjualan produk kepada pelanggan.

Kolom jumlah stok digunakan untuk memperbarui jumlah persediaan barang yang tersedia di toko.

Bagian *printer* digunakan untuk memilih perangkat *printer* yang akan dipakai dalam proses pencetakan *barcode* produk. Fitur ini membantu pengguna dalam mencetak ulang *barcode* setelah data barang diperbarui. Pada bagian tersebut tersedia pilihan *printer* yang dapat dipilih sesuai perangkat yang terhubung dengan aplikasi.

Tombol “Simpan Data” digunakan untuk menyimpan seluruh perubahan data barang ke dalam *database* aplikasi. Sistem akan melakukan validasi terhadap data yang dimasukkan pengguna. Jika data telah sesuai, maka perubahan data barang akan berhasil diperbarui dan langsung tersimpan di dalam sistem. Namun, apabila terdapat data yang belum lengkap atau tidak sesuai format, sistem akan menampilkan pesan kesalahan agar pengguna dapat memperbaiki data tersebut terlebih dahulu. Struktur tampilan halaman edit barang ini dirancang sederhana dan mudah digunakan agar proses pembaruan data produk dapat dilakukan dengan cepat, praktis, dan efisien oleh admin aplikasi.



k. Halaman Laporan Barang



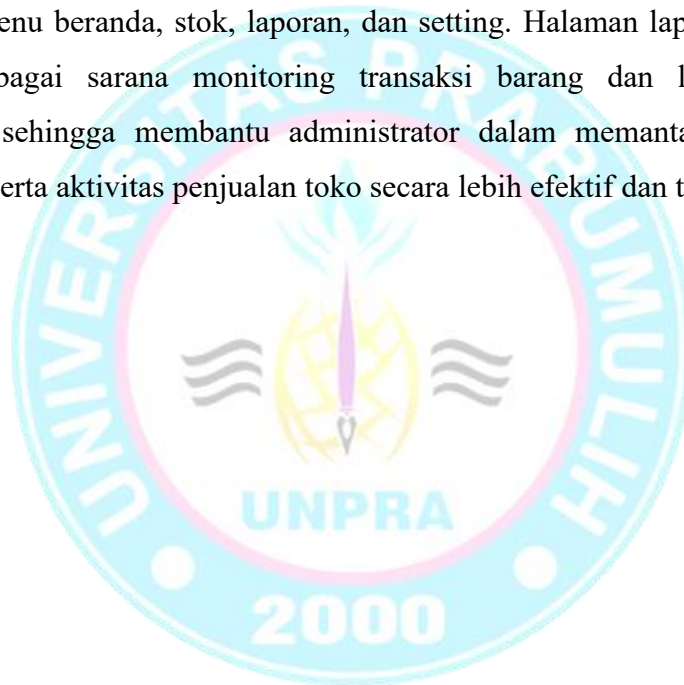
Sumber: Penulis, 2026

Gambar 4.24 Halaman Laporan Transaksi

Berdasarkan tampilan antarmuka pada gambar di atas, halaman laporan penjualan merupakan halaman yang digunakan oleh administrator untuk melihat dan memantau data transaksi barang dan layanan pada aplikasi kasir berbasis *Android*. Halaman ini menampilkan ringkasan serta detail transaksi yang telah dilakukan dalam periode tertentu sehingga memudahkan administrator dalam mengevaluasi hasil penjualan toko. Pada bagian atas halaman terdapat judul “Laporan Penjualan” beserta keterangan “Data transaksi barang & layanan”. Di sisi kanan atas terdapat tombol cetak PDF yang digunakan untuk menyimpan atau mencetak laporan penjualan dalam format PDF, serta ikon kalender yang digunakan untuk memilih tanggal laporan. Di bawahnya terdapat pilihan periode laporan, seperti harian, mingguan dan bulanan, yang berfungsi untuk menentukan rentang waktu data yang akan ditampilkan.

Bagian laporan barang menampilkan ringkasan hasil transaksi penjualan barang yang terdiri dari informasi pendapatan dan keuntungan. Di bawah ringkasan tersebut ditampilkan detail transaksi barang yang telah dilakukan, meliputi nama barang yang terjual, nama kasir yang melakukan transaksi, nilai transaksi, serta waktu transaksi. Selanjutnya terdapat bagian laporan layanan yang menampilkan data transaksi layanan atau jasa yang tersedia pada toko. Pada bagian ini ditampilkan total Pendapatan dari layanan yang telah terjual. Di bawahnya terdapat rincian transaksi layanan yang meliputi nama layanan, nama kasir yang melayani transaksi, nilai transaksi, serta waktu transaksi.

Pada bagian paling bawah halaman terdapat menu navigasi utama aplikasi yang terdiri dari menu beranda, stok, laporan, dan setting. Halaman laporan penjualan berfungsi sebagai sarana monitoring transaksi barang dan layanan secara menyeluruh, sehingga membantu administrator dalam memantau pendapatan, keuntungan, serta aktivitas penjualan toko secara lebih efektif dan terstruktur.



1. Halaman Cetak PDF



Laporan Penjualan
Data Transaksi Barang dan Layanan

Jenis Laporan: Bulanan
Tanggal: 27 May 2026

LAPORAN TRANSAKSI BARANG

Keterangan	Jumlah
Total Pendapatan Barang	Rp 555.000
Total Profit Barang	Rp 100.000

Barang	Kasir	Tanggal	Total	Profit
Charger Redmi	Sinta	25/05/2026 22:23	Rp 50.000	Rp 20.000
Type C	Sinta	25/05/2026 23:23	Rp 10.000	Rp 5.000

LAPORAN TRANSAKSI LAYANAN

Keterangan	Jumlah
Total Pendapatan Layanan	Rp 555.000

Layanan	Kasir	Tanggal	Total
Top up Gopay	Sinta	25/05/2026 22:23	Rp 50.000
Top up Dana	Sinta	27/05/2026 11:23	Rp 10.000

Dicetak Pada
7 June 2026 14:22

Sumber: Penulis, 2026

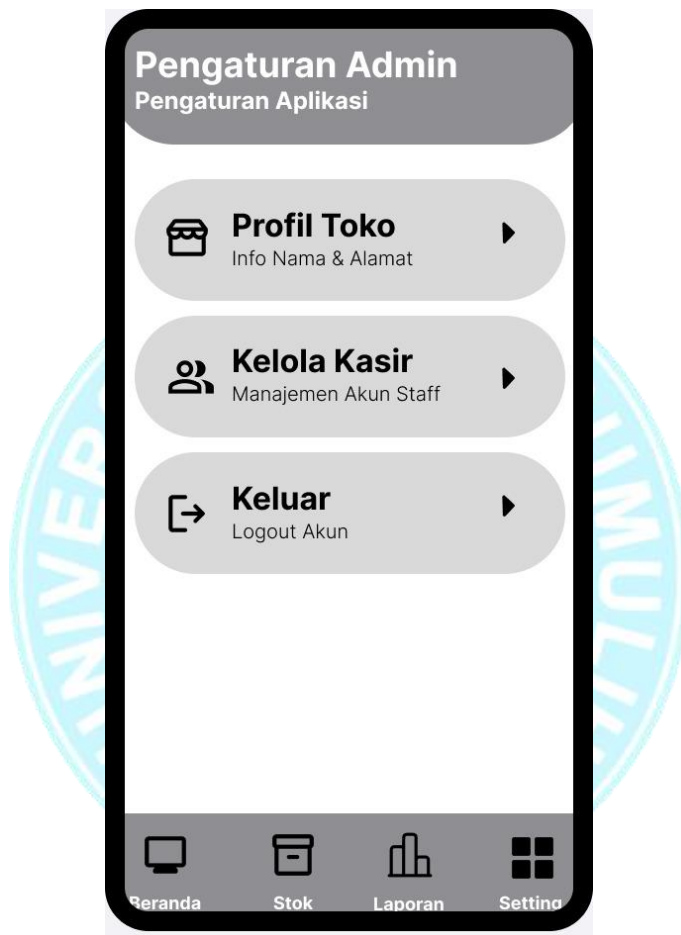
Gambar 4.25 Halaman Cetak PDF

Berdasarkan tampilan antarmuka pada gambar di atas, halaman laporan penjualan merupakan halaman yang digunakan untuk menampilkan data transaksi barang dan layanan pada aplikasi kasir berbasis *Android*. Pada bagian atas halaman terdapat judul “Laporan Penjualan” serta informasi jenis laporan dan tanggal laporan yang menunjukkan periode data yang ditampilkan.

Halaman ini terdiri dari dua bagian utama, yaitu Laporan Transaksi Barang dan Laporan Transaksi Layanan. Pada laporan transaksi barang ditampilkan informasi total pendapatan, total profit, serta tabel rincian transaksi yang memuat data barang, kasir, tanggal transaksi, total penjualan, dan profit. Pada laporan transaksi layanan ditampilkan total pendapatan layanan beserta tabel rincian transaksi yang berisi data layanan, kasir, tanggal transaksi, dan total pendapatan.

Pada bagian bawah halaman terdapat informasi waktu pencetakan laporan yang menunjukkan tanggal dan jam laporan dibuat. Halaman ini membantu pengguna dalam melihat dan memantau hasil transaksi barang maupun layanan secara lebih terstruktur dan mudah dipahami.

m. Halaman Pengaturan Admin



Sumber: Penulis, 2026

Gambar 4.26 Halaman Pengaturan

Berdasarkan tampilan antarmuka pada gambar di atas, halaman pengaturan admin merupakan halaman yang digunakan oleh administrator untuk mengelola pengaturan aplikasi pada aplikasi kasir berbasis *Android*. Halaman ini berfungsi untuk mengatur informasi toko, data akun kasir, serta akses keluar dari aplikasi. Pada bagian atas halaman terdapat judul “Pengaturan admin” serta keterangan “Pengaturan aplikasi” sebagai informasi mengenai fungsi halaman tersebut.

Pada halaman ini terdapat beberapa menu pengaturan yang dapat digunakan oleh administrator. Menu profil toko digunakan untuk melihat dan mengelola informasi toko, seperti nama toko dan alamat toko. Menu kelola kasir digunakan untuk mengatur dan mengelola akun staf kasir yang dapat mengakses aplikasi. Pada bagian bawah menu pengaturan terdapat menu keluar yang digunakan untuk melakukan proses *logout* dari akun administrator yang sedang digunakan. Pada bagian paling bawah halaman juga terdapat menu navigasi utama aplikasi yang terdiri dari menu beranda, stok, laporan, dan pengaturan. Halaman ini membantu administrator dalam mengelola pengaturan aplikasi serta data pengguna secara lebih mudah dan terstruktur.



n. Halaman Profil Toko



Sumber: Penulis, 2026

Gambar 4.27 Halaman Profil Toko

Berdasarkan tampilan antarmuka pada gambar di atas, halaman profil toko merupakan bagian dari menu pengaturan admin yang digunakan untuk menampilkan informasi identitas toko yang terdaftar pada aplikasi kasir berbasis *Android*. Halaman ini membantu admin dalam melihat data profil usaha secara lengkap dan terstruktur.

Pada halaman ini ditampilkan beberapa informasi penting mengenai toko, seperti nama toko, alamat, dan telepon. Informasi nama toko digunakan sebagai identitas utama usaha yang dikelola di dalam aplikasi. Informasi alamat digunakan untuk menunjukkan lokasi usaha secara lengkap. Informasi nomor telepon digunakan sebagai data kontak yang dapat dihubungi oleh pelanggan maupun pihak

terkait lainnya. Bagian informasi profil ditampilkan dalam bentuk panel khusus agar data toko dapat dibaca dengan lebih jelas dan mudah dipahami oleh pengguna. Setiap informasi juga dilengkapi dengan ikon yang membantu memperjelas fungsi dari data yang ditampilkan pada halaman tersebut.

Tombol “Edit Profil” digunakan untuk memperbarui informasi profil toko yang telah tersimpan di dalam sistem. Melalui tombol ini admin dapat mengubah data seperti nama toko, alamat, maupun nomor telepon apabila terjadi perubahan informasi usaha. Sistem akan menyimpan perubahan data secara otomatis setelah proses pembaruan berhasil dilakukan. Halaman profil toko ini dirancang agar admin dapat mengelola identitas usaha dengan lebih mudah, cepat, dan efisien sehingga informasi toko yang digunakan pada aplikasi selalu sesuai dengan kondisi usaha yang sebenarnya.



o. Halaman Edit Profil Toko



Sumber: Penulis, 2026

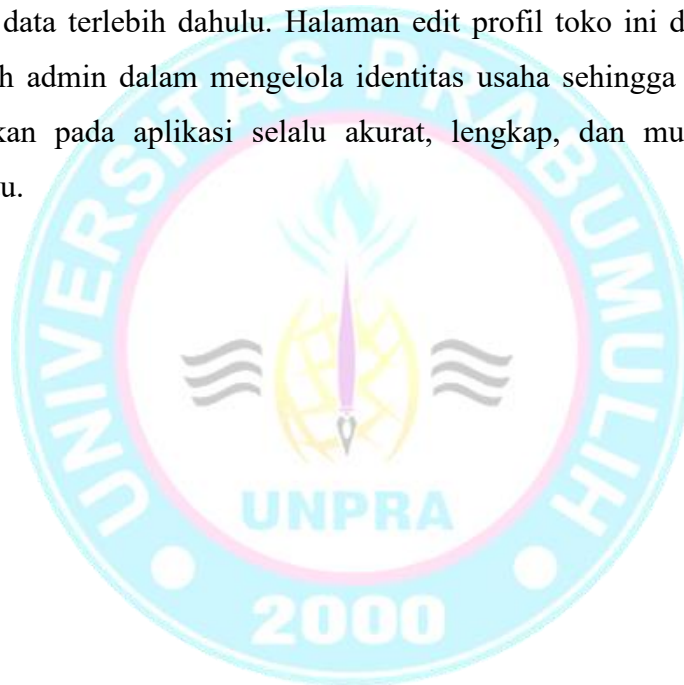
Gambar 4.28 Halaman Edit Profil Toko

Berdasarkan tampilan antarmuka pada gambar di atas, halaman edit profil toko merupakan halaman yang digunakan untuk memperbarui informasi identitas toko pada aplikasi kasir berbasis *Android*. Halaman ini membantu admin dalam mengubah data profil usaha agar informasi yang tersimpan di dalam sistem tetap sesuai dengan kondisi toko yang sebenarnya.

Pada halaman ini ditampilkan beberapa data profil toko yang dapat diperbarui oleh pengguna, meliputi nama toko, alamat, dan telepon. kolom nama toko digunakan untuk menampilkan dan mengubah identitas usaha yang dikelola di dalam aplikasi. Kolom alamat digunakan untuk memperbarui lokasi atau alamat lengkap toko. Kolom telepon digunakan untuk mengubah nomor kontak toko yang dapat digunakan dalam proses komunikasi maupun identifikasi usaha. Setiap

informasi profil juga dilengkapi dengan ikon pendukung yang membantu pengguna memahami fungsi dari masing-masing data yang ditampilkan pada halaman tersebut. Struktur tampilan halaman dibuat sederhana dan mudah dipahami agar proses pengelolaan data toko dapat dilakukan dengan lebih cepat dan praktis.

Tombol “Simpan” digunakan untuk menyimpan seluruh perubahan data profil toko ke dalam *database* aplikasi. Sistem akan melakukan validasi terhadap data yang dimasukkan pengguna sebelum proses penyimpanan dilakukan. Jika data telah sesuai, maka perubahan profil toko akan berhasil diperbarui secara otomatis di dalam sistem. Namun, apabila terdapat data yang belum lengkap atau tidak sesuai, sistem akan menampilkan pesan kesalahan agar pengguna dapat memperbaiki data terlebih dahulu. Halaman edit profil toko ini dirancang untuk mempermudah admin dalam mengelola identitas usaha sehingga informasi toko yang digunakan pada aplikasi selalu akurat, lengkap, dan mudah diperbarui sewaktu-waktu.



p. Halaman Manajemen Kasir



Sumber: Penulis, 2026

Gambar 4.29 Halaman Manajemen Kasir

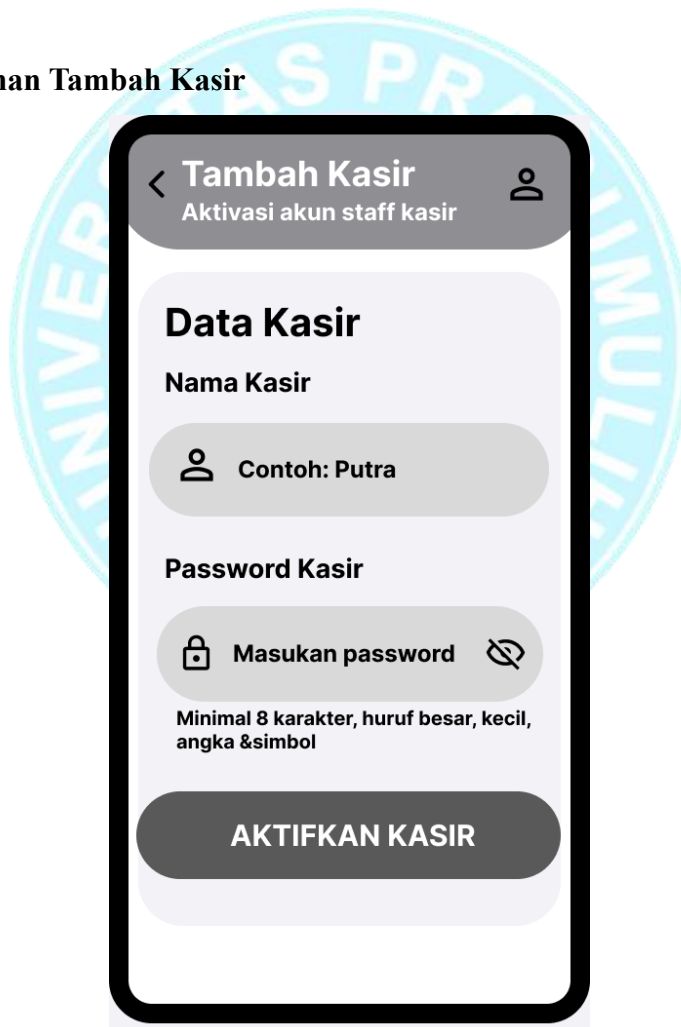
Berdasarkan tampilan antarmuka pada gambar di atas, halaman manajemen kasir merupakan halaman yang digunakan untuk mengelola seluruh data akun kasir pada aplikasi kasir berbasis *Android*. Halaman ini membantu admin dalam mengatur akun staf yang memiliki akses untuk melakukan transaksi penjualan di dalam sistem.

Pada halaman ini ditampilkan contoh daftar nama kasir yang telah terdaftar pada aplikasi, seperti Anggi Saputra, Yuki, Putra, dan Sinta. Setiap data kasir ditampilkan dalam bentuk kartu informasi agar mudah dibaca dan dikelola oleh admin. Informasi nama kasir digunakan sebagai identitas pengguna yang bertugas dalam proses transaksi penjualan di toko. Setiap data kasir juga dilengkapi dengan beberapa tombol aksi yang memiliki fungsi berbeda. Ikon pensil digunakan untuk mengubah atau memperbarui data akun kasir yang telah tersimpan di dalam sistem.

Ikon tempat sampah digunakan untuk menghapus akun kasir dari *database* aplikasi apabila akun tersebut sudah tidak digunakan lagi.

Tombol “Tambah Data” yang terdapat pada bagian bawah halaman digunakan untuk menambahkan akun kasir baru ke dalam sistem. Melalui tombol ini admin dapat membuat akun staf baru beserta data identitas dan hak akses yang diperlukan untuk melakukan transaksi pada aplikasi. Bagian paling bawah halaman terdapat menu navigasi utama aplikasi yang terdiri dari menu beranda, stok, laporan, dan setting. Struktur tampilan halaman manajemen kasir ini dirancang sederhana dan mudah dipahami agar proses pengelolaan akun staf dapat dilakukan dengan lebih cepat, praktis, dan efisien oleh admin toko.

q. Halaman Tambah Kasir



Tambah Kasir
Aktivasi akun staff kasir

Data Kasir

Nama Kasir

Contoh: Putra

Password Kasir

Masukan password

Minimal 8 karakter, huruf besar, kecil, angka & simbol

AKTIFKAN KASIR

Sumber: Penulis, 2026

Gambar 4.30 Halaman Tambah Kasir

Berdasarkan tampilan antarmuka pada gambar di atas, halaman tambah kasir merupakan halaman yang digunakan oleh administrator untuk menambahkan akun staf kasir pada aplikasi kasir berbasis *Android*. Halaman ini berfungsi untuk memberikan akses kepada staf kasir agar dapat menggunakan fitur transaksi yang tersedia di dalam sistem aplikasi. Pada bagian atas halaman terdapat judul “Tambah kasir” serta keterangan “Aktivasi akun staff kasir” sebagai informasi mengenai fungsi halaman tersebut.

Pada halaman ini tersedia beberapa *field input*, yaitu nama kasir dan *password* kasir. *Field* nama kasir digunakan untuk memasukkan identitas staf kasir yang akan ditambahkan ke dalam sistem. *Field password* kasir digunakan sebagai keamanan akun kasir agar akses aplikasi tetap aman. Pada bagian kolom *password* kasir tersedia ikon mata yang berfungsi untuk menampilkan atau menyembunyikan karakter *password* yang dimasukkan administrator.

Pada bagian bawah kolom *password kasir* terdapat informasi ketentuan *password*, yaitu minimal terdiri dari 8 karakter yang mengandung huruf besar, huruf kecil, angka, dan simbol. Setelah seluruh data terisi dengan benar, administrator dapat menekan tombol “Aktifkan kasir” untuk menyimpan dan mengaktifkan akun kasir baru ke dalam sistem aplikasi.

r. Halaman Edit Data Kasir

Sumber: Penulis, 2026

Gambar 4.31 Halaman Edit Data Kasir

Berdasarkan tampilan antarmuka pada gambar di atas, halaman edit kasir merupakan halaman yang digunakan oleh administrator untuk mengubah atau memperbarui data akun kasir pada aplikasi kasir berbasis *Android*. Halaman ini berfungsi untuk mengelola informasi akun kasir yang telah tersimpan di dalam sistem. Pada bagian atas halaman terdapat judul “Edit kasir” serta keterangan “Perbarui data akun kasir” sebagai informasi mengenai fungsi halaman tersebut.

Pada halaman ini tersedia beberapa *field* data, yaitu nama kasir dan *password* kasir. *Field* nama kasir digunakan untuk menampilkan atau mengubah identitas kasir yang terdaftar pada sistem. *Field password* kasir digunakan untuk memperbarui kata sandi akun kasir sebagai bentuk keamanan akses aplikasi. Pada bagian kolom *password* kasir tersedia ikon mata yang berfungsi untuk

menampilkan atau menyembunyikan karakter *password* yang dimasukkan administrator.

Pada bagian bawah kolom *password* kasir terdapat informasi ketentuan *password*, yaitu minimal terdiri dari 8 karakter yang mengandung huruf besar, huruf kecil, angka, dan simbol. Setelah data diperbarui, administrator dapat menekan tombol “Simpan perubahan” untuk menyimpan perubahan data akun kasir ke dalam sistem aplikasi.

s. Halaman *Dashboard* Kasir



Sumber: Penulis, 2026

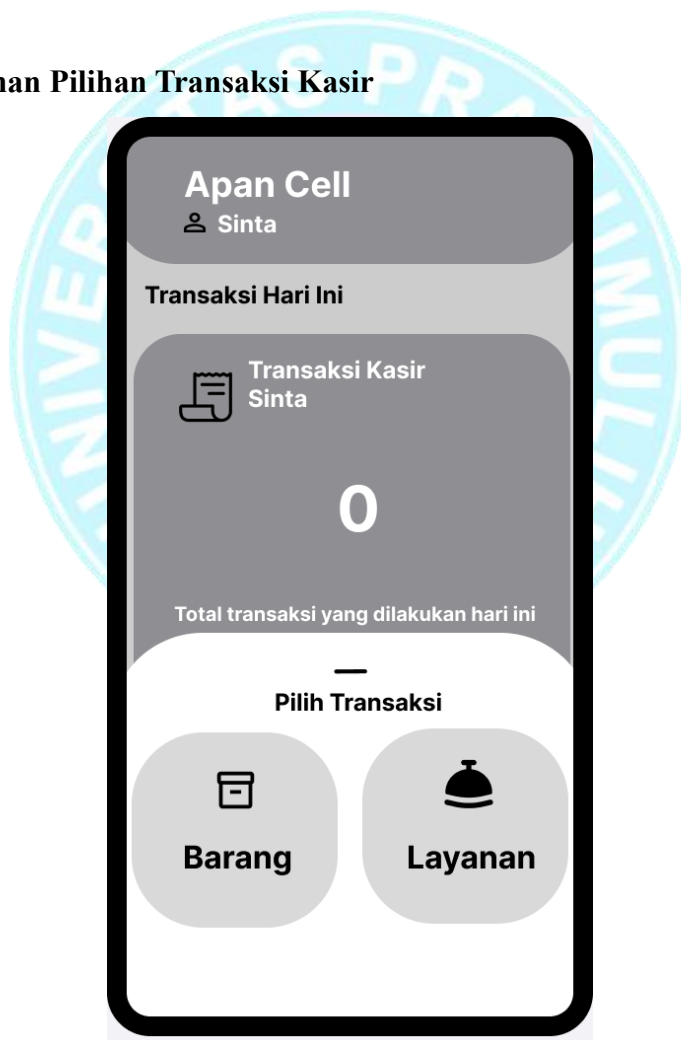
Gambar 4.32 Halaman *Dashboard* Kasir

Berdasarkan tampilan antarmuka pada gambar di atas, halaman *dashboard* kasir merupakan halaman utama yang tampil setelah kasir berhasil melakukan proses *login* ke dalam aplikasi kasir berbasis *Android*. Halaman ini digunakan untuk menampilkan informasi transaksi yang dilakukan oleh kasir pada hari tersebut. Pada

bagian atas halaman terdapat nama toko “Apan Cell” serta informasi nama kasir yang sedang menggunakan aplikasi.

Pada bagian tengah halaman terdapat informasi “Transaksi hari ini” yang menampilkan jumlah total transaksi yang telah dilakukan oleh kasir pada hari berjalan. Informasi tersebut membantu kasir dalam memantau aktivitas transaksi yang sedang berlangsung pada toko. Pada bagian paling bawah halaman terdapat menu navigasi yang terdiri dari menu beranda, transaksi, dan *logout*. Menu beranda digunakan untuk kembali ke halaman utama kasir. Menu transaksi digunakan untuk melakukan proses transaksi penjualan barang maupun layanan. Menu *logout* digunakan untuk keluar dari akun kasir yang sedang digunakan pada aplikasi.

t. Halaman Pilihan Transaksi Kasir



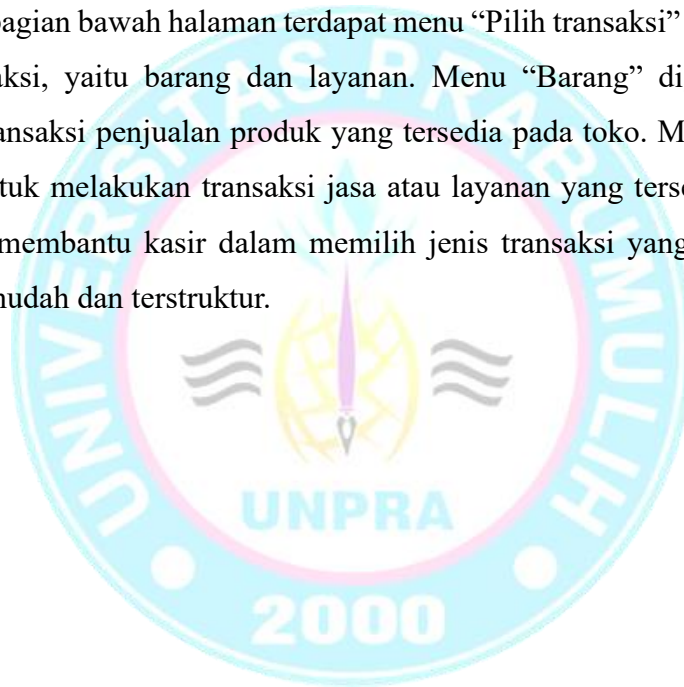
Sumber: Penulis, 2026

Gambar 4.33 Halaman Pilihan Transaksi Kasir

Berdasarkan tampilan antarmuka pada gambar di atas, halaman pilihan transaksi kasir merupakan halaman yang digunakan oleh kasir untuk memilih jenis transaksi pada aplikasi kasir berbasis *Android*. Halaman ini ditampilkan setelah kasir menekan menu “Transaksi” pada bagian navigasi aplikasi. Pada bagian atas halaman terdapat informasi nama toko, nama kasir, serta jumlah transaksi yang telah dilakukan pada hari tersebut.

Pada bagian tengah halaman ditampilkan informasi total transaksi kasir hari ini yang digunakan untuk membantu kasir memantau aktivitas transaksi yang sedang berlangsung. Informasi tersebut menunjukkan jumlah transaksi yang berhasil diproses oleh kasir pada hari berjalan.

Pada bagian bawah halaman terdapat menu “Pilih transaksi” yang berisi dua pilihan transaksi, yaitu barang dan layanan. Menu “Barang” digunakan untuk melakukan transaksi penjualan produk yang tersedia pada toko. Menu “Layanan” digunakan untuk melakukan transaksi jasa atau layanan yang tersedia pada toko. Halaman ini membantu kasir dalam memilih jenis transaksi yang akan diproses secara lebih mudah dan terstruktur.



u. Halaman *Checkout* Barang Kasir



Sumber: Penulis, 2026

Gambar 4.34 Halaman *Checkout* Barang Kasir

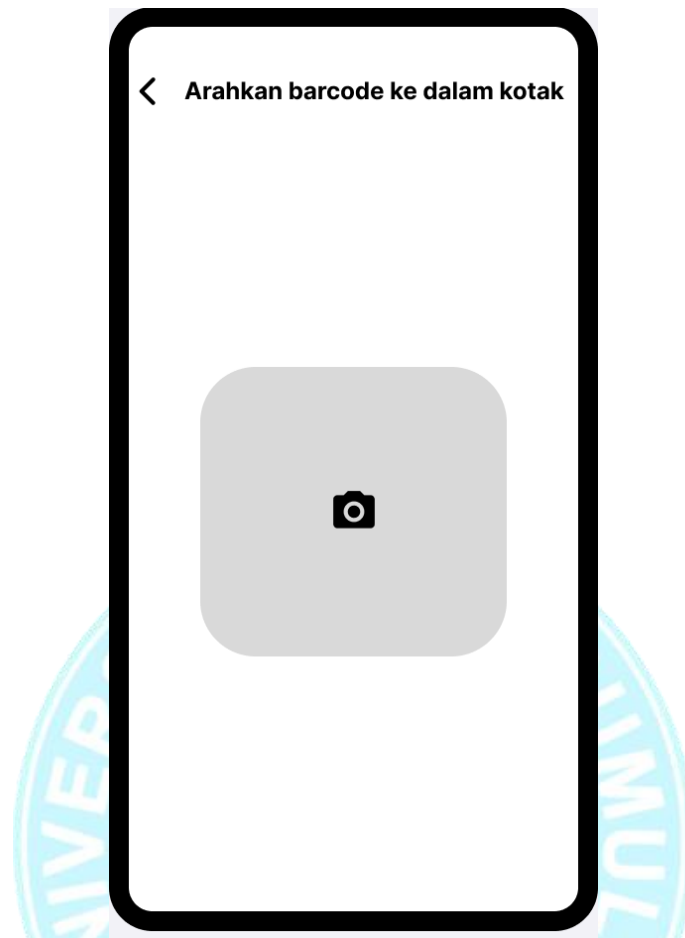
Berdasarkan tampilan antarmuka pada gambar di atas, halaman *checkout* kasir merupakan halaman yang digunakan oleh kasir untuk menyelesaikan proses transaksi penjualan barang pada aplikasi kasir berbasis *Android*. Halaman ini ditampilkan setelah kasir memilih barang yang akan dijual pada menu transaksi. Pada bagian atas halaman terdapat judul “*Checkout Kasir*”, informasi nama kasir yang sedang bertugas, tombol kembali untuk kembali ke halaman sebelumnya, serta ikon scan *barcode* yang digunakan untuk memindai *barcode* produk agar proses input barang dapat dilakukan dengan lebih cepat.

Pada bagian informasi transaksi ditampilkan jumlah produk yang dipilih dan tanggal transaksi yang sedang berlangsung. Karena belum terdapat produk yang

dimasukkan ke dalam keranjang transaksi, sistem menampilkan ikon keranjang belanja beserta keterangan “Belum ada barang” sebagai indikator bahwa belum ada item yang akan diproses. Di bagian bawah halaman tersedia kolom *input* nominal yang digunakan untuk memasukkan jumlah uang yang diterima dari pelanggan. Nominal yang dimasukkan akan digunakan oleh sistem untuk menghitung pembayaran dan kembalian pelanggan secara otomatis. Selanjutnya terdapat panel Total Bayar yang menampilkan jumlah produk yang dibeli beserta total biaya transaksi yang harus dibayarkan.

Halaman ini juga menyediakan fitur pilih *printer bluetooth* yang digunakan untuk memilih perangkat *printer bluetooth* yang akan digunakan dalam proses pencetakan struk transaksi. Setelah seluruh data transaksi telah lengkap dan pembayaran berhasil dilakukan, kasir dapat menekan tombol “Bayar” untuk menyimpan transaksi ke dalam sistem serta mencetak struk sebagai bukti pembayaran pelanggan. Halaman *checkout* kasir berfungsi sebagai tahap akhir dalam proses transaksi penjualan barang. Melalui halaman ini, kasir dapat memeriksa data transaksi, menerima pembayaran pelanggan, memilih perangkat *printer*, serta mencetak struk transaksi secara cepat, mudah, dan terstruktur sehingga mendukung kelancaran proses pelayanan di toko.

v. Halaman *Scan* Barang Kasir



Sumber: Penulis, 2026

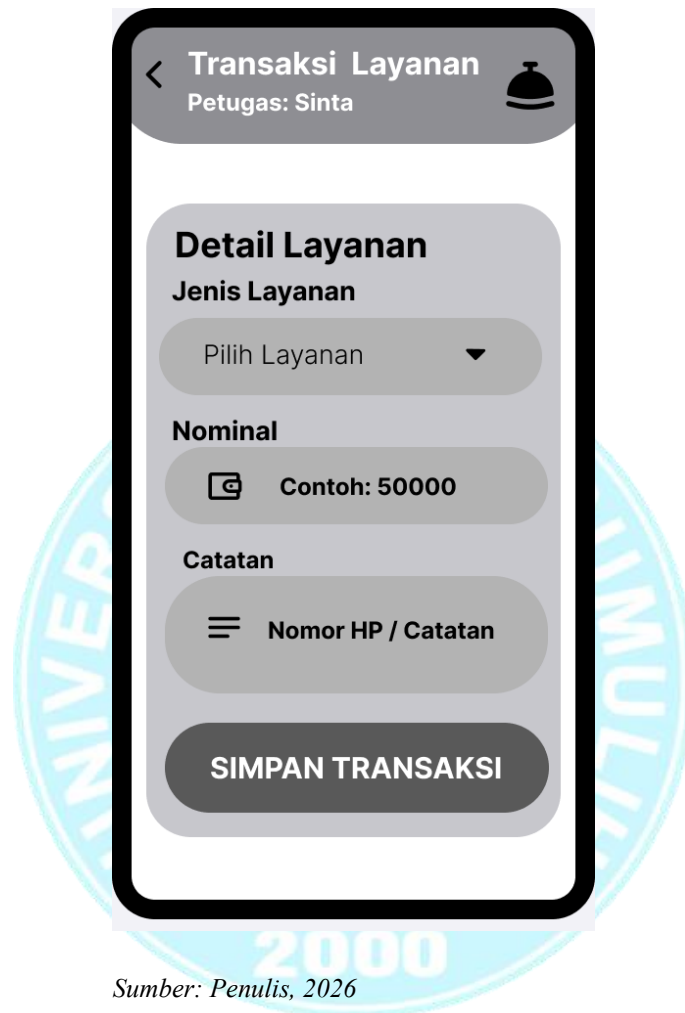
Gambar 4.35 Halaman *Scan* Barang Kasir

Berdasarkan tampilan antarmuka pada gambar di atas, halaman *scan* barang kasir merupakan halaman yang digunakan oleh kasir untuk melakukan proses pemindaian *barcode* produk pada aplikasi kasir berbasis *Android*. Halaman ini akan ditampilkan setelah kasir menekan ikon *scan barcode* pada halaman *checkout* kasir. Fitur ini digunakan untuk membantu proses *input* barang ke dalam transaksi secara otomatis menggunakan kamera perangkat.

Pada bagian atas halaman terdapat informasi “Arahkan *barcode* ke dalam kotak” sebagai petunjuk penggunaan fitur pemindaian *barcode*. Di bagian tengah halaman terdapat area pemindaian yang digunakan untuk mendeteksi dan membaca *barcode* barang yang diarahkan ke kamera perangkat. Halaman *scan* barang ini membantu kasir dalam melakukan proses transaksi penjualan secara lebih cepat,

praktis, dan akurat karena data barang dapat dimasukkan ke dalam sistem tanpa perlu melakukan *input* data secara manual.

w. Halaman Transaksi Layanan Kasir



Sumber: Penulis, 2026

Gambar 4.36 Halaman Transaksi Layanan Kasir

Berdasarkan tampilan antarmuka pada gambar di atas, halaman transaksi layanan kasir merupakan halaman yang digunakan oleh kasir untuk melakukan transaksi layanan atau jasa pada aplikasi kasir berbasis *Android*. Halaman ini akan ditampilkan setelah kasir memilih menu “Layanan” pada halaman transaksi kasir. Pada bagian atas halaman terdapat judul “Transaksi layanan”, informasi nama petugas, serta ikon keranjang yang digunakan untuk melihat data transaksi layanan yang telah dipilih.

Pada halaman ini tersedia beberapa field input, yaitu jenis layanan, nominal, dan catatan. *Field* jenis layanan digunakan untuk memilih layanan yang akan diproses, sedangkan field nominal digunakan untuk memasukkan jumlah pembayaran transaksi layanan. *Field* catatan digunakan untuk menambahkan informasi tambahan, seperti nomor *handphone* atau keterangan lain yang berkaitan dengan transaksi layanan. Pada bagian bawah halaman terdapat tombol “Simpan transaksi” yang digunakan untuk menyimpan data transaksi layanan yang telah diinput. Halaman ini membantu kasir dalam melakukan proses transaksi layanan secara lebih mudah, cepat, dan terstruktur.



BAB V

HASIL DAN PEMBAHASAN

5.1 Spesifikasi Perangkat

Dalam penelitian ini, proses pengembangan aplikasi didukung oleh penggunaan berbagai perangkat yang berperan penting pada setiap tahap pengerjaan. Perangkat tersebut digunakan mulai dari tahap perancangan sistem, penulisan kode program, hingga proses pengujian aplikasi agar sistem dapat berjalan sesuai dengan kebutuhan pengguna. Pemanfaatan perangkat yang tepat dapat membantu meningkatkan efektivitas proses pengembangan, mengurangi kemungkinan terjadinya kesalahan, serta mempercepat penyelesaian aplikasi. Selain itu, penggunaan perangkat yang sesuai juga berpengaruh terhadap kestabilan dan kinerja aplikasi yang dihasilkan. Adapun perangkat yang digunakan dalam penelitian ini terdiri dari dua jenis utama, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*), yang saling mendukung dalam proses pengembangan aplikasi secara keseluruhan.

5.1.1 Perangkat Keras yang digunakan

Perangkat keras yang digunakan dalam penelitian ini merupakan perangkat yang mendukung proses perancangan dan pengembangan aplikasi kasir berbasis *Android* untuk pengelolaan penjualan dan persediaan pada Apan Cell. Perangkat tersebut digunakan dalam proses pembuatan aplikasi, menjalankan program, melakukan pengujian sistem, serta memastikan seluruh fitur aplikasi dapat berfungsi dengan baik pada perangkat *Android*. Kinerja perangkat keras sangat mempengaruhi kelancaran proses pengembangan, kecepatan pengujian aplikasi, serta stabilitas sistem saat digunakan dalam aktivitas transaksi dan pengelolaan persediaan. Oleh karena itu, pemilihan perangkat keras yang sesuai menjadi salah satu faktor penting agar aplikasi yang dikembangkan dapat berjalan secara optimal dan efisien. Adapun spesifikasi perangkat keras yang digunakan dalam penelitian ini adalah sebagai berikut.

1. *Processor: Intel® Celeron® CPU N3350 @ 1.10 GHz*
2. *RAM: 6 GB DDR3*
3. *Storage: 466 GB*
4. *Handphone: Android 10*
5. *RAM: 2 GB*
6. *Storage: 32 GB*
7. *Printer Bluetooth: VSC Receipt Printer TM-58V (USB + Bluetooth)*

5.1.2 Perangkat Lunak yang digunakan

Perangkat lunak yang digunakan dalam penelitian ini berperan sebagai pendukung utama dalam proses perancangan dan pengembangan aplikasi kasir berbasis *Android* pada Apan Cell. Perangkat lunak tersebut digunakan untuk membuat program, mengelola basis data, mendesain antarmuka aplikasi, hingga melakukan pengujian sistem agar aplikasi dapat berjalan sesuai dengan kebutuhan pengguna. Penggunaan perangkat lunak yang tepat membantu proses pengembangan menjadi lebih efektif, mempermudah pengelolaan sistem, serta menjaga kestabilan aplikasi saat dijalankan. Oleh karena itu, perangkat lunak memiliki peranan penting dalam memastikan setiap fitur pada aplikasi dapat berfungsi dengan baik dan optimal. Adapun perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut.

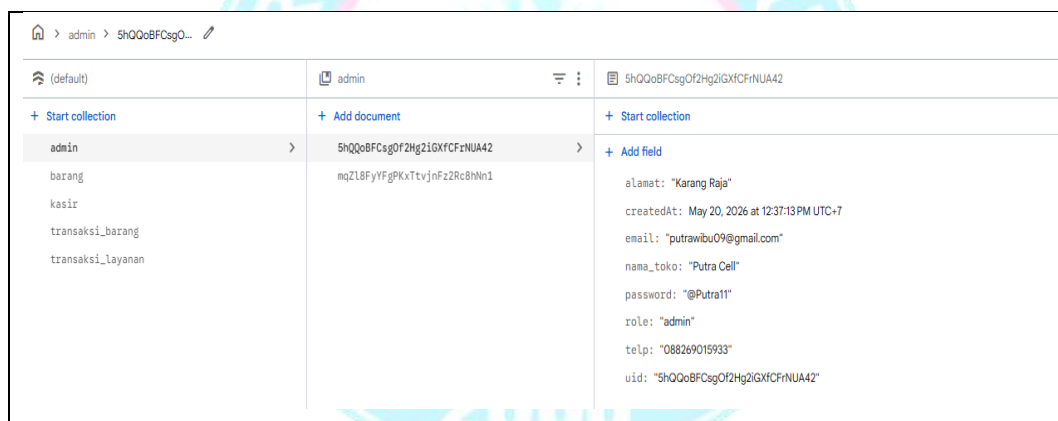
1. *Sistem Operasi: Windows 10 Pro 64-bit Version 22H2*
2. *Bahasa Pemrograman: Dart SDK 3.5.3*
3. *Framework: Flutter 3.24.3*
4. *Android SDK: Android SDK 35.0.0*
5. *Firebase_core: 2.30.0*
6. *Cloud_firestore: 4.15.0*
7. *Firebase_auth: 4.17.0*
8. *Code Editor: Visual Studio Code 1.121.0*
9. *Alat Bantu Design: Figma*

5.2 Implementasi Database

Database pada aplikasi kasir berbasis *Android* ini dirancang untuk menyimpan dan mengelola seluruh data yang digunakan dalam sistem, seperti data barang, transaksi penjualan, persediaan, data pengguna, serta laporan penjualan. Struktur *database* disusun secara terorganisir agar proses pengolahan dan pencarian data dapat berjalan lebih cepat, mudah, dan saling terhubung antar bagian sistem.

Penggunaan *database* juga membantu mengurangi terjadinya duplikasi data serta mempermudah pengelolaan informasi pada aplikasi kasir. Pada penelitian ini, *Firebase Firestore* digunakan sebagai media penyimpanan database karena mampu menyimpan data secara *real-time* dan terintegrasi dengan aplikasi *Android*. Selain itu, *Cloudinary* digunakan untuk menyimpan gambar produk atau barang yang terdapat pada aplikasi.

5.2.1 Implementasi Tabel Admin



Collection	Document ID	Fields
admin	ShQOoBFCsgOf2Hg2iGXFCFNUA42	- alamat: "Karang Raja" - createdAt: May 20, 2026 at 12:37:13 PM UTC+7 - email: "putrawibu09@gmail.com" - nama_toko: "Putra Cell" - password: "@Putra11" - role: "admin" - telp: "088269015933" - uid: "ShQOoBFCsgOf2Hg2iGXFCFNUA42"

Sumber: Penulis, 2026

Gambar 5.1 Tabel Admin

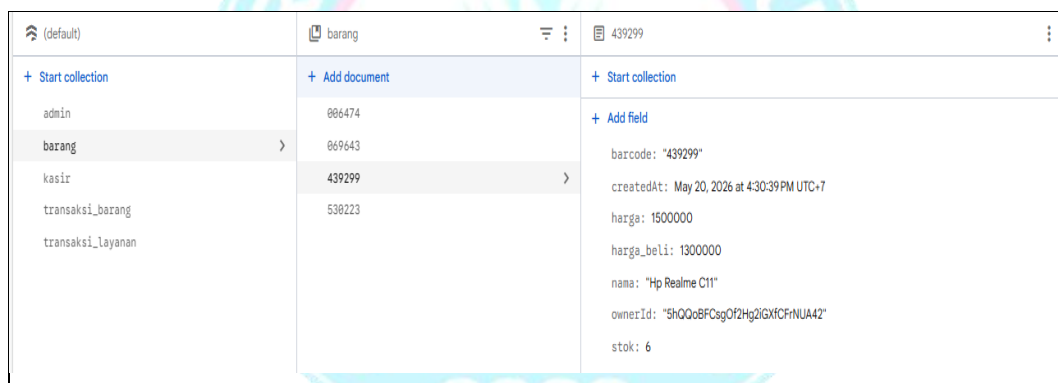
Tabel admin pada basis data *Firebase Firestore* digunakan untuk menyimpan data pengguna yang memiliki hak akses sebagai pengelola aplikasi kasir berbasis *Android* pada Apan Cell. Pada koleksi ini, setiap data admin disimpan dalam bentuk dokumen dengan identitas unik berupa *Document ID* yang dibuat secara otomatis oleh *Firebase*, sehingga memudahkan proses penyimpanan, pencarian, dan pengelolaan data pengguna di dalam sistem.

Struktur data pada koleksi admin terdiri dari beberapa atribut penting, seperti *uid*, *nama_toko*, *email*, *password*, *telp*, *alamat*, *role*, dan *createdAt*. Atribut *uid* digunakan sebagai identitas unik pengguna yang terhubung dengan sistem

autentikasi Firebase. Atribut *nama_toko* digunakan untuk menyimpan nama toko yang terdaftar pada aplikasi, sedangkan *email* dan *password* digunakan dalam proses *login* dan autentikasi pengguna. Selain itu, atribut *telp* digunakan untuk menyimpan nomor telepon admin, atribut *alamat* digunakan untuk menyimpan alamat toko, dan atribut *role* berfungsi untuk menentukan hak akses pengguna sebagai admin di dalam sistem. Atribut *createdAt* digunakan untuk menyimpan waktu saat data admin dibuat pada *database*.

Berdasarkan struktur tersebut, sistem dapat mengelola proses *autentikasi* serta pengaturan hak akses admin dengan lebih teratur dan aman. Penggunaan koleksi admin juga memudahkan aplikasi dalam mengelola informasi pengguna serta mendukung proses pengoperasian aplikasi kasir secara lebih efektif dan terstruktur.

5.2.2 Implementasi Tabel Barang



(default)	barang	439299
+ Start collection	+ Add document	+ Start collection
admin	006474	+ Add field
barang	069643	barcode: "439299"
kasir	439299	createdAt: May 20, 2026 at 4:30:39PM UTC+7
transaksi_barang	530223	harga: 1500000
transaksi_layanan		harga_beli: 1300000
		nama: "Hp Realme C11"
		ownerId: "ShQOoBFCsgOf2Hg2iGKfCFrNUA42"
		stok: 6

Sumber: Penulis, 2026

Gambar 5.2 Tabel Barang

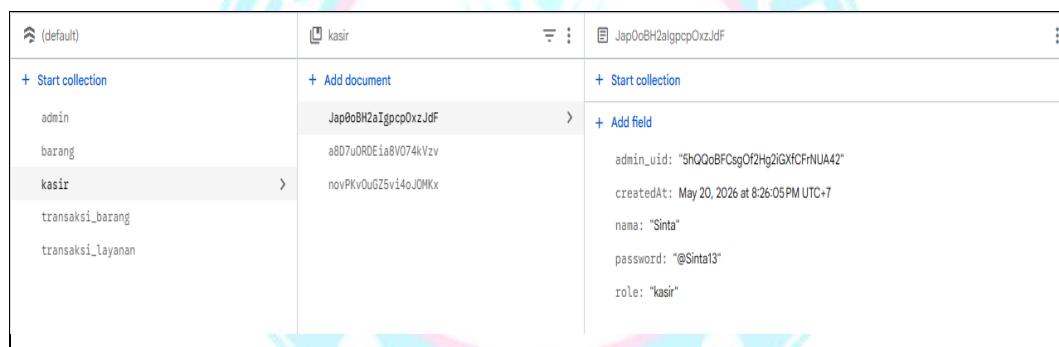
Tabel barang pada basis data *Firebase Firestore* digunakan untuk menyimpan seluruh data barang yang terdapat pada aplikasi kasir berbasis *Android* di Apan Cell. Pada koleksi ini, setiap barang disimpan dalam bentuk dokumen dengan identitas unik berupa *Document ID* yang dibuat berdasarkan *barcode* atau kode barang, sehingga memudahkan proses pencarian, pengelolaan, dan pengolahan data barang di dalam sistem.

Struktur data pada koleksi barang terdiri dari beberapa atribut penting, seperti *barcode*, *nama*, *harga*, *harga_beli*, *stok*, *ownerId*, dan *createdAt*. Atribut

barcode digunakan sebagai kode unik barang yang berfungsi untuk membedakan setiap produk yang tersimpan di dalam *database*. Atribut nama digunakan untuk menyimpan nama barang, sedangkan atribut harga dan harga_beli digunakan untuk menyimpan informasi harga jual dan harga beli barang. Selain itu, atribut stok digunakan untuk mengetahui jumlah persediaan barang yang tersedia, atribut *ownerId* digunakan sebagai identitas pemilik data barang yang terhubung dengan akun admin, dan atribut *createdAt* digunakan untuk menyimpan waktu saat data barang ditambahkan ke *database*.

Berdasarkan struktur tersebut, sistem dapat mengelola data barang dan persediaan secara lebih teratur, cepat, dan akurat. Penggunaan koleksi barang juga memudahkan proses transaksi penjualan, pemantauan stok barang secara *real-time*, serta membantu penyusunan laporan persediaan pada aplikasi kasir.

5.2.3 Implementasi Tabel Kasir



Collection	Document ID	Document Data
kasir	Jap0o8H2aIgcpcOxzJdF	admin_uid: "5hQ0oBPCsgOf2Hg2GKfCFrNUA42" createdAt: May 20, 2026 at 8:26:05 PM UTC+7 nama: "Sinta" password: "@Sinta13" role: "kasir"

Sumber: Penulis, 2026

Gambar 5.3 Tabel Kasir

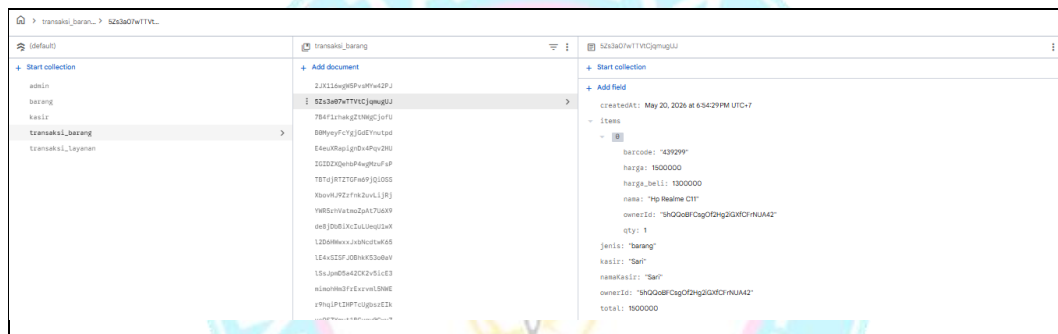
Tabel kasir pada basis data *Firebase Firestore* digunakan untuk menyimpan data pengguna yang memiliki hak akses sebagai kasir pada aplikasi kasir berbasis *Android* di Apan Cell. Pada koleksi ini, setiap data kasir disimpan dalam bentuk dokumen dengan identitas unik berupa *Document ID* yang dibuat secara otomatis oleh *Firebase*, sehingga memudahkan proses pengelolaan dan pencarian data kasir di dalam sistem.

Struktur data pada koleksi kasir terdiri dari beberapa atribut penting, seperti *admin_uid*, *nama*, *password*, *role*, dan *createdAt*. Atribut *admin_uid* digunakan untuk menghubungkan data kasir dengan akun admin yang membuat atau

mengelola data tersebut. Atribut nama digunakan sebagai identitas kasir di dalam aplikasi, sedangkan atribut *password* digunakan dalam proses autentikasi saat kasir melakukan *login* ke sistem. Selain itu, atribut *role* berfungsi untuk menentukan hak akses pengguna sebagai kasir, dan atribut *createdAt* digunakan untuk menyimpan waktu saat data kasir ditambahkan ke *database*.

Berdasarkan struktur tersebut, sistem dapat mengelola data pengguna kasir dengan lebih teratur dan aman. Penggunaan koleksi kasir juga memungkinkan aplikasi membedakan hak akses antara admin dan kasir, sehingga fitur yang dapat digunakan oleh masing-masing pengguna dapat disesuaikan dengan perannya di dalam sistem.

5.2.4 Implementasi Tabel Transaksi Barang



Sumber: Penulis, 2026

Gambar 5.4 Tabel Transaksi Barang

Tabel transaksi_barang pada basis data *Firestore* digunakan untuk menyimpan seluruh data transaksi penjualan barang pada aplikasi kasir berbasis *Android* di Apan Cell. Pada koleksi ini, setiap transaksi disimpan dalam bentuk dokumen dengan identitas unik berupa *Document ID* yang dibuat secara otomatis oleh *Firestore*, sehingga memudahkan proses penyimpanan, pencarian, serta pengelolaan data transaksi di dalam sistem.

Struktur data pada koleksi transaksi_barang terdiri dari beberapa atribut penting, seperti *createdAt*, *items*, *kasir*, *namaKasir*, *ownerId*, dan *total*. Atribut *createdAt* digunakan untuk menyimpan waktu terjadinya transaksi. Atribut *items* berisi daftar barang yang dibeli pada setiap transaksi, yang di dalamnya terdapat data seperti *barcode*, nama barang, harga, harga_beli, *qty*, jenis, dan *ownerId*.

Atribut *barcode* digunakan sebagai kode unik barang, atribut nama digunakan untuk menyimpan nama barang, atribut harga dan harga_beli digunakan untuk menyimpan harga jual serta harga beli barang, sedangkan atribut *qty* digunakan untuk menunjukkan jumlah barang yang dibeli dalam transaksi.

Atribut kasir dan namaKasir digunakan untuk menyimpan informasi pengguna yang melakukan transaksi. Atribut *ownerId* digunakan untuk menghubungkan data transaksi dengan akun admin pemilik toko. Atribut total digunakan untuk menyimpan jumlah keseluruhan pembayaran dari transaksi yang dilakukan. Berdasarkan struktur tersebut, sistem dapat mengelola data transaksi penjualan dengan lebih cepat, teratur, dan akurat. Penggunaan koleksi *transaksi_barang* membantu aplikasi dalam mencatat riwayat transaksi secara otomatis, memantau penjualan barang, serta mempermudah proses pembuatan laporan penjualan pada aplikasi kasir.

5.2.5 Implementasi Tabel Transaksi Layanan

Collection	Document ID	Document Data
transaksi_layanan	mXJDYsStcoInzVp6C6ZE	<pre> catatan: "082166216622" createdAt: "May 15, 2026 at 12:51:07PM UTC+7" isLayanan: true items: [{ harga: 5000 nama: "Top Up DANA" qty: 1 }] jenis: "layanan" kasir: "Anggi Saputra" name: "Layanan: Top Up DANA" nameKasir: "Anggi Saputra" ownerId: "mN6upfRg4XmCoeliUvRLwRkr" total: 5000 </pre>

Sumber: Penulis, 2026

Gambar 5.5 Tabel Transaksi Layanan

Tabel *transaksi_layanan* pada basis data *Firebase Firestore* digunakan untuk menyimpan seluruh data transaksi layanan pada aplikasi kasir berbasis *Android* di Apan Cell. Pada koleksi ini, setiap transaksi disimpan dalam bentuk dokumen dengan identitas unik berupa *Document ID* yang dibuat secara otomatis oleh *Firebase*, sehingga memudahkan proses penyimpanan, pencarian, serta pengelolaan data transaksi layanan di dalam sistem.

Struktur data pada koleksi transaksi_layanan terdiri dari beberapa atribut penting, seperti *createdAt*, *isLayanan*, *items*, *catatan*, *namaKasir*, *ownerId*, dan *total*. Atribut *createdAt* digunakan untuk menyimpan waktu terjadinya transaksi layanan. Atribut *isLayanan* digunakan sebagai penanda bahwa transaksi yang dilakukan termasuk ke dalam kategori layanan. Atribut *items* berisi data layanan yang diproses pada setiap transaksi, yang di dalamnya terdapat atribut seperti nama, harga, *qty*, jenis, kasir, *namaKasir*, dan *ownerId*. Atribut *nama* digunakan untuk menyimpan nama layanan, atribut *harga* digunakan untuk menyimpan biaya layanan, sedangkan atribut *qty* digunakan untuk menunjukkan jumlah layanan yang diproses dalam transaksi.

Atribut *catatan* digunakan untuk menyimpan informasi tambahan yang berkaitan dengan layanan, seperti nomor tujuan atau keterangan transaksi. Atribut *namaKasir* digunakan untuk menyimpan informasi pengguna yang melakukan transaksi layanan. Atribut *ownerId* digunakan untuk menghubungkan data transaksi dengan akun admin pemilik toko. Atribut *total* digunakan untuk menyimpan jumlah keseluruhan pembayaran dari transaksi layanan yang dilakukan.

Berdasarkan struktur tersebut, sistem dapat mengelola data transaksi layanan dengan lebih cepat, teratur, dan akurat. Penggunaan koleksi transaksi_layanan membantu aplikasi dalam mencatat riwayat layanan secara otomatis, mempermudah proses pengelolaan transaksi digital, serta mendukung pembuatan laporan transaksi layanan pada aplikasi kasir.

5.3 Tampilan Antar Muka

5.3.1 Tampilan Halaman Register

Sumber: Penulis, 2026

Gambar 5.6 Halaman Register

Halaman yang ditampilkan pada gambar merupakan halaman daftar akun admin toko pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh pengguna untuk melakukan pendaftaran akun admin toko sebelum dapat mengakses aplikasi. Pada bagian atas halaman terdapat ikon toko, judul “Daftar Toko”, serta keterangan “Buat Akun Admin Toko” sebagai informasi mengenai fungsi halaman tersebut.

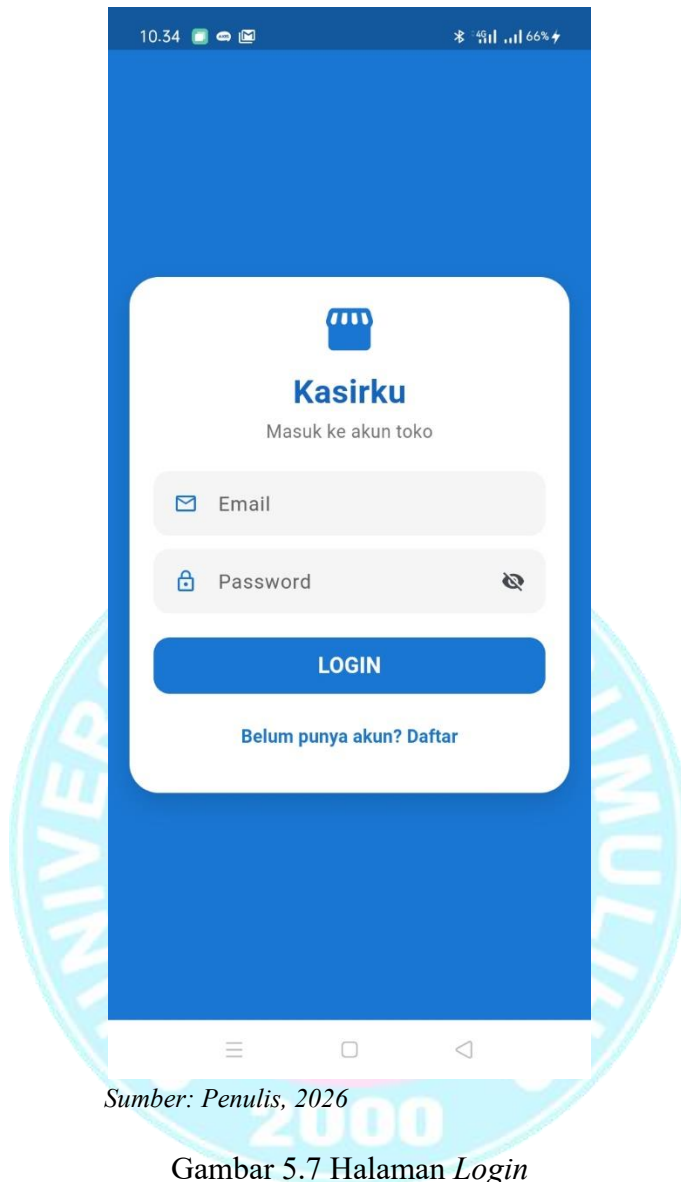
Pada halaman ini tersedia beberapa *field* input yang harus diisi oleh pengguna, yaitu nama toko, alamat, nomor telepon, *email*, *password*, dan

konfirmasi *password*. Data tersebut digunakan sebagai identitas akun admin toko di dalam aplikasi. *Field password* dan konfirmasi *password* dilengkapi dengan ikon untuk menampilkan atau menyembunyikan kata sandi agar pengguna dapat memastikan *password* yang dimasukkan sudah benar.

Di bawah *field password* terdapat informasi mengenai ketentuan *password*, yaitu minimal terdiri dari 8 karakter yang mengandung huruf besar, huruf kecil, angka, dan simbol. Selanjutnya tersedia tombol “Daftar” yang digunakan untuk memproses pendaftaran akun. Apabila data yang dimasukkan valid, maka akun admin toko akan berhasil dibuat. Pada bagian bawah halaman juga terdapat menu “Sudah punya akun? *Login*” yang digunakan untuk mengarahkan pengguna ke halaman *login* apabila sudah memiliki akun sebelumnya.



5.3.2 Tampilan Halaman *Login*



Sumber: Penulis, 2026

Gambar 5.7 Halaman *Login*

Halaman yang ditampilkan pada gambar di atas adalah halaman *login*, yang merupakan halaman *autentikasi* yang digunakan oleh administrator dan kasir untuk masuk ke dalam aplikasi kasir berbasis *Android*. Halaman ini menjadi akses awal sebelum pengguna dapat menggunakan fitur-fitur yang tersedia di dalam sistem. Pada bagian tengah halaman terdapat ikon toko, nama aplikasi “Kasirku”, serta keterangan “Masuk ke akun toko” yang menjelaskan fungsi dari halaman tersebut.

Pada halaman ini tersedia formulir login yang terdiri dari kolom *email* dan kolom *password*. Kolom *email* digunakan untuk memasukkan alamat *email* toko yang telah terdaftar pada sistem. Dalam implementasinya, administrator dan kasir

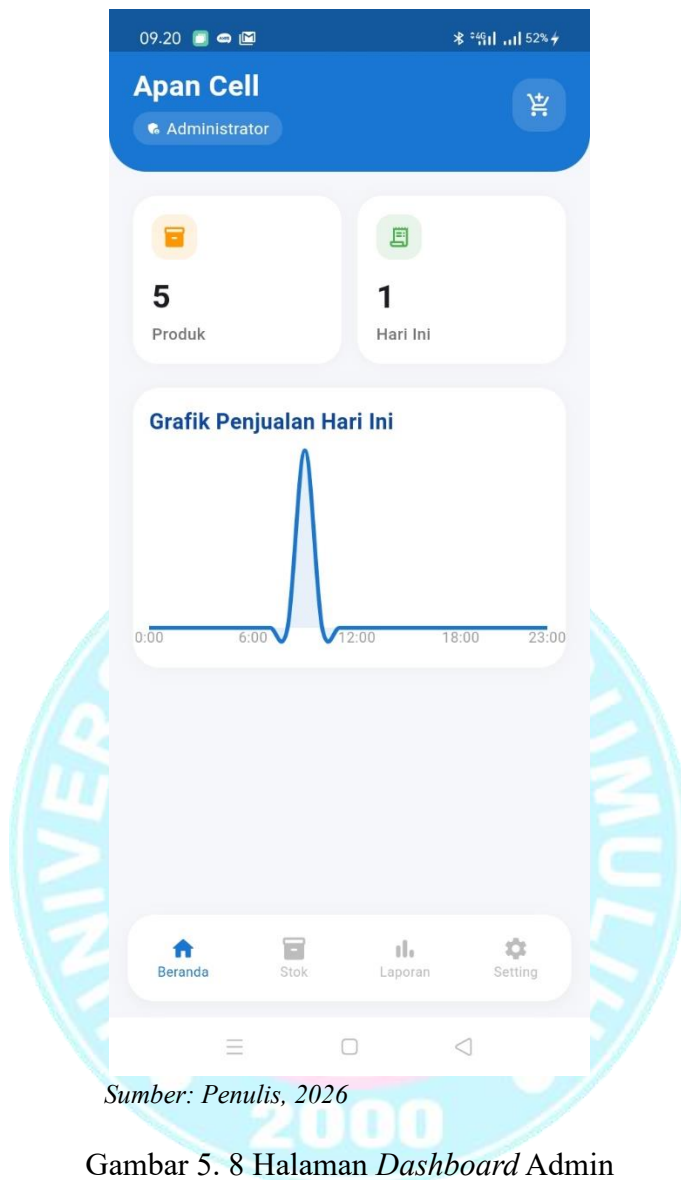
menggunakan *email* toko yang sama untuk masuk ke dalam aplikasi. Perbedaan hak akses ditentukan berdasarkan akun dan kata sandi yang digunakan oleh masing-masing pengguna. Akun administrator diperoleh melalui proses pendaftaran akun pada aplikasi, sedangkan akun kasir dibuat dan dikelola langsung oleh administrator melalui fitur manajemen kasir.

Kolom *password* digunakan untuk memasukkan kata sandi sesuai dengan akun yang dimiliki pengguna. Setiap akun kasir memiliki kata sandi yang berbeda meskipun menggunakan *email* toko yang sama. Pada bagian kolom *password* terdapat ikon mata yang berfungsi untuk menampilkan atau menyembunyikan karakter kata sandi sehingga pengguna dapat memastikan data yang dimasukkan sudah benar sebelum melakukan proses *login*.

Di bawah formulir *login* terdapat tombol “Login” yang digunakan untuk memproses data pengguna. Sistem akan melakukan verifikasi terhadap *email* dan *password* yang dimasukkan dengan data yang tersimpan pada basis data. Apabila data yang dimasukkan *valid*, pengguna akan diarahkan ke halaman utama sesuai dengan hak akses yang dimilikinya. Administrator akan diarahkan ke halaman utama admin yang memiliki akses penuh terhadap pengelolaan toko, sedangkan kasir akan diarahkan ke halaman utama kasir yang hanya dapat mengakses fitur transaksi sesuai izin yang diberikan oleh administrator.

Pada bagian bawah halaman terdapat tautan “Belum punya akun? Daftar” yang digunakan untuk mengarahkan pengguna ke halaman pendaftaran akun. Fitur pendaftaran ini hanya digunakan untuk membuat akun administrator toko. Sementara itu, akun kasir tidak dapat didaftarkan secara mandiri karena harus dibuat terlebih dahulu oleh administrator melalui sistem. Dengan mekanisme tersebut, pengelolaan pengguna dan keamanan akses aplikasi dapat dilakukan secara lebih terstruktur dan terkontrol.

5.3.3 Tampilan Halaman *Dashboard Admin*



Sumber: Penulis, 2026

Gambar 5. 8 Halaman *Dashboard Admin*

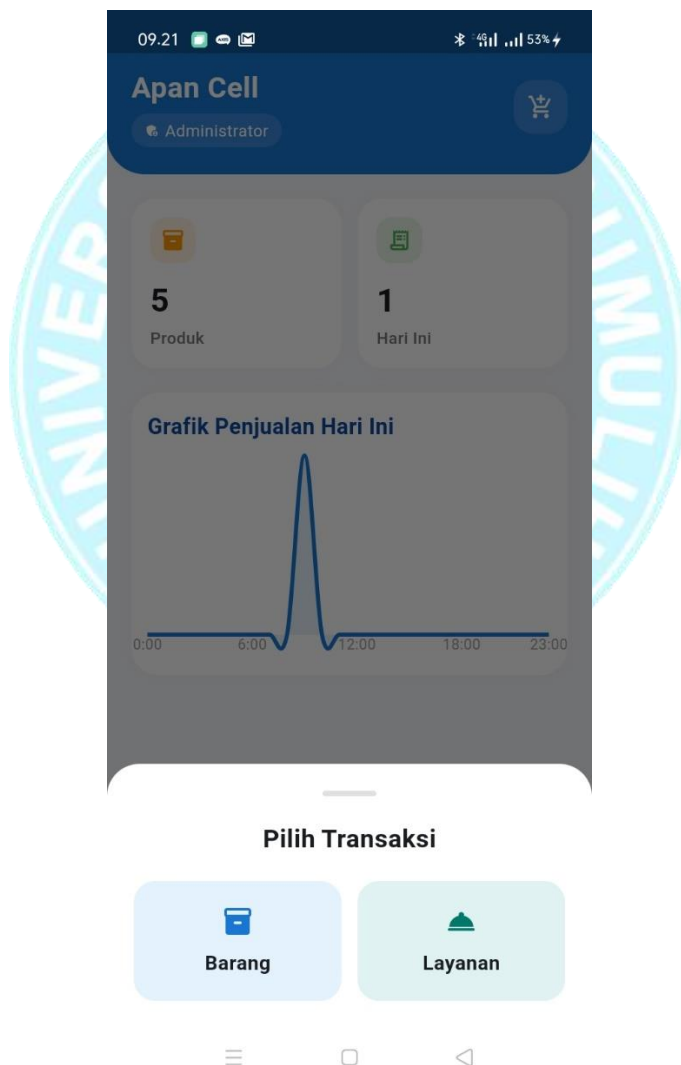
Halaman yang ditampilkan pada gambar merupakan halaman beranda administrator pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh administrator untuk melihat informasi utama terkait aktivitas penjualan dan data toko. Pada bagian atas halaman terdapat nama toko “Apan Cell”, informasi hak akses sebagai administrator, serta ikon keranjang yang digunakan untuk mengakses menu transaksi.

Pada halaman ini ditampilkan informasi jumlah produk dan jumlah transaksi hari ini dalam bentuk kartu informasi. Data tersebut digunakan untuk membantu administrator memantau aktivitas penjualan yang berlangsung pada toko. Halaman

ini juga menampilkan grafik penjualan hari ini yang digunakan untuk melihat perkembangan transaksi penjualan berdasarkan waktu transaksi yang terjadi pada hari tersebut.

Pada bagian bawah halaman terdapat menu navigasi, yaitu beranda, stok, laporan, dan *setting*. Menu beranda digunakan untuk kembali ke halaman utama administrator, menu stok digunakan untuk mengelola data barang, menu laporan digunakan untuk melihat laporan transaksi penjualan, sedangkan menu *setting* digunakan untuk mengelola pengaturan aplikasi dan akun pengguna.

5.3.4 Tampilan Halaman Transaksi Admin



Sumber: Penulis, 2026

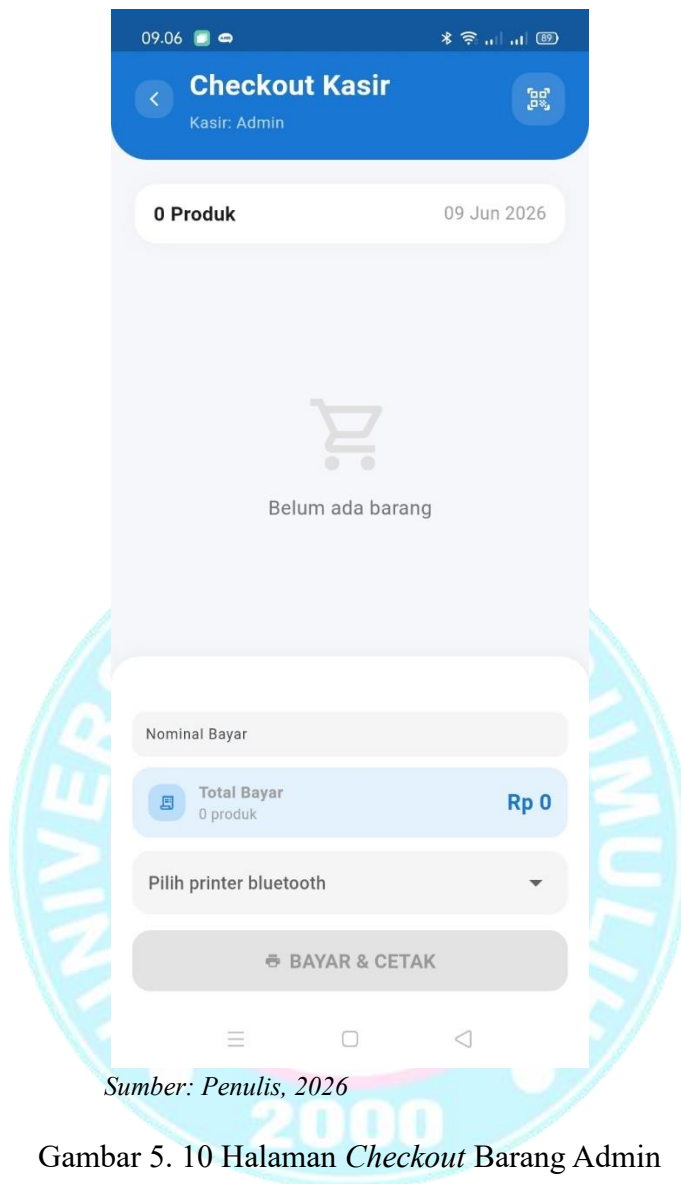
Gambar 5.9 Halaman Transaksi Admin

Halaman yang ditampilkan pada gambar merupakan halaman transaksi pada administrator aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh administrator untuk memilih jenis transaksi yang akan dilakukan pada aplikasi. Pada bagian bawah halaman terdapat menu “Pilih transaksi” yang berisi dua pilihan menu, yaitu barang dan layanan.

Menu barang digunakan untuk melakukan transaksi penjualan produk yang tersedia pada toko. Melalui menu ini administrator dapat memilih produk yang akan dibeli oleh pelanggan dan memproses transaksi penjualan barang. Menu layanan digunakan untuk melakukan transaksi jasa atau layanan yang tersedia pada toko. Kedua menu tersebut dibuat untuk memudahkan administrator dalam mengelola transaksi sesuai dengan jenis transaksi yang dilakukan.



5.3.5 Tampilan Halaman *Checkout* Barang Admin



Sumber: Penulis, 2026

Gambar 5. 10 Halaman *Checkout* Barang Admin

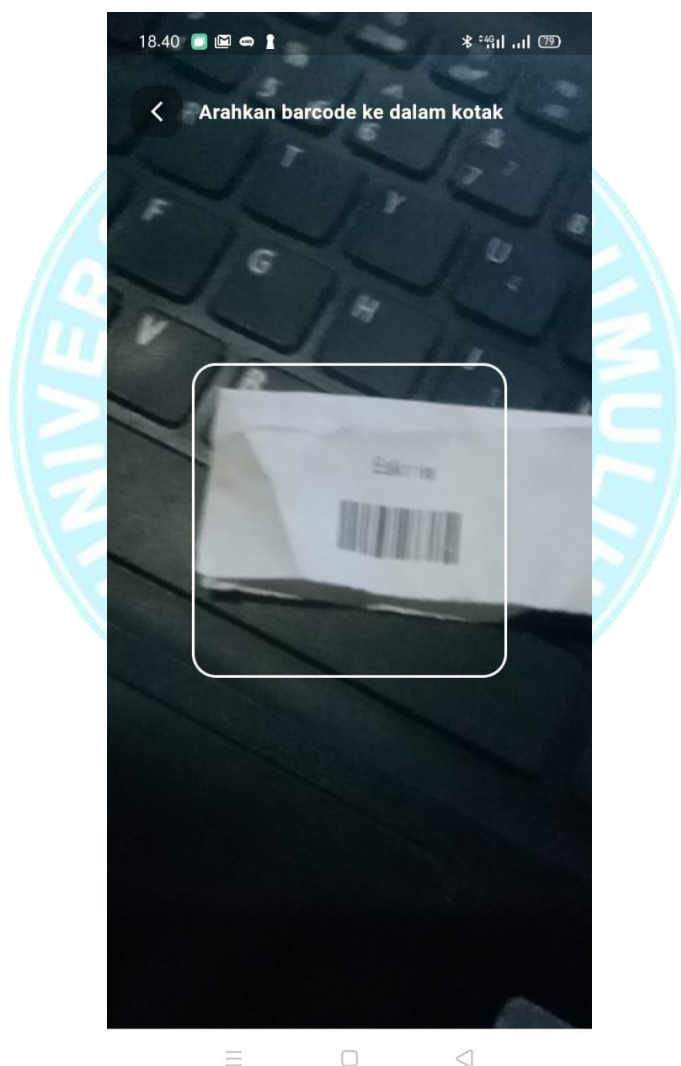
Halaman yang ditampilkan pada gambar merupakan halaman *checkout* admin pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh admin untuk melakukan proses pembayaran transaksi penjualan barang. Pada bagian atas halaman terdapat judul “Checkout kasir”, informasi nama kasir, serta ikon *scan barcode* yang digunakan untuk masuk ke halaman scan barang kasir.

Pada halaman ini ditampilkan informasi jumlah produk yang dipilih dalam transaksi serta tanggal transaksi yang sedang berlangsung. Apabila belum ada barang yang dipilih, maka sistem akan menampilkan informasi “Belum ada

barang”. Halaman ini juga menyediakan pilihan *printer bluetooth* yang digunakan untuk memilih perangkat *printer* dalam proses pencetakan struk transaksi.

Pada bagian bawah halaman ditampilkan informasi total pembayaran yang harus dibayarkan oleh pelanggan. Terdapat tombol “Bayar dan Cetak” yang digunakan untuk memproses transaksi serta mencetak struk pembayaran. Halaman ini membantu kasir dalam melakukan proses transaksi penjualan secara lebih cepat dan terstruktur.

5.3.6 Tampilan Halaman *Scan* Barang Admin



Sumber: Penulis, 2026

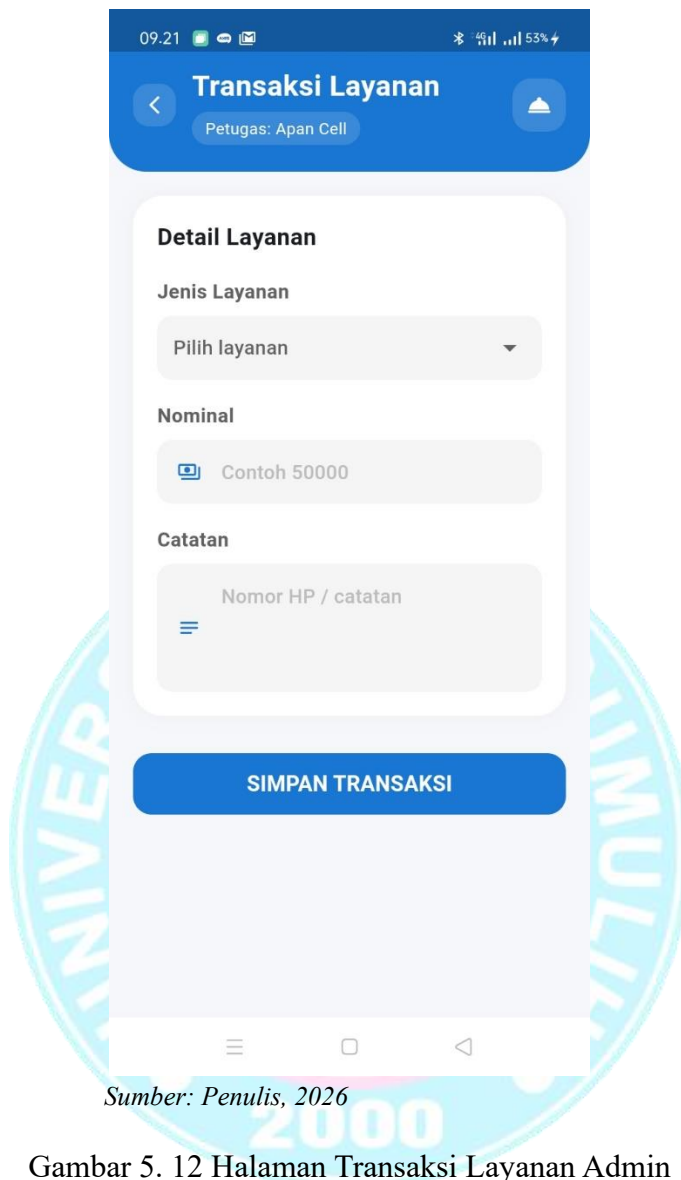
Gambar 5.11 Halaman *Scan* Barang Admin

Halaman yang ditampilkan pada gambar merupakan halaman scan barang admin pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh admin untuk melakukan proses pemindaian *barcode* barang saat transaksi penjualan berlangsung. Halaman ini akan ditampilkan ketika admin menekan tombol *scan* barang pada menu transaksi.

Pada bagian atas halaman terdapat informasi “Arahkan *barcode* ke dalam kotak” sebagai petunjuk penggunaan fitur pemindaian. Di bagian tengah halaman terdapat area kotak pemindaian yang digunakan untuk mendeteksi dan membaca *barcode* produk menggunakan kamera perangkat. Halaman *scan* barang ini membantu kasir dalam memasukkan data produk ke dalam transaksi secara lebih cepat tanpa perlu memasukkan data barang secara manual.



5.3.7 Tampilan Halaman Transaksi Layanan Admin



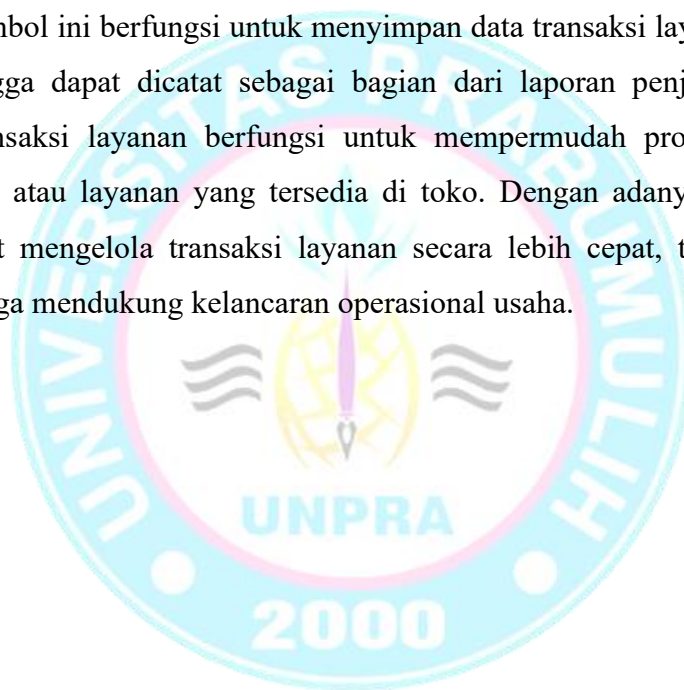
Sumber: Penulis, 2026

Gambar 5. 12 Halaman Transaksi Layanan Admin

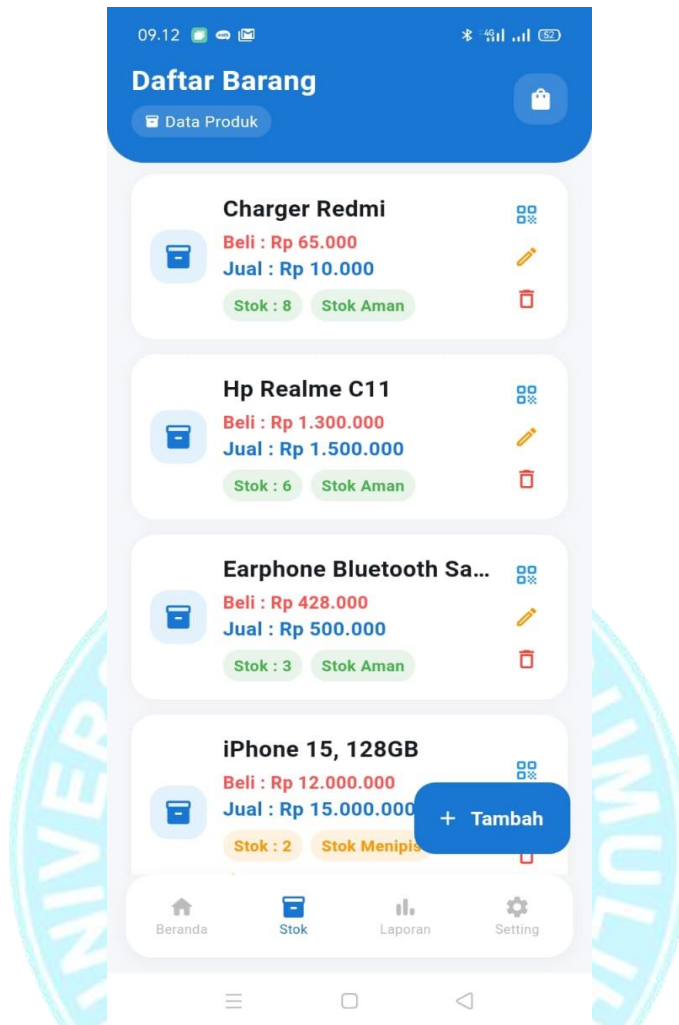
Halaman yang ditampilkan pada gambar di atas adalah halaman transaksi layanan, halaman ini digunakan oleh admin atau petugas untuk melakukan transaksi layanan atau jasa yang tersedia pada aplikasi kasir berbasis *Android*. Halaman ini digunakan untuk mencatat dan mengelola transaksi layanan yang diberikan kepada pelanggan. Pada bagian atas halaman terdapat judul “Transaksi Layanan”, informasi nama petugas yang sedang bertugas, tombol kembali untuk kembali ke halaman sebelumnya, serta ikon keranjang yang digunakan untuk melihat daftar transaksi layanan yang telah dipilih.

Pada bagian detail layanan tersedia beberapa kolom yang harus diisi sebelum transaksi disimpan. Kolom jenis layanan digunakan untuk memilih layanan yang akan diproses sesuai dengan layanan yang tersedia pada sistem. Kolom nominal digunakan untuk memasukkan nilai pembayaran atau biaya layanan yang dikenakan kepada pelanggan. Selanjutnya, kolom catatan digunakan untuk menambahkan informasi pendukung terkait transaksi, seperti nomor telepon pelanggan, nomor tujuan, atau keterangan lain yang diperlukan sesuai dengan jenis layanan yang dipilih.

Setelah seluruh data transaksi diisi dengan lengkap dan benar, pengguna dapat menekan tombol “Simpan Transaksi” yang terdapat pada bagian bawah halaman. Tombol ini berfungsi untuk menyimpan data transaksi layanan ke dalam sistem sehingga dapat dicatat sebagai bagian dari laporan penjualan layanan. Halaman transaksi layanan berfungsi untuk mempermudah proses pencatatan transaksi jasa atau layanan yang tersedia di toko. Dengan adanya halaman ini, petugas dapat mengelola transaksi layanan secara lebih cepat, terstruktur, dan akurat sehingga mendukung kelancaran operasional usaha.



5.3.8 Tampilan Halaman Daftar Barang



Sumber: Penulis, 2026

Gambar 5. 13 Halaman Daftar Barang

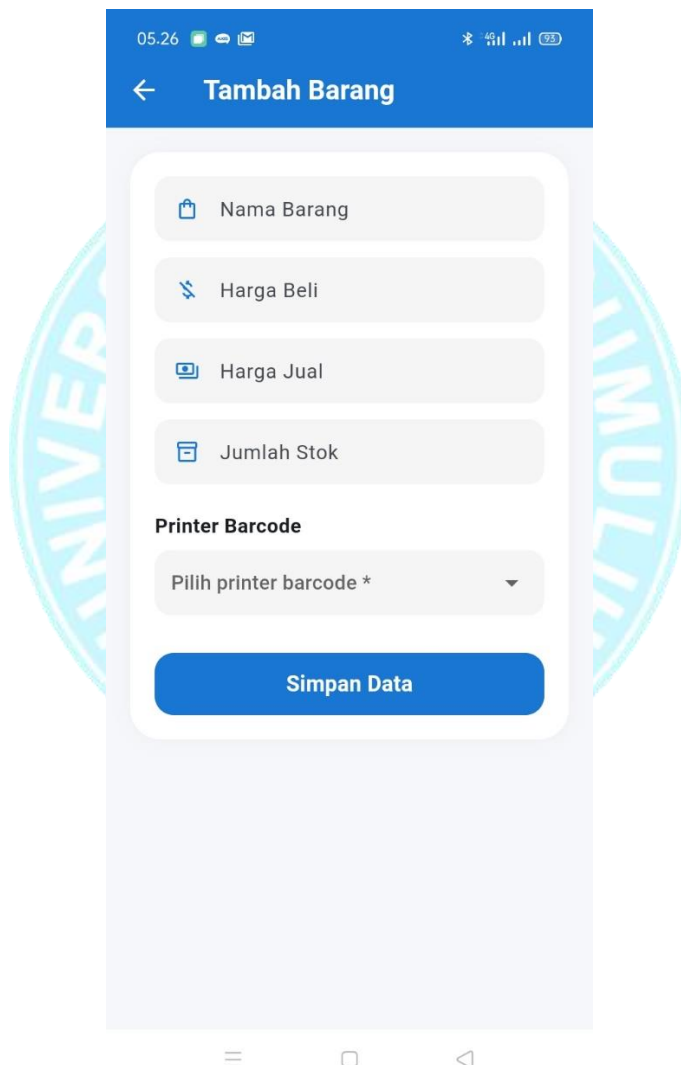
Halaman yang ditampilkan pada gambar merupakan halaman daftar barang pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh administrator untuk melihat dan mengelola data produk yang tersedia pada toko. Pada bagian atas halaman terdapat judul “Daftar barang”, informasi “Data produk”, serta ikon tas belanja yang digunakan untuk mengakses menu transaksi.

Pada halaman ini ditampilkan beberapa data barang yang tersedia di toko, seperti nama barang, harga beli, harga jual, jumlah stok, serta status persediaan barang. Informasi stok digunakan untuk membantu administrator mengetahui kondisi persediaan barang, seperti stok aman atau stok menipis. Setiap data barang

juga dilengkapi dengan ikon aksi yang digunakan untuk melihat detail *barcode*, mengubah data barang, dan menghapus data barang.

Di bagian bawah halaman terdapat tombol “Tambah” yang digunakan untuk menambahkan data barang baru ke dalam sistem. Halaman ini membantu administrator dalam mengelola data persediaan barang secara lebih terstruktur dan memudahkan proses pemantauan stok pada toko.

5.3.9 Tampilan Halaman Tambah Barang



Sumber: Penulis, 2026

Gambar 5.14 Halaman Tambah Barang

Halaman yang ditampilkan pada gambar merupakan halaman tambah barang, halaman ini digunakan oleh administrator untuk menambahkan data barang

baru ke dalam sistem aplikasi kasir berbasis *Android*. Pada bagian atas halaman terdapat tombol kembali dan judul “Tambah Barang” yang menunjukkan fungsi halaman tersebut.

Pada halaman ini tersedia beberapa kolom *input*, yaitu nama barang, harga beli, harga jual, dan jumlah stok. Data tersebut digunakan sebagai informasi barang yang akan disimpan ke dalam sistem. Selain itu, terdapat pilihan *printer barcode* yang digunakan untuk memilih perangkat *printer* yang akan digunakan dalam proses pencetakan *barcode* barang.

Di bagian bawah halaman terdapat tombol “Simpan Data” yang berfungsi untuk menyimpan data barang yang telah dimasukkan. Halaman ini membantu administrator dalam mengelola data barang dan persediaan stok secara lebih mudah, cepat, dan terstruktur.



5.3.10 Tampilan Halaman Edit Barang

05.26

← Edit Barang

Nama Barang
Hp Realme C11

Harga Beli
1300000

Harga Jual
1500000

Jumlah Stok
6

Printer Barcode
Pilih printer barcode *

Simpan Data

Sumber: Penulis, 2026

Gambar 5.15 Halaman Edit Barang

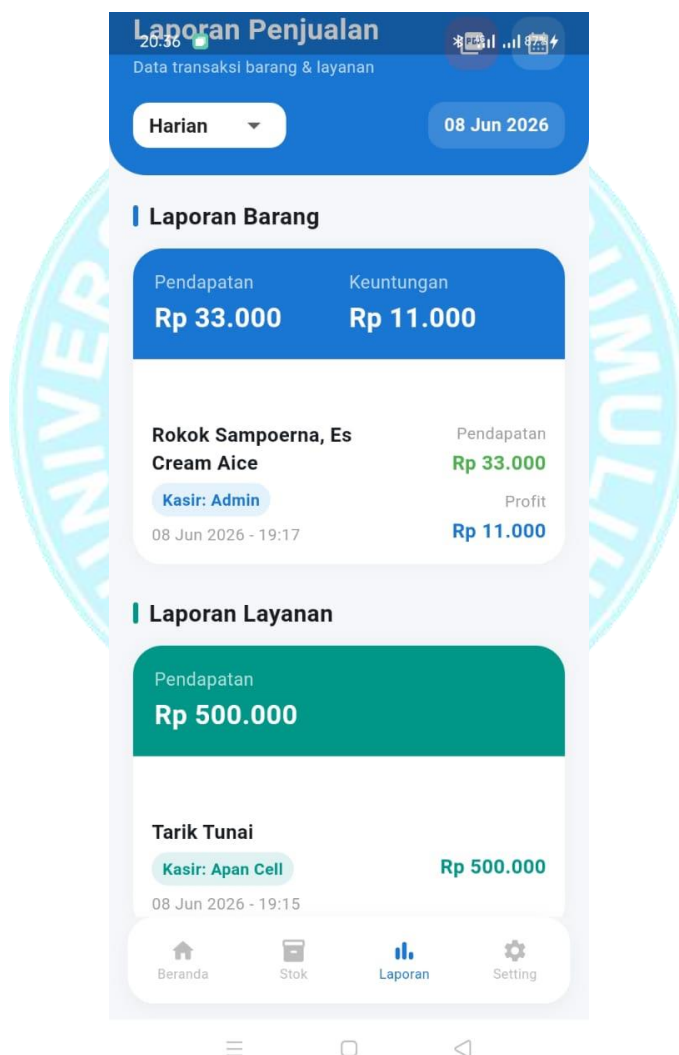
Halaman yang ditampilkan pada gambar merupakan halaman edit barang merupakan halaman yang digunakan oleh administrator untuk mengubah atau memperbarui data barang yang telah tersimpan di dalam sistem aplikasi kasir berbasis *Android*. Pada bagian atas halaman terdapat tombol kembali dan judul “Edit Barang” yang menunjukkan fungsi dari halaman tersebut.

Pada halaman ini ditampilkan informasi barang yang dapat diperbarui, yaitu nama barang, harga beli, harga jual, dan jumlah stok. Administrator dapat mengubah data tersebut sesuai dengan kondisi barang yang ada di toko. Selain itu,

tersedia pilihan *printer barcode* yang digunakan untuk memilih perangkat *printer* yang akan digunakan dalam proses pencetakan *barcode* barang.

Di bagian bawah halaman terdapat tombol “Simpan Data” yang berfungsi untuk menyimpan perubahan data barang yang telah diperbarui. Halaman ini membantu administrator dalam mengelola dan memperbarui data barang secara lebih mudah, cepat, dan terstruktur sehingga informasi stok dan harga barang tetap sesuai dengan kondisi yang sebenarnya.

5.3.11 Tampilan Halaman Laporan Penjualan



Sumber: Penulis, 2026

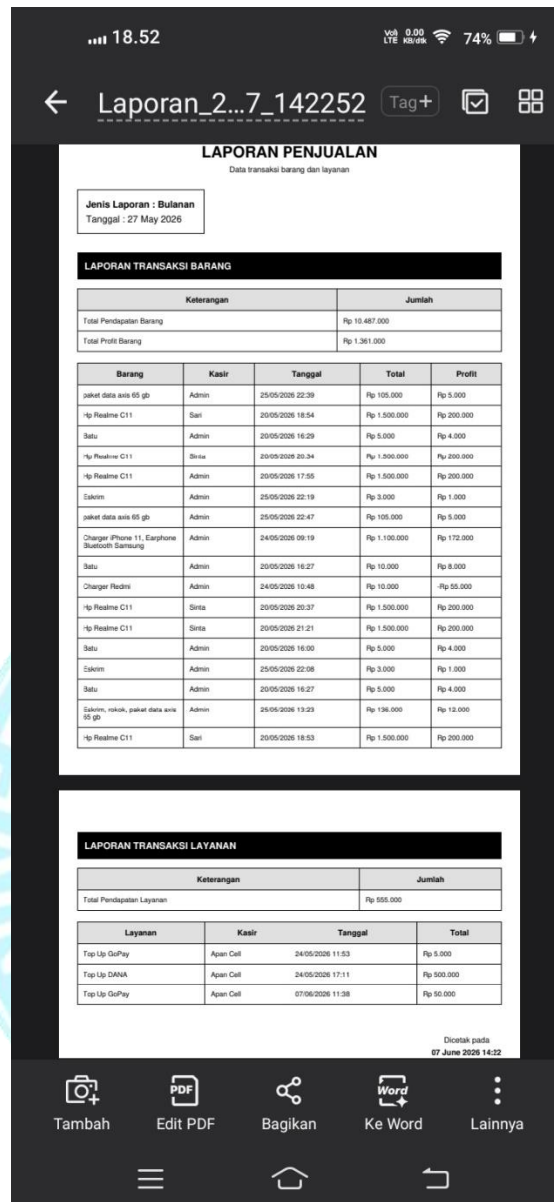
Gambar 5.16 Halaman Laporan Barang

Halaman yang ditampilkan pada gambar diatas merupakan halaman laporan yang digunakan oleh administrator untuk melihat data transaksi barang dan layanan yang telah dilakukan pada toko. Pada bagian atas halaman terdapat judul “Laporan Penjualan”, keterangan “Data transaksi barang & layanan”, tombol cetak PDF, pilihan periode laporan, serta pilihan tanggal yang digunakan untuk menampilkan laporan sesuai waktu yang diinginkan. Pada halaman ini tersedia pilihan periode laporan, yaitu harian, mingguan, atau bulanan. Sistem akan menampilkan data transaksi berdasarkan periode dan tanggal yang dipilih oleh pengguna. Pada gambar terlihat laporan dengan periode Harian pada tanggal 08 Jun 2026.

Bagian Laporan Barang menampilkan informasi total pendapatan dan keuntungan yang diperoleh dari transaksi penjualan barang. Selain itu, ditampilkan pula rincian transaksi yang meliputi nama barang, nama kasir yang melakukan transaksi, waktu transaksi, nilai pendapatan, dan keuntungan yang diperoleh dari transaksi tersebut. Di bawahnya terdapat bagian laporan layanan yang menampilkan informasi pendapatan dari transaksi layanan atau jasa yang tersedia di toko. Pada bagian ini juga ditampilkan rincian transaksi layanan, seperti nama layanan, nama petugas atau kasir yang melakukan transaksi, waktu transaksi, serta nilai transaksi yang diperoleh.

Pada bagian paling bawah halaman terdapat menu navigasi yang terdiri dari beranda, stok, laporan, dan setting. Halaman ini membantu administrator dalam memantau hasil penjualan barang dan layanan serta mempermudah proses pengelolaan laporan transaksi secara lebih terstruktur dan akurat.

5.3.12 Tampilan Halaman Cetak PDF



Sumber: Penulis, 2026

Gambar 5.17 Halaman Cetak PDF

Berdasarkan tampilan pada gambar di atas, halaman hasil cetak Laporan Penjualan merupakan dokumen laporan yang digunakan oleh administrator untuk melihat dan menyimpan data transaksi barang dan layanan dalam bentuk PDF. Dokumen ini dihasilkan melalui fitur cetak PDF pada menu laporan penjualan sehingga dapat digunakan sebagai arsip dan dokumentasi transaksi toko.

Pada bagian atas dokumen terdapat judul “laporan penjualan” beserta keterangan “data transaksi barang dan layanan”. Di bawahnya ditampilkan

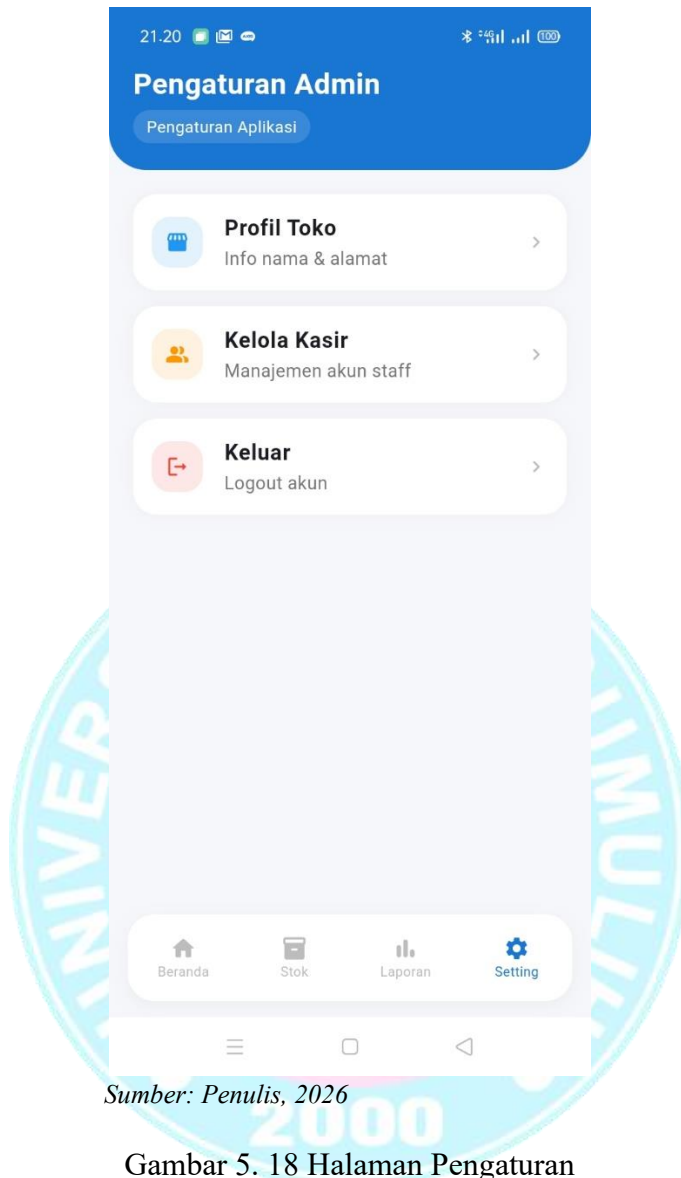
informasi Jenis Laporan dan Tanggal yang menunjukkan periode laporan yang dicetak. Pada gambar terlihat laporan yang dicetak menggunakan periode bulanan dengan tanggal 27 May 2026.

Bagian laporan transaksi barang menampilkan ringkasan hasil penjualan barang yang terdiri dari total pendapatan barang dan total profit barang. Selain itu, terdapat tabel yang berisi rincian transaksi barang, meliputi nama barang, nama kasir, tanggal transaksi, total penjualan, dan profit yang diperoleh dari setiap transaksi. Informasi tersebut digunakan untuk memantau hasil penjualan barang secara lebih rinci selama periode laporan.

Di bawahnya terdapat bagian laporan transaksi layanan yang menampilkan total pendapatan layanan beserta tabel rincian transaksi layanan. Tabel tersebut berisi informasi nama layanan, nama kasir, tanggal transaksi, dan total transaksi layanan yang telah dilakukan selama periode laporan.

Pada bagian bawah dokumen terdapat informasi waktu pencetakan laporan yang menunjukkan tanggal dan jam saat laporan dibuat. Dokumen laporan penjualan dalam format PDF ini membantu administrator dalam melakukan pencatatan, penyimpanan, dan pengelolaan data transaksi barang maupun layanan secara lebih rapi, terstruktur, dan mudah untuk didokumentasikan.

5.3.13 Tampilan Halaman Pengaturan



Sumber: Penulis, 2026

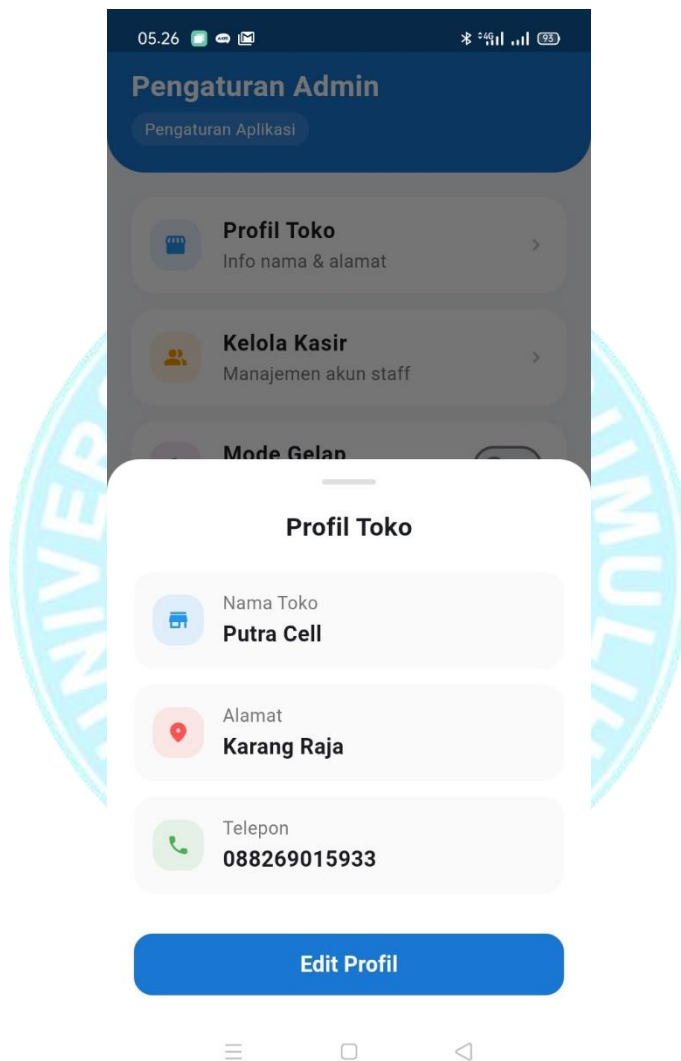
Gambar 5. 18 Halaman Pengaturan

Halaman yang ditampilkan pada gambar merupakan halaman pengaturan admin pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh administrator untuk mengelola pengaturan aplikasi dan informasi akun toko. Pada bagian atas halaman terdapat judul “Pengaturan admin” serta keterangan “Pengaturan aplikasi” sebagai informasi mengenai fungsi halaman tersebut.

Pada halaman ini terdapat menu “Profil toko” yang digunakan untuk melihat dan mengubah informasi toko, seperti nama toko dan alamat toko. Menu “Kelola kasir” digunakan untuk mengatur dan mengelola akun staf atau kasir yang dapat mengakses aplikasi.

Pada bagian bawah halaman terdapat menu “Keluar” yang digunakan untuk melakukan proses *logout* dari akun administrator. Halaman pengaturan ini membantu administrator dalam mengelola informasi aplikasi dan akun pengguna secara lebih mudah dan terstruktur.

5.3.14 Tampilan Halaman Profil Toko



Sumber: Penulis, 2026

Gambar 5.19 Halaman Profil Toko

Halaman yang ditampilkan pada gambar merupakan halaman profil toko pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh administrator untuk melihat informasi data toko yang telah tersimpan di dalam sistem. Pada

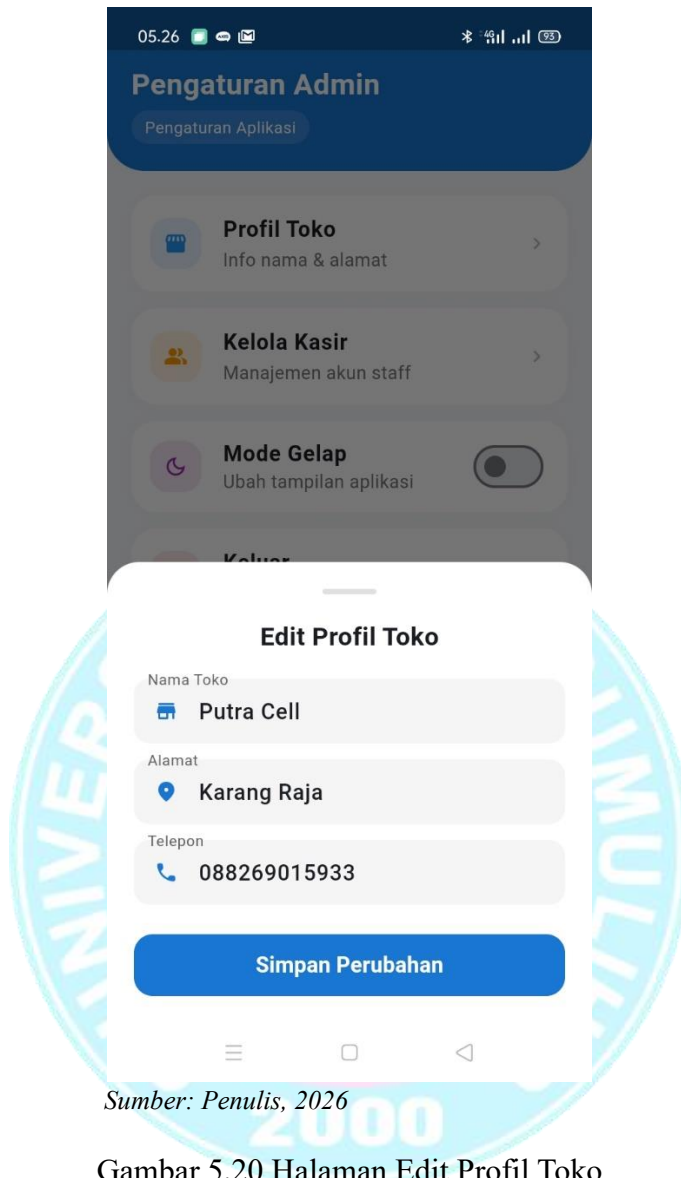
bagian atas halaman terdapat judul “Profil toko” sebagai informasi mengenai fungsi halaman tersebut.

Pada halaman ini ditampilkan beberapa informasi toko, yaitu nama toko, alamat, dan nomor telepon. Data tersebut digunakan sebagai identitas toko yang digunakan di dalam aplikasi kasir. Informasi yang ditampilkan membantu administrator dalam memastikan data toko telah sesuai dengan data yang digunakan pada sistem.

Pada bagian bawah halaman terdapat tombol “Edit profil” yang digunakan untuk mengubah atau memperbarui informasi profil toko. Halaman ini membantu administrator dalam mengelola data identitas toko secara lebih mudah dan terstruktur.



5.3.15 Tampilan Halaman Edit Profil Toko



Sumber: Penulis, 2026

Gambar 5.20 Halaman Edit Profil Toko

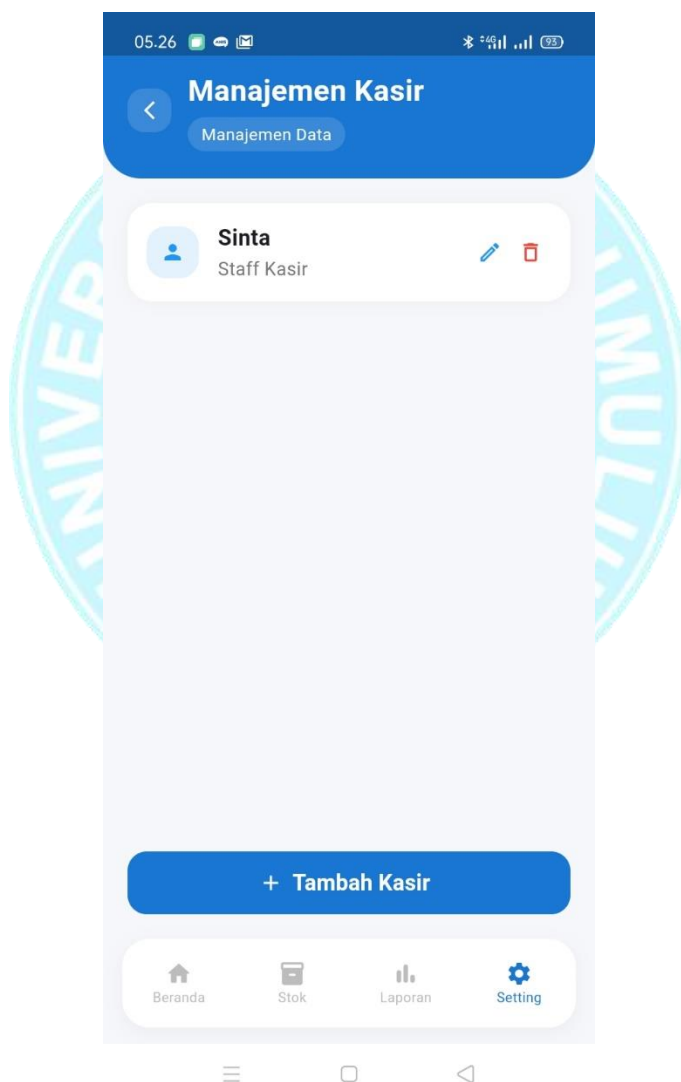
Halaman yang ditampilkan pada gambar merupakan halaman edit profil toko yang digunakan oleh administrator untuk mengubah atau memperbarui informasi profil toko yang tersimpan di dalam sistem aplikasi kasir berbasis *Android*. Halaman ini ditampilkan dalam bentuk jendela *pop-up* yang muncul di atas halaman pengaturan admin. Pada bagian atas *pop-up* terdapat judul “Edit Profil Toko” yang menunjukkan fungsi dari halaman tersebut.

Pada halaman ini ditampilkan informasi toko yang dapat diperbarui, yaitu nama toko, alamat, dan telepon. Administrator dapat mengubah data tersebut sesuai dengan informasi terbaru yang dimiliki oleh toko. Informasi yang diperbarui akan

digunakan sebagai identitas toko pada aplikasi serta dapat ditampilkan pada berbagai fitur, seperti laporan dan struk transaksi.

Di bagian bawah halaman terdapat tombol “Simpan Perubahan” yang berfungsi untuk menyimpan data profil toko yang telah diperbarui. Halaman ini membantu administrator dalam mengelola dan memperbarui informasi toko secara mudah, sehingga data yang tersimpan dalam sistem tetap akurat dan sesuai dengan kondisi toko yang sebenarnya.

5.3.16 Tampilan Halaman Kelola Kasir



Sumber: Penulis, 2026

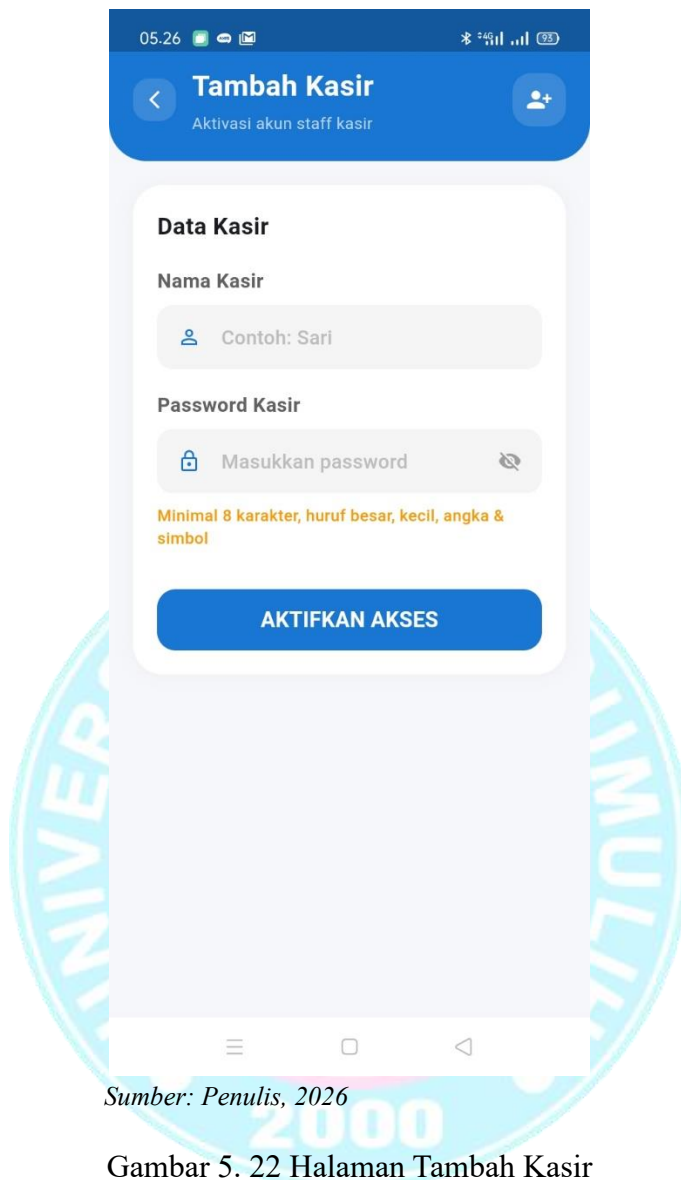
Gambar 5.21 Halaman Kelola Kasir

Halaman yang ditampilkan pada gambar merupakan halaman manajemen kasir pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh administrator untuk mengelola data akun kasir yang dapat mengakses aplikasi. Pada bagian atas halaman terdapat judul “Manajemen kasir” serta keterangan “Manajemen data” sebagai informasi mengenai fungsi halaman tersebut.

Pada halaman ini ditampilkan daftar data kasir yang telah terdaftar di dalam sistem, seperti nama kasir dan status sebagai staf kasir. Setiap data kasir dilengkapi dengan ikon untuk mengubah data kasir dan menghapus akun kasir dari sistem. Pada bagian bawah halaman terdapat tombol “Tambah kasir” yang digunakan untuk menambahkan akun kasir baru ke dalam aplikasi. Halaman ini membantu administrator dalam mengelola akun kasir secara lebih mudah dan terstruktur.



5.3.17 Tampilan Halaman Tambah Kasir



Sumber: Penulis, 2026

Gambar 5. 22 Halaman Tambah Kasir

Halaman yang ditampilkan pada gambar merupakan halaman tambah kasir yang digunakan oleh administrator untuk menambahkan dan mengaktifkan akun kasir pada aplikasi kasir berbasis *Android*. Pada bagian atas halaman terdapat judul “Tambah Kasir” serta keterangan “Aktivasi akun staff kasir” yang menjelaskan fungsi dari halaman tersebut. Pada halaman ini administrator dapat memasukkan nama kasir dan *password* kasir yang akan digunakan untuk *login* ke aplikasi. Data tersebut digunakan sebagai identitas akun kasir yang akan ditambahkan ke dalam sistem. Pada bagian *password* tersedia ikon mata yang berfungsi untuk

menampilkan atau menyembunyikan kata sandi sehingga administrator dapat memastikan data yang dimasukkan sudah benar.

Di bawah kolom *password* terdapat informasi mengenai ketentuan kata sandi, yaitu minimal terdiri dari 8 karakter yang mengandung huruf besar, huruf kecil, angka, dan simbol. Ketentuan ini diterapkan untuk meningkatkan keamanan akun kasir yang dibuat. Pada bagian bawah halaman terdapat tombol “Aktifkan Akses” yang digunakan untuk menyimpan data kasir dan mengaktifkan akun tersebut agar dapat digunakan untuk masuk ke aplikasi. Halaman ini membantu administrator dalam mengelola akun kasir secara mudah, aman, dan terstruktur.

5.3.18 Tampilan Halaman Edit Kasir

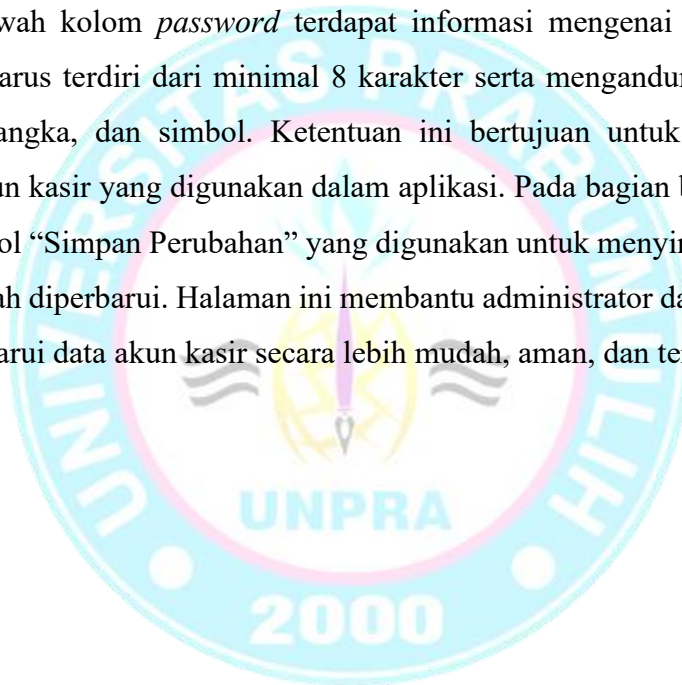
Sumber: Penulis, 2026

Gambar 5.23 Halaman Edit Kasir

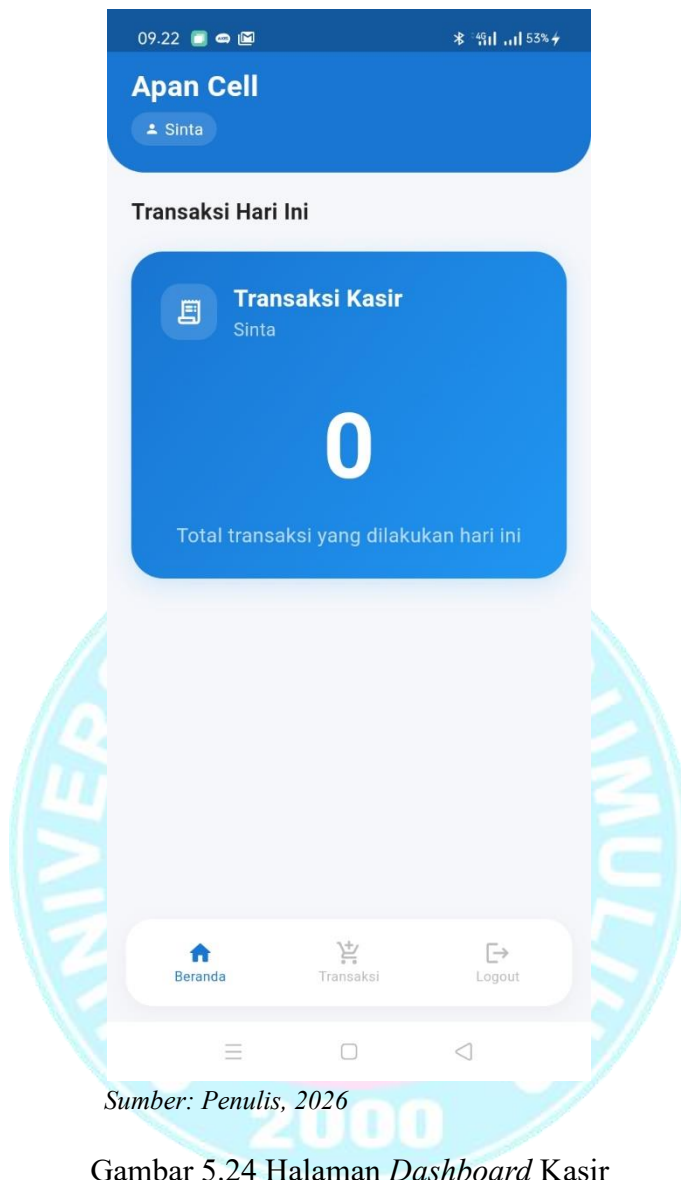
Halaman yang ditampilkan pada gambar diatas merupakan halaman edit kasir yang digunakan oleh administrator untuk mengubah atau memperbarui data akun kasir yang telah tersimpan di dalam sistem aplikasi kasir berbasis *Android*. Pada bagian atas halaman terdapat judul “Edit Kasir” serta keterangan “Perbarui data akun kasir” yang menunjukkan fungsi dari halaman tersebut.

Pada halaman ini ditampilkan informasi akun kasir yang dapat diperbarui, yaitu nama kasir dan *password* kasir. Administrator dapat mengubah data tersebut sesuai dengan kebutuhan atau informasi terbaru akun kasir. Pada bagian *password* tersedia ikon mata yang berfungsi untuk menampilkan atau menyembunyikan kata sandi sehingga administrator dapat memastikan data yang dimasukkan sudah benar.

Di bawah kolom *password* terdapat informasi mengenai ketentuan kata sandi, yaitu harus terdiri dari minimal 8 karakter serta mengandung huruf besar, huruf kecil, angka, dan simbol. Ketentuan ini bertujuan untuk meningkatkan keamanan akun kasir yang digunakan dalam aplikasi. Pada bagian bawah halaman terdapat tombol “Simpan Perubahan” yang digunakan untuk menyimpan data akun kasir yang telah diperbarui. Halaman ini membantu administrator dalam mengelola dan memperbarui data akun kasir secara lebih mudah, aman, dan terstruktur.



5.3.19 Tampilan Halaman *Dashboard* Kasir



Sumber: Penulis, 2026

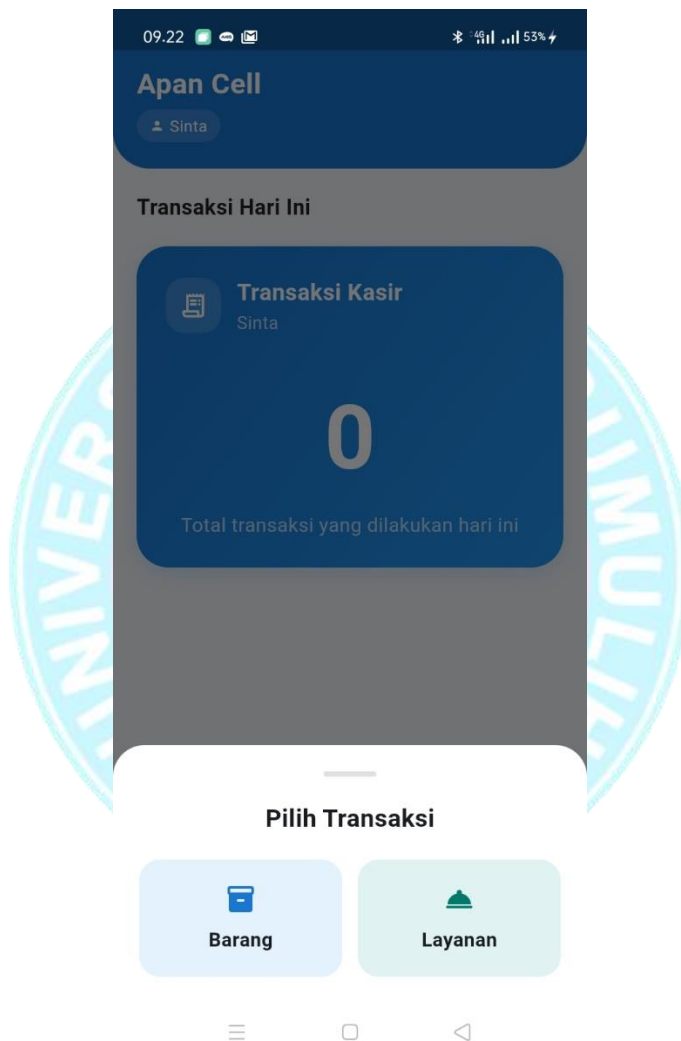
Gambar 5.24 Halaman *Dashboard* Kasir

Halaman yang ditampilkan pada gambar diatas merupakan halaman beranda kasir pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh staf kasir untuk melihat informasi transaksi yang dilakukan pada hari tersebut. Pada bagian atas halaman terdapat nama toko “Apan Cell” serta informasi nama kasir yang sedang menggunakan aplikasi.

Pada halaman ini ditampilkan informasi “Transaksi hari ini” yang berisi jumlah total transaksi yang telah dilakukan oleh kasir pada hari tersebut. Informasi tersebut membantu kasir dalam memantau aktivitas transaksi yang sedang berlangsung pada toko. Pada bagian bawah halaman terdapat menu navigasi, yaitu

beranda, transaksi dan *logout*. Menu beranda digunakan untuk kembali ke halaman utama kasir, menu transaksi digunakan untuk melakukan proses transaksi penjualan, sedangkan menu *logout* digunakan untuk keluar dari akun kasir yang sedang digunakan.

5.3.20 Tampilan Halaman Transaksi Kasir



Sumber: Penulis, 2026

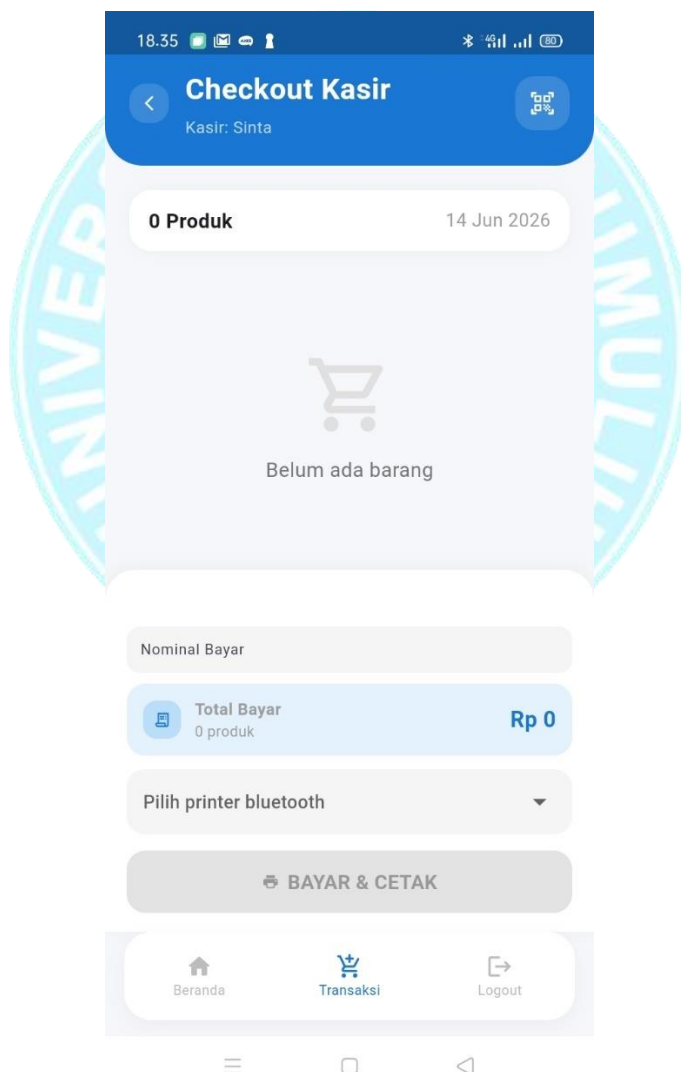
Gambar 5. 25 Halaman Transaksi Kasir

Halaman yang ditampilkan pada gambar merupakan halaman transaksi kasir pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh staf kasir untuk memilih jenis transaksi yang akan dilakukan pada aplikasi. Pada bagian bawah

halaman terdapat menu “Pilih transaksi” yang berisi dua pilihan menu, yaitu barang dan layanan.

Menu barang digunakan untuk melakukan transaksi penjualan produk yang tersedia pada toko. Melalui menu ini kasir dapat memilih produk yang akan dibeli oleh pelanggan dan memproses transaksi penjualan barang. Menu layanan digunakan untuk melakukan transaksi jasa atau layanan yang tersedia pada toko. Halaman ini membantu kasir dalam melakukan proses transaksi sesuai dengan jenis transaksi yang dipilih.

5.3.21 Tampilan Halaman *Checkout* Barang Kasir



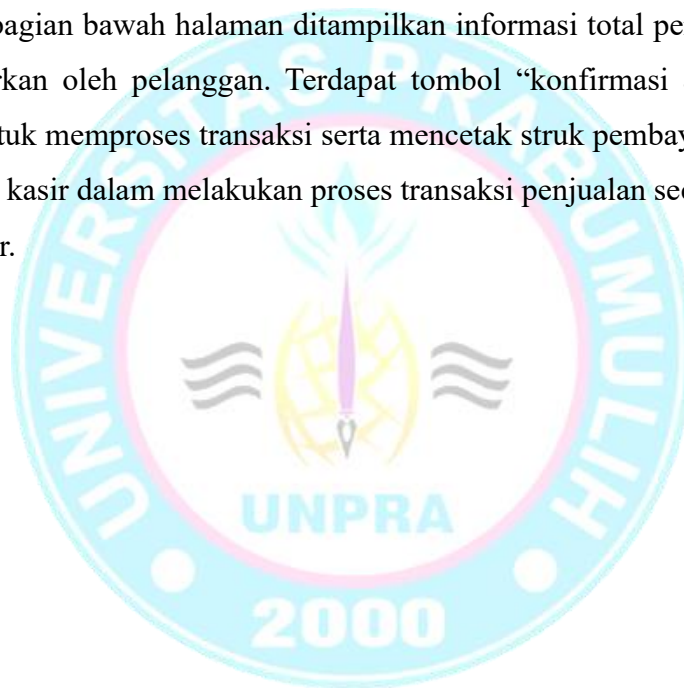
Sumber: Penulis, 2026

Gambar 5. 26 Halaman *Checkout* Kasir

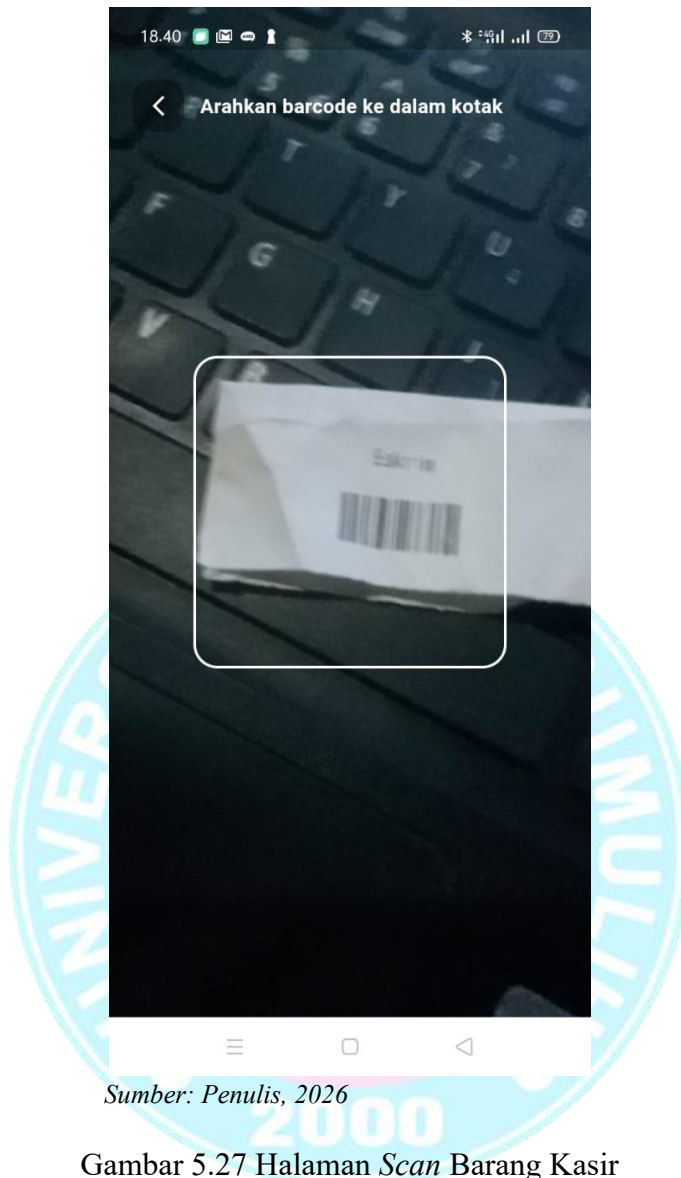
Halaman yang ditampilkan pada gambar merupakan halaman *checkout* kasir pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh staf kasir untuk melakukan proses pembayaran transaksi penjualan barang. Pada bagian atas halaman terdapat judul “*Checkout* kasir”, informasi nama kasir, serta ikon *barcode scanner* yang digunakan untuk memindai *barcode* produk saat proses transaksi.

Pada halaman ini ditampilkan informasi jumlah produk yang dipilih dalam transaksi serta tanggal transaksi yang sedang berlangsung. Apabila belum ada barang yang dipilih, maka sistem akan menampilkan informasi “Belum ada barang”. Halaman ini juga menyediakan pilihan *printer bluetooth* yang digunakan untuk memilih perangkat *printer* dalam proses pencetakan struk transaksi.

Pada bagian bawah halaman ditampilkan informasi total pembayaran yang harus dibayarkan oleh pelanggan. Terdapat tombol “konfirmasi & cetak” yang digunakan untuk memproses transaksi serta mencetak struk pembayaran. Halaman ini membantu kasir dalam melakukan proses transaksi penjualan secara lebih cepat dan terstruktur.



5.3.22 Tampilan Halaman *Scan* Barang Kasir



Sumber: Penulis, 2026

Gambar 5.27 Halaman *Scan* Barang Kasir

Halaman yang ditampilkan pada gambar merupakan halaman scan barang kasir pada aplikasi kasir berbasis *Android*. Halaman ini digunakan oleh staf kasir untuk melakukan proses pemindaian *barcode* barang saat transaksi penjualan berlangsung. Halaman ini akan ditampilkan ketika kasir menekan tombol *scan* barang pada menu transaksi kasir.

Pada bagian atas halaman terdapat informasi “arahkan *barcode* ke dalam kotak” sebagai petunjuk penggunaan fitur pemindaian. Di bagian tengah halaman terdapat area kotak pemindaian yang digunakan untuk mendeteksi dan membaca *barcode* produk menggunakan kamera perangkat. Halaman *scan* barang ini

membantu kasir dalam memasukkan data produk ke dalam transaksi secara lebih cepat tanpa perlu memasukkan data barang secara manual.

5.3.23 Tampilan Halaman Transaksi Layanan Kasir

Sumber: Penulis, 2026

Gambar 5. 28 Halaman Transaksi Layanan Kasir

Halaman yang ditampilkan pada gambar merupakan halaman Transaksi Layanan yang digunakan oleh staf kasir untuk melakukan transaksi layanan atau jasa yang tersedia pada aplikasi kasir berbasis *Android*. Pada bagian atas halaman terdapat tombol kembali, judul “Transaksi Layanan”, informasi nama petugas yang sedang bertugas, serta ikon keranjang yang digunakan untuk melihat daftar transaksi layanan yang telah dipilih.

Pada bagian detail layanan tersedia beberapa kolom yang harus diisi oleh kasir, yaitu jenis layanan, nominal, dan catatan. Kolom jenis layanan digunakan untuk memilih layanan yang akan diproses. Kolom nominal digunakan untuk memasukkan nilai transaksi atau biaya layanan yang dibayarkan pelanggan. Sementara itu, kolom catatan digunakan untuk menambahkan informasi pendukung, seperti nomor telepon pelanggan atau keterangan lain yang berkaitan dengan transaksi layanan.

Di bagian bawah halaman terdapat tombol “Simpan Transaksi” yang berfungsi untuk menyimpan data transaksi layanan ke dalam sistem. Setelah data berhasil disimpan, transaksi akan tercatat dan dapat ditampilkan pada laporan penjualan layanan. Halaman transaksi layanan membantu staf kasir dalam melakukan proses transaksi layanan secara lebih mudah, cepat, dan terstruktur sehingga pencatatan transaksi dapat dilakukan dengan lebih akurat.

5.4 Hasil Pengujian Sistem

5.4.1 Pengujian Fungsi Register

- a. Potongan *source code* fungsi register

```
Future<void> register() async {
    if (tokoC.text.trim().isEmpty) {
        _showSnack("Nama toko wajib diisi");
        return;
    }

    if (alamatC.text.trim().isEmpty) {
        _showSnack("Alamat wajib diisi");
        return;
    }

    if (telpC.text.trim().isEmpty) {
        _showSnack("Nomor HP wajib diisi");
        return;
    }

    if (!IsValidEmail) {
        _showSnack("Format email tidak valid");
        return;
    }
}
```

```

if (!isPasswordStrong) {
  _showSnack("Password lemah");
  return;
}

if (!isPasswordMatch) {
  _showSnack("Password tidak cocok");
  return;
}

await FirebaseAuth.instance
  .createUserWithEmailAndPassword(
    email: emailC.text.trim(),
    password: passwordC.text.trim(),
  );

  _showSnack("Registrasi berhasil");
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan setiap statement (baris kode) dieksekusi minimal satu kali. Target pengujian pada fungsi register adalah sebagai berikut:

1. Semua percabangan *if*.
2. Semua validasi *input* register.
3. Semua pesan *error* validasi.
4. Proses registrasi akun *Firebase Authentication*.
5. Pesan registrasi berhasil.
6. Semua *return* pada kondisi *error*.

c. Identifikasi *Statement*

Tabel 5.1 Identifikasi *Statement* Fungsi Register

No	Kode Program	Keterangan
S1	if (tokoC.text.trim().isEmpty)	Memeriksa nama toko kosong
S2	_showSnack("Nama toko wajib diisi")	Menampilkan pesan nama toko kosong
S3	if (alamatC.text.trim().isEmpty)	Memeriksa alamat kosong

S4	_showSnack("Alamat wajib diisi")	Menampilkan pesan alamat kosong
S5	if (telpC.text.trim().isEmpty)	Memeriksa nomor HP kosong
S6	_showSnack("Nomor HP wajib diisi")	Menampilkan pesan nomor HP kosong
S7	if (!isValidEmail)	Memeriksa format <i>email</i>
S8	_showSnack("Format email tidak valid")	Menampilkan pesan <i>email</i> tidak valid
S9	if (!isPasswordStrong)	Memeriksa kekuatan <i>password</i>
S10	_showSnack("Password lemah")	Menampilkan pesan <i>password</i> lemah
S11	if (!isPasswordMatch)	Memeriksa kecocokan <i>password</i>
S12	_showSnack("Password tidak cocok")	Menampilkan pesan <i>password</i> tidak cocok
S13	createUserWithEmailAndPassword()	Proses registrasi akun

Sumber: Penulis, 2026

d. *Test case* untuk *Statement Coverage*

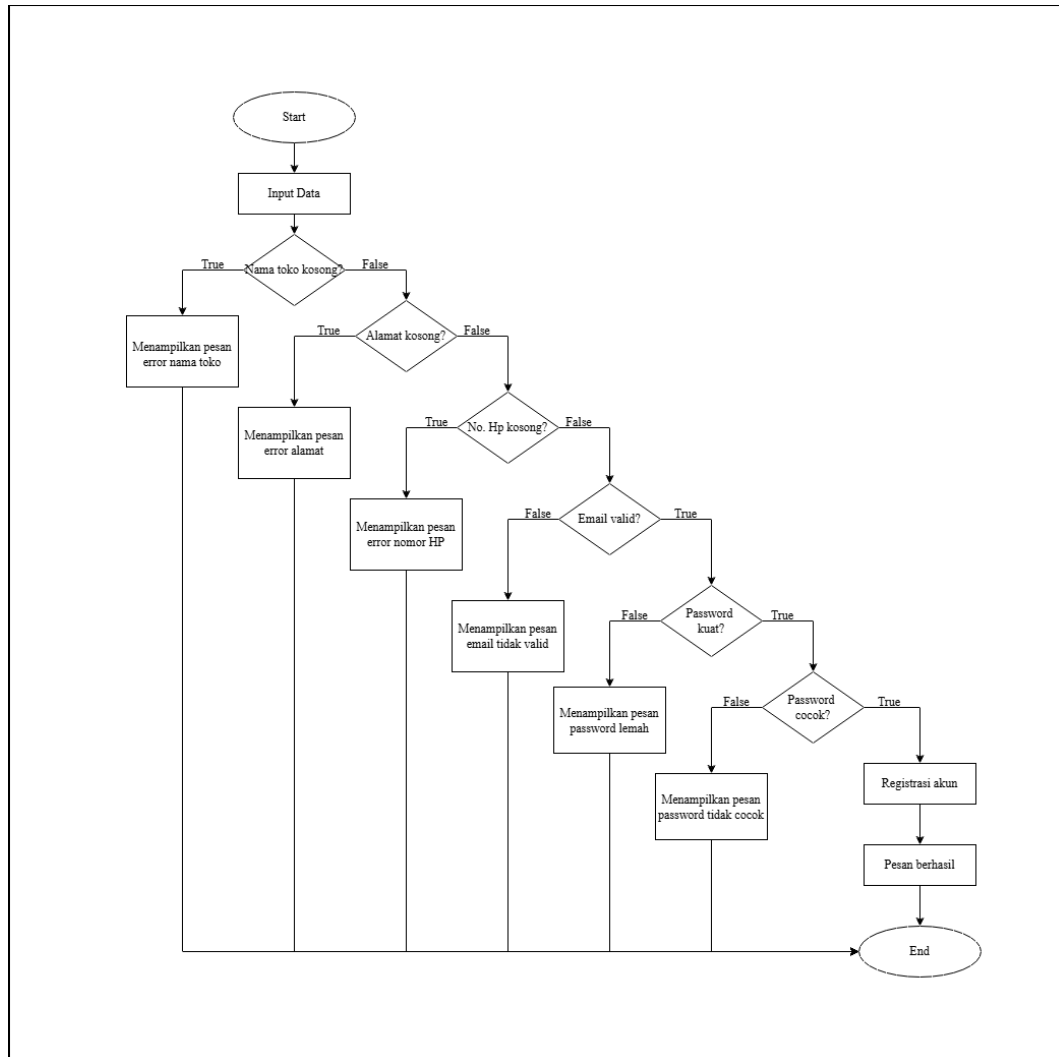
Tabel 5.2 *Test Case* untuk *Statement Coverage* fungsi Register

<i>Test Case</i>	<i>Nama Toko</i>	<i>Alamat</i>	<i>No HP</i>	<i>Email</i>	<i>Password</i>	<i>Jalur Eksekusi</i>	<i>Output</i>
TC1	kosong	kosong	kosong	kosong	kosong	S1 → S2 → S3	Validasi gagal
TC2	Toko Maju	kosong	0812345 6789	user@gm ail.com	Abc@1234	S1 → S4 → S5 → S6	Validasi alamat gagal
TC3	Toko Maju	Aceh	kosong	user@gm ail.com	Abc@1234	S1 → S4 → S7 → S8 → S9	Validasi nomor HP gagal
TC4	Toko Maju	Aceh	0812345 6789	salahgma il.com	Abc@1234	S1 → S4 → S7 → S10 → S11 → S12	Format <i>email</i> tidak valid
TC5	Toko Maju	Aceh	0812345 6789	user@gm ail.com	123	S1 → S4 → S7 → S10 → S13 → S14 → S15	<i>Passwor</i> <i>d</i> lemah

TC6	Toko Maju	Aceh	0812345 6789	user@gm ail.com	Abc@1111	S1 → S4	<i>Passwor</i> <i>d tidak</i> <i>cocok</i>
						→ S7 →	
						S10 → S13	
						→ S16 →	
						S17 → S18	
TC7	Toko Maju	Aceh	0812345 6789	user@gm ail.com	Abc@1234	S1 → S4	Registra si berhasil
						→ S7 →	
						S10 → S13	
						→ S16 →	
						S19 → S20	
						→ S21	

Sumber: Penulis, 2026



e. *Flowchart Register*

Sumber: Penulis, 2026

Gambar 5.29 *Flowchart Register*

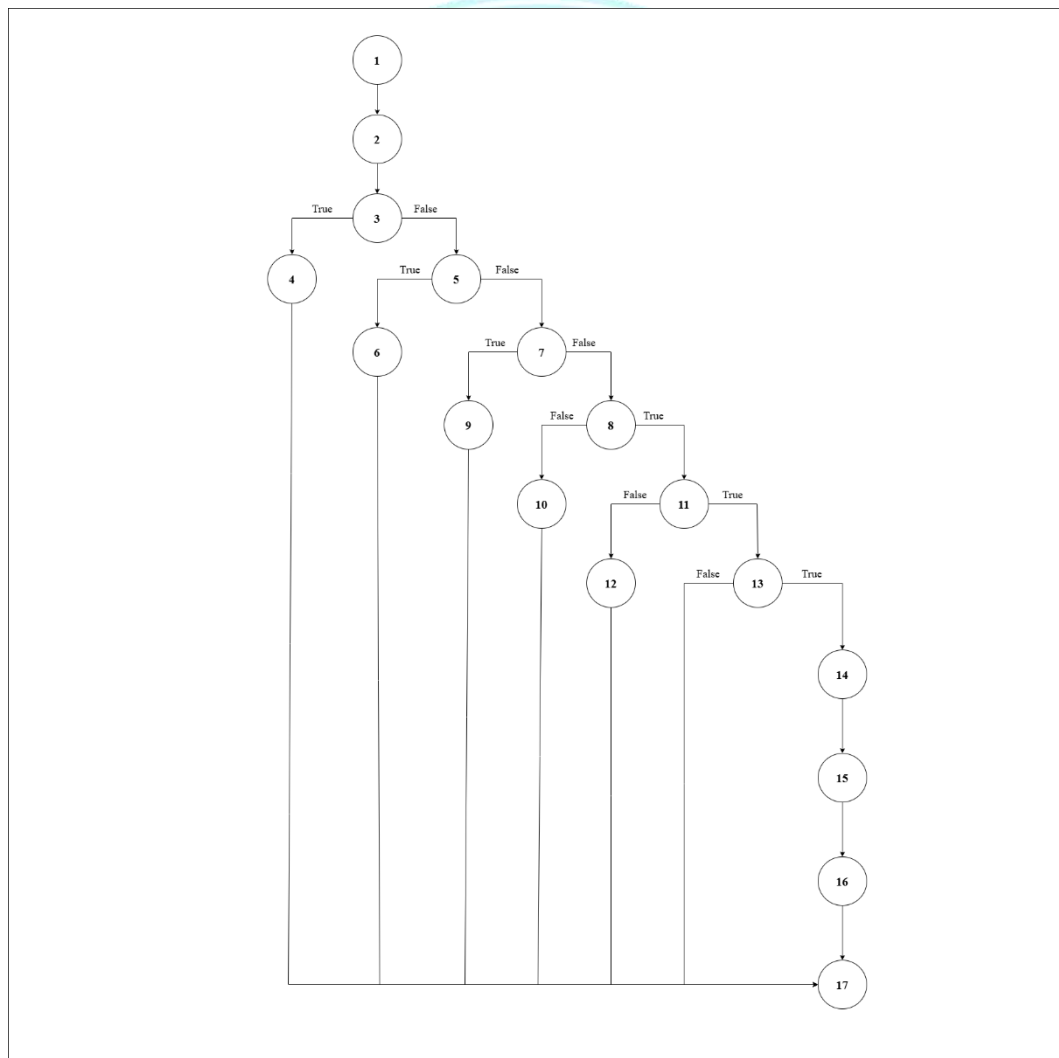
f. *Flowgraph Register*

Node dan Edge

Node:

1. *Start*
2. *Input data register*
3. *Validasi nama toko*
4. *Error nama toko kosong*
5. *Validasi alamat*
6. *Error alamat kosong*
7. *Validasi nomor HP*

8. Error nomor HP kosong
9. Validasi format *email*
10. Error *email* tidak valid
11. Validasi *password*
12. Error *password* lemah
13. Validasi konfirmasi *password*
14. Error *password* tidak cocok
15. Proses *Firestore Authentication*
16. Registrasi berhasil
17. End



Sumber: Penulis, 2026

Gambar 5.30 Flowgraph Register

g. Jalur *Independen Path*

1. *Path 1*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 17$
(Nama toko kosong)
2. *Path 2*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 17$
(Alamat kosong)
3. *Path 3*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 17$
(Nomor HP kosong)
4. *Path 4*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 10 \rightarrow 17$
(Format *email* tidak valid)
5. *Path 5*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 12 \rightarrow 17$
(*Password* lemah)
6. *Path 6*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 13 \rightarrow 14 \rightarrow 17$
(*Password* tidak cocok)
7. *Path 7*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 13 \rightarrow 15 \rightarrow 16 \rightarrow 17$
(Registrasi berhasil)

5.4.2 Pengujian Fungsi Login

a. Potongan *source code* fungsi *login*

```
Future<void> login() async {
  if (emailC.text.trim().isEmpty) {
    _showSnackBar("Email wajib diisi");
    return;
  }

  if (!isValidEmail) {
    _showSnackBar("Format email tidak valid");
    return;
  }

  if (passwordC.text.trim().isEmpty) {
    _showSnackBar("Password wajib diisi");
    return;
  }

  var adminQuery = await FirebaseFirestore.instance
    .collection('admin')
    .where('email', isEqualTo: emailInput)
    .get();
```

```

if (adminQuery.docs.isEmpty) {
  _showSnackBar("Akun toko tidak ditemukan");
  return;
}

var kasirQuery = await FirebaseFirestore.instance
  .collection('kasir')
  .where('password', isEqualTo: passInput)
  .get();

if (kasirQuery.docs.isNotEmpty) {
  _showSnackBar("Login Kasir Berhasil");
} else {
  _showSnackBar("Login Admin Berhasil");
}
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan setiap statement (baris kode) dieksekusi minimal satu kali. Target pengujian pada fungsi *login* adalah sebagai berikut:

1. Semua percabangan *if* dan *else*.
2. Semua validasi input *login*.
3. Semua proses *Firebase Authentication* dan *Firestore*.
4. Semua pesan *error login*.
5. Semua proses *login* admin dan kasir.
6. Semua *return* pada kondisi *error*.

c. Identifikasi *Statement*

Tabel 5.3 Identifikasi *Statement* fungsi *Login*

No	Kode Program	Keterangan
S1	if (emailC.text.trim().isEmpty)	Memeriksa <i>email</i> kosong

S2	_showSnackBar("Email wajib diisi")	Menampilkan pesan <i>email</i> kosong
S3	if (!isValidEmail)	Memeriksa format <i>email</i>
S4	_showSnackBar("Format email tidak valid")	Menampilkan pesan email tidak valid
S5	if (passwordC.text.trim().isEmpty)	Memeriksa <i>password</i> kosong
S6	_showSnackBar("Password wajib diisi")	Menampilkan pesan <i>password</i> kosong
S7	adminQuery.docs.isEmpty	Memeriksa akun admin tersedia
S8	_showSnackBar("Akun toko tidak ditemukan")	Menampilkan pesan akun tidak ditemukan
S9	kasirQuery.docs.isNotEmpty	Memeriksa <i>login</i> kasir
S10	_showSnackBar("Login Kasir Berhasil")	Menampilkan pesan <i>login</i> kasir berhasil
S11	_showSnackBar("Login Admin Berhasil")	Menampilkan pesan <i>login</i> admin berhasil

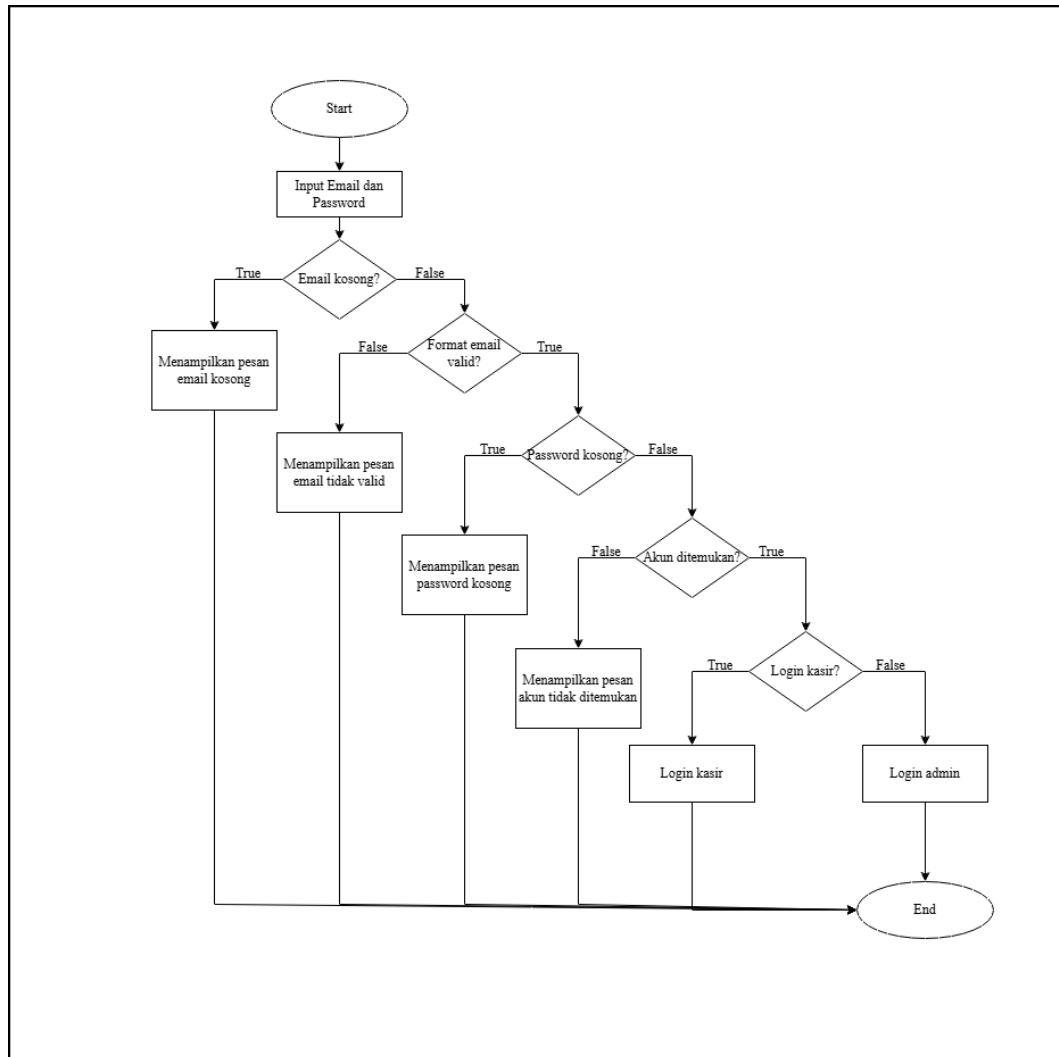
Sumber: Penulis, 2026

d. *Test Case* untuk *Statement Coverage*

Tabel 5.4 *Test Case* untuk *Statement Coverage* fungsi *Login*

<i>Test Case</i>	<i>Email</i>	<i>Password</i>	Jalur Eksekusi	<i>Output</i>
TC1	kosong	kosong	S1 → S2 → S3	Validasi gagal
TC2	salahgmail.com	Abc@1234	S1 → S4 → S5 → S6	Format <i>email</i> tidak valid
TC3	admin@gmail.com	kosong	S1 → S4 → S7 → S8 → S9	<i>Password</i> wajib diisi
TC4	user@gmail.com	Abc@1234	S1 → S4 → S7 → S10 → S11 → S12 → S13	Akun toko tidak ditemukan
TC5	kasir@gmail.com	Abc@1234	S1 → S4 → S7 → S10 → S11 → S14 → S15 → S16	<i>Login</i> kasir berhasil
TC6	admin@gmail.com	Abc@1234	S1 → S4 → S7 → S10 → S11 → S14 → S15 → S17	<i>Login</i> admin berhasil

Sumber: Penulis, 2026

e. *Flowchart Login*

Sumber: Penulis, 2026

Gambar 5.31 *Flowchart Login*

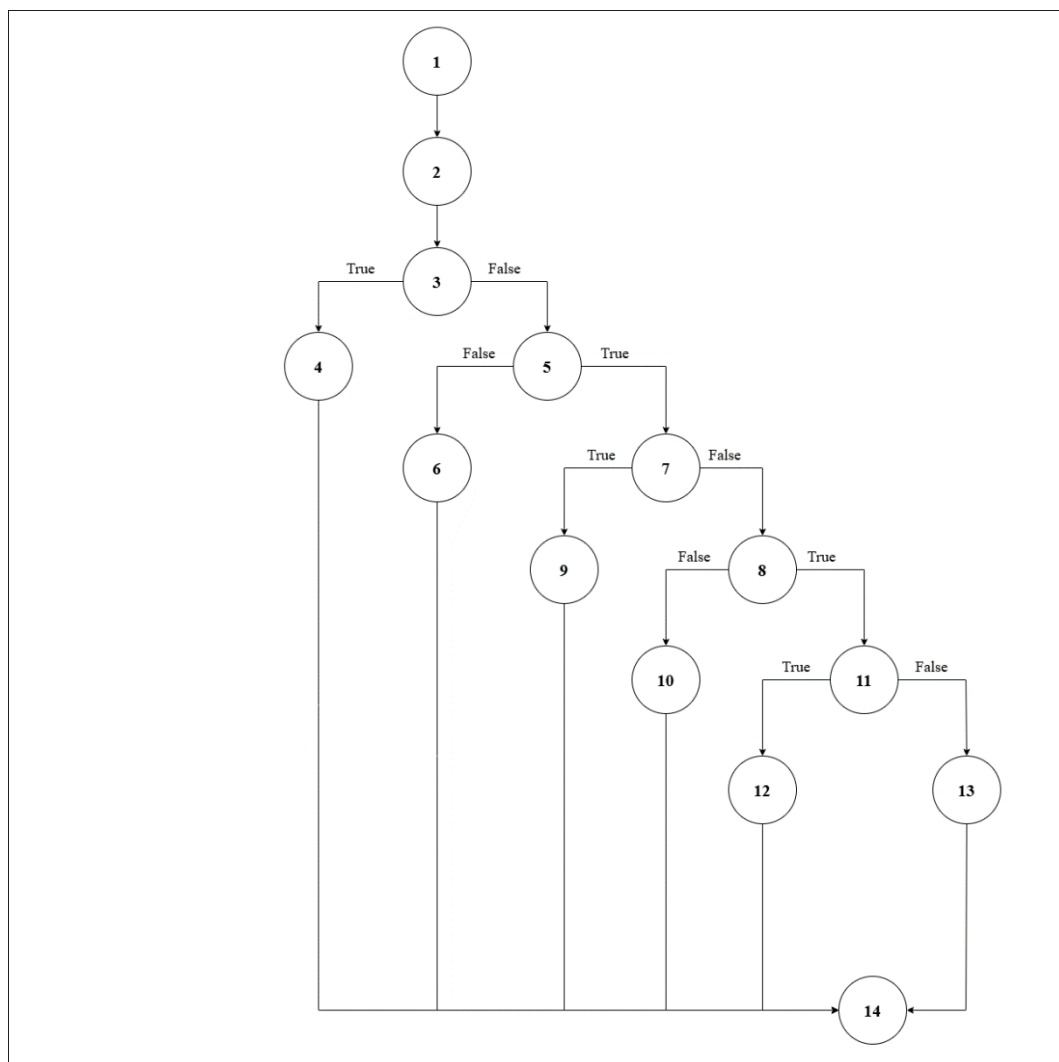
f. *Flowgraph Login*

Node dan Edge

Node:

1. *Start*
2. *Input email dan password*
3. *Validasi email kosong*
4. *Error email kosong*
5. *Validasi format email*
6. *Error format email tidak valid*

7. Validasi *password* kosong
8. *Error password* kosong
9. Validasi akun admin
10. *Error* akun tidak ditemukan
11. Validasi *login* kasir
12. *Login* kasir berhasil
13. *Login* admin berhasil
14. *End*



Sumber: Penulis, 2026

Gambar 5.32 Flowgraph Login

g. Jalur Idenpenden Path

1. Path 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 14$
(Email kosong)
2. Path 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 14$
(Format email tidak valid)
3. Path 3: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 14$
(Password kosong)
4. Path 4: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 10 \rightarrow 14$
(Akun tidak ditemukan)
5. Path 5: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 12 \rightarrow 14$
(Login kasir berhasil)
6. Path 6: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 13 \rightarrow 14$
(Login admin berhasil)

5.4.3 Pengujian Fungsi Tambah Barang

a. Potongan Source Code Fungsi Tambah Barang

```
Future<void> simpanDanCetak() async {

  if (namaC.text.isEmpty ||
      hargaBeliC.text.isEmpty ||
      hargaC.text.isEmpty ||
      stokC.text.isEmpty) {

    ScaffoldMessenger.of(context).showSnackBar(
      const SnackBar(
        content: Text(
          "Semua data wajib diisi",
        ),
      ),
    );

    return;
  }

  if (selectedDeviceMac == null) {

    ScaffoldMessenger.of(context).showSnackBar(
      const SnackBar(
```

```

        content: Text(
          "Pilih printer barcode terlebih dahulu",
        ),
      ),
    );

    return;
  }

  final String? uid =
    FirebaseAuth.instance.currentUser?.uid;

  if (uid == null) return;

  finalBarcode =
    DateTime.now()
      .millisecondsSinceEpoch
      .toString()
      .substring(7);

  await FirebaseFirestore.instance
    .collection('barang')
    .doc(finalBarcode)
    .set({
      'nama': namaC.text,
      'harga_beli': int.parse(hargaBeliC.text),
      'harga': int.parse(hargaC.text),
      'stok': int.parse(stokC.text),
    });

  await _printLogic(
    namaC.text,
    finalBarcode,
    int.parse(stokC.text),
  );

  ScaffoldMessenger.of(context).showSnackBar(
    const SnackBar(
      content: Text(
        "Barang berhasil disimpan",
      ),
    ),
  );
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan *if*.
2. Semua validasi input data barang.
3. Semua pesan *error* tambah barang.
4. Validasi *printer barcode*.
5. Validasi *user login*.
6. Proses *generate barcode*.
7. Proses penyimpanan data barang.
8. Proses cetak *barcode*.
9. Pesan barang berhasil disimpan.
10. Semua *return* pada kondisi *error*.

c. Identifikasi *Statement*

Tabel 5.5 Identifikasi *Statement* fungsi Tambah Barang

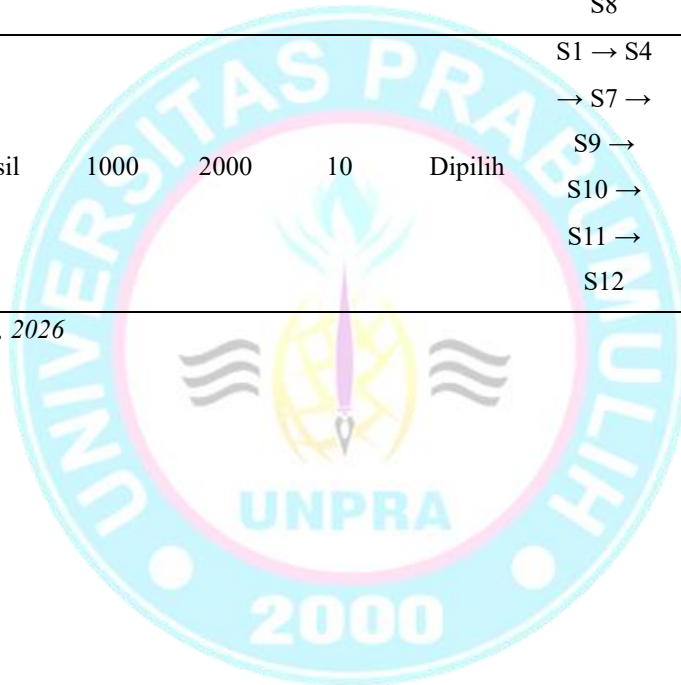
No	Kode Program	Keterangan
S1	<code>if (namaC.text.isEmpty...)</code>	Memeriksa data barang kosong
S2	<code>SnackBar("Semua data wajib diisi")</code>	Menampilkan pesan data kosong
S3	<code>if (selectedDeviceMac == null)</code>	Memeriksa <i>printer barcode</i>
S4	<code>SnackBar("Pilih printer barcode")</code>	Menampilkan pesan <i>printer</i> belum dipilih
S5	<code>if (uid == null)</code>	Memeriksa <i>user login</i>
S6	<code>finalBarcode = ...</code>	Membuat <i>barcode</i> otomatis
S7	<code>.set({...})</code>	Menyimpan data barang
S8	<code>_printLogic()</code>	Proses cetak <i>barcode</i>
S9	<code>SnackBar("Barang berhasil disimpan")</code>	Menampilkan pesan berhasil

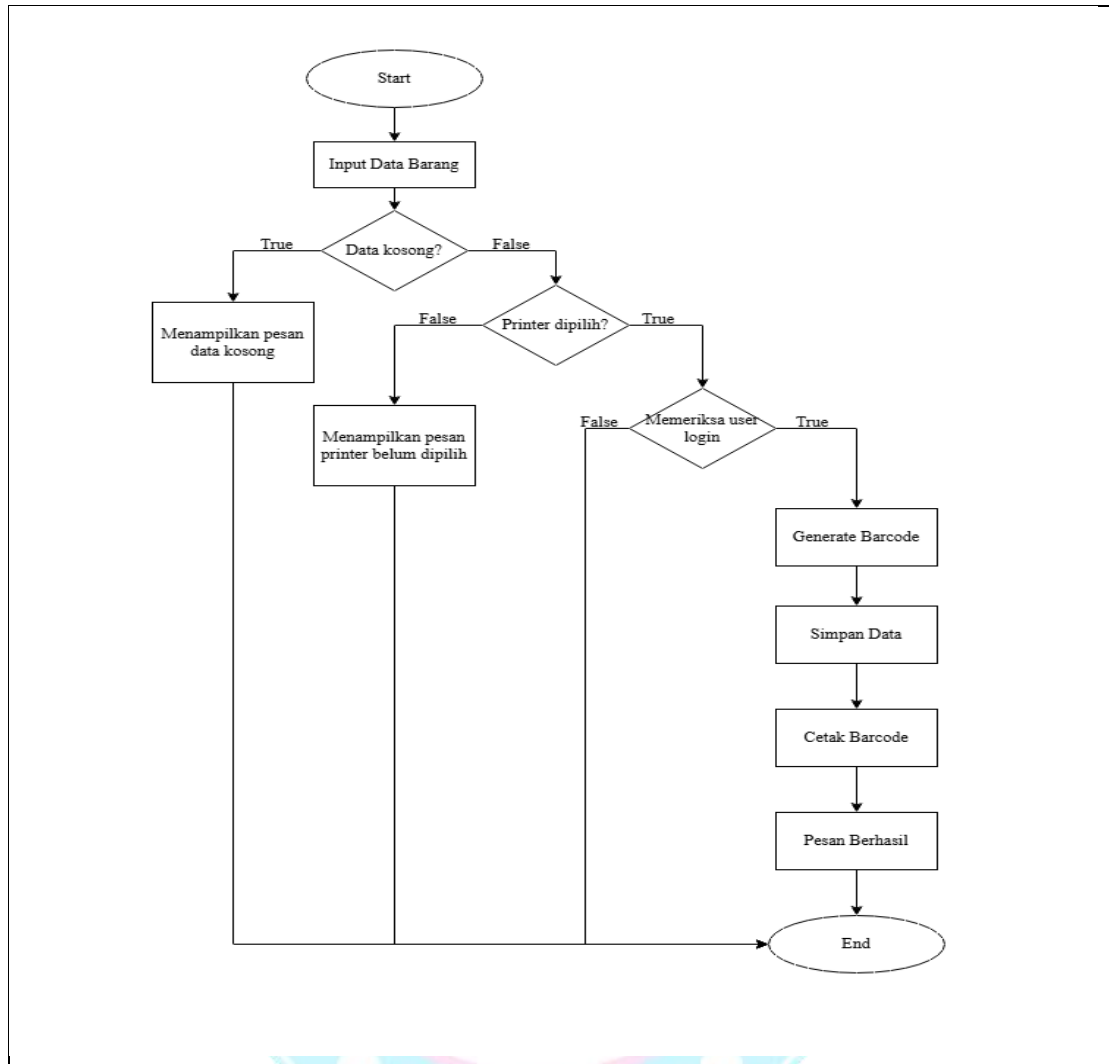
Sumber: Penulis, 2026

d. *Test Case* untuk *Statement Coverage*Tabel 5.6 *Test Case* untuk *Statement Coverage* fungsi Tambah Barang

<i>Test Case</i>	<i>Nama Barang</i>	<i>Harga Beli</i>	<i>Harga Jual</i>	<i>Stok</i>	<i>Printer Barcode</i>	<i>Jalur Eksekusi</i>	<i>Output</i>
TC1	kosong	kosong	kosong	kosong	Belum dipilih	S1 → S2 → S3	Validasi data gagal
TC2	Pensil	1000	2000	10	Belum dipilih	S1 → S4 → S5 → S6	<i>Printer barcode</i> belum dipilih
TC3	Pensil	1000	2000	10	Dipilih	S1 → S4 → S7 → S8	<i>User</i> belum login
TC4	Pensil	1000	2000	10	Dipilih	S1 → S4 → S7 → S9 → S10 → S11 → S12	Barang berhasil disimpan dan <i>barcode</i> dicetak

Sumber: Penulis, 2026



e. *Flowchart* Tambah Barang

Sumber: Penulis, 2026

Gambar 5.33 *Flowchart* Tambah Barang

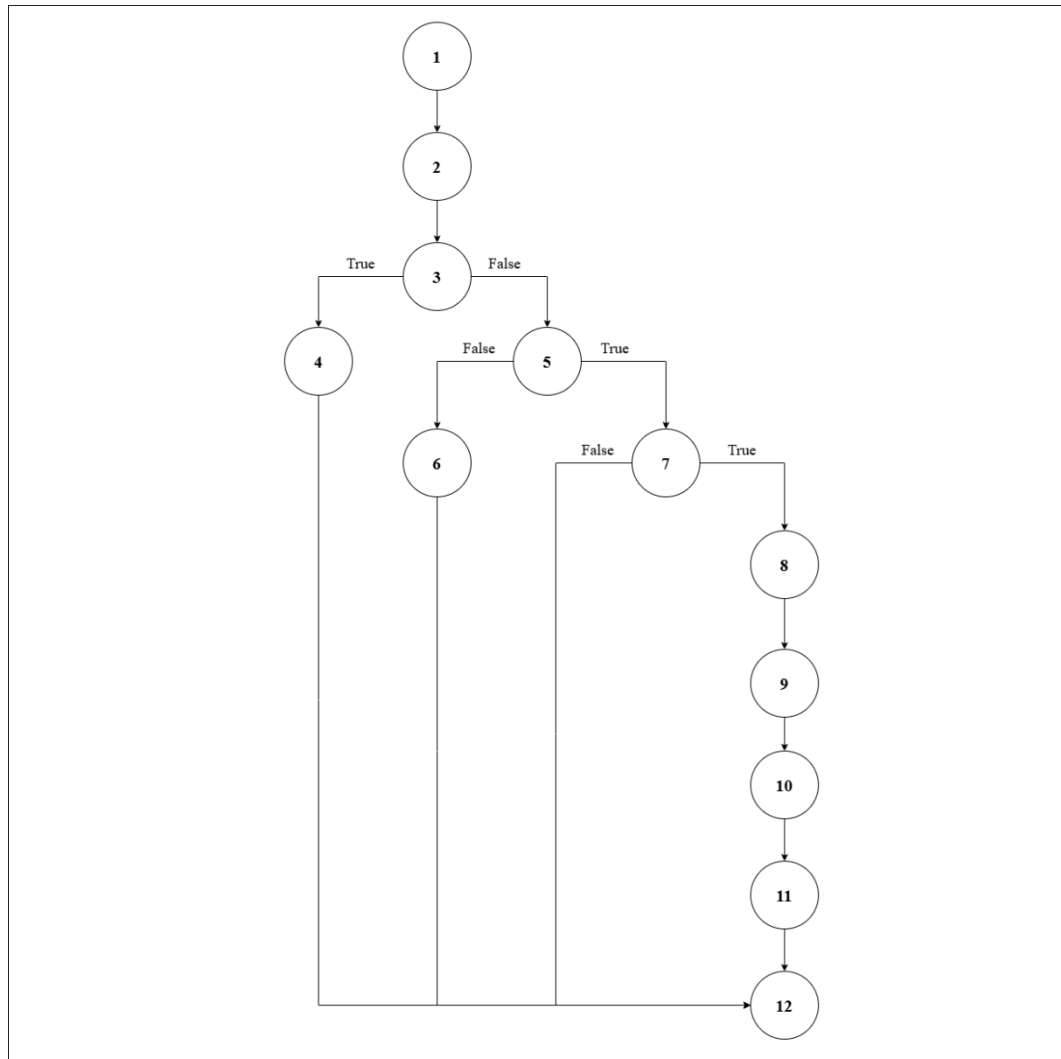
f. *Flowgraph* Tambah Barang

Node dan Edge

Node:

1. *Start*
2. *Input data barang*
3. *Validasi data barang kosong*
4. *Error data barang kosong*
5. *Validasi printer barcode*
6. *Error printer barcode belum dipilih*

7. Validasi *user login*
8. *Generate barcode* otomatis
9. Simpan data barang
10. Cetak *barcode*
11. Pesan barang berhasil disimpan
12. *End*



Sumber: Penulis, 2026

Gambar 5.34 *Flowgraph* Tambah Barang

g. Jalur Independen *Path*

1. *Path* 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 12$
(Data barang kosong)

2. *Path 2*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 12$
(*Printer barcode* belum dipilih)
3. *Path 3*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 12$
(*User* belum *login*)
4. *Path 4*: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 12$
(Tambah barang berhasil)

5.4.4 Pengujian Fungsi Edit Barang

- a. Potongan *Source Code* Fungsi Edit Barang

```
Future<void> simpanDanCetak() async {
```

```
    if (namaC.text.isEmpty ||  
        hargaBeliC.text.isEmpty ||  
        hargaC.text.isEmpty ||  
        stokC.text.isEmpty) {
```

```
        ScaffoldMessenger.of(context)
```

```
            .showSnackBar(
```

```
                const SnackBar(
```

```
                    content: Text(
```

```
                        "Semua data wajib diisi",
```

```
                    ),
```

```
                ),
```

```
            );
```

```
        return;
```

```
    }
```

```
    if (selectedDeviceMac == null) {
```

```
        ScaffoldMessenger.of(context)
```

```

        .showSnackBar(
        const SnackBar(
            content: Text(
                "Pilih printer barcode terlebih dahulu",
            ),
        ),
    );

    return;
}

final String? uid =
    FirebaseAuth.instance.currentUser?.uid;

if (uid == null) return;

setState(() => isProcessing = true);

try {

    String finalBarcode;

    if (widget.editData != null) {

        finalBarcode =
            widget.editData!['barcode'];

        await FirebaseFirestore.instance
            .collection('barang')
            .doc(widget.editData!.id)
            .update({

```

```

        'nama': namaC.text,
        'harga_beli':
            int.parse(hargaBeliC.text),
        'harga':
            int.parse(hargaC.text),
        'stok':
            int.parse(stokC.text),
    });
}

```

```

await _printLogic(
    namaC.text,
    finalBarcode,
    int.parse(stokC.text),
);

Navigator.pop(context);

} finally {

    setState(() => isProcessing = false);
}
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan if.
2. Semua validasi data barang.
3. Semua pesan *error* edit barang.

4. Validasi *printer barcode*.
5. Validasi *user login*.
6. Proses *loading* data.
7. Validasi mode edit barang.
8. Proses *update* data barang.
9. Proses cetak *barcode*.
10. Proses kembali ke halaman sebelumnya.
11. Proses *loading* selesai.
12. Semua *return* pada kondisi error.

c. Identifikasi *Statement*

Tabel 5.7 Identifikasi *Statement* fungsi Edit Barang

No	Kode Program	Keterangan
S1	if (namaC.text.isEmpty...)	Memeriksa input barang
S2	SnackBar("Semua data wajib diisi")	Menampilkan pesan data kosong
S3	if (selectedDeviceMac == null)	Memeriksa <i>printer barcode</i>
S4	SnackBar("Pilih printer barcode terlebih dahulu")	Menampilkan pesan <i>printer</i> belum dipilih
S5	if (uid == null)	Memeriksa <i>user login</i>
S6	setState(() => isProcessing = true)	Mengaktifkan <i>loading</i>
S7	if (widget.editData != null)	Memeriksa mode edit barang
S8	.collection('barang').doc(...).update({...})	Memperbarui data barang
S9	_printLogic(...)	Mencetak <i>barcode</i>
S10	Navigator.pop(context)	Kembali ke halaman sebelumnya
S11	setState(() => isProcessing = false)	Menonaktifkan <i>loading</i>

Sumber : penulis, 2026

d. Test Case untuk *Statement Coverage*

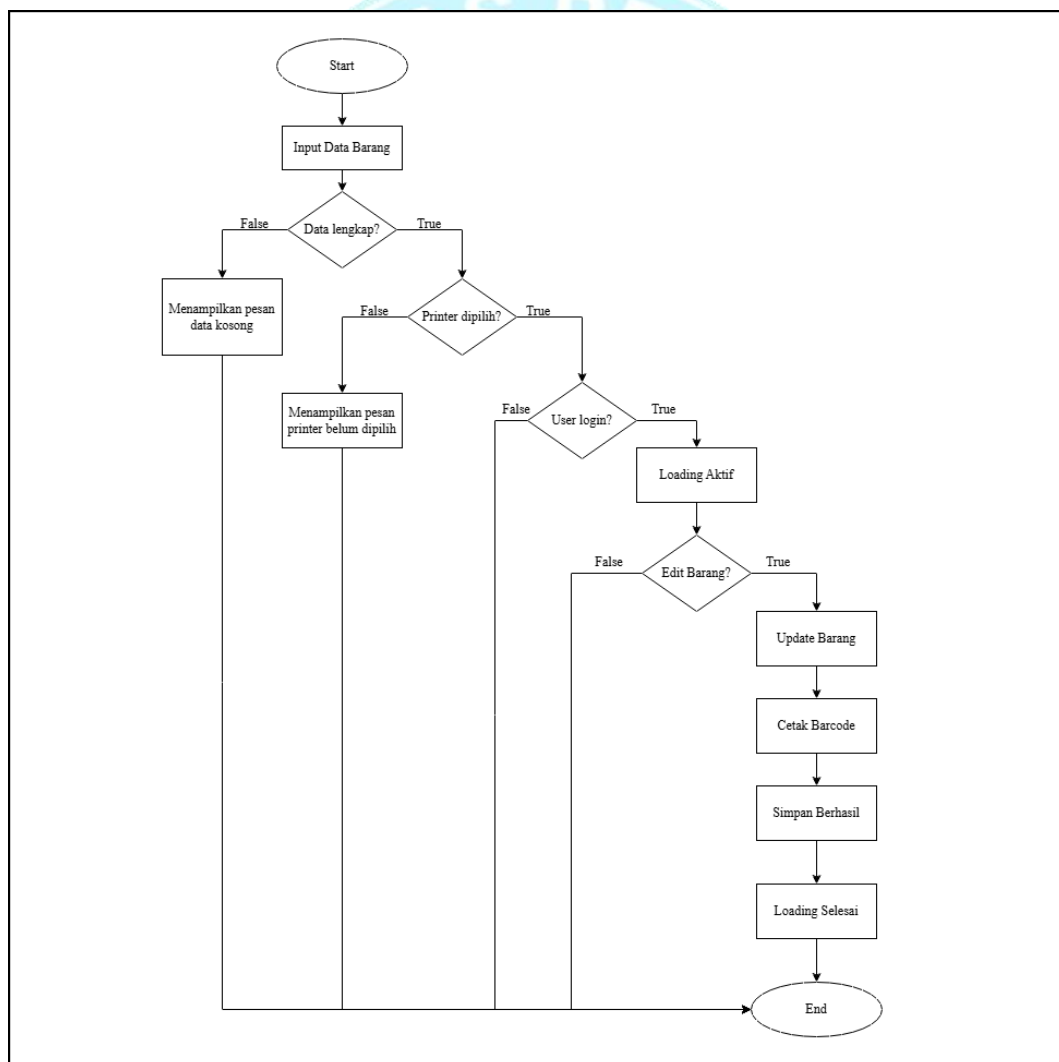
Tabel 5.8 Test Case untuk *Statement Coverage* fungsi Edit Barang

Test Case	Kondisi	Jalur Eksekusi	Output
TC1	Data barang tidak lengkap	S1 → S2 → S3	Sistem menampilkan pesan semua data wajib diisi

TC2	Printer <i>barcode</i> belum dipilih	S1 → S4 → S5 → S6	Sistem menampilkan pesan pilih <i>printer barcode</i> terlebih dahulu
TC3	User belum <i>login</i>	S1 → S4 → S7 → S8	Proses edit barang dihentikan
TC4	Data edit barang tidak ditemukan	S1 → S4 → S7 → S9 → S10	Proses edit barang dihentikan
TC5	Semua data barang valid	S1 → S4 → S7 → S9 → S11 → S12 → S13 → S14	Data barang berhasil diperbarui dan <i>barcode</i> dicetak

Sumber: Penulis, 2026

e. Flowchart Edit Barang



Sumber: Penulis, 2026

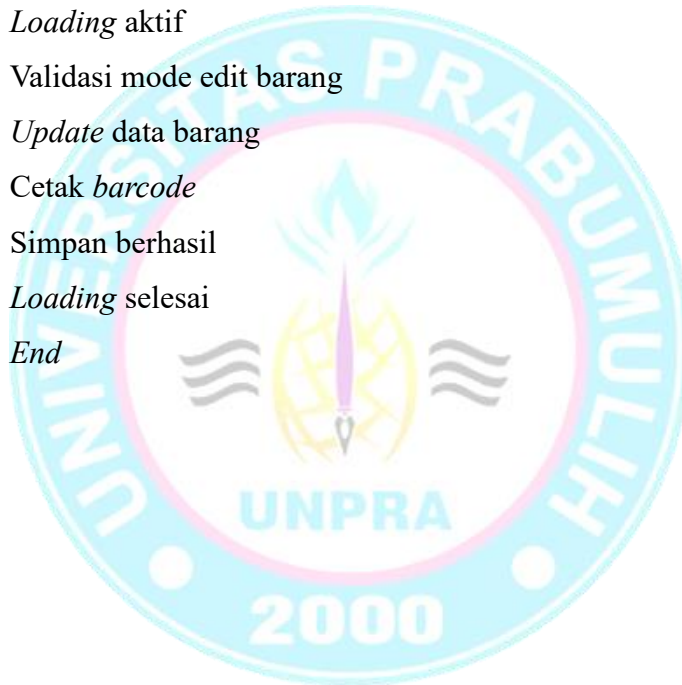
Gambar 5. 35 Flowchart Edit Barang

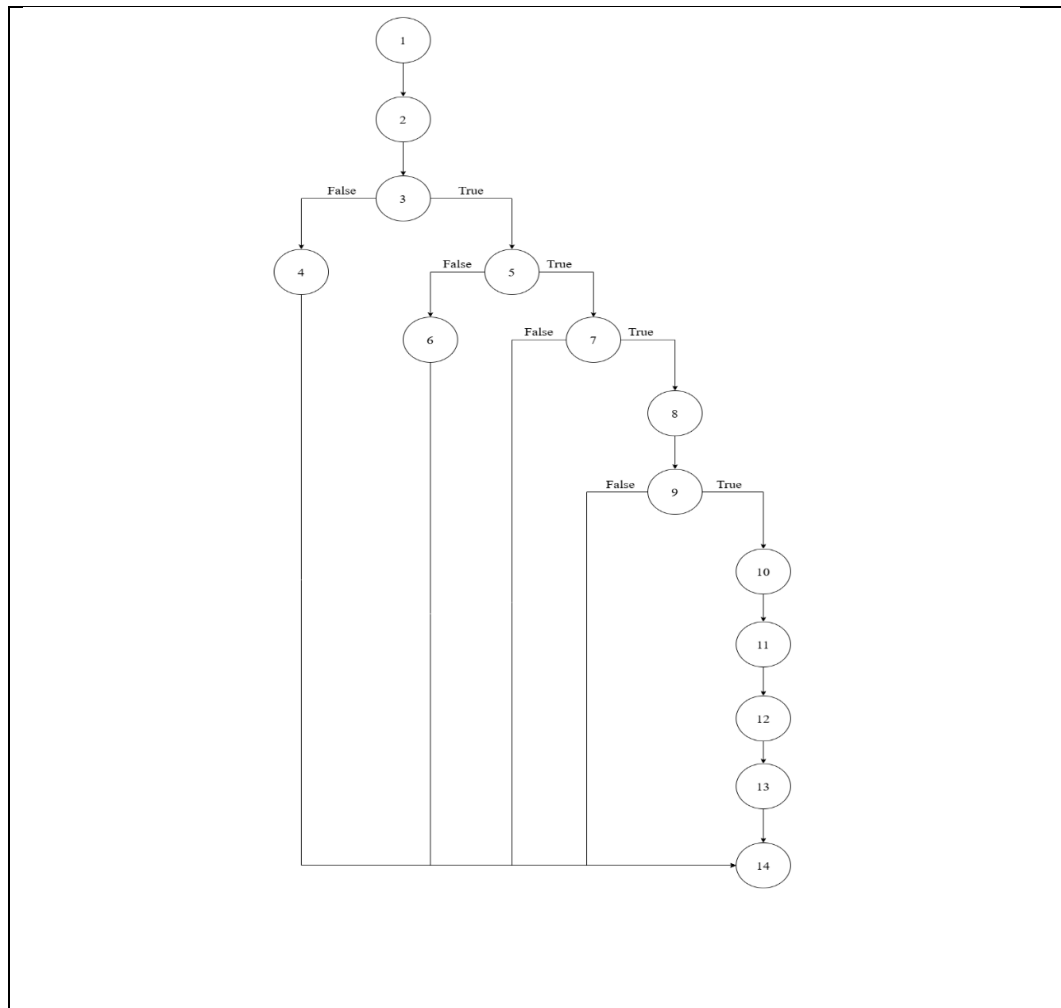
f. *Flowgraph* Edit Barang

Node dan Edge

Node:

1. *Start*
2. Input data barang
3. Validasi data barang
4. *Error* data barang tidak lengkap
5. Validasi *printer barcode*
6. *Error printer barcode* belum dipilih
7. Validasi *user login*
8. *Loading* aktif
9. Validasi mode edit barang
10. *Update* data barang
11. Cetak *barcode*
12. Simpan berhasil
13. *Loading* selesai
14. *End*





Sumber: Penulis, 2026

Gambar 5.36 *Flowgraph* Edit Barang

g. Jalur Independen *Path*

1. Path 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 14$
(Data barang tidak lengkap)
2. Path 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 14$
(Printer barcode belum dipilih)
3. Path 3: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 14$
(User belum login)
4. Path 4: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 14$
(Data edit barang tidak ditemukan)
5. Path 5: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 12 \rightarrow 13 \rightarrow 14$
(Edit barang berhasil)

5.4.5 Pengujian Fungsi Transaksi Barang

a. Potongan *Source Code* Fungsi Transaksi Barang

```
Future<void> bayar() async {

  if (selectedDeviceMac == null) {

    ScaffoldMessenger.of(context).showSnackBar(
      const SnackBar(
        content: Text(
          "Pilih printer dulu",
        ),
      ),
    );

    return;
  }

  try {

    var userDoc = await FirebaseFirestore.instance
      .collection('admin')
      .doc(currentUid)
      .get();

    if (!userDoc.exists) {
      return;
    }

    WriteBatch batch =
      FirebaseFirestore.instance.batch();

    batch.set(trxRef, {
      'total': total,
      'items': keranjang,
    });

    for (var item in keranjang) {

      batch.update(
        barangRef,
        {
          'stok': FieldValue.increment(
            -(item['qty'] as int),
          ),
        },
      );
    }
  }
}
```

```

    }

    await batch.commit();

    await _printReceipt(
      namaToko,
      penanggungJawab,
    );

    _showSuccessDialog(
      penanggungJawab,
    );

  } catch (e) {

    ScaffoldMessenger.of(context).showSnackBar(
      SnackBar(
        content: Text(
          "Error: $e",
        ),
      ),
    );
  }
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan *if* dan *catch*.
2. Semua validasi *printer bluetooth*.
3. Semua proses pengambilan data admin.
4. Semua proses penyimpanan transaksi.
5. Semua proses *update* stok barang.
6. Semua proses cetak struk transaksi.
7. Semua pesan transaksi berhasil.
8. Semua pesan *error* transaksi.
9. Semua *return* pada kondisi *error*.

c. Identifikasi *Statement*Tabel 5.9 Identifikasi *Statement* fungsi Transaksi Barang

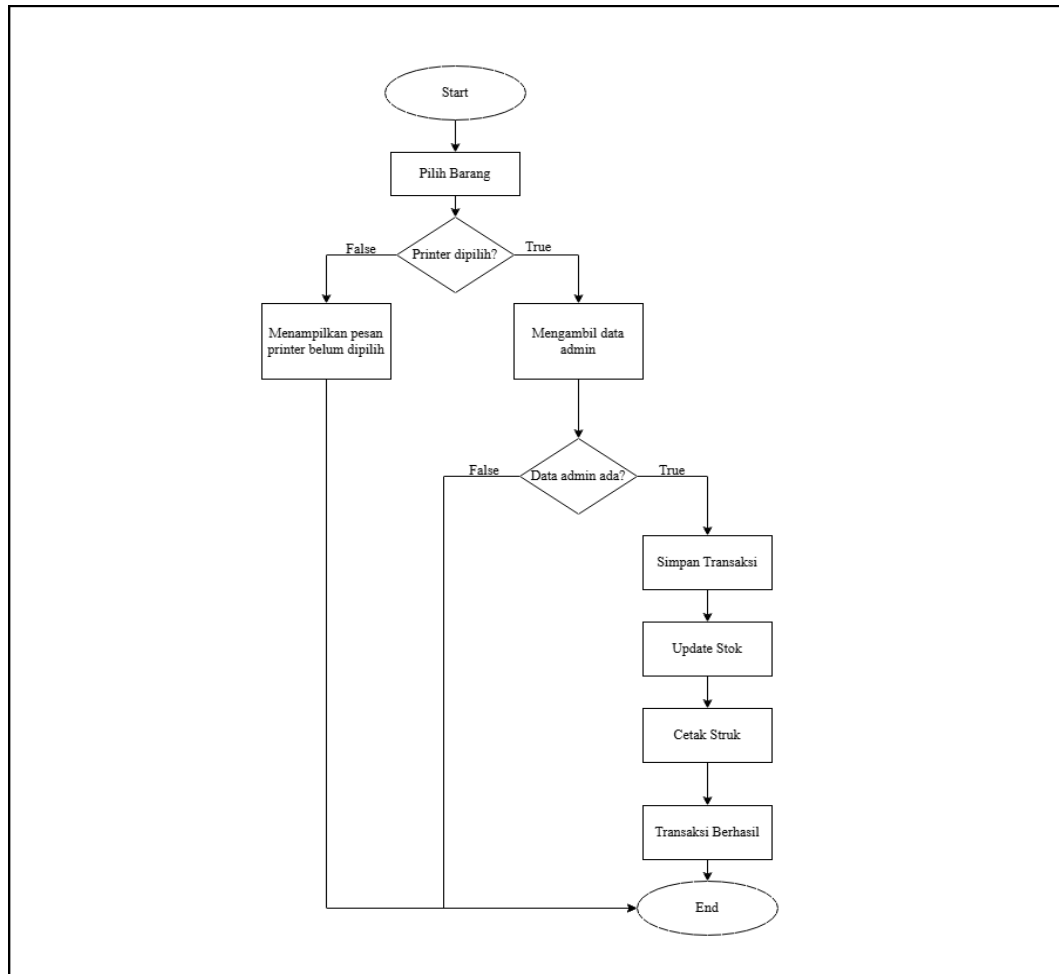
No	Kode Program	Keterangan
S1	if (selectedDeviceMac == null)	Memeriksa <i>printer bluetooth</i>
S2	SnackBar("Pilih printer dulu")	Menampilkan pesan <i>printer</i> belum dipilih
S3	userDoc = FirebaseFirestore...get()	Mengambil data admin
S4	if (!userDoc.exists)	Memeriksa data admin
S5	batch.set(trxRef, {...})	Menyimpan transaksi
S6	for (var item in keranjang)	Perulangan data barang
S7	batch.update(...)	Mengurangi stok barang
S8	batch.commit()	Menyimpan perubahan <i>database</i>
S9	_printReceipt()	Mencetak struk transaksi
S10	_showSuccessDialog()	Menampilkan transaksi berhasil
S11	catch (e)	Menangkap <i>error</i> program

Sumber: Penulis, 2026

d. *Test Case* untuk *Statement Coverage*Tabel 5.10 *Test Case* untuk *Statement Coverage* fungsi Transaksi Barang

<i>Test Case</i>	Kondisi	Jalur Eksekusi	Output
TC1	<i>Printer bluetooth</i> belum dipilih	S1 → S2 → S3	Sistem menampilkan pesan pilih <i>printer</i> dulu
TC2	Data admin tidak ditemukan	S1 → S4 → S5 → S6	Proses transaksi dihentikan
TC3	Semua data transaksi valid	S1 → S4 → S5 → S7 → S8 → S9 → S10 → S11 → S12 → S13	Transaksi berhasil dan struk tercetak

Sumber: Penulis, 2026

e. *Flowchart* Transaksi Barang

Sumber: Penulis, 2026

Gambar 5.37 *Flowchart* Transaksi Barang

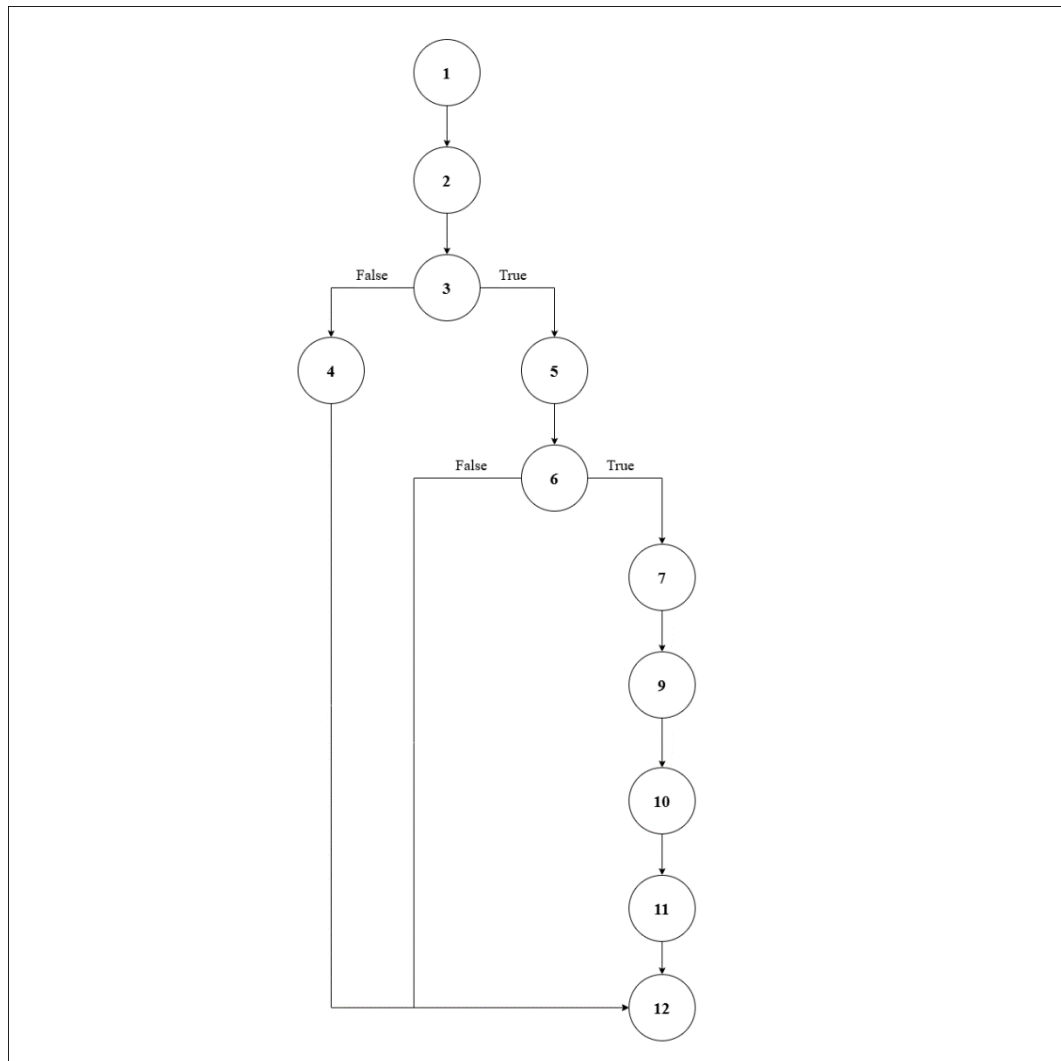
f. *Flowgraph* Transaksi Barang

Node dan Edge

Node:

1. *Start*
2. *Pilih barang*
3. *Cek Printer*
4. *Pesan Printer* belum terpilih
5. *Ambil data admin*
6. *Cek data admin*
7. *Simpan transaksi*
8. *Update stok barang*

- 9. Cetak struk
- 10. Transaksi berhasil
- 11. *Error*
- 12. *End*



Sumber: Penulis, 2026

Gambar 5.38 *Flowgraph* Transaksi Barang

g. Jalur Independen *Path*

- 1. *Path* 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 12$
(*Printer bluetooth* belum dipilih)
- 2. *Path* 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 12$
(Data admin tidak ditemukan)

3. *Path* 3: 1 → 2 → 3 → 5 → 6 → 7 → 8 → 9 → 10 → 12
(Transaksi berhasil dan struk tercetak)

5.4.6 Pengujian Fungsi Transaksi Layanan

- a. Potongan *Source Code* Fungsi Transaksi Layanan

```
Future<void> simpanLayanan() async {

  if (!_formKey.currentState!.validate() ||
    selectedLayanan == null) {

    _showSnackBar(
      "Mohon lengkapi data",
      isError: true,
    );

    return;
  }

  setState(() => isLoading = true);

  try {

    double nominal = double.parse(
      _nominalController.text.replaceAll('.', ''),
    );

    String pelaksana =
      widget.namaKasir ?? namaToko;

    await FirebaseFirestore.instance
      .collection('transaksi_layanan')
      .add({
        'total': nominal,
        'namaKasir': pelaksana,
        'catatan': _catatanController.text,
      });

    _showSuccessDialog(pelaksana);

  } catch (e) {

    _showSnackBar(
      "Terjadi kesalahan",
      isError: true,
    );
  }
}
```

```

    } finally {
        setState(() => isLoading = false);
    }
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan if dan catch.
2. Semua validasi data layanan.
3. Semua pesan *error* transaksi layanan.
4. Proses *loading* transaksi.
5. Proses pengambilan nominal layanan.
6. Proses penentuan pelaksana layanan.
7. Proses penyimpanan transaksi layanan.
8. Pesan transaksi layanan berhasil.
9. Proses *loading* selesai.
10. Semua return pada kondisi error.

c. Identifikasi *Statement*

Tabel 5.11 Identifikasi *Statement* fungsi Transaksi Layanan

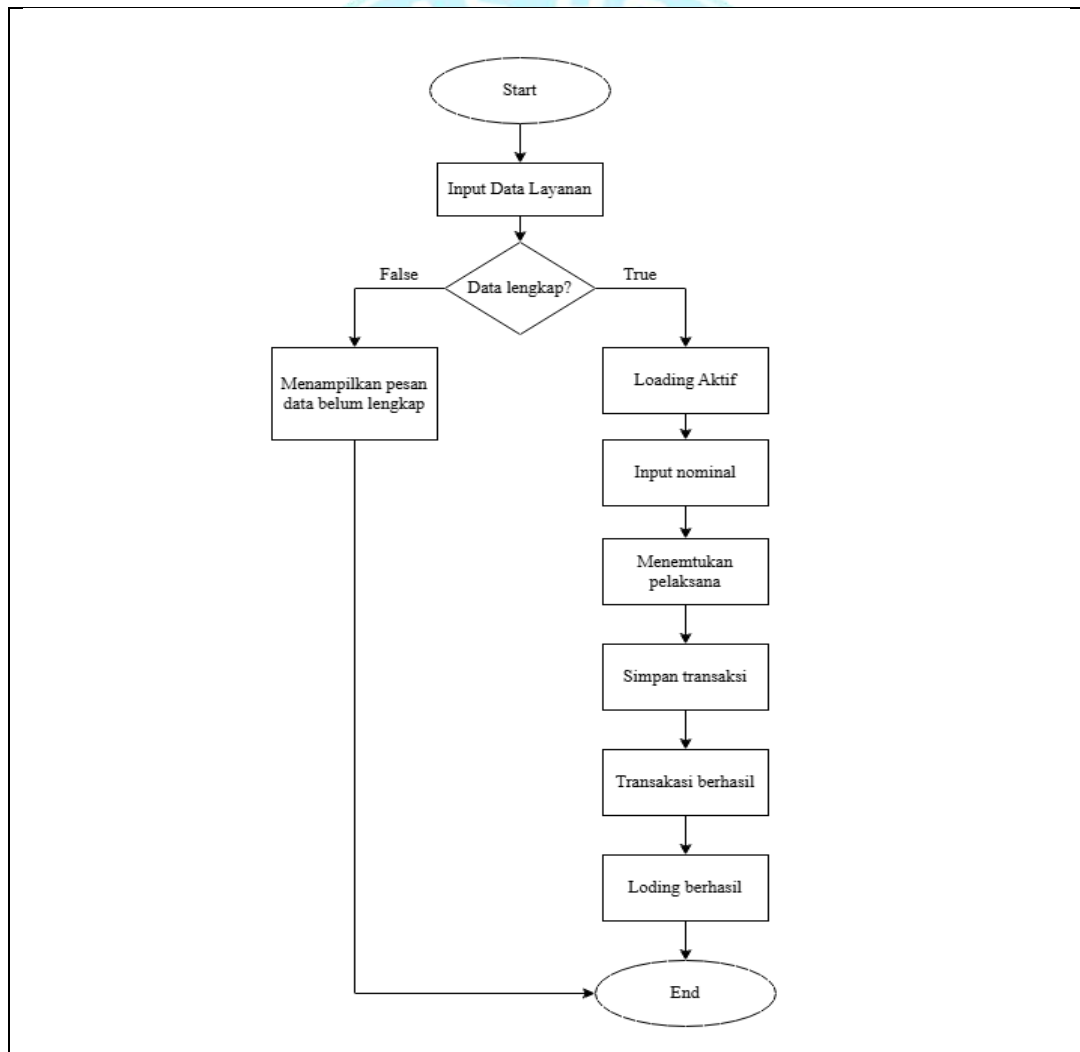
No	Kode Program	Keterangan
S1	if (!_formKey.currentState!.validate()...)	Memeriksa data layanan
S2	_showSnackBar("Mohon lengkapi data")	Menampilkan pesan data belum lengkap
S3	setState(() => isLoading = true)	Mengaktifkan loading
S4	double.parse(...)	Mengambil nominal layanan
S5	widget.namaKasir ?? namaToko	Menentukan pelaksana layanan
S6	.collection('transaksi_layanan').add({...})	Menyimpan transaksi layanan
S7	_showSuccessDialog(pelaksana)	Menampilkan transaksi berhasil
S8	catch (e)	Menangkap <i>error</i>
S9	_setState(() => isLoading = false)	Menonaktifkan loading

Sumber: Penulis, 2026

d. *Test Case* untuk *Statement Coverage*Tabel 5.12 *Test Case* untuk *Statement Coverage* fungsi Transaksi Layanan

<i>Test Case</i>	Kondisi	Jalur Eksekusi	Output
TC1	Data layanan tidak lengkap	S1 → S2 → S3	Sistem menampilkan pesan mohon lengkapi data
TC2	Semua data layanan valid	S1 → S4 → S5 → S6 → S7 → S8 → S11	Transaksi layanan berhasil disimpan

Sumber: Penulis, 2026

e. *Flowchart* Transaksi Barang

Sumber: Penulis, 2026

Gambar 5. 39 *Flowchart* Transaksi Layanan

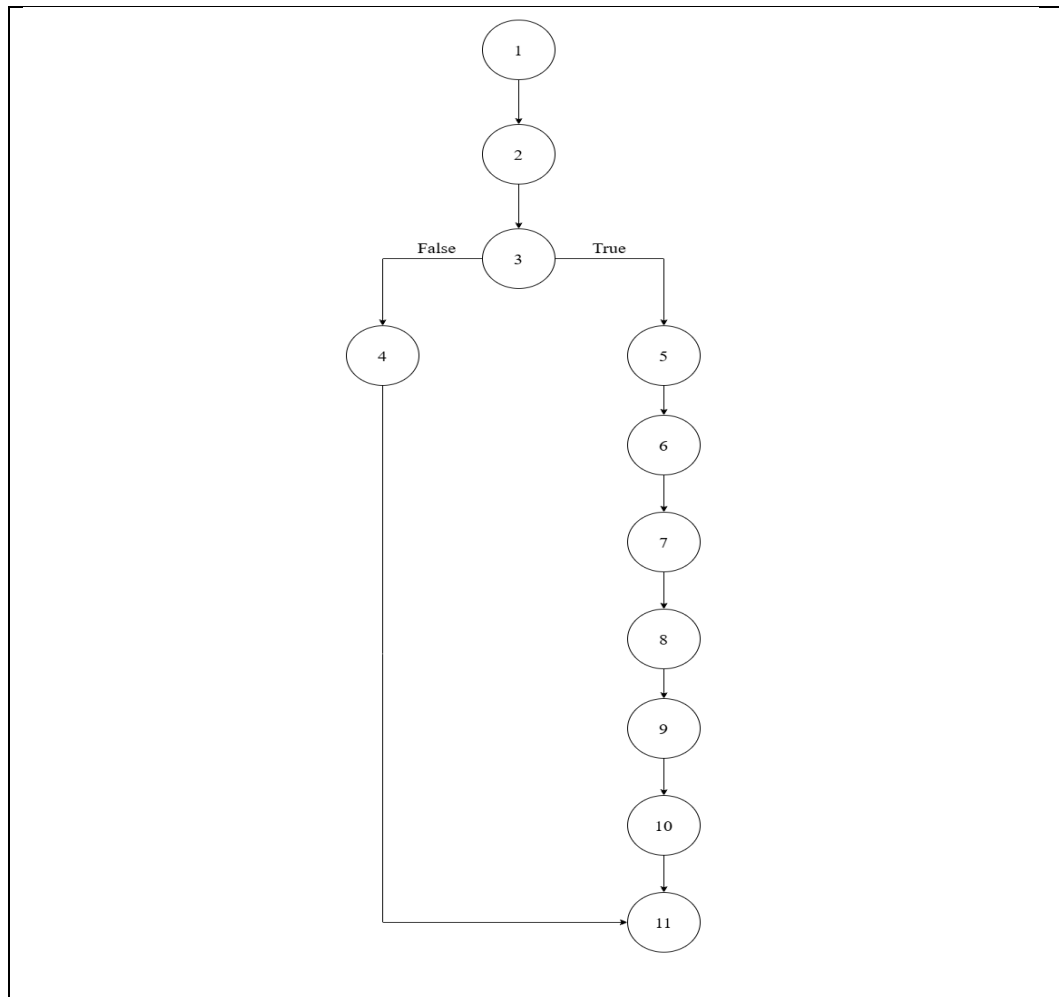
f. *Flowgraph* Transaksi Layanan

Node dan Edge

Node:

1. *Start*
2. Input data layanan
3. Validasi data layanan
4. *Error* data layanan tidak lengkap
5. *Loading* aktif
6. Ambil nominal layanan
7. Tentukan pelaksana layanan
8. Simpan transaksi layanan
9. Transaksi berhasil
10. *Loading* selesai
11. *End*





Sumber: Penulis, 2026

Gambar 5.40 *Flowgraph* Transaksi Layanan

g. Jalur Independen *Path*

1. *Path* 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 11$

(Data layanan tidak lengkap)

2. *Path* 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11$

(Transaksi layanan berhasil)

5.4.7 Pengujian Fungsi Tambah Kasir

a. Potongan *Source Code* Fungsi Tambah Kasir

```

Future<void> simpanKasir() async {
    if (namaC.text.trim().isEmpty ||
  
```

```

passwordC.text.trim().isEmpty) {

    _showSnackBar(
        "Nama dan Password wajib diisi",
        isError: true,
    );

    return;
}

if (!isValidPassword(
    passwordC.text.trim())) {

    _showSnackBar(
        "Password minimal 8 karakter",
        isError: true,
    );

    return;
}

final String? adminUid =
    FirebaseAuth.instance.currentUser?.uid;

if (adminUid == null) return;

setState(() => loading = true);

try {

    await FirebaseFirestore.instance
        .collection('kasir')
        .add({
            'nama': namaC.text.trim(),
            'password': passwordC.text.trim(),
            'role': 'kasir',
        });

    _showSnackBar(
        "Akses kasir berhasil diaktifkan",
        isError: false,
    );

} catch (e) {

    _showSnackBar(
        "Gagal menyimpan",
        isError: true,
    );
}

```

```

    );
  } finally {
    setState(() => loading = false);
  }
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan *if* dan *catch*.
2. Semua validasi data kasir.
3. Semua pesan *error* tambah kasir.
4. Validasi *password* kasir.
5. Validasi admin *login*.
6. Proses *loading* data.
7. Proses penyimpanan data kasir.
8. Pesan kasir berhasil ditambahkan.
9. Proses *loading* selesai.
10. Semua *return* pada kondisi *error*.

c. Identifikasi *Statement*

Tabel 5.13 *Identifikasi Statement* fungsi Tambah Kasir

No	Kode Program	Keterangan
S1	<code>if (namaC.text.trim().isEmpty...)</code>	Memeriksa data layanan
S2	<code>_showSnackBar("Nama dan Password wajib diisi")</code>	Menampilkan pesan data belum lengkap
S3	<code>if (!isValidPassword(...))</code>	Memeriksa validasi <i>password</i>
S4	<code>_showSnackBar("Password minimal 8 karakter")</code>	Menampilkan pesan <i>password</i> tidak valid
S5	<code>if (adminUid == null)</code>	Memeriksa admin <i>login</i>
S6	<code>.setState(() => loading = true)</code>	Mengaktifkan <i>loading</i>
S7	<code>.collection('kasir').add({...})</code>	Menyimpan data kasir

S8	<code>_showSnackBar("Akses kasir berhasil diaktifkan")</code>	Menampilkan pesan berhasil
S9	<code>catch (e)</code>	Menangkap <i>error</i>
S10	<code>setState() => loading = false)</code>	Menonaktifkan <i>loading</i>

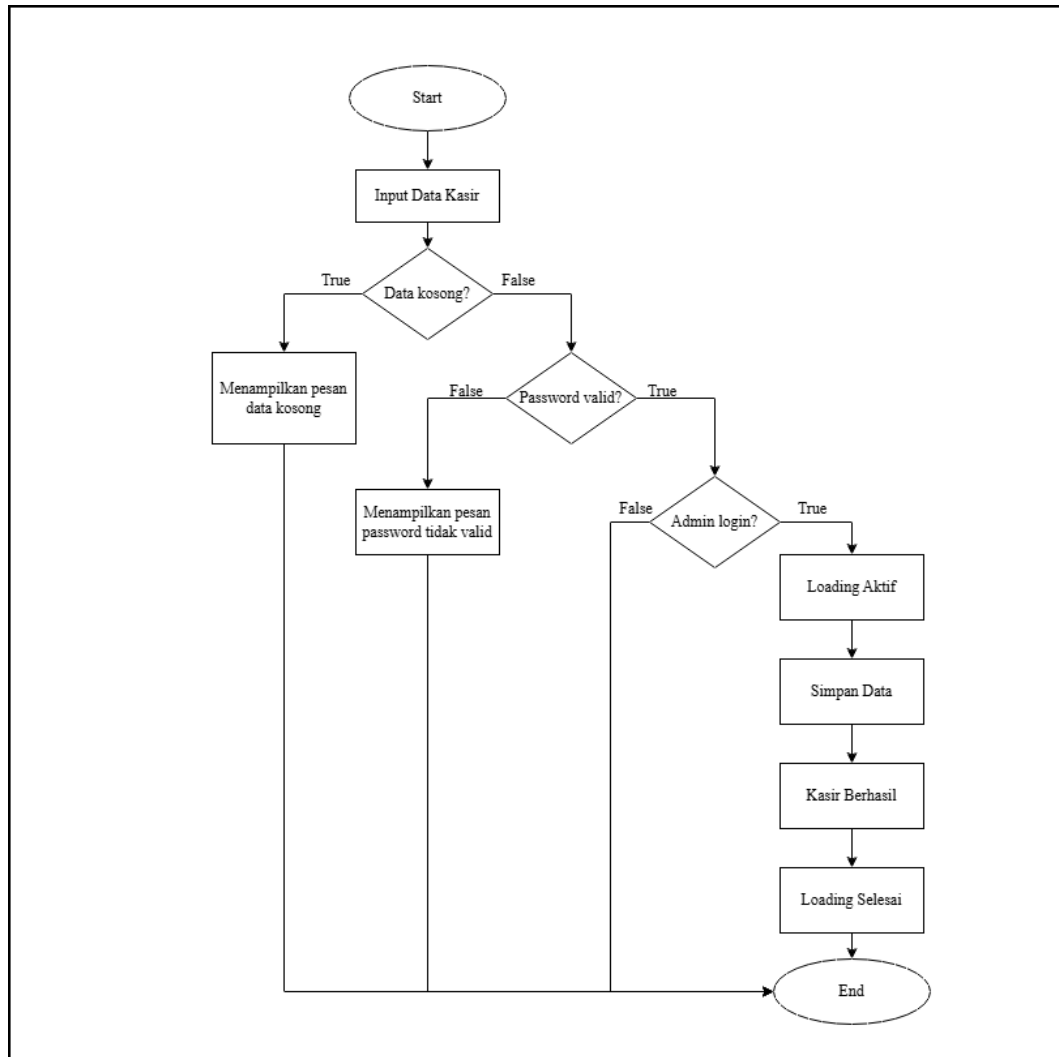
Sumber: Penulis, 2026

d. *Test Case* untuk *Statement Coverage*

Tabel 5.14 *Test Case* untuk *Statement Coverage* fungsi Tambah Kasir

<i>Test Case</i>	Kondisi	Jalur Eksekusi	<i>Output</i>
TC1	Nama atau <i>password</i> kosong	S1 → S2 → S3	Sistem menampilkan pesan wajib diisi
TC2	<i>Password</i> tidak valid	S1 → S4 → S5 → S6	Sistem menampilkan pesan <i>password</i> minimal 8 karakter
TC3	Admin belum <i>login</i>	S1 → S4 → S7 → S8	Proses tambah kasir dihentikan
TC4	Semua data kasir valid	S1 → S4 → S7 → S9 → S10 → S11 → S14	Kasir berhasil ditambahkan

Sumber: Penulis, 2026

e. *Flowchart* Tambah Kasir

Sumber: Penulis, 2026

Gambar 5.41 *Flowchart* Tambah Kasir

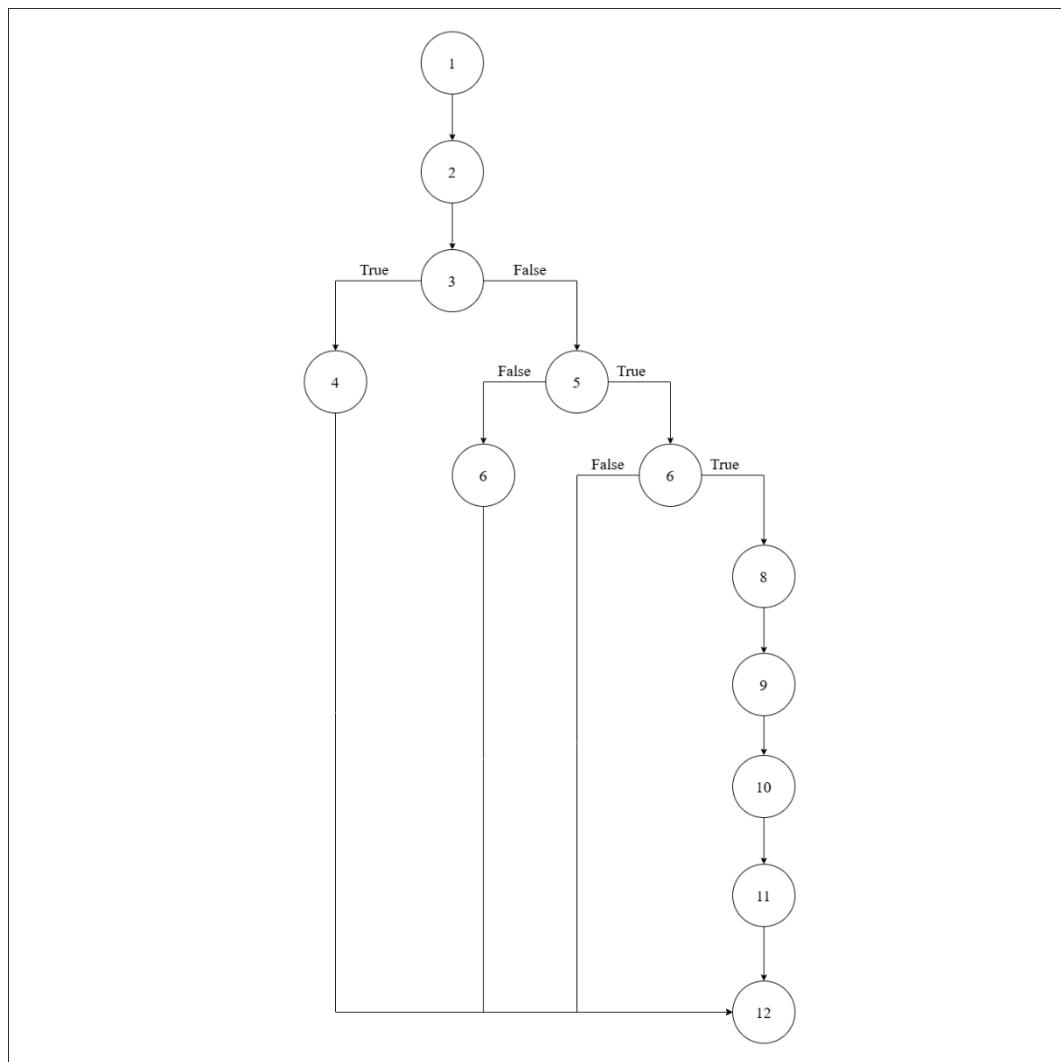
f. *Flowgraph* Tambah Kasir

Node dan Edge

Node:

1. *Start*
2. *Input data kasir*
3. *Validasi data kosong*
4. *Error data kosong*
5. *Validasi password*
6. *Error password tidak valid*

7. Validasi admin *login*
8. Loading aktif
9. Simpan data kasir
10. Kasir berhasil ditambahkan
11. *Loading* selesai
12. *End*



Sumber: Penulis, 2026

Gambar 5.42 *Flowgraph* Tambah Kasir

g. Jalur Independen *Path*

1. *Path* 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 12$
(Nama atau password kosong)

2. Path 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 12$
(Password tidak valid)
3. Path 3: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 12$
(Admin belum login)
4. Path 4: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11 \rightarrow 12$
(Tambah kasir berhasil)

5.4.8 Pengujian Fungsi Edit Kasir

- a. Potongan Source Code Fungsi Edit Kasir

onPressed: loading

? null

: () async {

```
if (namaC.text.trim().isEmpty ||
    passC.text.trim().isEmpty) {
```

```
ScaffoldMessenger.of(context)
```

```
.showSnackBar(
```

```
const SnackBar(
```

```
content: Text(
```

```
"Nama dan password wajib diisi",
```

```
),
```

```
),
```

```
);
```

```
return;
```

```
}
```

```
if (!isValidPassword(
```

```
passC.text.trim(),
```

```
)) {
```

```
ScaffoldMessenger.of(context)
```

```
.showSnackBar(
```

```
const SnackBar(
```

```
content: Text(
```

```
"Password tidak sesuai ketentuan",
```

```
),
```

```
),
```

```
);
```

```
return;
```

```
}
```

```

setState() {
  loading = true;
});

await FirebaseFirestore.instance
  .collection('kasir')
  .doc(widget.docId)
  .update({

    'nama': namaC.text.trim(),

    'password':
      passC.text.trim(),

    'updatedAt':
      FieldValue.serverTimestamp(),
  });

if (!context.mounted) {
  return;
}

Navigator.pop(context);

ScaffoldMessenger.of(context)
  .showSnackBar(
    SnackBar(
      content: const Text(
        "Data kasir berhasil diperbarui",
      ),
      backgroundColor: Colors.green,
    ),
  );
}

```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan if.
2. Semua validasi data kasir.
3. Semua pesan error edit kasir.
4. Validasi password kasir.
5. Proses loading data.

6. Proses update data kasir.
7. Validasi context aktif.
8. Proses kembali ke halaman sebelumnya.
9. Pesan data berhasil diperbarui.
10. Semua return pada kondisi error.

c. Identifikasi *Statement*

Tabel 5.15 Identifikasi *Statement* fungsi Edit Kasir

No	Kode Program	Keterangan
S1	if (namaC.text.trim().isEmpty...)	Memeriksa nama dan <i>password</i> kosong
S2	SnackBar("Nama dan password wajib diisi")	Menampilkan pesan data kosong
S3	if (!isValidPassword(...))	Memeriksa validasi <i>password</i>
S4	SnackBar("Password tidak sesuai ketentuan")	Menampilkan pesan <i>password</i> tidak valid
S5	setState(() { loading = true; })	Mengaktifkan <i>loading</i>
S6	.collection('kasir').doc(...).update({...})	Memperbarui data kasir
S7	if (!context.mounted)	Memeriksa <i>context</i> aktif
S8	Navigator.pop(context)	Kembali ke halaman sebelumnya
S9	SnackBar("Data kasir berhasil diperbarui")	Menampilkan pesan berhasil

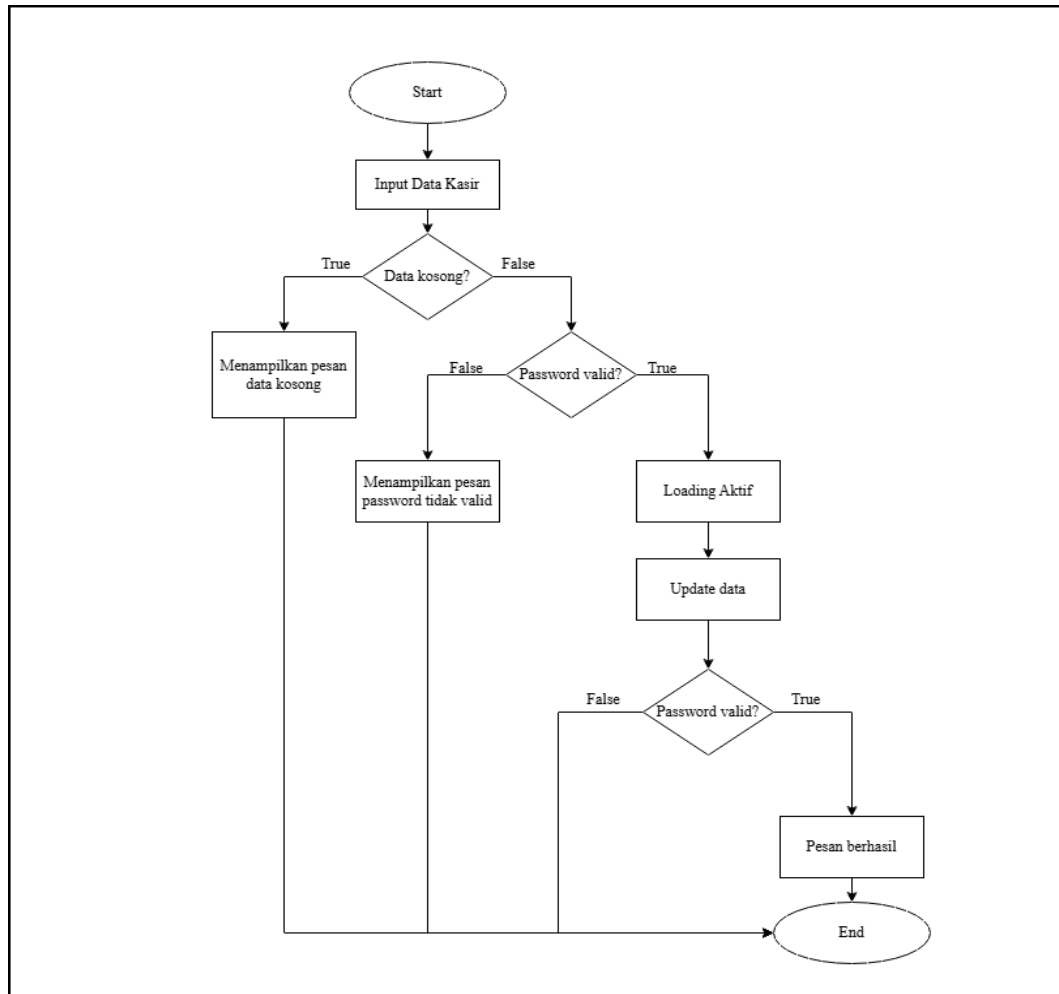
Sumber: Penulis, 2026

d. *Test Case* untuk *Statement Coverage*

Tabel 5. 16 *Test Case* untuk *Statement Coverage* fungsi Edit Kasir

<i>Test Case</i>	Kondisi	Jalur Eksekusi	<i>Output</i>
TC1	Nama atau <i>password</i> kosong	S1 → S2 → S3	Sistem menampilkan pesan wajib diisi
TC2	Password tidak valid	S1 → S4 → S5 → S6	Sistem menampilkan pesan <i>password</i> minimal 8 karakter
TC3	Context tidak aktif	S1 → S4 → S7 → S8 → S9 → S10	Proses edit kasir dihentikan
TC4	Semua data kasir valid	S1 → S4 → S7 → S8 → S9 → S11 → S12	Data kasir berhasil diperbarui

Sumber: Penulis, 2026

e. *Flowchart* Tambah Kasir

Sumber: Penulis, 2026

Gambar 5.43 *Flowchart* Edit Kasir

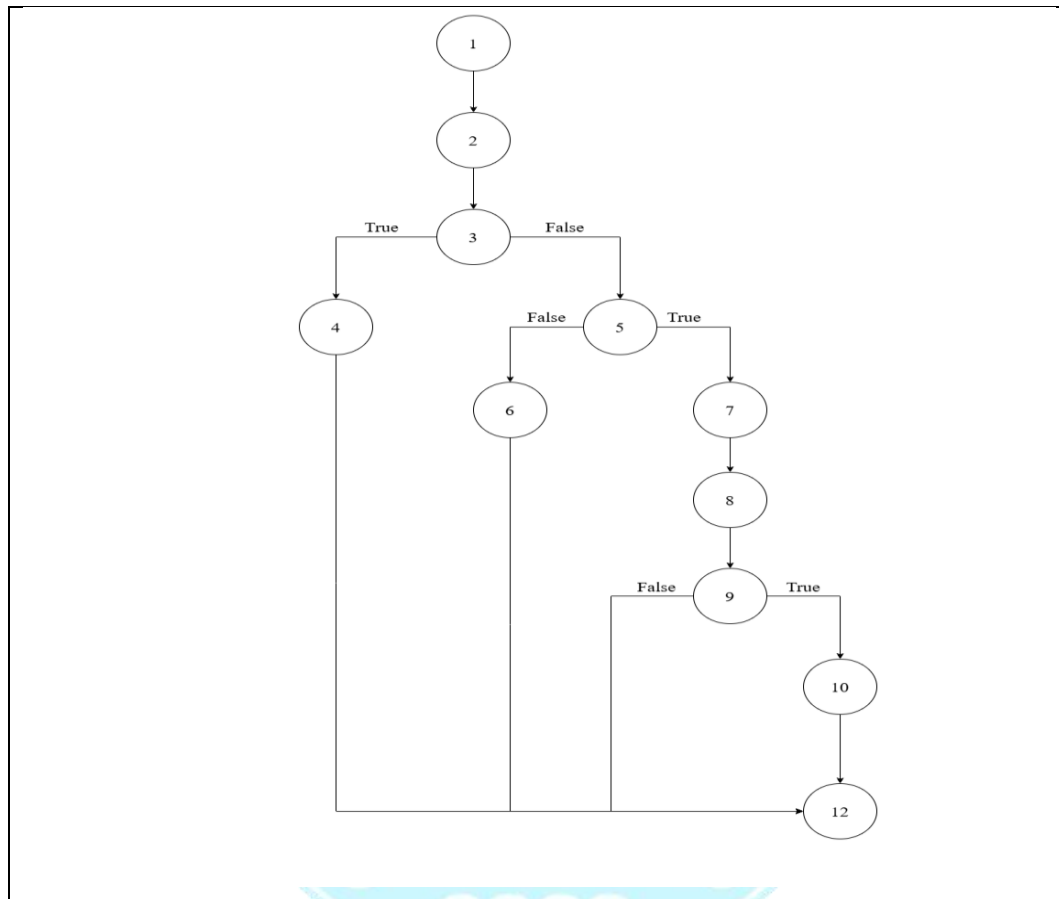
f. *Flowgraph* Edit Kasir

Node dan Edge

Node:

1. *Start*
2. *Input data kasir*
3. *Validasi data kosong*
4. *Error data kosong*
5. *Validasi password*
6. *Error password tidak valid*
7. *Loading aktif*

8. *Update data kasir*
9. Validasi *context* aktif
10. Pesan berhasil
11. *End*



Sumber: Penulis, 2026

Gambar 5.44 *Flowgraph* Edit Kasir

g. Jalur Independen *Path*

1. Path 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 11$
(Nama atau *password* kosong)
2. Path 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 11$
(*Password* tidak valid)
3. Path 3: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 11$
(*Context* tidak aktif)
4. Path 4: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10 \rightarrow 11$
(Edit kasir berhasil)

5.4.9 Pengujian Fungsi Edit Profil Toko

a. Potongan *Source Code* Fungsi Edit Profil Toko

```
onPressed: () async {

  var user =
    FirebaseAuth.instance.currentUser;

  if (user != null) {

    await FirebaseFirestore.instance
      .collection('admin')
      .doc(user.uid)
      .update({

        'nama_toko': namaC.text,

        'alamat': alamatC.text,

        'telp': telpC.text,
      });

    if (!mounted) {
      return;
    }

    Navigator.pop(context);
  }
}
```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan *if*.
2. Validasi *user login*.
3. Proses akses *collection* admin.
4. Proses pengambilan *document* admin.
5. Proses *update* data profil toko.
6. Validasi *context* aktif.
7. Proses menutup halaman edit profil.
8. Semua *return* pada kondisi *error*.

c. Identifikasi *Statement*Tabel 5.17 Identifikasi *Statement* fungsi Edit Profil Toko

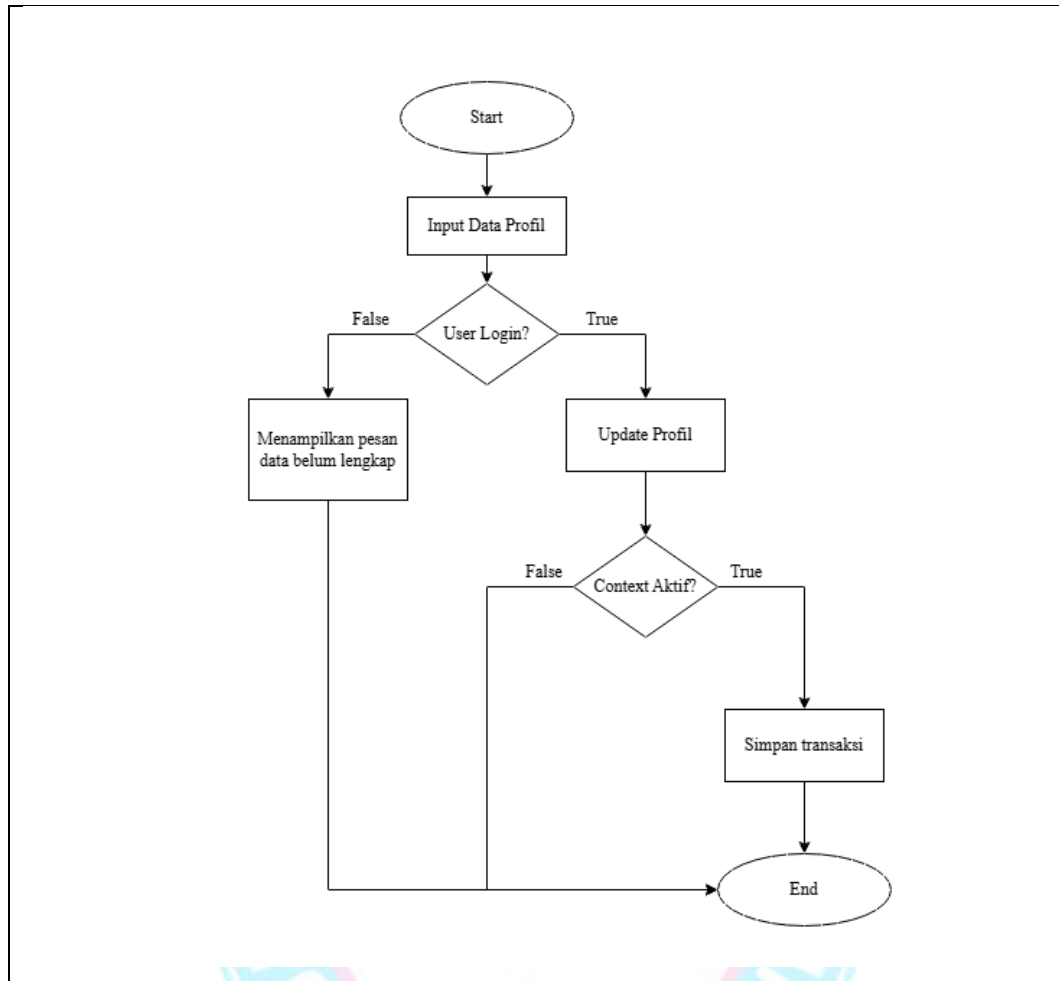
No	Kode Program	Keterangan
S1	FirebaseAuth.instance.currentUser	Mengambil data <i>user login</i>
S2	if (user != null)	Memeriksa <i>user login</i>
S3	.collection('admin')	Mengakses <i>collection</i> admin
S4	.doc(user.uid)	Mengambil <i>document</i> admin
S5	.update({...})	Memperbarui data profil toko
S6	if (!mounted)	Memeriksa <i>context</i> aktif
S7	Navigator.pop(context)	Menutup halaman edit profil

Sumber: Penulis, 2026

d. *Test Case* untuk *Statement Coverage*Tabel 5.18 *Test Case* untuk *Statement Coverage* fungsi Edit Profil Toko

<i>Test Case</i>	Kondisi	Jalur Eksekusi	<i>Output</i>
TC1	<i>User</i> belum login	S1 → S2	Proses edit profil dihentikan
TC2	<i>Context</i> tidak aktif	S1 → S2 → S3 → S4 → S5 → S6 → S7	Proses dihentikan
TC3	Data profil toko valid	S1 → S2 → S3 → S4 → S5 → S6 → S8	Data profil toko berhasil diperbarui

Sumber: Penulis, 2026

e. *Flowchart* Edit Profil Toko

Sumber: Penulis, 2026

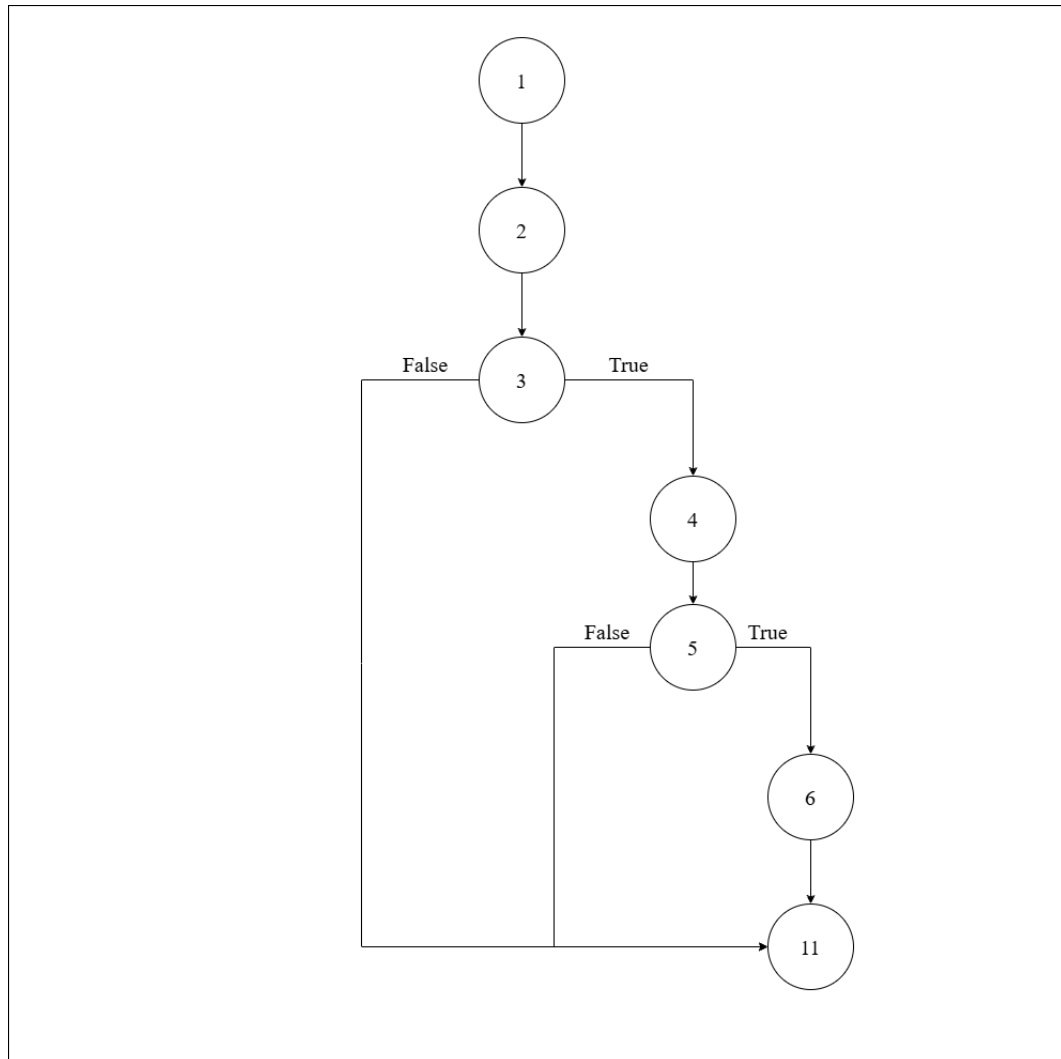
Gambar 5.45 *Flowchart* Edit Profil Toko

f. *Flowgraph* Edit Profil Toko

Node dan *Edge*

Node:

1. *Start*
2. *Input* data profil
3. Validasi *user login*
4. *Update* profil toko
5. Validasi *context* aktif
6. Tutup halaman edit profil
7. *End*



Sumber: Penulis, 2026

Gambar 5.46 *Flowgraph* Edit Profil Toko

g. Jalur Independen *Path*

1. Path 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 7$
(User belum login)
2. Path 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7$
(Context tidak aktif)
3. Path 3: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7$
(Edit profil toko berhasil)

5.4.10 Pengujian Fungsi *Logout*

a. Potongan *Source Code* Fungsi *Logout*

```
void logout(BuildContext context) async {

  bool confirm = await showDialog(
    context: context,
    builder: (context) => AlertDialog(
      title: const Text(
        "Keluar Akun",
      ),
    ),
  ) ?? false;

  if (!confirm) return;

  await FirebaseAuth.instance.signOut();

  if (!mounted) return;

  Navigator.pushAndRemoveUntil(
    context,
    MaterialPageRoute(
      builder: (_) => const LoginScreen(),
    ),
    (route) => false,
  );
}
```

Kode ini akan diuji menggunakan metode *White Box Testing – Statement Coverage*.

b. Tujuan *Statement Coverage*

Statement Coverage bertujuan untuk memastikan:

1. Semua percabangan *if*.
2. Proses menampilkan dialog konfirmasi *logout*.
3. Validasi konfirmasi *logout* pengguna.
4. Proses pembatalan *logout*.
5. Proses menghapus sesi *login Firebase Authentication*.
6. Validasi *context* aktif.
7. Proses perpindahan ke halaman *login*.
8. Semua *return* pada kondisi tertentu.

c. Identifikasi *Statement*Tabel 5.19 Identifikasi *Statement* fungsi *Logout*

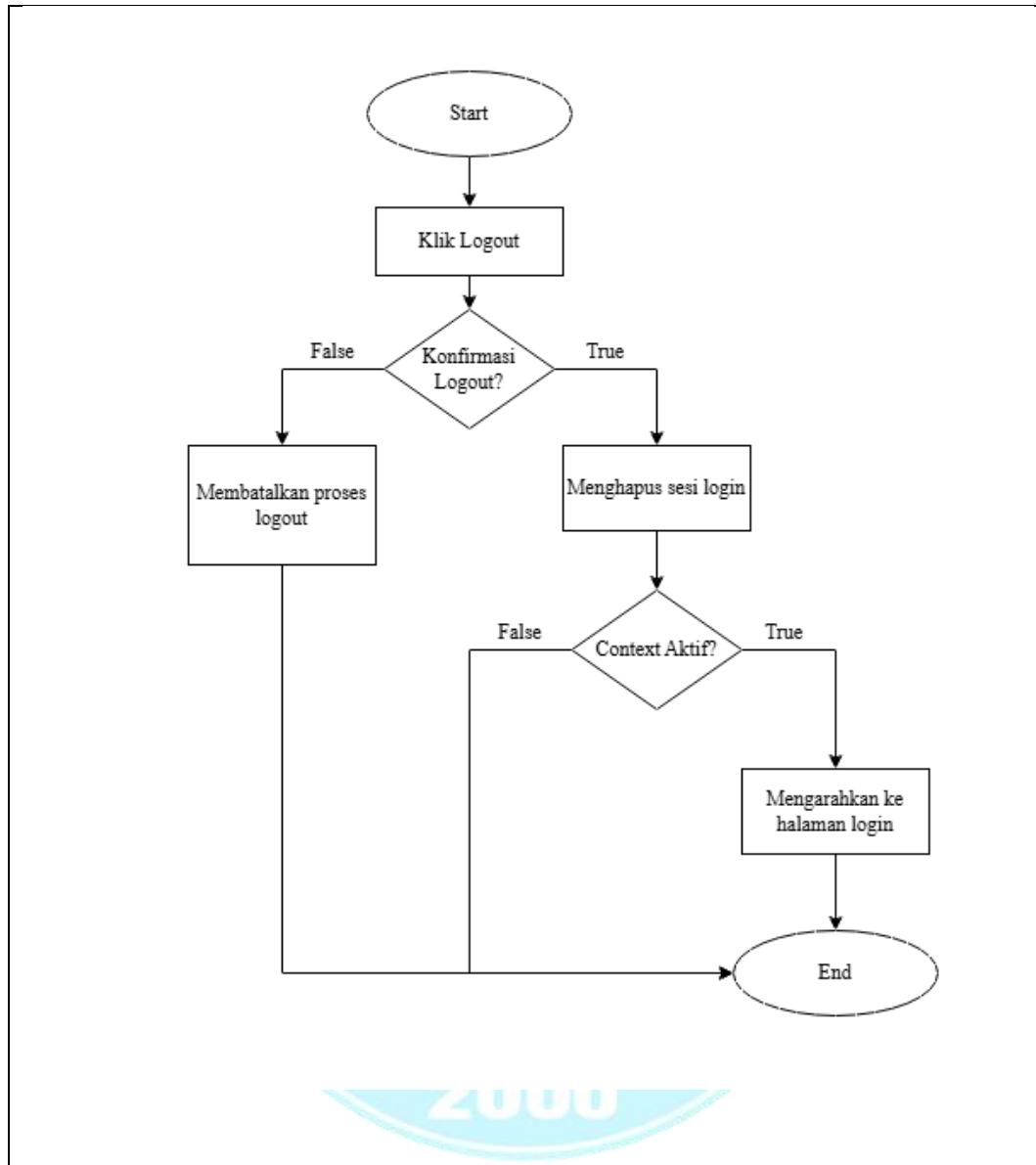
No	Kode Program	Keterangan
S1	showDialog(...)	Menampilkan dialog konfirmasi <i>logout</i>
S2	if (!confirm)	Memeriksa konfirmasi <i>logout</i>
S3	return	Membatalkan proses <i>logout</i>
S4	FirebaseAuth.instance.signOut()	Menghapus sesi <i>login</i>
S5	if (!mounted)	Memeriksa <i>context</i> aktif
S6	Navigator.pushAndRemoveUntil(...)	Mengarahkan ke halaman <i>login</i>

Sumber: Penulis, 2026

d. Test Case untuk *Statement Coverage*Tabel 5. 20 Test Case untuk *Statement Coverage* fungsi *Logout*

Test Case	Kondisi	Jalur Eksekusi	Output
TC1	Pengguna memilih batal <i>logout</i>	S1 → S2 → S3	Sistem membatalkan proses <i>logout</i>
TC2	<i>Context</i> tidak aktif setelah <i>sign out</i>	S1 → S2 → S4 → S5 → S6	Proses <i>logout</i> dihentikan
TC3	Pengguna mengonfirmasi <i>logout</i>	S1 → S2 → S4 → S5 → S7	Sistem berhasil <i>logout</i> dan berpindah ke halaman <i>login</i>

Sumber: Penulis, 2026

e. *Flowchart Logout*

Sumber: Penulis, 2026

Gambar 5.47 *Flowchart Logout*

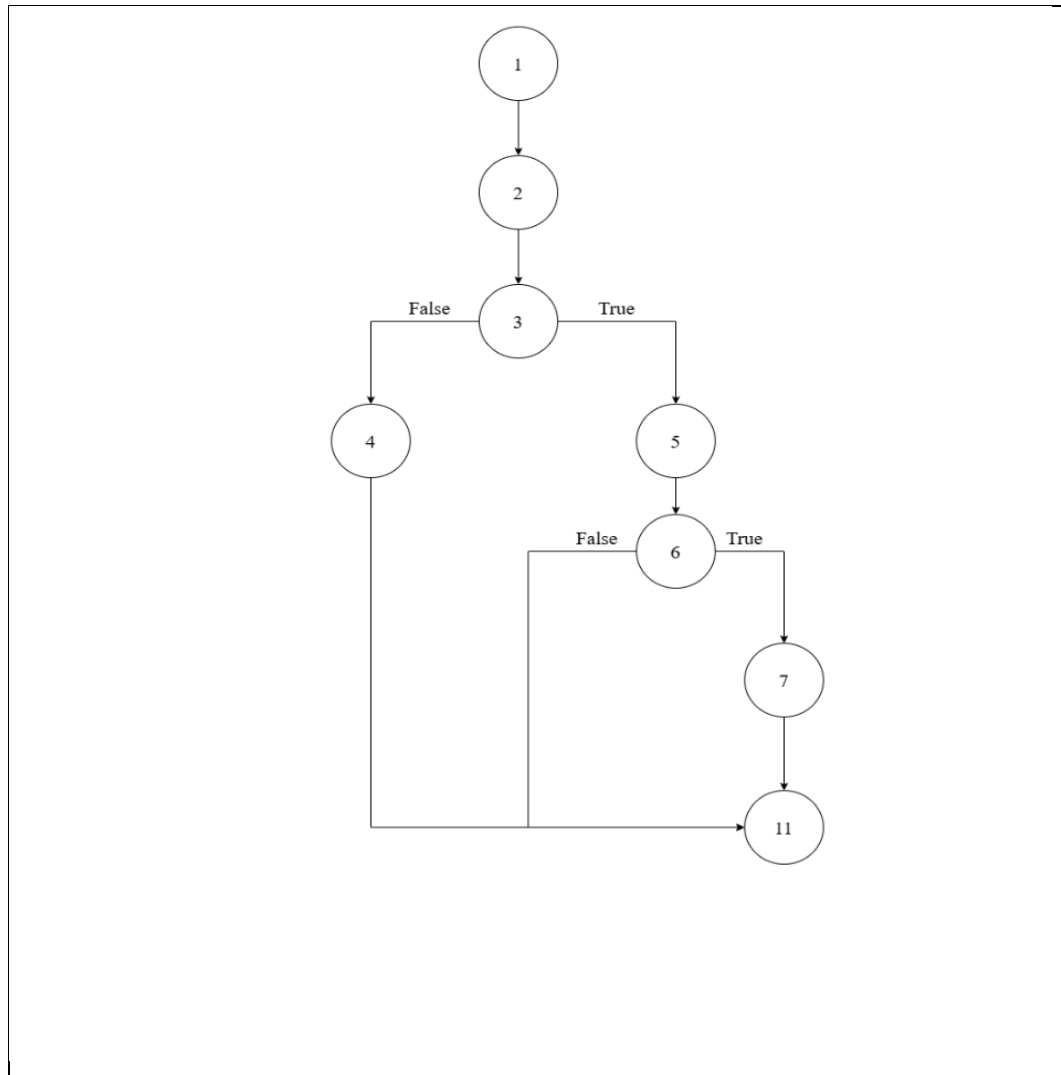
f. *Flowgraph Logout*

Node dan Edge

Node:

1. *Start*
2. *Klik logout*
3. *Validasi konfirmasi logout*
4. *Batal logout*

5. *Sign out*
6. Validasi *context* aktif
7. Pindah ke halaman *login*
8. *End*



Sumber: Penulis, 2026

Gambar 5.48 *Flowgraph Logout*

g. Jalur Independen *Path*

1. Path 1: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 8$
(*Logout* dibatalkan)
2. Path 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 8$
(*Context* tidak aktif)

3. Path 3: $1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8$
(Logout berhasil)



BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil penelitian perancangan, implementasi, serta pengujian yang telah dilakukan pada penelitian "Perancangan Aplikasi Kasir Berbasis *Android* untuk Pengelolaan Penjualan dan Persediaan pada Apan Cell", maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Aplikasi kasir berbasis *Android* berhasil dirancang dan dikembangkan untuk membantu proses pencatatan transaksi penjualan di Apan Cell secara lebih cepat dan akurat. Aplikasi yang dibangun menggunakan *flutter* dan *firebase* mampu mempermudah proses transaksi penjualan barang maupun layanan digital sehingga proses pencatatan tidak lagi dilakukan secara manual menggunakan buku.
2. Aplikasi yang dikembangkan telah mampu memantau dan mengelola persediaan barang secara *real-time*. Sistem dapat memperbarui jumlah stok secara otomatis setiap kali terjadi transaksi penjualan maupun perubahan data barang, sehingga memudahkan pemilik toko dalam mengetahui kondisi persediaan barang seperti *handphone*, aksesoris *handphone*, rokok, minuman, dan es krim secara lebih akurat.
3. Aplikasi berhasil menghasilkan laporan transaksi dan persediaan yang rapi, sistematis, informatif, dan mudah dipahami. Data transaksi yang tersimpan dalam *database* dapat diolah menjadi laporan penjualan dan laporan persediaan sehingga membantu pemilik toko maupun karyawan dalam melakukan pemantauan dan evaluasi terhadap aktivitas operasional toko.
4. Penerapan aplikasi kasir berbasis *Android* mampu meningkatkan efisiensi operasional Apan Cell dibandingkan dengan sistem pencatatan manual. Penggunaan aplikasi membantu mengurangi risiko kesalahan pencatatan, mempercepat proses transaksi, mempermudah pengelolaan stok barang, serta mempercepat penyusunan laporan sehingga kegiatan operasional toko menjadi lebih efektif dan terorganisir.

6.2 Saran

Meskipun aplikasi kasir berbasis *Android* untuk pengelolaan penjualan dan persediaan pada Apan Cell telah berhasil dirancang, diimplementasikan, dan diuji sesuai dengan kebutuhan penelitian, masih terdapat beberapa aspek yang dapat dikembangkan untuk meningkatkan kualitas sistem dan memperluas fungsionalitas aplikasi di masa mendatang. Oleh karena itu, beberapa saran yang dapat diberikan untuk pengembangan selanjutnya adalah sebagai berikut:

1. Penelitian selanjutnya diharapkan dapat melakukan implementasi dan penggunaan aplikasi dalam jangka waktu yang lebih panjang pada operasional toko sehingga efektivitas sistem dapat diketahui secara lebih mendalam.
2. Pengguna aplikasi diharapkan melakukan pengelolaan dan pembaruan data secara rutin agar informasi transaksi dan persediaan yang tersimpan dalam sistem tetap akurat dan sesuai dengan kondisi sebenarnya.
3. Aplikasi dapat terus disesuaikan dengan kebutuhan operasional Apan Cell yang berkembang sehingga mampu memberikan dukungan yang lebih optimal terhadap proses pengelolaan penjualan dan persediaan barang.
4. Penelitian selanjutnya diharapkan dapat mengembangkan sistem yang telah dibuat sehingga memiliki cakupan pengelolaan usaha yang lebih luas sesuai dengan kebutuhan pengguna.

DAFTAR PUSTAKA

- Ardian, Faradila, C., & Saputra, F. (2021). *Sistem Informasi Akuntansi Penjualan Pada Cv. Mitra Mobil Aceh Besar*. 8.
- Ardian Nugraha, E., & Aziz, R. A. (2024). *Sistem Manajemen Persediaan Stok Barang Berbasis Web Yang Meningkatkan Efisiensi Stok Barang*.
- Arianto, A., Amalia, D., Dana, R., Ajiz, A., & Hamonangan, R. (2022). *Rancang Bangun Aplikasi Persediaan Barang pada Toko Endo Tani*. 7.
- Arif, S. M. (2023). Pembuatan Website Informasi & Pendaftaran Webinar Umum Menggunakan Php & Mysql. *Journal of Information System, Applied, Management, Accounting*, 7(3), 789–796.
<https://doi.org/10.52362/jisamar.v7i3.1168>
- Asriani, Herdhiansyah, D., & Rismawan, Y. (2022). *Digital Marketing Produk Sagu*.
- Assegaf, S., Rasywir, E., Pratama, Y., Kurniabudi, Rohaini, E., Sharipuddin, Winanto, E. A., Hendri, Sutoyo, M. A. H., Jasmir, Wijaya, I. S., Meisak, D., & Akbar, N. (2024). *Penerapan Teknologi Informasi dalam Era Revolusi Industri di Indonesia*. Tohar Media.
- Cahyono, D. E., & Jayanti, A. (2022). *Implementasi Aplikasi Kasir Berbasis Web Pada Toko Ghafya Fruits Shop*.
- Darip, M. D., Rohman, A., & Aziz, A. (2024). Implementasi Aplikasi Manajemen Stok Dengan Metode FIFO Untuk Optimalisasi Inventori: Studi Kasus Aziz Helmet Store. *Software Development, Digital Business Intelligence, and Computer Engineering*, 3(1), 01–07.
<https://doi.org/10.57203/session.v3i1.2024.01-07>
- Fauzi, R., Nasution, H. N., Hastini, F., Zainy, A., & Simanjuntak, F. (2022). Perancangan Aplikasi Pariwisata Berbasis Android Di Kota Padang Sidempuan. *Jurnal Education And Development*, 11(1), 437–442.
<https://doi.org/10.37081/ed.v11i1.2687>
- Gunawan, A. (2024). *Mobile Programming Menggunakan Flutter dan Visual Studio Code Untuk Pemula*. www.penerbitlitnus.co.id

- Hafidh, F., Izzatul Y.A, R., & Arminarahmah, N. (2021). Penerapan Aplikasi Kasir Portabel pada Kedai Sari Kopi Banjarbaru. *J-Dinamika : Jurnal Pengabdian Masyarakat*, 6(1), 200–205. <https://doi.org/10.25047/j-dinamika.v6i1.1439>
- Hasim, M. S. F., & Arifin, A. (2025). Aplikasi Kasir Agen Portable Berbasis Mobile. *Jurnal Bisnis Mahasiswa*, 5(1), 275–286. <https://doi.org/10.60036/jbm.v5i1.341>
- Hidayat, F. (2022). *Aplikasi Kasir Berbasis Android Pada Warung Spesial Soto Sehat Mbak Jum*.
- Irawan, D. (2025). *Metode Penelitian Ekonomi & Bisnis*.
- Johari, mun. (2024). Aplikasi Kompas Kiblat Digital Untuk Penyandang Tunanetra Berbasis Android Study Kasus Di Pijat Tunanetra Santa 3 Rawamangun. *Universitas Saintek Muhammadiyah*, 10(2).
- Krischnan, R., Pradipta, T., & Widiono, S. (2024). Perancangan Aplikasi Point Of Sales Berbasis Android Pada Komedi Cafe. *Jurnal Teknologi Informasi*, 5(3). <https://doi.org/10.46576/djtechno>
- Lisa, L., Sutejo, H., & Kungkung, A. Y. (2025). *Sistem Informasi Manajemen Persediaan Barang Berbasis Website di PT. Rejo Mulyo Solution*. <https://doi.org/10.38035/jemsi.v6i3>
- Martin, R. S., & Dewanto, Y. (2023). Prototipe Kunci Pintu Otomatis Menggunakan Sensor Kamera Berbasis Raspberry. *Jurnal Teknologi Industri*, 12(1).
- Mulyana, N., Sulistyanto, A., & Yasin, V. (2021). Perancangan Sistem Informasi Pengelolaan Aset It Berbasis Web Pada Pt Mandiri Axa General Insurance. *Jurnal Manajemen Informatika Jayakarta*, 1(3), 243. <https://doi.org/10.52362/jmijayakarta.v1i3.498>
- Nugraha, F. W., & Kalifia, A. D. (2024). *Aplikasi E-Stock Mobile Dengan QR Code Untuk Pengelolaan Persediaan Barang Elektronik*. 13.
- Nurwicaksono, A. B. (2024). Perancangan Sistem Manajemen Kasir Berbasis Android Pada Blessing Of Shoes. *Journal of Information System Management (JOISM) e-ISSN*, 5(2), 2715–3088.
- Pamungkas, F. T. (2023). *Perancangan UI/UX Aplikasi Variety off Food Layanan Penjualan Makanan Secara Online Menggunakan Aplikasi Figma*. 6(1).

- Puspitasari, A. (2024). *Analisis Akuntansi Persediaan Barang Dagangan Pada Toko Baju Bekas Import (Studi Kasus Pada Toko Baju Bekas Import Tropical Trift)*. 4(2).
- Riswanto, B., & Fahrudin, I. (2024). Perancangan Aplikasi Kasir Aira Motor Berbasis Android Menggunakan Firebase Realtime Database. *Digital Transformation Technology*, 4(2), 776–784. <https://doi.org/10.47709/digitech.v4i2.4593>
- Riyansyah, A. (2021). Perancangan Sistem Informasi Penggajian Berbasis Web Menggunakan Php Dan Mysql Di Lion Parcel Cisokan Kota Bandung. *Infotech Journal*, 29–35. <https://doi.org/10.31949/infotech.v7i2.1373>
- Sadiah, H., Agus Sunandar, M., & Defriani, M. (2024). Rancang Bangun Sistem Informasi Manajemen Stok Barang Pada Toko Plastik Bimiu Dengan Metode First In-First Out (FIFO). *Jurnal Riset Sistem Informasi Dan Teknik Informatika*, 2(5). <https://doi.org/10.61132/mercurius.v2i4.312>
- Sahir, S. H. (2021). *Metodologi Penelitian*. www.penerbitbukumurah.com
- Setiawan, P. R., Ramadhan, R. A., & Labellapansa, A. (2022). *Pelatihan Pemrograman Flutter*. 03.
- Sinar, S., Pagala, M. arman Y., & Baso, A. (2024). Analisis Kinerja Badan Usaha Milik Desa Terhadap Peningkatan Usaha Rumah Tangga Petani (Home Industri) Didesa Lapeo Kecamatan Campalagian Kabupaten Polewali Mandar. *Jurnal Agroterpadu*, 3(2), 176. <https://doi.org/10.35329/ja.v3i2.5271>
- Sintawati, I. D. (2022). *Komparasi Metode RAD Dengan RUP Pada Pengembangan Sistem Informasi*. 7.
- Siregar, L. R. (2025). Rancang Bangun Aplikasi Kasir Berbasis Web pada Usaha Roti Karkus. *Bisnis Dan Teknologi*, 11(1).
- Sofi, N., & Dharmawan, R. (2022). Perancangan Aplikasi Bengkel Csm Berbasis Android Menggunakan Framework Flutter (Bahasa Dart). *JTS*, 1(2).
- Suhendri. (2024). *Mobile Programming With Android*. Penerbit Adab.
- Suparman, M., Rosada, M., Lutpi, M., Kamaliya, P., Sabaniah, F., Alfian, R. H., Ramadhan, F., Alfaro, I., & Rosdiana, M. (2023). *Mengenal Aplikasi Figma Untuk Membuat Content Menjadi Lebih Interaktif Di Era Society 5.0*. 1(6), 552–555. <https://jurnal.portalpublikasi.id/index.php/AJP/index>

- Suryana, T. (2021). *Belajar Bahasa Pemrograman Dart (1)*.
- Tangkere, A. J., Hasan Jan, A. Bin, & Karuntu, M. M. (2024). *Analisis Pengendalian Persediaan Pupuk (Studi Kasus Pada Toko Berlian Tani)*. 12, 685–696.
- Waruwu, M. (2024). *Pendekatan Penelitian Kualitatif: Konsep, Prosedur, Kelebihan dan Peran di Bidang Pendidikan*. 5.
<https://afeksi.id/jurnal/index.php/afeksi/>



LAMPIRAN

Lampiran 1 Surat Permohonan Penelitian



**FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH**

Prabumulih, 07 November 2025

Nomor : 070 /FASILKOM/UNPRA/Inf-S1/XI/2025
Lampiran : -
Perihal : Permohonan Penelitian dan Pengambilan Data

Kepada Yth,
Pemilik Apan Cell
Di
Tempat

Dalam rangka pelaksanaan skripsi mahasiswa Program Studi Informatika Fakultas Ilmu Komputer Universitas Prabumulih, maka kami harapkan bantuan dari Bpk/Ibu kiranya mengizinkan mahasiswa:

Nama : Maulidia Rosinta
Nim : 2022230012
Program Studi : Informatika

Untuk melaksanakan penelitian dan pengambilan data ditempat Bpk/Ibu Pimpin guna menyelesaikan skripsi. Atas bantuan dan kerjasamanya kami ucapkan terima kasih.

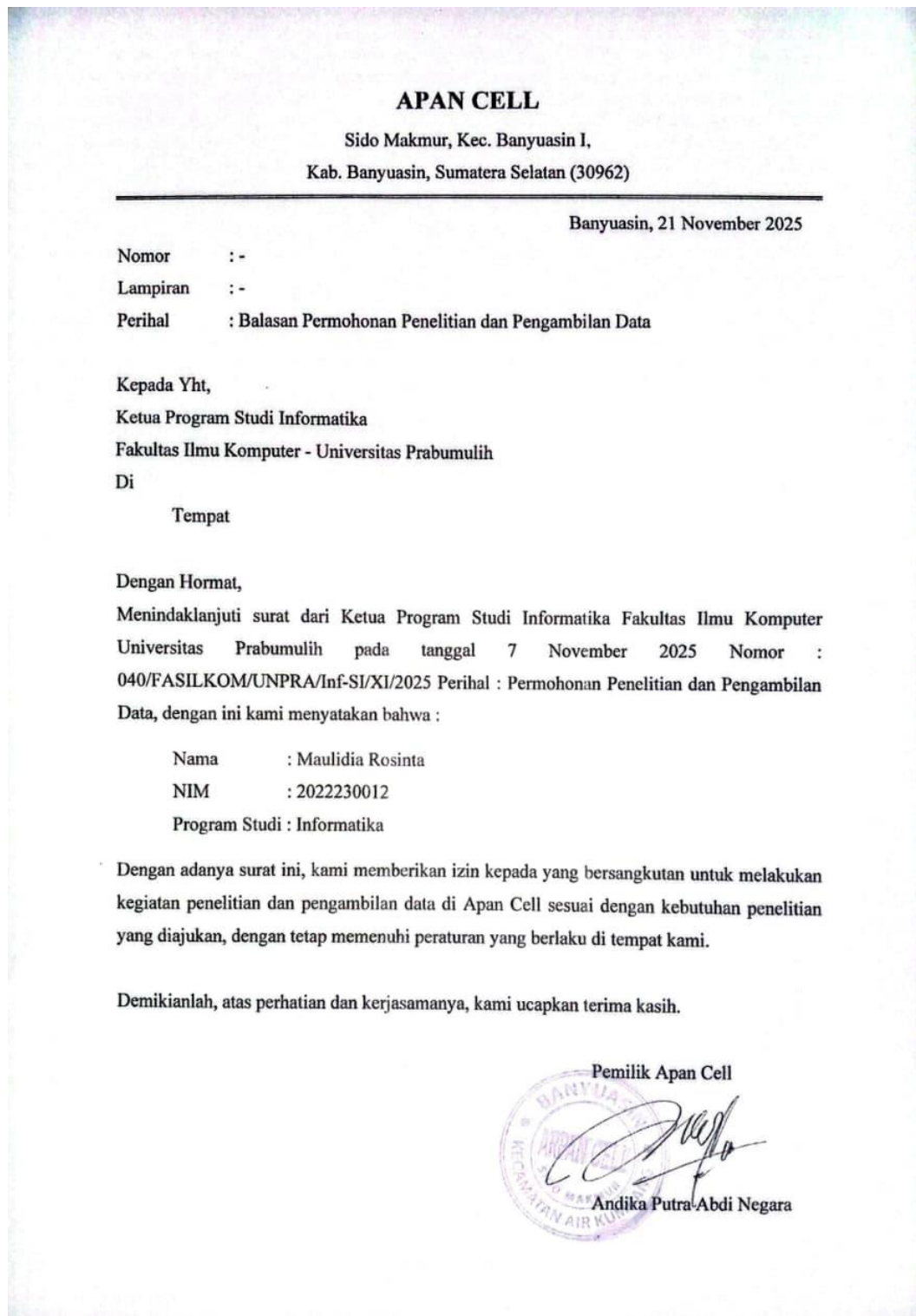
Ketua Program Studi,
Informatika



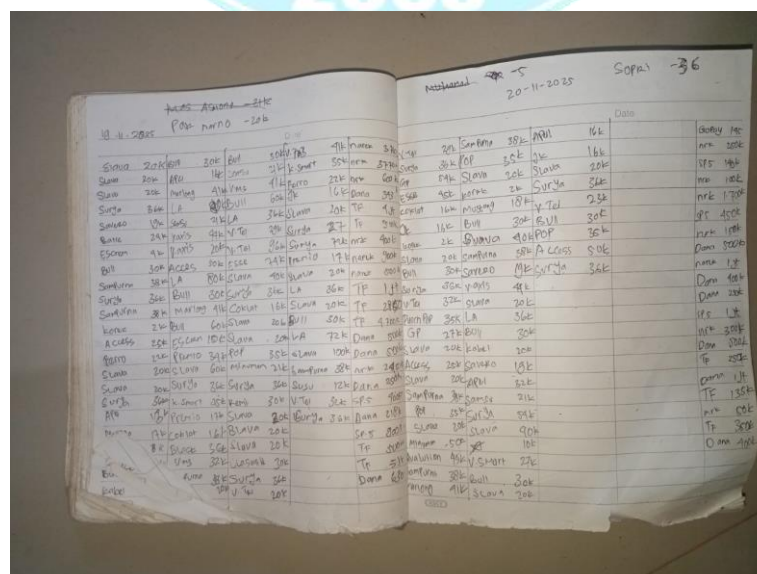
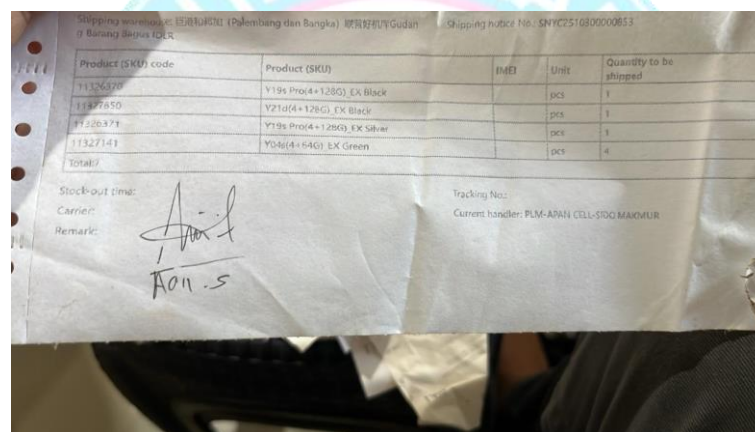
Iwan Setiawan, S.Kom., M.Kom.
NIDN.0209129101

Jalan Patra No.50 RT.01 RW.03 Kelurahan Sukaraja Kecamatan Prabumulih Selatan
Kota Prabumulih Sumatera Selatan Indonesia Telpn 0713-3224417 Fax 0713-322418
e-mail : mail@unpra.ac.id

Lampiran 2 Surat Balasan Permohonan Penelitian



Lampiran 3 Dokumentasi Wawancara & Observasi



Lampiran 4 Daftar Pertanyaan Wawancara

Draft Wawancara

Nama Narasumber : Andika Putra Abdi Negara
Jabatan : Pemilik Toko
Tempat Wawancara : Apan Cell
Tanggal Wawancara : 21 November 2025
Pewawancara : Maulidia Rosinta

No	Pertanyaan	Jawaban
1	Bisa Bapak jelaskan sejak kapan Apan Cell berdiri dan bagaimana perkembangannya sampai sekarang?	Apan Cell berdiri pada tahun 2021, saat pandemi Covid-19 sedang melanda. Awalnya usaha ini bukan konter HP, melainkan Apan <i>Sport</i> yang menjual perlengkapan olahraga. Namun karena kondisi pandemi membuat masyarakat kesulitan memenuhi kebutuhan digital seperti pulsa dan token listrik, akhirnya kami memutuskan untuk mengubah usaha menjadi konter HP. Dari situ, Apan Cell mulai berkembang sedikit demi sedikit sampai sekarang bisa menyediakan lebih banyak layanan bagi warga sekitar.
2	Apa alasan Bapak memilih membuka usaha konter HP ini?	Karena di daerah ini konter HP masih sangat terbatas, dan layanan yang mereka sediakan juga belum lengkap. Misalnya, untuk <i>top up e-wallet</i> , transfer antar <i>bank</i> , atau tarik tunai, dulu warga harus pergi cukup jauh. Melihat kebutuhan itu, saya berpikir usaha ini bisa menjadi peluang yang baik sekaligus membantu masyarakat sekitar.
3	Apa saja barang atau layanan yang tersedia di Apan Cell?	Untuk barang, kami menyediakan berbagai kebutuhan gadget seperti <i>handphone</i> , <i>casing</i> , baterai, <i>charger</i> , hingga <i>power bank</i> . Selain itu kami menyediakan rokok dan es cream di toko kami. Sedangkan layanannya cukup lengkap, mulai dari pengisian pulsa dan paket data, pembayaran tagihan, <i>top up Dana</i> dan <i>OVO</i> , transfer bank, sampai tarik tunai.
4	Bagaimana proses transaksi penjualan yang dilakukan setiap hari?	Sejauh ini proses transaksi kami masih dilakukan secara manual. Jadi setiap ada transaksi, semuanya dicatat di buku dan dilakukan secara langsung tanpa bantuan aplikasi.

5	Bagaimana cara pencatatan stok barang selama ini?	Pencatatan stok barang juga masih manual menggunakan buku. Saya mengecek stok secara berkala, walaupun memang memerlukan waktu lebih lama untuk memastikan jumlah barang tetap sesuai.
6	Kendala apa yang sering terjadi dalam proses penjualan dan pencatatan persediaan barang?	Kendalanya biasanya dari segi pencatatan yang masih manual, kadang bisa terjadi salah tulis atau ada barang yang terlewat tercatat. Selain itu, saat kondisi toko sedang ramai, pengecekan stok jadi kurang maksimal karena membutuhkan waktu.
7	Berapa rata-rata jumlah pelanggan yang datang setiap hari?	Kalau dihitung rata-rata, sekitar 200 pelanggan datang setiap hari, dari pagi hingga malam.
8	Bagaimana penilaian Bapak terhadap pencatatan manual yang digunakan saat ini? Apakah sudah membantu atau masih terdapat kendala dalam penerapannya?	Sebenarnya pencatatan manual masih membantu, tetapi saya merasa kurang efektif untuk jangka panjang. Karena selain memakan waktu, risiko salah catat juga cukup besar. Kalau ada aplikasi yang bisa membantu, pasti pekerjaan akan lebih mudah dan cepat.
9	Apakah Bapak bersedia menggunakan aplikasi kasir untuk membantu usaha ini?	Tentu saja saya bersedia. Kalau aplikasi kasir bisa membantu pekerjaan di toko, saya sangat senang menggunakan aplikasi tersebut di Apan Cell.
10	Apakah Bapak bersedia jika nantinya dilakukan uji coba aplikasi yang dibangun peneliti?	Tentu saja, selama proses uji coba dilakukan dengan penjelasan terlebih dahulu dan tidak mengganggu aktivitas kerja di toko.
11	Apa harapan Bapak untuk perkembangan Apan Cell kedepannya?	Harapan saya, semoga Apan Cell bisa terus berkembang dan semakin maju. Minimal bisa bertahan 10 sampai 15 tahun ke depan.

Menyetujui

Narasumber



Andika Putra Abdi Negara

Banyuasin, 21 November 2025

Pewawancara


Maulidia Rosinta

Lampiran 5 Daftar Wawancara Penyerahan Aplikasi

Draft Wawancara

Nama Narasumber : Andika Putra Abdi Negara
Jabatan : Pemilik Toko
Tempat Wawancara : Apan Cell
Tanggal Wawancara : 02 Juli 2026
Pewawancara : Maulidia Rosinta

No	Pertanyaan	Jawaban
1	Bagaimana pendapat Bapak mengenai tampilan aplikasi yang telah dibuat?	Tampilan aplikasi terlihat rapi, sederhana, dan mudah dipahami sehingga pengguna tidak mengalami kesulitan saat mengoperasikannya.
2	Apakah menu-menu pada aplikasi sudah sesuai dengan kebutuhan toko?	Ya, menu yang tersedia sudah sesuai dengan kebutuhan operasional toko dan memudahkan pengguna dalam mengakses fitur yang diperlukan.
3	Apakah aplikasi mudah digunakan?	Ya, aplikasinya mudah digunakan. Alur penggunaan aplikasi juga sangat jelas.
4	Bagaimana pendapat Bapak mengenai proses login pada aplikasi?	Proses login berjalan dengan baik dan cukup mudah digunakan. Hal ini membantu menjaga keamanan akses ke dalam aplikasi.
5	Apakah fitur-fitur yang tersedia sudah berfungsi dengan baik?	Ya, seluruh fitur yang telah diuji dapat berfungsi dengan baik sesuai dengan tujuan penggunaannya.
6	Apakah aplikasi kasir ini sudah bisa membantu pekerjaan ditoko, dibandingkan dengan sistem manual sebelumnya?	Ya, aplikasi ini sudah bisa membantu mempercepat proses kerja, mengurangi kesalahan pencatatan, dan membuat pengelolaan data menjadi lebih teratur.
7	Apakah terdapat kendala saat menggunakan aplikasi?	Setelah saya mencoba aplikasinya, aplikasi ini bisa berjalan tanpa kendala.

-
- 8 Bagaimana kecepatan respon aplikasi saat digunakan? Respon aplikasi cukup cepat sehingga proses pengoperasian berjalan dengan lancar.
-
- 9 Fitur apa yang menurut Bapak paling bermanfaat? Fitur pengelolaan data dan pembuatan laporan sangat membantu karena mempermudah proses pencarian informasi dan penyusunan laporan.
-
- 10 Apakah Bapak memiliki saran untuk pengembangan aplikasi ke depannya? Secara keseluruhan aplikasi sudah baik. Saya harap agar aplikasi ini dapat ditambahkan fitur baru sesuai kebutuhan toko ke depannya.
-

Menyetujui

Narasumber



Andika Putra Abdi Negara

Banyuasin, 02 Juli 2026

Pewawancara



Maulidia Rosinta



**FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH**

**KESEDIAAN SEBAGAI PEMBIMBING SKRIPSI
PROGRAM STUDI INFORMATIKA**

Yang bertanda tangan di bawah ini:

Nama : Nur Aini H, S.Kom., M.Si., M.Kom.
NUPTK : 2434764664300022
Program Studi : Informatika
Fakultas : Ilmu Komputer

Bersedia menjadi pembimbing I Skripsi, dan

Nama : Iwan Setiawan, S.Kom., M.Kom.
NUPTK : 1541769670130303
Program Studi : Informatika
Fakultas : Ilmu Komputer

Bersedia menjadi pembimbing II Skripsi

Dari mahasiswa Program Studi Informatika Fakultas Ilmu Komputer Universitas Prabumulih di bawah ini:

Nama : Maulidia Rosinta
NIM : 2022230012
Judul Skripsi : Perancangan Aplikasi Kasir Berbasis *Android* Untuk Pengelolaan Penjualan Dan Persediaan Pada Apan Cell.

Demikian surat kesediaan membimbing ini kami buat, atas perhatiannya kami ucapkan terima kasih.

Prabumulih, Juni 2026

Dosen Pembimbing I

Dosen Pembimbing II

(Nur Aini H, S.Kom., M.Si., M.Kom)
NIDN. 0202018602

(Iwan Setiawan, S.Kom., M.Kom)
NIDN. 0209129101



FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : MAULIDIA ROSINTA
NIM : 2022230012
PROGRAM STUDI : INFORMATIKA
NAMA PEMBIMBING I : NUR AINI H, S.Kom., M.Kom
JUDUL : PERANCANGAN APLIKASI KASIR BERBASIS ANDROID UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN PADA APAN CELL

NO	TGL. KONSULTASI	TOPIK POKOK YANG DIBICARAKAN	TANDA TANGAN PEMB. I	TGL MENGHADAP KEMBALI
1.	04/02/2026	Bab 1 Ren : - Batas • Muli - Tug penelitian - Batas Muli	✓	05/02/2026
2.	09/02/2026	Ren : - Rumi Muli	✓	10/02/2026
3.	25/02/2026	Acc Bab 1	✓	27/02/2026
4.	25/02/2026	Bab 11 Ren : petisi rami di unt penelitian	✓	
5.	2/03/2026	Acc Bab 11	✓	
6.	3/03/2026	Bab 11 Revisi	✓	4/03/2026

Mulai Bimbingan : 04/02/2026

Batas Akhir Bimbingan :

PERHATIAN !
TIDAK BOLEH HILANG
SETIAP BIMBINGAN HARUS DIBAWA

Mengetahui Kaprodi Informatika

IWAN SETIAWAN, S.Kom.M.Kom.



FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : MAULIDIA ROSINTA
NIM : 2022230012
PROGRAM STUDI : INFORMATIKA
NAMA PEMBIMBING I : NUR AINI H, S.Kom., M.Kom
JUDUL : PERANCANGAN APLIKASI KASIR BERBASIS ANDROID UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN PADA APAN CELL

NO	TGL. KONSULTASI	TOPIK POKOK YANG DIBICARAKAN	TANDA TANGAN PEMB. I	TGL MENGHADAP KEMBALI
7.	6/4/2026	Bab IV Peneri : - Aher fyth bja - kebtha puiol. - puiol 4st - use.	7.	
8	25/5/2026	Ace Bab IV	7.	
9.	26/5/2026	Bab V Peneri : - deth - puiol 10/UX - puiol 8st	7.	
10.	8/6/2026	Puiol : puiol 8st	7.	
11.	11/6/2026.	Ace Bab V	7.	12/6/2026

Mulai Bimbingan :

Batas Akhir Bimbingan :

PERHATIAN !
TIDAK BOLEH HILANG
SETIAP BIMBINGAN HARUS DIBAWA

Mengetahui Kaprodi Informatika

IWAN SETIAWAN, S.Kom.M.Kom.



FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : MAULIDIA ROSINTA
NIM : 2022230012
PROGRAM STUDI : INFORMATIKA
NAMA PEMBIMBING II : IWAN SETIAWAN, S.Kom., M.Kom
JUDUL : PERANCANGAN APLIKASI KASIR BERBASIS ANDROID UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN PADA APAN CELL

NO	TGL. KONSULTASI	TOPIK POKOK YANG DIBICARAKAN	TANDA TANGAN PEMBIMBING	TGL MENGHADAP KEMBALI
1	4-2-2026	BAB I Acc		
		BAB II Acc		
		BAB III a. Rancangan Sistem		
2	4-2-2026	BAB III Acc		
3	22-4-2026	BAB IV a. Rdd Judul. b. Pengguna c. Hardware d. game class diagram e. Database		

Mulai Bimbingan : 4-2-2026
Batas Akhir Bimbingan :

Mengetahui Kaprodi Informatika

PERHATIAN !
TIDAK BOLEH HILANG
SETIAP BIMBINGAN HARUS DIBAWA

IWAN SETIAWAN, S.Kom.M.Kom.



FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH

KARTU BIMBINGAN SKRIPSI

NAMA : MAULIDIA ROSINTA
NIM : 2022230012
PROGRAM STUDI : INFORMATIKA
NAMA PEMBIMBING II : IWAN SETIAWAN, S.Kom., M.Kom
JUDUL : PERANCANGAN APLIKASI KASIR BERBASIS ANDROID UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN PADA APAN CELL

NO	TGL. KONSULTASI	TOPIK POKOK YANG DIBICARAKAN	TANDA TANGAN PEMB. II	TGL MENGHADAP KEMBALI
4	6/5/2026	BAB IV a. Bold passage b. Delah c. Tabel min 4.		
5	19/5/2026	BAB IV ACC		
6	25/5/2026	BAB V a. ganti data di Bab IV b. pengisian		
7	29/5/2026	BAB V a. Database.		
8	3/6/2026	BAB V ACC		

Mulai Bimbingan :

Mengetahui Kaprodi Informatika

Batas Akhir Bimbingan :

PERHATIAN !
TIDAK BOLEH HILANG
SETIAP BIMBINGAN HARUS DIBAWA

IWAN SETIAWAN, S.Kom.M.Kom.



NAMA : MAULIDIA ROSINTA
NIM : 2022230012
PROGRAM STUDI : INFORMATIKA
NAMA PEMBIMBING II : IWAN SETIAWAN, S.Kom., M.Kom
JUDUL : PERANCANGAN APLIKASI KASIR BERBASIS ANDROID UNTUK PENGELOLAAN PENJUALAN DAN PERSEDIAAN PADA APAN CELL

[illegible]

**PERHATIAN !
TIDAK BOLEH HILANG
SETIAP BIMBINGAN HARUS DIBAWA**

IWAN SETIAWAN, S.Kom.M.Kom.



**FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH**

PENILAIAN REVIEW SKRIPSI

Nama : Maulidia Rosinta
NIM : 2022230012
Judul Skripsi : Perancangan Aplikasi Kasir Berbasis *Android* Untuk Pengelolaan
Penjualan Dan Persediaan Pada Apan Cell.
Hari/Tanggal :
Saran dan masukan :

gcp
13/6/2026

Rekomendasi*:

- a. Dilanjutkan tanpa perbaikan
- b. Dilanjutkan dengan perbaikan
- c. Ditolak

Prabumulih, 22 Juni 2026
Ketua Sidang

(Nur Aini H, S.Kom., M.Si., M.Kom)
NUPTK. 2434764664300022

**pilih salah satu*



**FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH**

PENILAIAN REVIEW SKRIPSI

Nama : Maulidia Rosinta
NIM : 2022230012
Judul Skripsi : Perancangan Aplikasi Kasir Berbasis *Android* Untuk Pengelolaan
Penjualan Dan Persediaan Pada Apan Cell.
Hari/Tanggal : KMN/22/6/2026

Saran dan masukan :

IKUTI ARAHAN PENGUJI DENGAN
KONSULTASI KE PEMBIMBING DAHULU

ACC SIAP JILID

13-7-2026

Rekomendasi*:

- a. Dilanjutkan tanpa perbaikan
- ☒ b. Dilanjutkan dengan perbaikan
- c. Ditolak

Prabumulih, 22-6-.....2026
Penguji I

(Iwan Setiawan, S.Kom., M.Kom)
NUPTK. 1541769670130303

*pilih salah satu



**FAKULTAS ILMU KOMPUTER
UNIVERSITAS PRABUMULIH**

PENILAIAN REVIEW SKRIPSI

Nama : Maulidia Rosinta
NIM : 2022230012
Judul Skripsi : Perancangan Aplikasi Kasir Berbasis *Android* Untuk Pengelolaan
Penjualan Dan Persediaan Pada Apan Cell.

Hari/Tanggal :

Saran dan masukan :

- Usecase diagram

- Wawancara lanjutan ke obyek

8

7

09/26/2022

ace sup

Rekomendasi*:

- Dilanjutkan tanpa perbaikan
- Dilanjutkan dengan perbaikan
- Ditolak

Prabumulih, 22.06.2026
Penguji II

(Fajriyah, S.Kom. M.Kom)
NUPTK. 1159766667237103

*pilih salah satu